Scanning system 101, or other devices, may traverse the map to identify vectors originating in the Internet and reaching the devices and systems in cloud infrastructure 106. In some embodiments, generating the map may include enumerating Internet-accessible services that are capable of serving as an Internet proxy. In some embodiments, a user may control scanning system 101 (e.g., via user device 102) to display the map on a graphical user interface.

[0084] In some embodiments, generating the map may include enumerating properties of all assets, including: virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, load balancer 115, log files or databases, API gateway resources, API gateway REST APIs, Autoscaling groups, CloudTrail logs, CloudFront services, volumes, snapshots, VPCs, subnets, route tables, network ACLs, VPC endpoints, NAT gateways, ELB and ALB, ECR repositories, ECS clusters, services, and tasks, EKS, S3 bucket and Glacier storage, SNS topics, IAM roles, policies, groups, users, KMS keys, and Lambda functions. In some embodiments, generating the map in step 227 may further include analyzing devices or systems for a subset of risk related situations, including determinations of compromise situations (e.g., where an attacker has already gained access), imminent compromise situations (e.g., a known attack vector exists and can be used, such as a data store that is exposed to the public Internet without authentication), hazardous situations (e.g., a serious security implication, but no full attack vector exists), or informational situations (e.g., when storage 111A-111D has a limited amount of free space, or an unexploitable vulnerability exists).

[0085] Generating the map in step 227 may include recording information such a region identifier, site identifier, datacenter identifier, physical address, network address, workload name, or any other identifier which may be acquired via an API provided through a cloud service provider's system.

[0086] In step 229, scanning system 101 may provide alerts, e.g., to user device 102, indicating any situations found during the process of generating the map. For example, scanning system 101 may send one or more of an email, a popup alert, a text message, or other notification to user device 102.

[0087] After performing step 229, the process may return to Fig. 2A and step 205.

[0088] **Fig. 2D** is a schematic block diagram illustrating an exemplary embodiment of a process 230 for assessing, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2D are executed using software and hardware of scanning system 101. In other embodiments, the steps in Fig. 2D may be executed by other devices. In some embodiments, scanning system 101 may execute the steps in Fig. 2D against a snapshot created in step 223 of Fig. 2C and stored at scanning system 101. In some embodiments, the steps in Fig. 2D may be performed in an order other than those depicted in Fig. 2D. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one

WIZ, Inc. EXHIBIT - 1004 (3 of 4)
WIZ, Inc. v. Orca Security LTD.
0794

step of Fig. 2D may be used to provide context or other information for use in another step of Fig. 2D.

[0089] Process 230 begins with step 231. In step 231, scanning system 101 may perform a step of vulnerability scanning. In some embodiments, step 231 comprises extracting everything in the snapshot, including operating system packages, installed software applications, libraries, and program language libraries such as Java archives, Python packages, Go modules, Ruby gems, PHP packages, and Node.js modules, or other software applications.

[0090] In some embodiments, step 231 may determine library versions, software versions, and other identifying characteristics of software and operating systems in the snapshot. Scanning system 101 may then try to match them to known vulnerabilities stored in a vulnerability database (e.g., one of databases 103A-103D). The vulnerability database, in some embodiments, may include vulnerability data from: NVD, WPVulnDB, US-CERT, Node.js Security Working Group, OVAL – Red Hat, Oracle Linux, Debian, Ubuntu, SUSE, Ruby Advisory Database, JVN, Safety DB(Python), Alpine secdb, PHP Security Advisories Database, Amazon ALAS, RustSec Advisory Database, Red Hat Security Advisories, Microsoft MSRC, KB, Debian Security Bug Tracker, Kubernetes security announcements, Exploit Database, Drupal security advisories, JPCERT. The vulnerability database may also include other vulnerability data (including, e.g., manually-added vulnerability data or vulnerability sources not listed above).

[0091] In step 233, scanning system 101 may perform a step of configuration scanning. In step 233, scanning system 101 may gather configuration information—such as a list of users of each system or device (e.g., virtual machines 107A-D), each system's or device's services, password hashes, and application-specific configurations for software/services such as Apache, Nginx, SSH, and other services. In some embodiments, scanning system 101 may perform a first analysis on all information collected in step 233 to remove sensitive information (e.g., social security numbers, passwords, birthdates) before proceeding to review the configuration information. In some embodiments, step 233 may comprise verifying adherence of the systems or devices in cloud infrastructure 106 to standards or benchmarks established by an external entity, such as the Center for Internet Security.

[0092] In some embodiments, scanning system 101 may perform a benchmarking process to detect misconfigurations of any services based on the information gathered in step 233. For example, scanning system 101 may determine the software version of each service and examine it against known vulnerabilities (e.g., stored in database 103A).

[0093] In some embodiments, scanning system 101 may determine bugs or other configuration risks that might only be exploitable from internal machines, because such bugs and risks can facilitate an attacker's lateral movement.

[0094] In some embodiments, scanning system 101 may evaluate network misconfigurations and their implications. Scanning system 101 may query devices and systems capable of routing traffic (e.g., load balancer 115, routers, switches, firewalls, and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations.

[0095] In step 235, scanning system 101 may perform a step of malware scanning. In some embodiments, scanning system 101 may perform malware scanning across all filesystems in the snapshot (e.g., gathered from virtual machines 107A-107D or storage 111A-111D). Scanning system 101 may use multiple malware scanning software solutions to perform a malware scan against the filesystems, including one sourced from another vendor, such as bucketAV, Trend Micro Cloud One, Sophos Cloud Optix, Crowdstrike Falcon CWP, or others. In some embodiments, malware scanning in step 235 comprises utilizing signatures, heuristics, or sandboxing capabilities to deduce whether there is an infection on the machine.

[0096] In step 237, scanning system 101 may perform a step of lateral movement scanning. An attacker who establishes a network foothold usually attempts to move laterally from one resource to another in search of rich targets such as valuable data. Stolen passwords and keys unlock access to servers, files, and privileged accounts. In some embodiments, scanning system 101 may gather keys from each scanned system or device (e.g., virtual machines 107A-107D or storage 111A-111D). In some embodiments, scanning system 101 searches for passwords, scripts, shell history, repositories, or other data that may contain passwords, cloud access keys, SSH keys, or other key/password/access information that provide unchecked access to important resources. In some embodiments, scanning system 101 searches for such keys/passwords/access information and calculates a "hash" (a mathematical fingerprint) of each string. Scanning system 101 then attempts to match the hashed strings to hashes of strings that that are stored on different systems or devices. This will be used to detect the potential lateral movement between assets.

[0097] In step 239, scanning system 101 may perform a step of key/password scanning. As one example situation, suppose there is a weak or unprotected password stored (in plain text) in storage 111A. For example, if a personal email account has been compromised, the passwords may be known about in advance. Scanning system 101 may search the snapshot for similar usernames or login names, and, either using known dictionaries or the account owner's previously leaked passwords (stored in, e.g., database 103A), may attempt to login to one or more systems or devices in cloud infrastructure 106, and may record the result thereof.

[0098] In some embodiments, scanning system 101 may perform a "fuzzy search" on any usernames found in a password database (e.g., database 103A) to determine existing targets for password testing. In some embodiments, the fuzzy search uses the Damerau-Levenshtein edit distance algorithm.

[0099] As one example, scanning system 101 may determine that a leaked username includes the email address johndoe@acme.com. Scanning system 101 may try to match passwords from john.doe@acme.net, johndoe@gmail.com, and other variations. Scanning system 101 may perform a similar process against leaked passwords (e.g., if a leaked password is "Victory@19," scanning system may attempt to log in using "Victory@20."

[00100] In step 241, scanning system 101 may perform a step of sensitive information scanning. In some embodiments, scanning system 101 may search the snapshot for sensitive information, such as personally identifiable information (PII), Social Security numbers, healthcare information, or credit card numbers. In some embodiments, scanning system 101 may search data repository history as well. This is because it is not uncommon for an entire production environment repository to be cloned, with no one remembering the copy contains sensitive information. In some situations, detecting sensitive data not secured is critical in adherence to data privacy regulations.

[00101] To be certain that such alerts do not constitute false positives, in some embodiments, scanning system 101 may perform statistical scans against the data. For example, it is possible for a random number to resemble a Social Security number, yet it is extremely unlikely for the majority of a file with thousands of numbers to be valid Social Security Number by pure chance.

[00102] In step 243, scanning system 101 may perform a step of container scanning. In some embodiments, scanning system 101 may apply one or more of the preceding steps of Fig. 2D against containerized environments. In some embodiments, in order to do so, scanning system 101 reconstructs a container runtime layered file system (LFS) before recursively running one or more of steps 231-241 on the reconstructed file system.

[00103] Scanning system 101 may read network information from the snapshot in order to determine which services within which containers are exposed externally and within which ports they are accessible. For example, scanning system 101 may identify a port on which a vulnerable application is accessible based on known software vulnerabilities for a versions of software application. Scanning system 101 may query network accessibility information via an API provided through a cloud service provider's system and may use it to identify specific vulnerabilities susceptible to attack.

[00104] After performing step 243 on one or more containers, the process may return to Fig. 2A and step 207 (as discussed above).

[00105] **Foundational Techniques**

[00106] Aspects of this disclosure may include establishing a trusted relationship between a source account in a cloud environment and a scanner account. A trusted relationship, as used herein, may refer to a secure communication channel between at least two accounts (e.g., domains). The secure communications channel may include, for example, an

administration link, communication link, a connection or communication network configured with security protocols, or any other secure relationship between two networked entities.. A trusted relationship between two accounts may enable user accounts and global groups to be used in a domain other than the domain where the accounts are defined. A source account, as used herein, may refer to a domain, a location on a network server, or anything used to access system resources. A cloud environment, as used herein, may refer to a platform implemented on, hosted on, and/or accessing servers that may be reached via the Internet or other shared network. A scanner account, as used herein, may refer to any type of account associated with a scanner. A scanner, as used herein, may refer to a device for examining, reading, or monitoring electronically accessible information. Fig. 1 illustrates one example of a scanner, which is depicted in an exemplary manner as scanning system 101.

[00107] By way of example, establishing a trusted relationship between a source account in a cloud environment and a scanner account may include an integration process (creating a connection between an account on scanning system 101 and an account on cloud infrastructure 106) discussed with reference to Fig. 2A. By way of one example, scanning system 101 of Fig. 1 may create a trusted relationship between a source account in a cloud environment and a scanner account.

[00108] Aspects of this disclosure may include using the established trust relationship, utilizing at least one cloud provider API to identify workloads in the source account. A cloud provider API, as used herein, may refer to any application program interface that allows an end user to interact with a cloud provider's service. A cloud provider, may include any entity through whom information is accessible over the Internet or other shared network. Workloads, as used herein, may refer to systems, devices, or resources in a network (or available via a network) such as a cloud infrastructure. By way of example, Fig. 1 illustrates examples of a workload: virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115. In some embodiments, scanning system 101 of Fig. 1 may perform a scanning operation on workload (e.g., systems, devices, resources, etc.) in cloud infrastructure 106.

[00109] In some embodiments, scanning system 101 of Fig. 1 may use an API to detect workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115 in the source account.

[00110] Aspects of this disclosure may involve using the at least one cloud provider API to query a geographical location of at least one of the identified workloads. Querying, as used herein, may refer to a request for data or information from a resource. A geographical location, as used herein, may refer to the data center where this workload is served, a position on Earth, a location in a network, an address of a server or other hardware, or any other information capable of identifying a locus, region, or position. Internet geolocation involves software capable

of deducing a geographic position of a device connected to the Internet. For example, the device's IP address may be used to determine the country, city, or ZIP code, determining its geographical location. By way of example, scanning system 101 of Fig. 1 may query any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 for their geographical location using a cloud provider API.

[00111] In some embodiments, the geographic location may include an identifier of a physical site. An identifier, as used herein, may refer to anything that allows for recognition of an object, location, or resource. A physical site, as used herein, may refer to an actual location. By way of example, scanning system 101 of Fig. 1 may query any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 for an identifier of their physical site (e.g., address, postal code, or any other physical descriptor) using a cloud provider API.

[00112] In some embodiments, scanning system 101 of Fig. 1 may use an out-of-band process to reach cloud workloads through a runtime storage layer, in combination with information gathered from cloud provider APIs.

[00113] Aspects of this disclosure may include receive an identification of the geographic location. An identification of the geographic location, as used herein, may refer to a recognition, association, or classification of the geographic location. By way of example, scanning system 101 of Fig. 1 may receive a geographical location, after querying for it, by interrogating one or more workloads, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, or keystores 113A-113D of cloud infrastructure 106. For example, the scanning system 101 of Fig. 1 may receive address, postal code, or any other physical descriptor of any workloads in cloud infrastructure 106. In some embodiments, scanning system 101 of Fig. 1 may receive a listing locating the workloads of cloud infrastructure 106 on a two-dimensional graphical representation.

[00114] In some embodiments, the identification of the geographic location may include an identification of a data center, at least one of a data center name, an Internet Protocol (IP) address, a name of the cloud provider, or a unique identity. A data center, as used herein, may refer to a facility that centralizes an organization's shared IT operations and equipment for the purposes of storing, processing, and/or disseminating data and applications. A data center, for example, may include a dedicated space within a building, or a group of buildings used to house computer systems and associated components, such as telecommunications and storage systems. An identification of a data center, as used herein, may refer to a recognition, association, or classification of a data center. A data center name, as used herein, may refer to a series of letters or numbers, a word, or set of words or numbers, an address, or a code identifying the data center. An Internet Protocol (IP) address, as used herein, may refer to a

unique string of characters that identifies a computer using the Internet Protocol to communicate over a network. A cloud provider, as used herein, may refer to a company that offers any components of cloud computing (e.g., infrastructure as a service (IaaS), software as a service (SaaS) or platform as a service (PaaS)). A unique identity, as used herein, may refer to a distinctive numeric or alphanumeric string that is associated with a single entity within a given system.

[00115] By way of example, scanning system 101 of Fig. 1 may query for and receive a geographical location such as a data center name/city and state, data center address, IP address / specific cloud identifier of the workload, name of cloud provider (e.g., Amazon Web Services), or a specific identifier of information of any of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00116] In some embodiments, scanning system 101 of Fig. 1 may receive a listing indicating locations of the workloads of cloud infrastructure 106 on a two-dimensional graphical representation.

[00117] In some other embodiments, scanning system 101 may receive additional information (in addition to geographic location information) such as whether the workload is Internet-facing and easily accessible to attackers, or if it is private (and may be less critical).

[00118] Aspects of this disclosure may include using the cloud provider APIs to access block storage volumes of the at least one workload. Block storage volumes, as used herein, may refer to data files maintained on Storage Area Networks (SANs) or in cloud-based storage environments. Each storage volume may act as an individual hard drive configured by a storage administrator. Accessing block storage volumes, as used herein, may refer to obtaining, examining, or retrieving data.

[00119] As described with relation to process 210 of Fig. 2B (beginning with step 211 to initiate a connection to cloud infrastructure 106), scanning system 101 may send a second message to user device 102, instructing the user to generate a role. A role may refer to a type associated with a node that may be assigned or defined. The second message may include instructions for the user to follow to generate the role. In step 213, a user may provide (e.g., via a keyboard at user device 102) a role definition to the cloud service provider's system. In some embodiments, the role definition includes read-only permissions and permissions to read a block storage layer (containing block storage volumes). In some embodiments, scanning system 101 provides a role formation template (e.g., an Amazon Web Services CloudFormation Template) for use with cloud infrastructure 106 to create the necessary role. In step 213, the user may utilize user device 102, for example, by copying and pasting a URL of the template, downloading and uploading the template to the cloud service provider's system, or selecting the template from a list of templates.

[00120] By way of example, scanning system 101 of Fig. 1 may access the block storage volume of any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00121] Aspects of this disclosure may include determining a file-system of the at least one workload. A file-system, as used herein, may refer to a method and/or a data structure that the operating system uses to control how data is stored and retrieved.

[00122] By way of example, scanning system 101 of Fig. 1 may identify the type of file-system of any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106. Some file systems may include NTFS, FAT32, ext3, exFAT, HFS, HFS+, HPFC, UFS, ZFS.

[00123] Aspects of this disclosure may include mounting the block storage volumes on a scanner based on the determined file-system. Mounting the block storage volumes, as used herein, may refer to a process by which the operating system makes files and directories (such as block storage volumes) on a storage device available for users to access via the computer's file system. A scanner, as used herein, may refer to a device for examining, reading, or monitoring electronically accessible information. By way of example, Fig. 1 illustrates one example of a scanner, scanning system 101 of Fig. 1. In some embodiments, mounting may include selecting a driver corresponding to the determined file system. In some embodiments, to mount the block storage volumes on the scanner, may include creating a snapshot of the block storage volumes; and mounting the snapshot of the block storage volumes on the scanner. A snapshot of the block storage volumes, as used herein, may refer to a current state or earlier state of the block storage volumes.

[00124] In some embodiments, the at least one processor may be further configured to encrypt the snapshot of the block storage volumes and mount the encrypted snapshot of the block storage volumes on the scanner. Encrypting a snapshot, as used herein, may refer to a process of encoding information (such as a snapshot). This process may convert the original representation of the information, known as plaintext, into an alternative form known as ciphertext.

[00125] In some embodiments, the scanner may use a privileged account to log in and determine how secure each host is from an inside vantage point. While authenticated scans can successfully discover potential vulnerabilities, they may be limited to the extent they require a privileged account on each scanned host. Furthermore, scans use significant system resources during the test procedures and require opening ports that by themselves pose a security risk.

[00126] Aspects of this disclosure may include activating a scanner at the geographic location. Activating a scanner, as used herein, may refer to starting up or initiating a device for examining, reading, or monitoring data. By way of example, scanning system 101 of Fig. 1 may

start up a scanner (a different scanner than scanning system 101) at the geographic location identified by scanning system 101.

[00127] Aspects of this disclosure may include reconstructing from the block storage volumes a state of the workload. Reconstructing, as used herein, may refer to rebuilding or reforming a state of the workload. A state of the workload, as used herein, may refer to a condition of a workload at a certain time. By way of example, scanning system 101 of Fig. 1 may rebuild the block storage volumes as an earlier version of any of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00128] In some embodiments, the reconstructed state of the workload may include at least two of an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content. An indication of an installed application, as used herein, may refer to an identifier of a software. A version of an installed application, as used herein, may refer to a particular form, variant, edition, or revision of a piece or pieces of software. An operating system configuration, as used herein, may refer to a manner in which components are arranged to make up an operating system or a computer system. An application configuration, as used herein, may refer to a manner in which components are arranged to make up an application or a computer system. A profile configuration, as used herein, may refer to a manner in which components are arranged to make up a profile or a computer system. A log, as used herein, may refer to a detailed list of an application information, system performance, or user activities. Database content, as used herein, may refer to any collection of data, or information, that is specially organized for rapid search and retrieval by a computer. In some embodiments, scanning system 101 of Fig. 1 may reconstruct any workload, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 from the block storage volumes to an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content.

[00129] Aspects of this disclosure may include assessing the reconstructed state of the workload to extract insights. Assessing the reconstructed state of the workload, as used herein, may refer viewing, utilizing, or evaluating the reconstructed state of the workload. Insights, as used herein, may refer to information or knowledge attained based on an analysis of some data. Extracting insights, as used herein, may refer to pulling information or data about, for example, a vulnerability associated with the workload or a composition of installed applications associated with the workload. By way of example, scanning system 101 of Fig. 1 may evaluate the reconstructed state of any of workloads, such as virtual machines 107A-107D, databases 109A-

109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 to pull valuable data and information.

[00130] In some embodiments, insights may include at least one of a vulnerability associated with the workload or a composition of installed applications associated with the workload. A vulnerability, as used herein, may refer to any weakness within an organization's information systems, internal controls, or system processes that can be exploited. A composition of installed applications associated with the workload, as used herein, may refer to a structure or group of accessible computer programs. By way of example, scanning system 101 of Fig. 1 may evaluate the reconstructed state of any workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 to pull valuable data and information regarding a vulnerability with the workload.

[00131] In some embodiments, a scanner may be deployed at the geographical location. Deploying a scanner, as used herein, may refer to installing, activating, utilizing, or bringing a scanner into effective action . By way of example, Fig. 1 illustrates one example of a scanner, scanning system 101 of Fig. 1 that is deployed.

[00132] In some embodiments, scanning system 101 of Fig. 1 may update the reconstructed state of the workload based on at least one change to the block storage volumes . Updating the reconstructed state of the workload, as used herein, may refer to overhauling or refurbishing the reconstructed state of the workload. A change to the block storage volumes, as used herein, may refer to any shift, alteration, or modification to the block storage volumes.

[00133] **Fig. 3** illustrates a block diagram of method 300 for insight extraction, consistent with disclosed embodiments. In some embodiments, the method may include ten (or more or less) steps:

[00134] Block 302: Establish a trusted relationship between a source account in a cloud environment and a scanner account. In some embodiments, establishing a trusted relationship between a source account in a cloud environment and a scanner account may include an integration process (creating a connection between an account on scanning system 101 and an account on cloud infrastructure 106) discussed with reference to Fig. 2A. By way of one example, scanning system 101 of Fig. 1 may create a trusted relationship between a source account in a cloud environment and a scanner account.

[00135] Block 304: Using the established trust relationship, utilize at least one cloud provider API to identify workloads in the source account. In some embodiments, scanning system 101 of Fig. 1 may use an API to detect workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115 in the source account.

[00136] Block 306: Use the at least one cloud provider API to query a geographical location of at least one of the identified workloads. In some embodiments, scanning system 101 of Fig. 1 may query any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 for their geographical location using a cloud provider API.

[00137] Block 308: Receive an identification of the geographic location. In some embodiments, scanning system 101 may indicate in the map additional information regarding the workload (e.g., a system or device). For example, scanning system 101 may provide: an identification of a data center (e.g., a name or number associated with the data center where the workload is located), at least one of a data center name (e.g., a name or address associated with the data center where the workload is located), Internet Protocol (IP) address, name of the cloud provider, or a unique identity (e.g., any other information related to the workload that may be used).

[00138] Block 310: Use the cloud provider APIs to access block storage volumes of the at least one workload. In some embodiments, scanning system 101 of Fig. 1 may access the block storage volume of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106. In some embodiments, accessing the block storage may include taking the mean of generating snapshots.

[00139] Block 312: Determine a file-system of the at least one workload. In some embodiments, scanning system 101 of Fig. 1 may identify the type of file-system any of workloads (virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106).

[00140] Block 314: Mount the block storage volumes on a scanner based on the determined file-system. In some embodiments, to mount the block storage volumes on the scanner, may include instructing the operating system to map a directory structure logically to a physical storage device. A storage volume may be mounted after it is attached and formatted for use by a server's operating system. Mounting may include creating a snapshot of the block storage..

[00141] Block 316: Activate a scanner at the geographic location. In some embodiments, scanning system 101 of Fig. 1 may start up a scanner (a different scanner than scanning system 101) at the geographic location identified by scanning system 101.

[00142] Block 318: Reconstruct from the block storage volumes a state of the workload. In some embodiments, scanning system 101 of Fig. 1 may rebuild the block storage volumes to an earlier version of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00143] Block 320: Assess the reconstructed state of the workload to extract insights. In some embodiments, scanning system 101 of Fig. 1 may evaluate the reconstructed state of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 to derive valuable data and information.

[00144] **Vulnerability Management Techniques**

[00145] **Fig. 4** depicts a cybersecurity system performing a side scanning function to protect against potential vulnerabilities. Referring to Fig. 4, a processor 400 configured to identify in a cloud environment one or more block storage volumes 401A-C communicatively connected to a cloud provider API where the block storage volumes 401A-C may be contained. When a block storage volume, e.g., 401A is identified, the processor 400 may be configured to perform the function of identifying the software installed 403A-C in or relating to block storage volume 401. Upon identifying installed software 403A-C, the processor may be further configured to identify the version of software 405A-C operating in the installed software 403A-C. Based on the installed software 403A-C and the version of the software 405A-C, the processor may be configured to identify a listing of known vulnerabilities 407A-C that may relate to the installed software 403A-C and the version installed 405A-C. When the listing of known vulnerabilities 407 is obtained, the system may list the known vulnerabilities 409A-C for that software version 411A-C, which can be communicated to the processor 400 of the end user. This information may be processed and displayed to an end user using processor 400.

[00146] **Fig. 5** of the disclosed embodiments describes a method of operating a cybersecurity system performing a side scanning function to protect against potential vulnerabilities. Referring to Fig. 5, a processor unit 500 may be configured to perform the method. When initiated either manually or automatically, processor unit 500 may be communicatively connected with a cloud provider API 501 to initiate a cybersecurity function as disclosed above. Contained digitally on a memory device and accessed by a cloud provider API 501 is a series of block storage volumes 503. Processor unit 500 may be capable of performing a vertical and horizontal scan for network access or security information of a block storage volume 503 to detect potential vulnerabilities to the block storage volume 503 or any processing unit 500 designed to access said block storage volume 503.

[00147] Based on the accessed block storage volume 500, processor unit 500 with the disclosed embodiments of a cybersecurity system may perform an identification of the type of software installed 505 connected with block storage volume 503. Following identification of the type of software installed 505, the processing unit may then identify an installed version 507 of said installed software 505 for a version comprised of a unique identifier based on a combination of letters, numbers, or similar unique identifiers.

[00148] Upon recognition of an installed software version 507 of the installed software 505, processing unit 500 with the disclosed embodiments of the disclosed scanning system 101

may provide a list of known vulnerabilities 509 for review by said scanning system 101 and/or its end user and maintainer of said block storage volumes 503. Based on the list of known vulnerabilities 509, the disclosed scanning system 101 or its end user may identify one or more ports of accessibility to said block storage volume 503 that may be accessed by a known, associated vulnerability 513 from listed vulnerabilities 509 to determine an avenue for potential vulnerability 513 to access and infiltrate block storage volume 503.

[00149] In some embodiments, a cyber security scanning system for a cloud environment 101 may include a processor 500 to operate said system. This processor 500 may include central processing units and other similar computer-enabling equipment for processing and executing commands based on the information inputted to said system. The processor 500 may be communicatively connected to a computer network or series of networks to accomplish said cyber security function.

[00150] As an example embodiment, a processor unit 500 may be configured to use a cloud-provider API that may communicate with one or more specified computer-readable media across a digital network. This can be accomplished through internet protocols, internet control message protocols, transmission control protocol, or user datagram protocol. Cloud provider API 501 may be one of several forms of middleware, interface, middle layer, or other systems of interfacing applications. A processor 300 may be one or more computer processing units, central processing unit, server, microcomputer, mainframe, and any other manifestation of digital computing.

[00151] Further to one of several possible embodiments, a cloud-provider API 501 may be configured to access a block-storage volume 503 of a workload maintained in a cloud-storage environment. This may be accomplished through a system of computer-readable media communicatively connected. Said block-storage volume 303 may be contained on a Storage Area Network (SAN) or similar cloud-based memory storage environment. The block storage volume 503 may be contained in smaller storage volumes with an associated identifier unique to that portion of said block storage volume 503. In some embodiments, the block-storage volume 503 of a workload may have multiple paths for the storage volume to be reaggregated and retrieved quickly.

[00152] Among several embodiments, a scanning system 101 may comprise a system for identifying an installed software application in the accessed block-storage volume 503. This identification of installed software may be accomplished by accessing installed software 505 files through signature verification, root license, or authorized user lists. The installed software 505 may be located and identified within applications such as file storage, database storage, and virtual machine file system volumes. The identification of said installed software application 507 may be processed, analyzed, and communicated to the scanning system 101 for

processing, cataloging, and protection through encryption and various methods of layered cyber defense.

[00153] Further, the scanning system 101 described herein may include functionality to analyze installed software applications to determine the associated software version. The software application version 507 may identify the software version based on unique version name, unique version number, and may be based on unique states of the currently installed computer software 505.

[00154] One of many embodiments disclosed above may include the scanning system 101 having the ability to access a data structure of known software vulnerabilities 509 for a plurality of versions of software applications. The known software vulnerabilities 509 may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software. The scanning system 101 may be able to access and identify software vulnerabilities for mitigation, rectification, correction, and fortification.

[00155] In one embodiment, a cybersecurity system may also perform scanning according to scanning system 101 by performing a lookup of the identified installed software version 507 in the data structure to identify known vulnerabilities 509. This function can be performed by the scanning system 101 according to Fig. 1 performing a query of the installed software 505 for unique version number or designator and comparing to, amongst many things, a set of likely or potential vulnerabilities to that software version for potential deficiencies or cybersecurity threats known or suspected to similar software types and versions. This query may be performed according to a predetermined set of values, to include previously identified unique version numbers or designators that may contain the known list of previously identified vulnerabilities.

[00156] Among many embodiments, embodiments of the disclosed scanning system 101 may query the cloud provider API 501 to determine network accessibility information 511 related to the workload. In order to accomplish this query of the cloud provider API 301, the scanning system 101 may involve index of search results and display of said search results, followed by processing and grouping search results. Network accessibility information 511 may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, or Hadoop nodes, among others.

[00157] If the system detects a vulnerable application 513, one embodiment may identify one or more port on which said vulnerable application is accessible. In one of several embodiments, the scanning system 101 may detect a vulnerable application in one or more computation process. In another embodiment, the cybersecurity system may perform a network accessibility query in a separate process. Further, a disclosed embodiment may perform these separate functions in subsequent and sequential steps of the same process. A person having skill would understand that an authorized user or an authorized scanning system 101 can perform these functions concurrently and subsequently by an authorized user or an authorized cybersecurity system while performing the same function as the disclosed embodiment.

[00158] Upon gathering network accessibility information and the identified port to identify one or more vulnerabilities susceptible to attack from outside the workload, a disclosed embodiment would have the functionality to perform processes to gather, display, and mitigate a discovered vulnerability in order to minimize the likelihood and effectiveness of a cyber threat outside of and attempting to access a workload through a known or previously encountered type of cyber threat. This functionality may include collecting and organizing the vulnerabilities according to type or category of vulnerability, displaying the gathered data for an end user or maintainer, and implementing security features automatically or manually by a user or maintainer such as security patches, password or passcode changes and suggestions for users to do the same, and malicious code eradication.

[00159] As one of several possible embodiments, the scanning system 101 may also, upon identification of one or more vulnerabilities, implement remedial actions via one or more processors. Remedial actions may include, among other things, notification to an end user of an identified threat, compensation through a revised security code to mitigate the potential threat, publication of the identified threat and vulnerability in a log or record of detected vulnerabilities, and communication of the sensed vulnerability and threat to a server operator or maintainer to fortify the protections of workloads existing on similar environments.

[00160] In a similar embodiment to above, the remedial measure may include transmission of an alert to a device associated with an administrator. The alert may be, amongst others, written, auditory, and visual for processing and use by an administrator of said scanning system 101. Said administrator may take action based on the received alert, to include eliminating the cyber threat through mitigation measures, change in cybersecurity posture, or removing the workload from the cybersecurity threat environment.

[00161] In another embodiment, a query of the cloud provider API 501 to determine network accessibility information related to the workload may be performed by at least one processor 500 configured to examine data sources associated with the workload. The data examined may include user data, system processing data, accessibility data, clock cycles, storage input/output, or similar data processors. A query of the cloud provider API 501 may be

automated or manually-initiated. Based on the said query, the network accessibility information 511 related to the workload may change based on the data sources associated with said workload.

[00162] In the above processor and similar embodiments, further configuration may include a process to determine network accessibility information 511 based on the examined data sources. The examined data sources may include various cloud-based workloads, internet protocols, transmission communication protocols, or other methods and systems of memory and data storage.

[00163] As an exemplary embodiment, network accessibility information 511 includes at least one of: data from an external data source, cloud provider information, or at least one network capture log. These embodiments of an external data source may include data from the operating environment of the cloud-based environment, an external operating system for a computer processing unit 500, or other similar computer readable media. Cloud provider information may further include information that may identify the network accessibility information 511 vertically or horizontally to fully describe the associated workload. Network capture logs may be automated or manually updated to include possible vulnerabilities 513 and threats to the cloud-based storage medium.

[00164] Further, the disclosed embodiments may include an installed software application 505, with the at least one processor 500 configured to extract data from at least one of operating system packages, libraries, or program language libraries. This data may be extracted through a system query, random access algorithm, or similar automated process. Operating system packages may include systems operable on Microsoft, Apple, Linux, and similar operating systems. Libraries may consist of a series of files, folders, and databases of information stored on one of any indexed data repositories. A program language library may contain several of an exemplary program languages including but not limited to Javascript, Swift, Scala, Go, Python, Elm, Ruby, C#, C++ and other similar sources of software code.

[00165] As another exemplary embodiment, a scanning system 101 may also include a processor 500 configured to identify installed software application 505 based on the extracted data. The processor 500 may perform only this function or this function among many to accomplish the layered cybersecurity defense described herein this disclosed cloud-based security environment. The identification of installed software by said processor may include identifying the software by unique version number or designator, recognized source code, metadata associated with the installed software application files, or similar software-identifying information.

[00166] One embodiment may include an additional function wherein the installed software application 505 that has been identified includes one or more scripts. These scripts

may be processed through various computer readable languages to include Javascript, C#, C++ and other forms of computer code.

[00167] One embodiment contemplated by the disclosed scanning system 101 may also include data structure includes aggregated vulnerability data 309. This aggregated vulnerability data 509 may be compiled by an end user or maintainer from within the cloud-based environment of the current data structure as well as compilation from similar data sets and aggregation of common threats to data structures likely to experience similar vulnerabilities. This aggregation of vulnerability data 509 may be contained within the data structure and it may be collectively aggregated to provide for a more robust and layered cybersecurity defense posture.

[00168] In an embodiment of the disclosed scanning system 101, the aggregated vulnerability data 509 may include data from one or more third-party vendors. These vendors may include operators of the cloud-based server environment, providers of networking and internet communication, methods of layered authentication, and other similar providers of services directly related and in communication with the cloud-based cybersecurity system.

[00169] As an additional exemplary embodiment, the aggregated vulnerability data 309 may include data collected by a scanner. This scanner may involve use of continuous or periodic monitoring of the workload. The scanner may perform security screenings of the various workloads vertically or horizontally to identify network identification information, port accessibility, and associated vulnerabilities. Any scan performed may be communicated to the scanning system 101 that may be responsible for performing and logging the results of the scan and may be able to initiate follow-on processes and protocols to protect the data contained in the workload that is the subject of scanning.

[00170] An embodiment of the disclosed scanning system 101 may also include aggregated vulnerability data 509 that may include at least one of an advisory, an exploit, a security announcement, or a known bug. An advisory may include notification to a system maintainer or user of the potential vulnerability, may log notice of the advisory, and may recommend possible user or maintainer actions to potentially address said advisory. An exploit may further consist of an automated system response designed to take advantage of the sensed vulnerability data. The exploit can be further reflected in the aggregated vulnerability data and protocols can be written into the cybersecurity infrastructure to prevent said exploit from gaining access and permissions to unauthorized areas of the workload storage environment.

[00171] A person having ordinary skill in the art would appreciate the above described embodiments are among many potential embodiments, to include a method of operating a scanning system 101 similar to the one described above. A disclosed embodiment contemplates this method to be accomplished through manual user operation, automated

computer processes, or similar manners of operation. These manners of operation and those contemplated similar to them would allow the scanning system 101 described and disclosed to execute its operations as the system above describes.

[00172] One of several embodiments of the disclosed method of operating a scanning system 101 may include a cyber security scanning system for a cloud environment 101 may include a processor 500 to operate said system. This processor may include central processing units and other similar computer-enabling equipment for processing and executing commands based on the information inputted to said system. The processor may be communicatively connected to a computer network or series of networks to accomplish said cyber security function.

[00173] As an example embodiment, a processor unit 500 may be configured to use a cloud-provider API 501 that may communicate with one or more specified computer-readable media across a digital network. This can be accomplished through internet protocols, internet control message protocols, transmission control protocol, or user datagram protocol. Cloud provider API 501 may be one of several forms of middleware, interface, middle layer, or other systems of interfacing applications. A processor may be one or more computer processing units, central processing unit, server, microcomputer, mainframe, and any other manifestation of digital computing.

[00174] Further to one of several possible embodiments, a cloud-provider API 501 may be configured to access a block-storage volume 503 of a workload maintained in a cloud-storage environment. This may be accomplished through a system of computer-readable media communicatively connected. Said block-storage volume 503 may be contained on a Storage Area Network (SAN) or similar cloud-based memory storage environment. The block storage volume 503 may be contained in smaller storage volumes with an associated identifier unique to that portion of said block storage volume 503. In some embodiments, the block-storage volume 503 of a workload may have multiple paths for the storage volume to be reaggregated and retrieved quickly.

[00175] Among several embodiments, a method of operating a scanning system 101 may comprise a system for identifying an installed software application 505 in the accessed block-storage volume 503. This identification of installed software 505 may be accomplished by accessing installed software files through signature verification, root license, or authorized user lists. The installed software 505 may be located and identified within applications such as file storage, database storage, and virtual machine file system volumes. The identification of said installed software application 505 may be processed, analyzed, and communicated to the method of operating a scanning system 101 for processing, cataloging, and protection through encryption and various methods of layered cyber defense.

[00176] Further, a method of operating a scanning system 101 described herein may include functionality to analyze installed software applications 505 to determine the associated software version 507. The software application version 507 may identify the software version based on unique version name, unique version number, and may be based on unique states of the currently installed computer software.

[00177] One of many embodiments disclosed above may include the method of operating a scanning system 101 having the ability to access a data structure of known software vulnerabilities 509 for a plurality of versions of software applications 507. The known software vulnerabilities 509 may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software. The cyber security system may be able to access and identify software vulnerabilities 309 for mitigation, rectification, correction, and fortification.

[00178] In one embodiment, a method of operating a scanning system 101 may also perform scanning according to scanning system 101 by performing a lookup of the identified installed software version 507 in the data structure to identify known vulnerabilities 309. This function can be performed by the scanning system 101 according to Fig. 1 performing a query of the installed software 505 for unique version number 507 or designator and comparing to, amongst many things, a set of likely or potential vulnerabilities to that software version for potential deficiencies or cybersecurity threats known or suspected to similar software types and versions. This query may be performed according to a predetermined set of values, to include previously identified unique version numbers or designators that may contain the known list of previously identified vulnerabilities.

[00179] Among many embodiments, embodiments of the disclosed method of operating a scanning system 101 may query the cloud provider API 501 to determine network accessibility information 511 related to the workload. In order to accomplish this query of the cloud provider API 501, the scanning system 101 may involve index of search results and display of said search results, followed by processing and grouping search results. Network accessibility information 511 may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, or Hadoop nodes, among others.

[00180] If the system detects a vulnerable application, one embodiment may identify one or more port on which said vulnerable application is accessible. In one of several embodiments, the scanning system 101 may detect a vulnerable application in one or more computation

process. In another embodiment, the scanning system 101 may perform a network accessibility query in a separate process. Further, a disclosed embodiment may perform these separate functions in subsequent and sequential steps of the same process. A person having skill would understand that an authorized user or an authorized cybersecurity system may perform these functions concurrently and subsequently while performing the same function as the disclosed embodiment.

[00181] Upon gathering network accessibility information 511 and the identified port 513 to identify one or more vulnerabilities susceptible to attack from outside the workload, a disclosed embodiment would have the functionality to perform processes to gather, display, and mitigate a discovered vulnerability in order to minimize the likelihood and effectiveness of a cyber threat outside of and attempting to access a workload through a known or previously encountered type of cyber threat. This functionality may include collecting and organizing the vulnerabilities according to type or category of vulnerability, displaying the gathered data for an end user or maintainer, and implementing security features automatically or manually by a user or maintainer such as security patches, password or passcode changes and suggestions for users to do the same, and malicious code eradication.

[00182] A person having ordinary skill in the art would similarly understand that the above disclosed system can be disclosed using a suitable non-transitory computer-readable medium performing each of the disclosed functions. The disclosed embodiment may include a non-transitory computer readable medium with a scanning system 101 similar to the one described above on a non-transitory computer-readable medium. A disclosed embodiment contemplates this system to be accomplished through a medium that may contain a central processing unit, virtual machine, or a similar non-transitory medium. These manners of operation and those contemplated similar to them would allow the scanning system 101 described and disclosed to execute its operations as the system above describes.

[00183] One of several embodiments of the disclosed non-transitory computer readable medium with a scanning system 101 may include a cyber security scanning system for a cloud environment 101 may include a processor 500 to operate said system. This processor 500 may include central processing units and other similar computer-enabling equipment for processing and executing commands based on the information inputted to said system. The processor 500 may be communicatively connected to a computer network or series of networks to accomplish said cyber security function.

[00184] As an example embodiment, a processor unit 300 may be configured to use a cloud-provider API 501 that may communicate with one or more specified computer-readable media across a digital network. This can be accomplished through internet protocols, internet control message protocols, transmission control protocol, or user datagram protocol. Cloud provider API 501 may be one of several forms of middleware, interface, middle layer, or other

systems of interfacing applications. A processor 500 may be one or more computer processing units, central processing unit, server, microcomputer, mainframe, and any other manifestation of digital computing.

[00185] Further to one of several possible embodiments, a cloud-provider API 501 may be configured to access a block-storage volume 503 of a workload maintained in a cloud-storage environment. This may be accomplished through a system of computer-readable media communicatively connected. Said block-storage volume 503 may be contained on a Storage Area Network (SAN) or similar cloud-based memory storage environment. The block storage volume 503 may be contained in smaller storage volumes with an associated identifier unique to that portion of said block storage volume 503. In some embodiments, the block-storage volume 503 of a workload may have multiple paths for the storage volume to be reaggregated and retrieved quickly.

[00186] Among several embodiments, a non-transitory computer readable medium with a scanning system 101 may comprise a system for identifying an installed software application 505 in the accessed block-storage volume 503. This identification of installed software 505 may be accomplished by accessing installed software files through signature verification, root license, or authorized user lists. The installed software 505 may be located and identified within applications such as file storage, database storage, and virtual machine file system volumes. The identification of said installed software application 505 may be processed, analyzed, and communicated to the method of operating a scanning system 101 for processing, cataloging, and protection through encryption and various methods of layered cyber defense.

[00187] Further, a non-transitory computer readable medium with a scanning system 101 described herein may include functionality to analyze installed software applications 505 to determine the associated software version 507. The software application version 507 may identify the software version based on unique version name, unique version number, and may be based on unique states of the currently installed computer software.

[00188] One of many embodiments disclosed above may include the non-transitory computer readable medium with a scanning system 101 having the ability to access a data structure of known software vulnerabilities 509 for a plurality of versions of software applications 507. The known software vulnerabilities 509 may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software. The scanning system 101 may be able to access and identify software vulnerabilities for mitigation, rectification, correction, and fortification.

[00189] In one embodiment, a non-transitory computer readable medium with a scanning system 101 may also perform scanning according to scanning system 101 by performing a lookup of the identified installed software version in the data structure to identify known vulnerabilities 509. This function can be performed by the scanning system 101 according to Fig. 1 performing a query of the installed software for unique version number 307or designator and comparing to, amongst many things, a set of likely or potential vulnerabilities 309 to that software version 307 for potential deficiencies or cybersecurity threats known or suspected to similar software types and versions. This query may be performed according to a predetermined set of values, to include previously identified unique version numbers or designators that may contain the known list of previously identified vulnerabilities 509.

[00190] Among many embodiments, embodiments of the disclosed non-transitory computer readable medium with a scanning system 101 may query the cloud provider API 501 to determine network accessibility information 511 related to the workload. In order to accomplish this query of the cloud provider API 501, the scanning system 101 may involve index of search results and display of said search results, followed by processing and grouping search results. Network accessibility information 511 may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, or Hadoop nodes, among others.

[00191] If the system detects a vulnerable application, one embodiment may identify one or more ports 515 on which said vulnerable application is accessible. In one of several embodiments, the scanning system 101 may detect a vulnerable application in one or more computation processes. In another embodiment, the scanning system 101 may perform a network accessibility query in a separate process. Further, a disclosed embodiment may perform these separate functions in subsequent and sequential steps of the same process. A person having skill would understand that an authorized user or an authorized cybersecurity system can perform these functions concurrently and subsequently while performing the same function as the disclosed embodiment.

[00192] Upon gathering network accessibility information 511 and the identified port 515 to identify one or more vulnerabilities susceptible to attack from outside the workload, a disclosed embodiment would have the functionality to perform processes to gather, display, and mitigate a discovered vulnerability in order to minimize the likelihood and effectiveness of a cyber threat outside of and attempting to access a workload through a known or previously encountered type of cyber threat. This functionality may include collecting and organizing the vulnerabilities according to type or category of vulnerability, displaying the gathered data for an end user or maintainer, and implementing security features automatically or manually by a user

or maintainer such as security patches, password or passcode changes and suggestions for users to do the same, and malicious code eradication.

[00193] Another disclosed embodiment may include a method of using a cloud provider API 501, accessing block storage volume 503 of a workload maintained in a cloud storage environment. The method may include manual user operation, automated system operation, systematic and random operating parameters for the system to provide its cybersecurity and similar security functions. The accessed block storage volume may be contained as a collection of block units organized together, may be a series of individual blocks that can be reorganized to form a new storage volume, and may have the ability to be disaggregated and reaggregated as necessary to accomplish its storage and cybersecurity functions.

[00194] The disclosed method may further comprise a system that may analyze the identified installed software application 505 to determine an associated software version 507. In analyzing an installed software application 305, the disclosed embodiment may include a method of querying the installed software application to access a software version 507 that may include a series of letters, numbers, and other identifying characters to differentiate the version of software currently operative and its associated identifying characteristics and capabilities.

[00195] A method of the disclosed may also include accessing a data structure of known software vulnerabilities 509 for a plurality of versions of software applications 507. The disclosed method may determine a data structure of known software vulnerabilities 509 of one among many operative versions of software installed in the monitored storage system as compared to historical data of similar storage systems.

[00196] The disclosed method as described herein may further include performing a lookup of the identified installed software version 507 in the data structure to identify known vulnerabilities 509. The lookup of identified installed software 507 may be previously indexed based on unique version identifier, and the indexed information may include known vulnerability information 509, network accessibility information 511, network protocols, and administrator identifier information. Based on the said indexed known vulnerabilities 509, the scanning system 101 disclosed may take actions to mitigate a cybersecurity threat and reinforce cybersecurity defenses at the type of cybersecurity threat, as previously discussed.

[00197] Further to the above, a disclosed method may also include querying the cloud provider API 501 to determine network accessibility information 511 related to the workload. A query of the cloud provider API 501 to determine network accessibility information 511 related to the workload may be automatically or manually initiated. Network accessibility information 511 similar to the above embodiment may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, and Hadoop nodes, among others.

[00198] A disclosed method of one embodiment also may provide a method of identifying at least one port 515 on which the vulnerable application is accessible. The identification of one port 515 on which the vulnerable application is accessible may be for the purpose of identifying a cyber weakness, and it may include the ability to change the port currently accessible to mitigate the said cyber weakness.

[00199] Yet another disclosed embodiment of a disclosed method may include using the network accessibility information 511 and the identified at least one port 515 to identify one or more vulnerabilities 513 susceptible to attack from outside the workload. The vulnerability 513 identified as susceptible to attack from outside the workload may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software.

[00200] A method of the disclosed embodiment may further include implementing a remedial action in response to the identified one or more vulnerabilities 513. Said remedial action may include, among other things, notification to an end user of an identified threat, compensation through a cybersecurity patch, publish of the identified threat and vulnerability in a log or record of detected vulnerabilities, and communication of the sensed vulnerability and threat to a server operator and maintainer to fortify the protections of workloads existing on similar environments.

[00201] The disclosed method may also include wherein the remedial measure includes transmitting an alert to a device associated with an administrator. Said alert may be, amongst others, written, auditory, and visual for processing and use by an administrator of said cybersecurity system. Said alert may further be logged and catalogued for future identification of known threats and vulnerabilities to similar software application versions.

[00202] A disclosed method of the present embodiment may also provide querying the cloud provider API 501 to determine the network accessibility information 311 related to the workload and examining data sources associated with the workload. A query of the cloud provider API 501 to determine network accessibility information 511 related to the workload may be automatically or manually initiated. Network accessibility information 511 similar to the above embodiment may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, and Hadoop nodes, among others.

[00203] The method of the above disclosed embodiment may further determine network accessibility information 511 by determining the network accessibility information 511 based on

the examined data sources. The data examined may include user data, system processing data, accessibility data, clock cycles, storage input/output, and similar processes. A query of the cloud provider API 501 may be automated or manually-initiated. Based on the said query, the network accessibility information 511 related to the workload may change based on the data sources associated with said workload.

[00204] In the method of one disclosed embodiment, network accessibility information 511 may include at least one of data from an external data source, cloud provider information, or at least one network capture log. These embodiments may include data from the operating environment of the cloud-based environment, an external operating system for a computer processing unit, or other similar computer readable media. Network capture logs may be automated or manually updated to include possible vulnerabilities and threats to the cloud-based storage medium.

[00205] A method of the disclosed embodiment may also include identifying the installed software application 507 by extracting data from at least one of OS packages, libraries, or program language libraries. This data may be extracted through a system query, random access algorithm, or similar automated process. Operating system packages may include systems operable on Microsoft, Apple, Linux, and similar operating systems. Libraries may consist of a series of files, folders, and databases of information stored on one of any indexed data repositories. A program language library may contain several of an exemplary program languages including but not limited to Javascript, Swift, Scala, Go, Python, Elm, Ruby, C#, C++ and other similar sources of software code.

[00206] Another method of the disclosed embodiment may identify the installed software application 507 based on the extracted data. The method of the processor 500 may perform only this function or this function among many to accomplish the layered cybersecurity defense described herein this disclosed cloud-based security environment.

[00207] A further disclosed method of the present embodiment may include at least one processor 500 further configured to identify a version 507 of the installed software application 505. The method of the processor may perform only this function or this function among many to accomplish the layered cybersecurity defense described herein this disclosed cloud-based security environment.

[00208] **Prioritization Techniques**

[00209] Fig. 6 is a block diagram of method 600 for risk prioritization, consistent with disclosed embodiments.

[00210] Aspects of this disclosure may include a cloud-based cybersecurity system for assessing internet exposure of a cloud-based workload with embodiments configured to access at least one cloud provider API to determine a plurality of entities capable of routing and/or filtering traffic in a virtual cloud environment associated with a target account containing the

workload. Routing traffic, as used herein, may refer to the process of selecting a path for traffic (e.g., cells, blocks, frames, packets, calls, messages, or other units of data) in a network or between or across multiple networks. The cloud-based cybersecurity system, in some embodiments, may comprise at least one processor configured to assess internet exposure as described herein.

[00211] A cloud provider API, as used herein, may refer to any application program interface that allows the end user to interact with a cloud provider's service. Workloads, as used herein, may refer to devices in a network. By way of example, workloads may include systems, devices, resources in a cloud infrastructure. By way of example, Fig. 1 illustrates examples of a workload, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, or load balancer 115. In some embodiments, scanning system 101 of Fig. 1 may perform a scanning operation on any workload (e.g., systems, devices, resources, etc.) in cloud infrastructure 106.

[00212] In some embodiments, scanning system 101 of Fig. 1 may use an API to detect workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115 in the source account. In some embodiments, workloads may exist as virtual machines, while in other embodiments, workloads may exist as discrete, physical devices. For example, in some embodiments, a layer 3 routing system could be implemented as a physical device (e.g., a router or switch) or a virtual device (e.g., a routing system instantiated as a virtual device on a computer).

[00213] Scanning system 101 may query devices and systems capable of routing and filtering traffic (e.g., load balancer 115, routers, switches, firewalls, security groups, API Gateways and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations. These devices can be cloud native and / or unmanaged (e.g. NGINX proxy)

[00214] A cloud environment, as used herein, may refer to a platform implemented on, hosted on, and/or accessing servers that are accessed over the Internet. A scanner account, as used herein, may refer to any type of account associated with a scanner. A scanner, as used herein, may refer to a device for examining, reading, or monitoring something. By way of example, Fig. 1 illustrates one example of a scanner, scanning system 101 of Fig. 1. A target account containing the workload, as used herein, may refer to a location on a network server selected as an aim of connection, for example. In some embodiments, scanning system 101 of Fig. 1 may determine entities capable of routing traffic.

[00215] In some embodiments, the plurality of entities includes a virtual network appliance. A virtual network appliance, as used herein, may refer to a machine focused on virtualizing network functionality. In some embodiments, a typical network virtual appliance may

include functionality directed to various layer four to seven. That is, such appliances may implement functionality associated with OSI model layers 4-7 (Transport, Session, Presentation, and Application). Such functionality may include, for example, a firewall, WAN optimizer, application delivery controllers, routers, load balancers, IDS/IPS, proxies, or SD-WAN edge device. In some embodiments, scanning system 101 of Fig. 1 may determine a virtual network appliance capable of routing traffic.

[00216] In some embodiments, the virtual network appliance is at least one of a load balancer, a firewall, a proxy, or a router. A load balancer, as used herein, may refer to any device that acts as a reverse proxy and distributes network or application traffic across a number of servers. In some embodiments, load balancer 115 may comprise one or more systems that balance incoming requests between the different systems and devices of cloud infrastructure 106. For example, load balancer 115 may be configured to determine usage (e.g., processor load, used storage capacity) of systems or devices in cloud infrastructure 106 to assist in determining where to route an incoming request from network 105 to store data, perform processing, or retrieve data.

[00217] Load balancer 115 may be configured to receive an incoming request from user device 102. Upon receipt of the request, load balancer 115 may consult a data store (part of or separate from load balancer 115; not pictured) to determine usage or forecasted usage of various systems or devices in cloud infrastructure 106, and may forward the request to the systems or devices having the lowest usage or forecasted usage. A firewall, as used herein, may refer to a device or system that monitors incoming and outgoing network traffic and permits or blocks data packets based on a set of security rules. A proxy, as used herein, may refer to a device or system that translates traffic between networks or protocols. In some embodiments, a proxy may be implemented as an intermediary a device or system separating end-user devices from destinations devices (e.g., web servers) by routing traffic through them. Proxies (or proxy servers) may be configured to provide varying levels of functionality, security, and privacy depending on use case, needs, or company policy. A router, as used herein, may refer to a device or system that connects a local network to the internet.

[00218] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to query the at least one cloud provider API to determine at least one networking configuration of the entities. Querying, as used herein, may refer to a request for data or information from a resource. A cloud provider API, as used herein, may refer to any application program interface that allows an end user to interact with a cloud provider's service. A cloud provider may include any entity through whom information is accessible over the Internet or other shared network. At least one networking configuration of the entities, as used herein, may refer to a manner in which components are arranged to make up an operating

system, networking system, or a computer system. In some embodiments, scanning system 101 of Fig. 1 may send a request to determine at least one networking configuration of the entities.

[00219] In some embodiments, the networking configuration is at least one of a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration.

[00220] A routing configuration, as used herein, may refer to a manner in which components are arranged to make up a routing system. For example, a routing configuration may provide for dynamic or static "routes" between devices on a network or on networks.

[00221] A proxy configuration, as used herein, may refer to a manner in which components are arranged to make up a proxy system. For example, a proxy configuration may provide for a configuration of how a client can route traffic through a central point (e.g., a proxy server), in order to relay multiple clients' requests from a single point on a network.

[00222] A load balancing configuration, as used herein, may refer to a manner in which components are arranged to make up a load balancing system. For example, a loan balancing configuration may define how resources such as uplinks or distributed applications may be utilized by devices on the same (or another) network, in order to balance usage of multiple redundant uplinks or distributed application nodes.

[00223] A firewall configuration, as used herein, may refer to a manner in which components are arranged to make up a firewall system. For example, a firewall configuration may comprise rules that determine what traffic may be forwarded, which devices are permitted to communicate with one another, or what protocols may be used for communication between devices.

[00224] A VPN (virtual private network) configuration, as used herein, may refer to a manner in which components are arranged to make up a VPN system. A VPN may refer to a type of network that extends a private network across a public network and enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network. For example, a VPN configuration may provide rules or other information by which data is sent or received over a single, encrypted route, such that intermediate points cannot determine the content, source, or destination of the data.

[00225] In some embodiments, scanning system 101 of Fig. 1 may send a request to determine at least a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration of the entities.

[00226] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to build a graph connecting the plurality of entities based on the networking configuration. A graph connecting the plurality of entities, as used herein, may refer to a diagram showing the relation between variable quantities (e.g., entities). In some embodiments, the graph may include any data structures that includes the connections from

one entity to another. In some embodiments, scanning system 101 of Fig. 1 may construct a graph or diagram linking the plurality of entities based on the networking configuration. In some embodiments, graph includes a data structure sequentially connecting entities. A data structure sequentially connecting entities, as used herein, may refer to an organization, management, and/or storage format that enables efficient access and modification of data such as entities.

[00227] In some embodiments, a graph may include directional vectors indicating directions of dataflow. Directional vectors, as used herein, may refer to a quantity having direction as well as magnitude (indicating, for example, what kinds of traffic are allowed to flow, what ports are open, what protocols are allowed), especially as determining the position of one point in space relative to another. Directions of dataflow, as used herein, may refer to the movement of data through a system comprised of software, hardware or a combination of both along a particular course. In some embodiments, scanning system 101 of Fig. 1 may construct a graph including a data structure connecting entities sequentially or a graph including directional vectors showing the direction of data flow.

[00228] In some embodiments, to build the graph, the at least one processor may be configured to identify individual entities as nodes and connect the nodes. Nodes, as used herein, may refer to a device in a network (e.g., devices in cloud infrastructure 100). Connecting nodes, as used herein, may refer to linking or bonding devices or data points in a larger network. In some embodiments, the graph includes port numbers associated with the workload. Port numbers, as used herein, may refer to a way to identify a specific process to which an internet or other network message is to be forwarded when it arrives at a server and/or which services are running on a server. In some embodiments, the graph includes a path from the at least one source to the workload. A path from the at least one source to the workload, as used herein, may refer to a string of characters used to uniquely identify a location in a directory structure.

[00229] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to access a data structure identifying services publicly accessible via the Internet and capable of serving as an internet proxy. Services capable of serving as an internet proxy, as used herein, may refer to a system or router that provides a gateway between users and the internet (e.g., network 105). In some embodiments, scanning system 101 of Fig. 1 may access a configuration detecting services over the internet and capable of serving as an internet proxy.

[00230] Aspects of this disclosure may include embodiments configured to integrate the identified services into the graph. Integrating identified services into a graph, as used herein, may refer to the act of bringing together smaller components (the identified services) into a single system that functions as one (the graph). For example, scanning system 101 may

integrate information on services available in each node into the data structure, associated with each node in the data structure.

[00231] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to traverse the graph to identify at least one source originating via the Internet and reaching the workload. Traversing the graph, as used herein, may refer to a device, such as scanning system 101, accessing the systems represented by the nodes in the graph, to determine which systems are accessible and how. For example, scanning system 101 may analyze the graph to detect sources coming from the internet and reaching the workload, by accessing a first system, and traversing between systems in a depth-first, breadth-first, or other manner, until a path to an external network (e.g., the Internet) is reached.

[00232] In some embodiments, the at least one source may be a potential malicious source. In some embodiments, the source may include a code in any part of a software system or script that is intended to cause undesired effects, security breaches or damage to a system. (In some embodiments, the Internet at large may be considered to be a potentially malicious source.)

[00233] Aspects of this disclosure may include embodiments configured to output a risk notification associated with the workload. Outputting a risk notification, as used herein, may refer to sending out data such as a message indicating a possibility of danger. In some embodiments, scanning system 101 of Fig. 1 may produce a notification indicating a risk that is related with the workload. Examples of such risk notifications may include, for example, an electronic mail message, a visual alert, an audio alert, an audio-visual alert,

[00234] In some embodiments, the risk notification includes one or more resolution recommendations. Resolution recommendations, as used herein, may refer to a suggestion or proposal as to the best course of action in view of the identified risk.

[00235] **Fig. 6** illustrates a block diagram of method 600 performed by a processor (e.g., a processor associated with scanning system 101), consistent with disclosed embodiments. In some embodiments, the processor may operate on instructions stored in a non-transitory computer readable medium. In some embodiments, the method may include seven (or more or less) steps:

[00236] Block 602: Access at least one cloud provider API to determine a plurality of entities capable of routing traffic in a virtual cloud environment associated with a target account containing the workload. In some embodiments, scanning system 101 of Fig. 1 may determine entities capable of routing traffic, consistent with the above-disclosed embodiments.

[00237] Block 604: Query the at least one cloud provider API to determine at least one networking configuration of the entities. In some embodiments, scanning system 101 of Fig. 1

may send a request to determine at least one networking configuration of the entities, consistent with the above-disclosed embodiments.

[00238] Block 606: Build a graph connecting the plurality of entities based on the networking configuration. In some embodiments, scanning system 101 of Fig. 1 may construct a graph or diagram linking the plurality of entities based on the networking configuration, consistent with the above-disclosed embodiments.

[00239] Block 608: Access a data structure identifying services publicly accessible via the Internet and capable of serving as an internet proxy. In some embodiments, scanning system 101 of Fig. 1 may access a configuration detecting services over the internet and capable of serving as an internet proxy, consistent with the above-disclosed embodiments.

[00240] Block 610: Integrate the identified services into the graph. In some embodiments, scanning system 101 of Fig. 1 may incorporate the services into the graph, consistent with the above-disclosed embodiments.

[00241] Block 612: Traverse the graph to identify at least one source originating via the Internet and reaching the workload. In some embodiments, scanning system 101 of Fig. 1 may analyze the graph to detect sources coming from the internet and reaching the workload, consistent with the above-disclosed embodiments.

[00242] Block 614: Output a risk notification associated with the workload. In some embodiments, scanning system 101 of Fig. 1 may produce a notification indicating a risk that is related with the workload, consistent with the above-disclosed embodiments.

[00243] **Techniques for Malware Detection Using Secondary System**

[00244] Aspects of this disclosure may provide a technical solution to the challenging technical problem of malware detection on a primary system using a secondary system other than the primary system. In existing technologies, malware detection system may be run on a primary system that itself may be susceptible to malware. The malware may detect existence of the malware detection system and deceive it to avoid being detected. To resolve such a technical problem, a secondary system may be provided to host the malware detection system, which may isolate the malware detection system from potential exposure or influence from the malware that may reside on the primary system. The disclosed technical solutions may increase success rate of malware detection and thus enhance security level of the primary system.

[00245] A cyber security system, as used herein, may refer to a system including any combination of software and hardware for enhancing security of a device, a platform, or another system in a network environment. By way of example, the cyber security may be implemented as a system including scanning system 101 in Fig. 1. As another example, scanning system 101 may include the disclosed cyber security system as a subset. A cloud environment, as used herein, may refer to a computing environment running on a cloud. By way of example, the cloud environment may include or be part of cloud infrastructure 106 in Fig. 1.

[00246] Consistent with disclosed embodiments, at least one processor may be configured to utilize a cloud provider API to access a block storage volume of a workload maintained on a target account in a target system of a cloud storage environment. Utilizing, as used herein, may refer to an operation of using, deploying, enacting, enabling, activating, allocating, invoking, calling, or any operation of putting a thing into use in a computer environment. An API refers to an application programming interface herein. A cloud provider in this disclosure may refer to a cloud service provider (e.g., a service provider of cloud computing, cloud storage, or any services provided to users on demand over a network). A cloud provider API, as used herein, may refer to an API provided, prepared, enabled, activated, programmed, or written by a cloud provider. A block storage volume in this disclosure may refer to a storage volume in a block storage scheme. In some embodiments, a block storage volume may be connected, disconnected, or reconnected to a system (e.g., a physical computer, a virtual machine, a network interface, a cloud, or any combination of hardware or software modules) without interfering (e.g., shutting down, halting, rebooting, or any manner of interrupting) the operation status of the system or other running tasks on the system. For example, a block storage volume may be implemented as a virtual disk. In some embodiments, a block storage volume can be associated with an account of a user of the system, the system that it is connected to, or both.

[00247] Block storage, as used herein, may refer to a data storage scheme or model in which data is saved to storage media in fixed-sized raw data chunks (referred to as “blocks”). The raw data blocks may contain no metadata. Each block may be associated with a unique address as metadata assigned to the block. Storage blocks may be controlled by an operating system (OS) and may be accessed by a protocol (e.g., iSCSI, Fibre Channel, or Fibre Channel over Ethernet).

[00248] A storage volume, as used herein, may refer to an identifiable unit (e.g., a physical unit or a logical, virtual unit) of data storage. The storage volume may be mounted to a device via an operation system and be configured with a specific file system (e.g., New Technology File System) assigned a system-unique name or number that identifies the storage volume. The storage volume may represent a named, logical area of storage that enables users and applications to access data on the underlying device. By way of example, a storage volume may be a logical disk that represents a named, logical area of a physical storage device (e.g., a hard disk drive, solid-state drive, compact disc read-only memory, digital video disk, floppy disk, or any other type of storage device). In some embodiments, a logical storage volume may span multiple physical storage devices (e.g., hard disks) and appear as a single, contiguous storage area that works like a physical volume. Storage volumes may be flexibly configured, such as being expanded, contracted, mirrored, stripped, or adapted to support multiple disks (e.g., redundant array of independent disks).

[00249] A workload, as used herein, may refer to a specific application, service, capability, or a specific amount of work that can be run on a cloud resource, system, or infrastructure. By way of example, a workload may be a virtual machine, a database, a container, a Hadoop node, an application, a storage object, a load balancer, or an IAM (Identity and Access Management) configuration. A cloud storage environment in this disclosure may refer to a computing environment of a cloud computing model (referred to as "cloud storage") that stores data on the Internet through a cloud computing provider that manages and operates data storage as a service. A cloud storage environment may be managed and controlled by an operating system. The operation system may be susceptible to malware and security vulnerabilities, and may become a primary system (or referred to as a "target system" herein) that needs malware detection. By way of example, the cloud storage environment may include or be part of cloud infrastructure 106 in Fig. 1.

[00250] A target system of a cloud storage environment in this disclosure may refer to an operating system or a computing system with an installed operating system, which manages and controls the cloud storage environment or a subsystem of the target system. The target system may maintain one or more accounts for users of the cloud storage environment. A target account of the target system in this disclosure may refer to an account of a user of a cloud infrastructure (e.g., a cloud environment or a cloud storage environment).

[00251] By way of example, to utilize the cloud provider API to access the block storage volume (e.g., existing in storage 111A-111D in Fig. 1) of the workload, the at least one processor (e.g., a processor in scanning system 101 of Fig. 1) may communicate with the cloud storage environment (e.g., cloud infrastructure 106 in Fig. 1) via sending and receiving data (e.g., data packets) over a network (e.g., network 105 in Fig. 1), and enable one or more APIs provided by the cloud provider for accessing the block storage volume. For example, the communication and API enabling may be performed through the process of integration as described in step 201 of Fig. 2. In some embodiments, the at least one processor may create a connection (e.g., by logging in) between a first account on scanning system 101 and a second account on cloud infrastructure 106, then cause the second account to authorize access privileges to the first account via the cloud provider API for accessing storage 111A-111D in cloud infrastructure 106.

[00252] Consistent with disclosed embodiments, the at least one processor may be configured to utilize a scanner at a location of the block storage volume and on a secondary system other than the target system. A scanner, as used herein, may refer to an application, a program, a service, a process, a thread, a function, or any executable codes for performing a scanning process on a computer system to obtain information (e.g., related to structures, vulnerabilities, security issues, or any information of the computer system).

[00253] A location of the block storage volume in this disclosure may refer to a system (e.g., a physical computer, a virtual machine, a network interface, a cloud, a data center, or any combination of hardware or software modules) or an identifier associated with the system that the block storage volume is connected to. In some embodiments, the location of the block storage volume may include at least one of the target account, a secondary system account, a cloud provider account, or a third party account. By way of example, the secondary system account may be an account connected with the target account (e.g., connected by an integration process similar to step 201 in Fig. 2A). The cloud provider account may be an account connected to at least one of the target account or the secondary system account. The third party account may be an account hold by a third party and connected to at least one of the target account, the secondary system account, or the cloud provider account.

[00254] In some embodiments, the at least one processor may determine the location of the block storage volume based on at least one of the target account, a secondary system account, a cloud provider account, or a third party account. By way of example, to determine the location of the block storage volume, the at least one processor may consult a lookup table, a list, or a database that stores a relationship record between an account (e.g., a target account, a secondary system account, a cloud provider account, or a third party account) and a system (e.g., a physical computer, a virtual machine, a network interface, a cloud, or any combination of hardware or software modules) that the block storage volume is connected to. Based on the relationship record, the at least one processor may identify the system as the location using the account as a key.

[00255] A secondary system, as used herein, may refer to a computer system running at an environment or conditions not affected by another system (referred to as a "primary system" or "target system" herein). In some embodiments, the secondary system may have an operating system different from an operating system of the target account. By way of example, the target account may have a WINDOWS® operating system, and the secondary system may have a LINUX® operating system.

[00256] In some embodiments, the secondary system includes at least one of a virtual machine (e.g., any of virtual machines 107A-107D in Fig. 1), a container, or a serverless function. A container, as used herein, may refer to an OS-level virtualization instance assigned with resources by a non-virtual computer operating system (OS). A non-virtual computer operating system (OS) may create multiple isolated user space instances that may function like real computers from the point of view of programs running in them. A computer program running on a non-virtual operating system can access all available resources (e.g., connected devices, files, folders, network shares, CPU power, or any other software or hardware capabilities) of the computer. In contrast, a computer program running inside of a container can only access the resources assigned to the container.

[00257] A serverless function, as used herein, may refer to a computer function hosted in a cloud environment that may allocate resources on demand to perform a function (e.g., scanning) for a user of the cloud environment. The user of the serverless function may be free from concerns of capacity planning, configuration, management, maintenance, fault tolerance, or scaling of containers, virtual machines, or physical servers. When a serverless function is not running, no computing resources may be allocated to it (e.g., using no provisioned server, thus named "serverless"). By way of example, a serverless function may be implemented as any combination of Lambda functions, event sources, and other computing resources. A serverless function may be invoked by the user of a cloud infrastructure (e.g., a cloud environment or a cloud environment).

[00258] By way of example, to utilize the scanner at the location of the block storage volume (e.g., existing in storage 111A-111D in Fig. 1), the at least one processor (e.g., a processor in scanning system 101 of Fig. 1) may communicate with the secondary system (e.g., one or more of virtual machines 107A-107D in Fig. 1) other than the target system (e.g., the main system operating and managing storage 111A-111D) and activate the secondary system for scanning. For example, to activate the secondary system, the at least one processor may transmit a scanner (e.g., in the form of a computer program or executable codes) to the secondary system, and the secondary system may allocate computing resources (e.g., CPU powers, memory space, network ports, or any other software or hardware resources) in preparation of running the scanner. The scanner may be programmed to be stored at the location of the block storage volume or to start the scanning at the location of the block storage volume. For example, the block storage volume may be connected to a particular system in cloud infrastructure 106 in Fig. 1, and the scanner may be stored at that system (e.g., on storage connected to the system). When the scanner begins scanning, the scanner may start the scanning at that system before scanning other systems.

[00259] In some embodiments, to utilize the scanner, the at least one processor may suspend an operation of the scanner after the scan of the block storage volume. Suspending, as used herein, may refer to an operation of pausing, stopping, ceasing, freezing, halting, hanging, terminating, killing, or any manner or temporarily or permanently setting aside an ongoing process or computer program. By way of example, after the scanner completes scanning the block storage volume, the at least one processor may detect the completion by an active manner (e.g., by periodically polling the secondary system for a status of the scanning) or a passive manner (e.g., by receiving a message from the secondary system indicating the completion of the scanning). The at least one process may then send data to the secondary system, the data including a signal or computer-executable codes for suspending the operation of the scanner. After receiving the data, the secondary system may suspend the scanner and release the computing resources previously allocated to the scanner.

[00260] In some embodiments, to utilize the scanner, the at least one processor may modify a pre-utilized scanner at the location of the block storage volume based on information related to the target account to obtain a modified scanner, and then utilize the modified scanner. A pre-utilized scanner in this disclosure may refer to a scanner that has been previously utilized (e.g., on the same secondary system) and not destroyed. In some embodiments, the information related to the target account may include data or parameters indicating a status or a change of the target account, such as, for example, newly created files, deleted files, changed files, or any data indicating a state of the target account. For example, the pre-utilized scanner at the block storage volume may use a configuration file. The configuration file may include data or parameters indicating a previous status (e.g., list of existing files, public network ports, Internet-accessible data channels, or any operation state parameters) of the target account. The information related to the target account may include data indicating a current status of the target system. The current status of the target system may be different from the previous status of the target system (e.g., including changed files, changed public network ports, changed Internet-accessible data channels, or any changed operation state parameters). The at least one processor may modify the pre-utilized scanner by modifying the configuration file to reflect the current status of the target system. Then, the at least one processor may utilize the modified scanner.

[00261] Consistent with disclosed embodiments, the at least one processor may be configured to scan the block storage volume for malicious code, using the secondary system. Scanning, as used herein, may refer to an operation of traversing, checking, inspecting, reading, examining, searching, or any manner of looking up. Malicious code, as used herein, may refer to an application, a computer program, a service, a process, a thread, a function, a script, or any type of executable codes that may be executed on a computer without authorization and with a malicious or harmful intent (e.g., manipulating the computer into dangerous behaviors, writing changes or add-ons to existing computer programs or files, stealing computer files, copying sensitive information without authorization, damaging the computer itself, disabling one or more functions of the computer, or attacking another computer using the computer as a disguise). For example, malicious code may include attack scripts, computer viruses, computer worms, Trojan horses, backdoors, or any other malicious executable content. In some embodiments, the malicious code may include a rootkit. A rootkit in this disclosure may refer to any type of computer software designed to enable access to a computer or an area of its software without authorization.

[00262] By way of example, to scan the block storage volume (e.g., existing in storage 111A-111D in Fig. 1) for malicious code, the secondary system (e.g., one or more of virtual machines 107A-107D in Fig. 1) may run the utilized scanner with allocated computing resources to look up databases (e.g., a blacklist) of known malicious files, look up databases storing

patterns of malicious files and identify the patterns, look up databases of cryptographic hashes that identify malicious files, run potential malicious code in a sandbox (e.g., a confined and isolated environment for running programs without causing changes outside the confined and isolated environment), determine whether there are scanned files that are not in a database (e.g., a whitelist) of known good files, detect changes over time in the scanned environment, or any combination thereof.

[00263] In some embodiments, scanning the block storage volume may include scanning disk-backed memory. A disk-backed memory in this disclosure may refer to a storage space implemented on a disk (e.g., a hard disk) and functioning like a volatile memory from the perspective of a running program. By way of example, scanning the block storage volume may include scanning files stored in the disk-backed memory. In some embodiments, the disk-backed memory may include at least one of a page file or a cache file. A page file, as used herein, may refer to a file containing one or more pages (also referred to as "memory pages" or "virtual pages"), each of the one or more pages being a fixed-length contiguous block of virtual memory and described by a single entry in a page table. A page file may be stored on a disk (e.g., a hard disk) for memory management of a virtual memory used by an operating system. A cache file, as used herein, may refer to a file storing data so that future requests for the stored data can be served faster. The data stored in a cache file may be the result of a computation or a copy of the data stored elsewhere.

[00264] Consistent with disclosed embodiments, the at least one processor may be configured to identify malicious code based on the scan. By way of example, if the secondary system scans the block storage volume by consulting databases of known malicious files and finds a scanned file matches a record in the databases, the at least one processor may identify the matched scanned file as the malicious code. In another example, if the secondary system scans the block storage volume by consulting databases storing patterns of malicious files and identifies a pattern of a scanned file matches one or more of the stored patterns, the at least one processor may identify the scanned file as the malicious code. In yet another example, if the secondary system scan the block storage volume by consulting databases of cryptographic hashes that identify malicious files and finds that a cryptographic hash of a scanned file matches one or more of the cryptographic hashes that identify malicious files, the at least one processor may identify the scanned file as the malicious code.

[00265] In yet another example, if the secondary system scans the block storage volume by running potential malicious code in a sandbox and finds that running a scanned file causes malicious or harmful changes or behavior (e.g., an attempt to modify a system setting, an attempt to access files outside of a set of directories, an attempt to perform a network mapping process), the at least one processor may identify the scanned file as the malicious code. In yet another example, if the secondary system scans the block storage volume to determine whether

there are scanned files that are not in a database of known good files and finds that a scanned file is not in the database of known good files, the at least one processor may identify the scanned file as the malicious code. As yet another example, if the secondary system scans the block storage volume to detect changes over time in the scanned environment and finding that a scanned file causes a malicious or harmful change in the scanned environment over time, the at least one processor may identify the scanned file as the malicious code. It should be noted that the identification of the malicious code based on the scan may be implemented in various manners and is not limited to the examples described herein.

[00266] Consistent with disclosed embodiments, the at least one processor may be configured to output from the secondary system, a notification of a presence of malicious code in the target system. In some embodiments, the notification may be a pop-up window, a pop-pup dialog, a flashing symbol, an email, a text message, a mobile application notice, a phone call, a warning sound, a tactile feedback (e.g., a vibration), or any type of communications. By way of example, the secondary system may trigger a communication channel between the secondary system and the target system and feed one or more parameters to the communication channel, in which the one or more parameters represent the presence of the malicious code. The secondary system may then send data representing the notification via the triggered communication channel including the one or more parameters to the target system. After receiving the data, the target system may parse the data and identify the one or more parameters. The target system may then display the notification (e.g., as a pop-up window on a screen) that present the presence of the malicious code.

[00267] Consistent with disclosed embodiments, the at least one processor may be further configured to identify a source of the malicious code. The notification of the presence of malicious code outputted from the secondary system in the target system may include information related to the identified source of the malicious code. In some embodiments, the source of the malicious code may include identity of a developer or distributor of the malicious code, the landing location (e.g., a network port or a vulnerable access point) where the malicious code is imported to the target system, source codes of the malicious code, or any information or data indicating the origination of the malicious code. The information related to the identified source of the malicious code may include a text, a picture, a symbol, a video, a link, a voice, or any type of data for presenting and describing the identified source of the malicious code.

[00268] By way of example, **Fig. 7** is a block diagram illustrating an exemplary process 300 of cyber security scanning for a cloud environment, consistent with the disclosed embodiments. While the block diagram may be described below in connection with certain implementation embodiments presented in other figures, those implementations are provided for illustrative purposes only, and are not intended to serve as a limitation on the block diagram.

For example, the steps in Fig. 7 may be executed in any order, steps may be duplicated, or steps may be omitted. In some embodiments, the process 300 may be performed by at least one processor (e.g., a CPU) of a computing device or system (e.g., scanning system 101 in Fig. 1) to perform operations or functions described herein, and may be described hereinafter with reference to Figs. 1-2D by way of example. In some embodiments, some aspects of the process 300 may be implemented as software (e.g., program codes or instructions) that are stored in a memory or a non-transitory computer-readable medium. In some embodiments, some aspects of the process 300 may be implemented as hardware (e.g., a specific-purpose circuit). In some embodiments, the process 700 may be implemented as a combination of software and hardware.

[00269] Fig. 7 includes process blocks 702-710. At block 702, at least one processor may utilize a cloud provider API to access a block storage volume of a workload maintained on a target account in a target system of a cloud storage environment.

[00270] At block 704, the at least one processor may utilize a scanner at a location of the block storage volume and on a secondary system other than the target system. In some embodiments, the secondary system may include at least one of a virtual machine, a container, or a serverless function. In some embodiments, the secondary system may have an operating system different from an operating system of the target account.

[00271] In some embodiments, to utilize the scanner, the at least one processor may suspend an operation of the scanner after the scan of the block storage volume. In some embodiments, to utilize the scanner, the at least one processor may modify a pre-utilized scanner at the location of the block storage volume based on information related to the target account to obtain a modified scanner, and the utilize the modified scanner.

[00272] At block 706, the at least one processor may scan the block storage volume for malicious code, using the secondary system. In some embodiments, to scan the block storage volume, the at least one processor may scan disk-backed memory. For example, the disk-backed memory includes at least one of a page file or a cache file. In some embodiments, the malicious code includes a rootkit.

[00273] At block 708, the at least one processor may identify malicious code based on the scan. At block 710, the at least one processor may output from the secondary system, a notification of a presence of malicious code in the target system.

[00274] Consistent with disclosed embodiments, besides blocks 702-710, the at least one processor may further determine the location of the block storage volume based on at least one of the target account, a secondary system account, a cloud provider account, or a third party account.

[00275] Consistent with disclosed embodiments, besides blocks 302-310, the at least one processor may further identify a source of the malicious code. The notification outputted in block 710 may include information related to the identified source of the malicious code.

[00276] **Forward and Rearward Facing Attack Vector Visualization Techniques**

[00277] **Fig. 8** is a schematic block diagram illustrating an exemplary embodiment of a system for performing visualization of forward and backward facing threats.

[00278] In one of many embodiments, the system may include a graphical user interface system for providing forward and backward facing attack vector visualizations. A graphical user interface system may include systems similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows 10 Mobile, PalmOS-Web OS, or Firefox OS. Forward facing attack vectors may be those external entities that may access an analyzed asset. Backward facing attack vectors may include the possible impacts to an analyzed asset and those impacts' possible effect. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of attack vectors.

[00279] In an embodiment, the system may comprise at least one processor configured to identify assets 801 in a cloud environment. The at least one processor may include a processor such as user device 102 of Fig. 1. The at least one processor may include a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service.

[00280] Another embodiment contemplated may consist of at least one processor configured to identify risks 803 associated with each of the identified asset. The identified risks 803 associated with each of the identified assets may be listed sequentially based on probability of risk, severity of risk, or similar method of ordering the risks. The identified risks 303 associated with an identified asset may be scored based on the level of risk to the asset and may be categorized as a high risk, medium risk, or low risk. Once the level of risk can be determined by scanning system 101, the level of risk can be conveyed and displayed to a user device 102 with a representative graphical display.

[00281] An exemplar embodiment may additionally comprise at least one processor configured to identify relationships 805 between at least some of the identified assets, the relationships including at least one of a trust 807A, a network connectivity 807B, or a mechanism of network proxying 807C. A trust 807A may establish a relationship and may

involve cryptography, digital signatures, electronic certificates, or similar method of establishing a trusting relationship. A network connectivity 807B may include wireless or wired connections and may be established once a relationship of trust 807A is established. A mechanism of network proxying 807C can be accommodated once trust 807A and network connectivity 807B are established through any number of connected servers to accomplish proxying.

[00282] A further embodiment may also comprise a processor configured to receive an identification 809 of a specific asset under investigation. The specific asset under investigation may be a physical or virtual machine that has software operating a specific version that is vulnerable to any number of associated risks. The identification 809 of a specific asset under investigation may involve an IP address, universally unique identifier, or similar system to uniquely or semi-uniquely identify the specific asset under investigation.

[00283] A system among several embodiments may include a processor configured to perform a forward analysis 811A of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment. A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. The analysis performed by scanning system 101 may determine the origin of the forward threat, the nature of the threat to the analyzed system, and the seriousness of the threat originating from the Internet. For example, scanning system 101 may identify at least one possible attack vector to the workload currently being analyzed. The network outside of the cloud environment may identify new or existing threats and attack vectors currently threatening the associated system and workload.

[00284] An exemplary embodiment may further comprise a processor configured to perform a backward analysis 811B of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset. A backward analysis 811B of the associated machine may look downstream to related workloads and systems connected to the analyzed systems. A scan downstream may involve, among several embodiments, the downstream exposure risk including an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset. The at least one exposure risk may include one or more entry point to access the exposed asset and any related machines potentially vulnerable to attack from the analyzed and exposed asset.

[00285] Another embodiment may comprise a processor configured to output a signal 813 to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points. The display of a forward or

backward path to a specific asset may be one of several visual displays such as linear, graphical, or through computer-generated images. The visualization of a plurality of entry points and lateral movement risks with the plurality of entry points may include visualization of the plurality of paths may sequence the paths based on likelihood of access via that path or the severity of potential threat in a path of access to an asset.

[00286] An example embodiment may comprise a system wherein the network outside the cloud environment includes the Internet. The network outside the cloud environment may include access to the Internet in order for the system to access a plurality of analyzed and potentially vulnerable assets.

[00287] Another embodiment may comprise a system wherein the assets in the cloud environment include at least one of a virtual machine, a network appliance, a storage appliance, a compute instance, or an engine instance. A virtual machine may include an emulated version of a computer - including an operating system, memory, storage, graphics processing - such that it can be indistinguishable from a standard (non-virtual) machine to a running program. A network appliance may include a plurality of servers connected through the Internet in order to establish a system and network of stored assets and workloads. A storage appliance may be comprised by a server designed to store workloads and assets to be analyzed. A compute instance may involve input and processing of data entries that cause the analyzed asset to be accessed and run. An engine instance may include a search algorithm run to analyze and search the analyzed workloads or virtual machines to determine the software and data stored on the said virtual machine or stored instance.

[00288] One among many embodiments may comprise a system wherein identifying the assets in a cloud environment includes identifying the assets based on at least one of an identity and access management policy, an organization policy, or an access policy. An identity and access management policy may comprise a software tool or similar method of establishing identity of users attempting to access a system to verify identities of users and control access to an associated workload or system functionality. An organization policy may include a policy applied to all users by the controlling organization of a workload. An access policy may determine the authorized users of an associated system.

[00289] One possible embodiment may comprise a system wherein the presentation of the forward and backward paths indicates alternative paths connecting between the specific asset and an upstream asset or a downstream asset. The indication will be via a visualization showing specific alternative paths connecting specific assets or it may visualize a plurality of alternative paths connecting specific assets. Alternative paths connecting between the specific asset and an upstream asset or a downstream asset may involve a direct connection, such as via the Internet, or an indirect connection via one or more series of servers acting as proxy servers between the analyzed asset and the upstream or downstream asset.

[00290] An exemplary embodiment may further comprise a system wherein the visualization includes a presentation of the alternative paths. The alternative paths may be presented visually through a graphical representation, sequential list of alternative paths, or a computer-generated image that represents the plurality of alternative paths.

[00291] Another embodiment may comprise a system wherein the presentation of the forward and backward paths indicates port numbers for each pathway. The port numbers may be indicated by a series of alphanumeric characters (e.g., “HTTP” or “80”) or other identifier.

[00292] One among several exemplary embodiments may further comprise a system wherein the visualization of the entry points indicates at least one entry point of risk. The visualization may indicate one or more points of entry to a given analyzed system. An entry point of risk may involve a port of access to the workload, and may involve a one or more entry points visualized for use by user device 102.

[00293] Another exemplary embodiment may comprise a system wherein the at least one processor is further configured to monitor network activities of the assets in a cloud environment. The processor may monitor continuously, intermittently, or as directed by an authorized user to direct monitoring of the analyzed network activities of the assets. The cloud environment may be, among several embodiments, the cloud infrastructure 106.

[00294] An embodiment contemplated may comprise a system further configured to detect a potential risk associated with the specific asset based on a network activity of the specific asset. The potential risk may be associated to a specific asset based on historical threats to the specific asset or a list of possible threats to the specific asset based on its current network activities.

[00295] Another embodiment may comprise a system further configured to detect a potential risk associated with the specific asset based on a network activity of an upstream asset of the specific asset. The potential risk associated with the specific asset may be identified by scanning system 101 and determine whether a potential risk can be classified as high risk, medium risk, or low risk. The potential risk may be identified based on the specific asset, the network activity performed by the specific asset, or the specific asset type or related software of the upstream asset.

[00296] One of many embodiments may comprise a system further configured to detect a potential risk associated with the specific asset based on a network activity of a downstream asset of the specific asset. The potential risk may be identified based on the specific asset, the network activity performed by the specific asset, or the specific asset type or related software of the upstream asset. For example, scanning system 101 may perform a scan on an asset being analyzed, and based on the one or more entry points of risk, may determine if the downstream asset of the specific asset poses a potential risk to the analyzed system.

[00297] Another exemplary embodiment may comprise a method for generating a graphical user interface for providing forward and backward facing attack vector visualizations. The method may include a method of generating a graphical user interface similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows 10 Mobile, PalmOS-Web OS, or Firefox OS. A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. Backward analysis 811B may include the possible impacts to an analyzed asset and those impacts' possible effect. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of attack vectors.

[00298] One exemplar embodiment may further comprise a method of identifying assets 801 in a cloud environment. A method may include at least one processor that may include a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service.

[00299] One of several embodiments may comprise a method of identifying risks 803 associated with each of the identified assets 801. The method of identifying risks 803 associated with each of the identified assets 801 may be listed sequentially based on probability of risk, severity of risk, or similar method of ordering the risks. The identified risks 803 associated with an identified asset may be scored based on the level of risk to the asset and may be categorized as a high risk, medium risk, or low risk. Once the level of risk can be determined by scanning system 101, the level of risk can be conveyed and displayed to a user device 102 with a representative graphical display.

[00300] Another embodiment contemplated may comprise a method for identifying relationships 305 between at least some of the identified assets 801, the relationships including at least one of a trust 807A, a network connectivity 807B, or a mechanism of network proxying 807C. A trust 807A may establish a relationship and may involve cryptography, digital signatures, electronic certificates, or similar method of establishing a trusting relationship. A network connectivity 807B may include wireless or wired connections and may be established once a relationship of trust 807A is established. A mechanism of network proxying 807C can be accommodated once trust 807A and network connectivity 807B are established through any number of connected servers to accomplish proxying.

[00301] An embodiment may further comprise a method of receiving an identification of a specific asset 809 under investigation. The specific asset under investigation may be a physical

or virtual machine that has software operating a specific version that is vulnerable to any number of associated risks 803. The identification 809 of a specific asset under investigation may involve an IP address, universally unique identifier, or similar system to uniquely or semi-uniquely identify the specific asset under investigation.

[00302] An embodiment may also comprise a method of performing a forward analysis 811A of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment. A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. A forward analysis may involve analysis of the specific asset under identification to identify at least one possible Internet-originating attack vector to the asset. The analysis performed by scanning system 101 may determine the origin of the forward threat, the nature of the threat to the analyzed system, and the seriousness of the threat originating from the Internet. For example, scanning system 101 may identify at least one possible attack vector to the workload currently being analyzed. The network outside of the cloud environment may identify new or existing threats and attack vectors currently threatening the associated system and workload.

[00303] One embodiment may comprise a method of performing a backward analysis 811B of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset. Backward analysis 811B may include the possible impacts to an analyzed asset and those impacts' possible effect. A scan downstream may involve, among several embodiments, the downstream exposure risk including an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset. The at least one exposure risk may include one or more entry point to access the exposed asset and any related machines potentially vulnerable to attack from the analyzed and exposed asset.

[00304] An embodiment may comprise a method of outputting a signal 813 to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points. The display of a forward or backward path to a specific asset may be one of several visual displays such as linear, graphical, or through computer-generated images. The visualization of a plurality of entry points and lateral movement risks with the plurality of entry points may include visualization of the plurality of paths may sequence the paths based on likelihood of access via that path or the severity of potential threat in a path of access to an asset.

[00305] Another embodiment may comprise a method wherein the network outside the cloud environment includes the Internet. The network outside the cloud environment may include access to the Internet in order for the system to access a plurality of analyzed and potentially vulnerable assets.

[00306] One of several embodiments may comprise a method wherein the assets in the cloud environment include at least one of a virtual machine, a network appliance, a storage appliance, a compute instance, or an engine instance. A virtual machine may include an emulated version of a computer - including an operating system, memory, storage, graphics processing - such that it can be indistinguishable from a standard (non-virtual) machine to a running program. A network appliance may include a plurality of servers connected through the Internet in order to establish a system and network of stored assets and workloads. A storage appliance may be comprised by a server designed to store workloads and assets to be analyzed. A compute instance may involve input and processing of data entries that cause the analyzed asset to be accessed and run. An engine instance may include a search algorithm run to analyze and search the analyzed workloads or virtual machines to determine the software and data stored on the said virtual machine or stored instance.

[00307] An embodiment may further comprise a method wherein identifying assets in a cloud environment includes identifying the assets based on at least one of an identity and access management policy, an organization policy, or an access policy. An identity and access management policy may comprise a software tool to verify identities of users and control access to an associated workload or system functionality. An organization policy may include a policy applied to all users by the controlling organization of a workload. An access policy may determine the authorized users of an associated system.

[00308] An embodiment may also comprise a method wherein the presentation of the forward and backward paths indicates alternative paths connecting between the specific asset and an upstream or a downstream asset. The indication may be via a visualization showing specific alternative paths connecting specific assets or it may visualize a plurality of alternative paths connecting specific assets.

[00309] Another exemplar embodiment may comprise a method wherein the visualization includes a presentation of the alternative paths. The alternative paths may be presented visually through a graphical representation, sequential list of alternative paths, or a computer-generated image that represents the plurality of alternative paths.

[00310] Another embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations.

[00311] One embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor 800, are configured to

perform operations identifying assets 801 in a cloud environment. the instructions stored on a non-transitory computer-readable medium may include at least one processor 800 that may include a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service.

[00312] An embodiment may further comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor 300 to perform operations comprising identifying risks 803 associated with each of the identified asset. The instructions performed of identifying risks 803 associated with each of the identified assets 301 may be listed sequentially based on probability of risk, severity of risk, or similar method of ordering the risks.

[00313] Another embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising identifying relationships 805 between at least some of the identified assets 301, the relationships including at least one of a trust 807A, network connectivity 807B, or a mechanism of network proxying 807C.

[00314] An exemplar embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising receiving an identification of a specific asset 809 under investigation. The instructions may include receiving information identifying the specific asset 809. The specific asset 809 under investigation may be a physical or virtual machine that has software operating a specific version that is vulnerable to any number of associated risks 803.

[00315] One among many embodiments may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising performing a forward analysis 811A of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment. . A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. And the scanning system 101 may identify at least one possible attack vector to the workload currently being analyzed. The network outside of the cloud environment may identify new or existing threats and attack vectors currently threatening the associated system and workload.

[00316] An embodiment may further comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising performing a backward

analysis 811B of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset. Backward analysis 811B may include the possible impacts to an analyzed asset and those impacts' possible effect. The at least one exposure risk may include one or more entry point to access the exposed asset and any related machines potentially vulnerable to attack from the analyzed and exposed asset.

[00317] An embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising outputting a signal 813 to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points. The display of a forward or backward path to a specific asset may be one of several visual displays such as linear, graphical, or through computer-generated images. The visualization of a plurality of entry points and lateral movement risks with the plurality of entry points may include visualization of the plurality of paths may sequence the paths based on likelihood of access via that path or the severity of potential threat in a path of access to an asset.

[00318] **Passive Key Identification Techniques**

[00319] Aspects of this disclosure may provide a technical solution to the challenging technical problem of identifying access keys (e.g., passwords, Secure Shell (SSH) keys, or cloud keys) to compute resources (e.g., machines, containers, storages, or any hardware or software component) in a cloud environment without using the access keys to access the compute resources. In existing technologies, scanning systems may identify access keys to compute resources and may need to verify the identified keys by actually accessing the compute resources using the access keys. In such cases, the actual accesses may generate logs or records (e.g., for successful or failed accesses) that can constitute activity patterns similar to an attacker to the system. Such activity patterns may distract a system administrator (e.g., an individual or a computer program) to misidentify the scanning system as an attacker. Further, actual accesses may create loads on both the scanning system and the compute resources and may reduce the computing resources and power for other tasks. To resolve such a technical problem, a cryptographic analysis may be performed to identify a first set of fingerprints of the identified keys, and trust configurations of the compute resources may be analyzed to identify a second set of fingerprints of the compute resources. The first and second set of fingerprints may be compared to match keys with the compute resources without using the keys to access the compute resources. By doing so, the scanning system may perform its

function without raising false alarms to the system administrator, while reducing computing costs to the cloud environment.

[00320] A cyber security system, as used herein, may refer to a system including any combination of software and hardware for enhancing security of a device, a platform, or another system in a network environment. By way of example, the cyber security may be implemented as a system including scanning system 101 in Fig. 1. As another example, scanning system 101 may include the disclosed cyber security system as a subset. A compute resource, as used herein, may refer to a database, a virtual machine, a storage, a keystore, a scanning system, a load balancer, a server, a computer, a container, or any physical or virtual component integrated in or communicatively connected to a cloud infrastructure. By way of example, the compute resource may be scanning system 101, any of databases 103A-103D, any of virtual machines 107A-107D, any of databases 109A-109D, any of storage 111A-111D, any of keystores 113A-113D, or load balancer 115 in cloud infrastructure 106 in Fig. 1.

[00321] A key to access a compute resource, as used herein, may refer to computerized data that includes a credential for granting permission to an accessor to access the compute resource, or a service or function provided by the compute resource. By way of example, the key may include at least one of a password, a remote access key, a script containing a password, a cloud key (e.g., a credential file for accessing a cloud service, such as an AWS credential file), an Secure Shell (SSH) key, an AWS key, a private component of a private-public key pair, or any type of credential that provide checked or unchecked access to the compute resource. Matching a key to a compute resource, as used herein, may refer to confirming, verifying, validating, or any operation or process of determining that the key is valid for an accessor (e.g., another compute resource) to access the compute resource. For example, to match a key to a compute resource, an accessor may test by using the key to access the compute resource and determines that the key is matched to the compute resource if such access is successful. As another example, to match a key to a compute resource, an accessor may perform methods or algorithms to verify that the key is matched to the compute resource without using the key to access the compute resource.

[00322] Consistent with disclosed embodiments, at least one processor may be configured to analyze a cloud environment to identify a plurality of keys to the compute resources in the cloud environment. In some embodiments, the at least one processor may analyze the cloud environment to identify the plurality of keys to the compute resources in the cloud environment in any of steps 231, 237, or 239 in FIG. 2D. In some embodiments, the at least one processor may analyze the cloud environment to identify the plurality of keys to the compute resources in the cloud environment in any step described in association with FIGS. 2A-2D. In some embodiments, at least one of the plurality of keys may include at least one of a password (e.g., a combination of alphanumeric characters and symbols), a script (e.g., a

PYTHON® script or a shell script) containing a password, a cloud key, or an Secure Shell (SSH) key. For example, a cloud key may be a key for accessing a cloud resource.

[00323] A cloud environment, as used herein, may refer to a computing environment running on a cloud. By way of example, the cloud environment may include or be part of cloud infrastructure 106 in Fig. 1. Analyzing a cloud environment, as used herein, may refer to an operation or a process of separating or dividing the cloud environment into parts or components (e.g., physical components or logic components), then determining nature and relationship of the parts or components based on data associated with the parts or components.

[00324] By way of example, if the cloud environment is cloud infrastructure 106 in Fig. 1, to analyze cloud infrastructure 106, the at least one processor may access cloud infrastructure 106 (e.g., via network 105 or via internal connections in cloud infrastructure 106) and obtain a list of compute resources in cloud infrastructure 106 (e.g., scanning system 101, any of databases 103A-103D, any of virtual machines 107A-107D, any of databases 109A-109D, any of storage 111A-111D, any of keystores 113A-113D, or load balancer 115). For example, the at least one processor may obtain the list of compute resources by reading a configuration file from a database (e.g., one of databases 109A-109D), or visit available compute resources one after another through communicative connections between the compute resources to generate the list of compute resources. After obtaining the list of compute resources in cloud infrastructure 106, the at least one processor may determine the nature of the compute resources, such as their types, numbers, functions, or any other characteristics or features. The at least one processor may further determine the relationship between the compute resources, such as communicative connections, access privileges, data input/output directions, or controls between the compute resources.

[00325] Identifying a key to a compute resource, as used herein, may refer to an operation or a process of locating, recognizing, or any operation or process or analyzing computerized data or information to determine that the computerized data or information is a key to the compute resources. By way of example, the at least one processor may analyze cloud infrastructure 106 in Fig. 1 to recognize a list of compute resources in cloud infrastructure 106 (e.g., scanning system 101, any of databases 103A-103D, any of virtual machines 107A-107D, any of databases 109A-109D, any of storage 111A-111D, any of keystores 113A-113D, or load balancer 115). Then, the at least one processor may read computerized data stored in a compute resource and identify that some of the computerized data are one or more keys to other compute resources in cloud infrastructure 106. For example, to recognize the keys, the at least one processor may read and compare the computerized data with records stored in keystores 113A-113D, and determine that the computerized data is a key if it matches a record in any of keystores 113A-113D. As another example, the at least one processor may read the computerized data and check its syntaxes, text string patterns, file formats, file properties,

encryption manners, library versions, software versions, or any other characteristics or features of the computerized data, and determine that the computerized data is a key if its checked characteristics or features fit a predetermined pattern of a key or fit an entry in a dictionary of keys. For example, the dictionary of keys may be stored in a keystore (e.g., any of keystores 113A-113D).

[00326] In some embodiments, the plurality of keys may be stored in at least one workload. A workload, for example, may refer to a specific application, service, capability, or a specific amount of work that can be run on a cloud resource, system, or infrastructure. By way of example, a workload may be a virtual machine, a database server, a container, a Hadoop node, an application, a storage server, a load balancer, or an IAM (Identity and Access Management) configuration. By way of example, with reference to FIG. 1, cloud infrastructure 106 may include workloads such as scanning system 101, databases 103A-103D, virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115.

[00327] Consistent with disclosed embodiments, at least one processor may be configured to perform a cryptographic analysis on the plurality of keys to identify a first set of fingerprints that uniquely identify each of the plurality of keys. The first set of fingerprints may be non-functional. In some embodiments, the at least one processor may perform the cryptographic analysis in step 237 of FIG. 2D. In some embodiments, the at least one processor may perform the cryptographic analysis in any step described in association with FIGS. 2A-2D.

[00328] A cryptographic analysis on data, as used herein, may refer to an operation or process of identifying ciphertext, ciphers and cryptosystems contained in the data to discover hidden aspects of the data that may improve or weaken security of the data or compute resources associated with the data. In some embodiments, to perform the cryptographic analysis on the plurality of keys, the at least one processor may perform an algorithm on the plurality of keys to convert the plurality of keys into another form of data. For example, the at least one processor may perform a hashing process on the plurality of keys (e.g., by inputting values of the keys to a hash function) to obtain corresponding hash values of respective keys.

[00329] A fingerprint of a key, as used herein, may refer to non-functional data (e.g., a numeric value, an alphanumeric string, or any combination of letters, numbers, or symbols) generated based on the key and may uniquely identify the key. By way of example, a fingerprint may be generated by performing a hashing process on a key or a part of the key. The hashing process may include, for example, an MD5 algorithm, an SHA-1 algorithm, an SHA-2 algorithm, an SHA-3 algorithm, a RIPEMD-160 algorithm, a BLAKE2 algorithm, a BLAKE3 algorithm, or any type of cryptographic hash algorithms. In such cases, the fingerprint may be a hash value of the key, a part (e.g., a truncated part) of the hash value of the key, a hash value of a part of the key, or a part (e.g., a truncated part) of the hash value of the part of the key. The hash value

may be unique (e.g., non-overlap with any hash value of any other key) and non-functional (e.g., unable to be used as a key to access a compute resource). To identify the first set of fingerprints, the at least one processor may perform the hashing process on the plurality of keys to obtain a plurality of hash values, and search the cloud environment to check whether there exists any compute resource that stores the plurality of hash values.

[00330] In some embodiments, at least one of the first set of fingerprints may be non-identical to any key of the plurality of keys. For example, if the first set of fingerprints are hash values, they may be non-identical to the plurality of keys.

[00331] Consistent with disclosed embodiments, at least one processor may be configured to analyze trust configurations of the compute resources to identify a second set of fingerprints for each of the compute resources. In some embodiments, the at least one processor may analyze the trust configurations in step 233 of FIG. 2D. In some embodiments, the at least one processor may analyze the trust configurations in any step described in association with FIGS. 2A-2D.

[00332] A trust configuration, as used herein, may refer to data (e.g., a stored file, a database entry, a value stored in a data structure, or any computerized information) that stores a pre-established trust relationship between at least two compute resources. For example, a trust configuration may be a file storing a trust policy (e.g., a public key infrastructure (PKI), a digital signature system, or an AWS trust policy). Analyzing a trust configuration, as used herein, may refer to an operation or a process of separating or dividing compute resources associated with the trust configuration into units or groups, then determining nature and relationship of the units or groups based on relationship data associated with the units or groups. For example, the relationship data may include flags, indicators, or any type of value that indicates a first compute resource has an access privilege to a second compute resource.

[00333] A fingerprint for a compute resource, as used herein, may refer to non-functional data (e.g., a numeric value, an alphanumeric string, or any combination of letters, numbers, or symbols) generated based on a key for accessing the compute resource and may uniquely identify the key. By way of example, a fingerprint for a compute resource may be generated by applying a cryptographic hash function (e.g., an SHA-1 or SHA-2 function) to a key for accessing the compute resource to obtain a sequence of bytes (e.g., a hash value or a modified hash value), and the sequence of bytes may be the fingerprint for the compute resource. The key may be uniquely associated with the compute resource.

[00334] For example, to identify the second set of fingerprints, the at least one processor may analyze a trust configuration file (e.g., a file storing fingerprints and their associated data) by reading the byte stream of the trust configuration file to determine whether any sequence of bytes that fits to a predetermined pattern (e.g., an alphanumeric string pattern). The sequences

of bytes that fit to the predetermined pattern may be determined as the second set of fingerprints.

[00335] Consistent with disclosed embodiments, at least one processor may be configured to compare the first set of fingerprints with the second set of fingerprints to match keys with the compute resources without using the keys to access the compute resources. In some embodiments, the at least one processor may compare the first set of fingerprints with the second set of fingerprints in any step described in association with FIGS. 2A-2D. Comparing a first fingerprint and a second fingerprint, as used herein, may refer to performing a textual, numerical, or semantic comparison of the first fingerprint and the second fingerprint, or performing an algorithmic conversion of at least one of the first fingerprint or the second fingerprint to further perform a textual, numerical, or semantic comparison of the conversion results. By way of example, a first compute resource may store a key (e.g., a public key) for accessing a second compute resource. The at least one processor may analyze the cloud environment to identify the key from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. A third compute resource (e.g., may be the same as or different from the second compute resource) may store in its trust configurations a second fingerprint of the key. The second fingerprint may be generated by a different processor at a different time. The at least one processor may then analyze the trust configurations of the third compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other. If the first fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource. As can be seen, after comparison, if the first set of fingerprints are matched with the second set of fingerprints, keys may be matched with the compute resources, and the at least one processor needs not use the keys to access the compute resources to confirm that the keys are matched to the compute resources.

[00336] Consistent with disclosed embodiments, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the first set of fingerprints. A multi-machine interaction, as used herein, may refer to data input/output, operation control, status polling, scanning, or any type of an interaction between a plurality of machines in a cloud environment. Analyzing a multi-machine interaction, as used herein, may refer to an operation or a process of separating or dividing machines in the cloud environment into units or groups, then determining nature and relationship of the units or groups based on relationship data associated with the units or groups. For example, the relationship data may be the first set of fingerprints.

[00337] By way of example, to analyze the multi-machine interaction, the at least one processor may compare the first set of fingerprints with the second set of fingerprints. If a first fingerprint of the first set of fingerprints matches with a second fingerprint of the second set of fingerprints, the at least one processor may determine that a first compute resource associated with the first fingerprint may have an inter-machine interaction (e.g., having an ability to access the other) with a second compute resource associated with the second fingerprint. By way of example, a first compute resource may store a key (e.g., a private key) for accessing a second compute resource. The at least one processor may analyze the cloud environment to identify the key from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. A third compute resource (e.g., may be the same as or different from the second compute resource) may store in its trust configurations a second fingerprint of the key. The at least one processor may then analyze the trust configurations of the third compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other (e.g., being the same). If the first fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource, and may further determine an inter-machine interaction in which the first compute resource has ability to access the second compute resource.

[00338] Consistent with disclosed embodiments, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the plurality of keys. For example, the at least one processor may use the plurality of keys as the relationship data for determining nature and relationship of compute resources in the cloud environment. By way of example, to analyze the multi-machine interaction, the at least one processor may compare a first key for accessing a first one of the compute resources and a second key for accessing a second one of the compute resources. If the first key matches with (e.g., is identical to) the second key, the at least one processor may determine that the first one of the compute resources may have an interaction (e.g., having an ability to access the other) with the second one of the compute resources.

[00339] By way of example, **Fig. 9** is a block diagram illustrating an exemplary process 900 of matching keys with compute resources in a cloud environment, consistent with the disclosed embodiments. While the block diagram may be described below in connection with certain implementation embodiments presented in other figures, those implementations are provided for illustrative purposes only, and are not intended to serve as a limitation on the block diagram. For example, the steps in Fig. 9 may be executed in any order, steps may be duplicated, or steps may be omitted. In some embodiments, the process 900 may be performed by at least one processor (e.g., a CPU) of a computing device or system (e.g., scanning system

101 in Fig. 1) to perform operations or functions described herein, and may be described hereinafter with reference to Figs. 1-2D by way of example. In some embodiments, some aspects of the process 900 may be implemented as software (e.g., program codes or instructions) that are stored in a memory or a non-transitory computer-readable medium. In some embodiments, some aspects of the process 300 may be implemented as hardware (e.g., a specific-purpose circuit). In some embodiments, the process 300 may be implemented as a combination of software and hardware.

[00340] Fig. 9 includes process blocks 902-908. At block 902, at least one processor may analyze a cloud environment to identify a plurality of keys to the compute resources in the cloud environment. In some embodiments, the plurality of keys may be stored in at least one workload. In some embodiments, at least one of the plurality of keys may include at least one of a password (e.g., a combination of alphanumeric characters and symbols), a script (e.g., a PYTHON® script or a shell script) containing a password, a cloud key, or an Secure Shell (SSH) key.

[00341] At block 904, the at least one processor may perform a cryptographic analysis on the plurality of keys to identify a first set of fingerprints that uniquely identify each of the plurality of keys. The first set of fingerprints may, in some embodiments, be non-functional. In some embodiments, at least one of the first set of fingerprints may be non-identical to the key.

[00342] At block 906, the at least one processor may analyze trust configurations of the compute resources to identify a second set of fingerprints for each of the compute resources. As discussed above, to identify the second set of fingerprints, in some embodiments, the at least one processor may analyze a trust configuration file (e.g., a file storing fingerprints and their associated compute resources) by reading the byte stream of the trust configuration file to determine whether any sequence of bytes that fits to a predetermined pattern (e.g., an alphanumeric string pattern). The sequences of bytes that fit to the predetermined pattern may be determined as the second set of fingerprints.

[00343] At block 908, the at least one processor may compare the first set of fingerprints with the second set of fingerprints to match keys with the compute resources without using the keys to access the compute resources. As discussed above, in some embodiments, a first compute resource may store a key (e.g., a private component of a private-public key pair) to a second compute resource, and the second compute resource may store in its trust configurations a second fingerprint of the key. The at least one processor may analyze the cloud environment to identify the key from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. The at least one processor may then analyze the trust configurations of the second compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other (e.g., being identical). If the first

fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource.

[00344] Consistent with disclosed embodiments, besides blocks 902-908, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the first set of fingerprints. By way of example, to analyze the multi-machine interaction, the at least one processor may compare the first set of fingerprints with the second set of fingerprints. As discussed above, in some embodiments, a first compute resource may store a key (e.g., a private key) for accessing a second compute resource. The at least one processor may analyze the cloud environment to identify the private component from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. A third compute resource (e.g., may be the same as or different from the second compute resource) may store in its trust configurations a second fingerprint of the key. The at least one processor may then analyze the trust configurations of the third compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other (e.g., being the same). If the first fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource, and may further determine an inter-machine interaction in which the first compute resource has ability to access the second compute resource.

[00345] Consistent with disclosed embodiments, besides blocks 902-908, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the plurality of keys. By way of example, to analyze the multi-machine interaction, the at least one processor may compare a first key for accessing a first one of the compute resources and a second key for accessing a second one of the compute resources.

[00346] **Hybrid Ephemeral Scanner Techniques**

[00347] Aspects of this disclosure may include accessing a primary account maintained in a cloud environment. A primary account as used herein, may refer to a principal or main identity created for a person in a computer or computing system. A primary account may also be created for machine entities, such as service accounts for running programs, system accounts for storing system files and processes, and root and administrator accounts for system administration. A cloud environment, as used herein, may refer to a platform implemented on, hosted on, and/or accessing servers that are accessed over the Internet. An example of a cloud environment is cloud infrastructure 106 in Fig. 1. In some embodiments, scanning system 101 of Fig. 1 may gain access to an account within cloud infrastructure 106.

[00348] Aspects of this disclosure may include receiving information defining a structure of the primary account, wherein the structure includes a plurality of assets.

Information as used herein, may refer to data received. Structure of the primary account as used herein, may refer to a set up or of the primary account. A plurality of assets as used herein, may refer to one or more data, devices, or components within an organization's system. For example, the plurality of assets may include any of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, load balancer 115, log files or databases, API gateway resources, API gateway REST APIs, Autoscaling groups, CloudTrail logs, CloudFront services, volumes, snapshots, VPCs, subnets, route tables, network ACLs, VPC endpoints, NAT gateways, ELB and ALB, ECR repositories, ECS clusters, services, and tasks, EKS, S3 bucket and Glacier storage, SNS topics, IAM roles, policies, groups, users, KMS keys, and Lambda functions.

[00349] In some embodiments, scanning system 101 of Fig. 1 may receive information relating to the structure of an account. In such an embodiment, the structure of the account may include virtual machines 107A-D and databases 109A-109D, for example. In some embodiments, the information may be acquired from another device, received via a one way communication, received in response to a request, retrieved from a storage device, or generated.

[00350] In some embodiments, the information may exclude raw data of the primary account. Raw data as used herein, may refer to a collection of data as gathered before it has been processed, cleaned, or analyzed. For example, raw data may include usage data, passwords, and cache.

[00351] Aspects of this disclosure may include deploying, inside the primary account, at least one ephemeral scanner configured to scan at least one block storage volume and to output metadata defining the at least one block storage volume. In some embodiments, the output may exclude raw data of the primary account. In some embodiments, the ephemeral scanner may be deployed inside a secondary account for which trust has been established with a primary account. For example, a secondary account on another system may have an existing trust relationship (such as a cloud trust policy) with a first account. The ephemeral scanner may be operated inside of that secondary account to scan block storage associated with the first account. Each account may be hosted on the same system or on different systems.

[00352] Further, in some embodiments, the at least one ephemeral scanner may be configured to periodically scan at least one block storage volume. For example, the scanner may be configured to scan at least one block storage volume once per hour, once per day, once per week, or more or less often.

[00353] Further, in some embodiments, the at least one ephemeral scanner may be periodically deployed to scan at least one block storage volume. For example, the scanner may be deployed for scanning at least one block storage volume once per hour, once per day, once per week, or more or less often.

[00354] Outputting metadata as used herein, may refer to producing a set of data that describes and gives information about other data. In some embodiments, scanning system 101 of Fig. 1 may scan a block storage volume and output data related to it (e.g., vulnerability information, configuration information, malware information, risk analysis information, and sensitive information). Periodically scanning as used herein, may refer to conducting a scan occurring or recurring at regular intervals. In some embodiments, the ephemeral scanner is configured to perform vulnerability scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for security vulnerabilities on the device. In some embodiments, the ephemeral scanner is configured to perform configuration scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for configuration information related to the device. In some embodiments, the ephemeral scanner may scan the block storage volume in order to look for security issues. In some embodiments, the ephemeral scanner is configured to perform malware scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for malware information related to the device. As discussed above with respect to Fig. 2D, in step 235, scanning system 101 may perform a step of malware scanning. In some embodiments, scanning system 101 may perform malware scanning across all filesystems in the snapshot (e.g., gathered from virtual machines 107A-107D or storage 111A-111D). Scanning system 101 may use multiple malware scanning software solutions to perform a malware scan against the filesystems, including one sourced from another vendor, such as bucketAV, Trend Micro Cloud One, Sophos Cloud Optix, Crowdstrike Falcon CWP, or others. In some embodiments, malware scanning in step 235 comprises utilizing signatures, heuristics, or sandboxing capabilities to deduce whether there is an infection on the machine.

[00355] In some embodiments, the ephemeral scanner is configured to perform lateral-movement risk analysis of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for lateral-movement risk information related to the device. For example, in some embodiments, scanning system 101 may perform a “backward” analysis of the specific asset to identify exposure risk to assets downstream of the specific asset, wherein the downstream exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset.

[00356] Further, as discussed above with respect to Fig. 2D, in step 237, scanning system 101 may perform a step of lateral movement scanning. An attacker who establishes a network foothold usually attempts to move laterally from one resource to another in search of rich targets such as valuable data. Stolen passwords and keys unlock access to servers, files, and privileged accounts. In some embodiments, scanning system 101 may gather keys from

each scanned system or device (e.g., virtual machines 107A-107D or storage 111A-111D). In some embodiments, scanning system 101 searches for passwords, scripts, shell history, repositories, or other data that may contain passwords, cloud access keys, SSH keys, or other key/password/access information that provide unchecked access to important resources. In some embodiments, scanning system 101 searches for such keys/passwords/access information and calculates a "hash" (a mathematical fingerprint) of each string. Scanning system 101 then attempts to match the hashed strings to hashes of strings that that are stored on different systems or devices. This will be used to detect the potential lateral movement between assets.

[00357] In some embodiments, the ephemeral scanner is configured to perform sensitive information scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for sensitive information related to the device. As discussed above with respect to Fig. 2D, in step 241, scanning system 101 may perform a step of sensitive information scanning. In some embodiments, scanning system 101 may search the snapshot for sensitive information, such as personally identifiable information (PII), Social Security numbers, healthcare information, or credit card numbers. In some embodiments, scanning system 101 may search data repository history as well. This is because it is not uncommon for an entire production environment repository to be cloned, with no one remembering the copy contains sensitive information. In some situations, detecting sensitive data not secured is critical in adherence to data privacy regulations.

[00358] In some embodiments, the ephemeral scanner is configured to perform container scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for container information related to the device. As discussed with respect to Fig. 2D, in step 243, scanning system 101 may perform a step of container scanning. In some embodiments, scanning system 101 may apply one or more of the preceding steps of Fig. 2D against containerized environments. In some embodiments, in order to do so, scanning system 101 reconstructs a container runtime layered file system (LFS) before recursively running one or more of steps 231-241 on the reconstructed file system.

[00359] In some embodiments, the ephemeral scanner is configured to perform keys and password scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for keys and password information related to the device. As discussed above with respect to Fig. 2D, in step 239, scanning system 101 may perform a step of key/password scanning. As one example situation, suppose there is a weak or unprotected password stored (in plain text) in storage 111A. For example, if a personal email account has been compromised, the passwords may be known about in advance. Scanning system 101 may search the snapshot for similar usernames or login names, and, either using known dictionaries or the account owner's previously leaked

passwords (stored in, e.g., database 103A), may attempt to login to one or more systems or devices in cloud infrastructure 106, and may record the result thereof.

[00360] In some embodiments, defining the at least one block storage volume includes presenting risk data without sharing consumer data or data that was used to identify the risk data. Risk data as used herein, may refer to information relating to causing harm. Consumer data as used herein, may refer to a user's personal information or any information trail user's leave behind as a result of their computer or Internet use. Data used to identify the risk as used herein, may refer to information for determining possible issues. In some embodiments, scanning system 101 of Fig. 1 may present information without sharing a user's personal information or any information trail user's leave behind as a result of their computer or Internet use.

[00361] In some embodiments, the metadata defining the at least one block storage volume includes at least one of: an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, or a profile configuration. An indication of an installed application as used herein, may refer to a sign or signal of a stored program. A version of an installed application as used herein, may refer to an older or newer form of a stored program. An operating system configuration as used herein, may refer to one or more computer system settings that have been set by default automatically or manually by a given program or the user. An application configuration as used herein, may refer to one or more computer program settings that have been set by default automatically or manually by a given program or the user. A profile configuration as used herein, may refer to one or more computer program file settings that have been set by default automatically or manually by a given program or the user.

[00362] Aspects of this disclosure may include receiving a transmission of the metadata from the at least one ephemeral scanner. In some embodiments, the transmission may exclude raw data of the primary account. A transmission of metadata as used herein, may refer to movement of a set of data that describes and gives information about other data. In some embodiments, scanning system 101 of Fig. 1 may receive data excluding any raw data (e.g., data as gathered before it has been processed, cleaned or analyzed).

[00363] Aspects of this disclosure may include analyzing the received metadata to identify a plurality of cybersecurity vulnerabilities. A plurality of cybersecurity vulnerabilities as used herein, may refer to a weakness in a system. For example, a cybersecurity vulnerability may be exploited by cybercriminals to gain unauthorized access to a computer system. In some embodiments, scanning system 101 of Fig. 1 may analyze the received data in order to identify any weakness in a system (e.g., to avoid cybercriminals gaining unauthorized access to a computer system). In some embodiments, vulnerability information may include information stored in a system.

[00364] Aspects of this disclosure may include correlating each of the identified plurality of cybersecurity vulnerabilities with one of the plurality of assets. In some embodiments, scanning system 101 of Fig. 1 may connect a security risk with an asset in the cloud environment. In such embodiments, the connection or correlation of vulnerability to asset may allow the system to address the vulnerability.

[00365] Aspects of this disclosure may include generating a report correlating the plurality of cybersecurity vulnerabilities with the plurality of assets. A report as used herein, may refer to a document containing information. For example, a report may notify the administrator of a website or application about a problem such as a security issue or vulnerability in the system that should be addressed. In some embodiments, scanning system 101 of Fig. 1 may create a report providing information related to the vulnerability and asset. In such an embodiment, the report may allow for the vulnerability to be addressed and resolved.

[00366] In such embodiments, the system may allow scanners (e.g., ephemeral scanners, scanning system 101) to run inside an account, including an Amazon Web Services, Azure, or GCP account. In this mode, a cloud cybersecurity service may generate ephemeral scanners inside an account that performs the same actions as a SAAS node but are logically hosted inside the account.

[00367] In some embodiments, the at least one processor is further configured to receive a transmission of updated metadata defining the at least one block storage volume in response to at least one change to the at least one block storage volumes. Updated metadata as used herein, may refer to a renewed set of data that describes and gives information about other data.

[00368] **Fig. 10** is a block diagram of a method 1000 for deployment of ephemeral scanners, consistent with disclosed embodiments. In some embodiments, the method may include seven (or more or less) steps:

[00369] Block 1002: Access a primary account maintained in a cloud environment. In some embodiments, scanning system 101 of Fig. 1 may gain access to an account within cloud infrastructure 106 using a password stored in a database, for example.

[00370] Block 1004: Receiving information defining a structure of the primary account, wherein the structure includes a plurality of assets. In some embodiments, the information may exclude raw data of the primary account. In some embodiments, scanning system 101 of Fig. 1 may receive data relating to the structure of an account. In such an embodiment, the structure of the account may include virtual machines 107A-D and databases 109A-109D, for example.

[00371] Block 1006: Deploying, inside the primary account, at least one ephemeral scanner configured to periodically scan at least one block storage volume and to output metadata defining the at least one block storage volume. In some embodiments, the output may exclude raw data of the primary account. In some embodiments, scanning system 101 of Fig. 1

may scan a block storage volume and output data related to it (e.g., vulnerability information, configuration information, malware information, risk analysis information, and sensitive information).

[00372] Block 1008: Receiving a transmission of the metadata from the at least one ephemeral scanner. In some embodiments, the transmission may exclude raw data of the primary account. In some embodiments, scanning system 101 of Fig. 1 may receive data excluding any raw data (e.g., data as gathered before it has been processed, cleaned or analyzed).

[00373] Block 1010: Analyzing the received metadata to identify a plurality of cybersecurity vulnerabilities. For example, a cybersecurity vulnerability may be exploited by cybercriminals to gain unauthorized access to a computer system. In some embodiments, scanning system 101 of Fig. 1 may analyze the received data in order to identify any weakness in a system (e.g., to avoid cybercriminals gaining unauthorized access to a computer system).

[00374] Block 1012: Correlating each of the identified plurality of cybersecurity vulnerabilities with one of the plurality of assets. In some embodiments, scanning system 101 of Fig. 1 may connect a security risk with an asset in the cloud environment. In such embodiments, the connection or correlation of vulnerability to asset may allow the system to address the vulnerability.

[00375] Block 1014: Generating a report correlating the plurality of cybersecurity vulnerabilities with the plurality of assets. For example, a report may notify the administrator of a website or application about a problem such as a security issue or vulnerability in the system that should be addressed. In some embodiments, scanning system 101 of Fig. 1 may create a report providing information related to the vulnerability and asset. In such an embodiment, the report may allow for the vulnerability to be addressed and resolved.

[00376] **Risk Information Aggregation Techniques**

[00377] **Fig. 11** represents a schematic block diagram 1100 illustrating an exemplary embodiment of a method for providing a dashboard aggregating risk information.

[00378] Assets operating in a cloud environment face myriad cybersecurity risks that vary in type and nature of threat with increasing regularity. In order to combine the varied and multiple cybersecurity risks an asset may encounter in a cloud environment into a single view to allow for faster and more flexible processing, prioritization and mitigation, a system and method of aggregating cybersecurity risks into a single dashboard is needed. Aggregation of multiple cybersecurity risks into a single view for an administrator is needed. In one of many embodiments, an exemplary embodiment may include a graphical user interface system for providing comprehensive cloud environment risk inventory visualization. A graphical user interface system may include systems similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows

10 Mobile, PalmOS-Web OS, or Firefox OS. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of risk inventories. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D.

[00379] An exemplary embodiment may also include at least one processor. The at least one processor may be part of a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations.

[00380] In an embodiment may include a processor configured to cause a display to present a plurality of asset categories (step 1101). A display may be graphical, sequential, or multi-dimensional. A plurality may include at least one asset category to be displayed. Asset categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category.

[00381] An account category may further include listing of assets based on type of account associated with that user. A type of account may include active user account, passive user account, administrator account, maintainer account, a system account, superuser account, or a guest user account.

[00382] A container category may include one or more containers currently listed in the system. A container may involve specific versions of programming language runtimes, libraries required to run software, or another method of packaging applications abstracted from the cloud environment.

[00383] A database category may include one or more databases managed and maintained in relation to the system. Databases may include a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database.

Databases may be implemented using ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[00384] An image category may contain one or more images based on image type, size, and content. Images may include images of one or more virtual machines.

[00385] A managed service category may include one or more managed services listed in the above system, to include at least client or customer-owned systems being managed by a third party entity.

[00386] A messaging service category may include systems/programs communicating with other systems/programs using services such as Google Cloud Pub and Sub communications, AWS SQS, or similar queues found in Information Systems.

[00387] A monitoring category may include centralized or decentralized methods of security monitoring systems.

[00388] A network category may include one or more systems of establishing a form of communicative connectivity between systems, e.g., TCP-IP services.

[00389] A storage category may list one or more methods of data storage, to include Direct Attached Storage, Network Attached Storage, SSD Flash Drive Arrays, Hybrid Flash Arrays, Hybrid Cloud Storage, Backup Software, Backup Appliances, Cloud Storage, or similar. Storage may include data structures, instructions, or any other data to be contained in a storage medium.

[00390] A user category may include one or more users of a given system along with identifying information relating to its unique or semi-unique identifier, level of authorized access and permissions, and storage volumes accessible by said user.

[00391] An access category may include one or more levels of access permission and a listing of users or devices granted access at a given level of access permission to a system.

[00392] A virtual machine category may list one or more virtual machines accessible to a system and the levels and types of access granted to a given virtual machine.

[00393] A serverless category may list one or more storage volumes stored on a serverless computing system such as Kubeless, Fission, Google App Engine, AWS Lambda, Google Cloud Functions, IBM Cloud Function, IBM Cloud Code Engine, Microsoft Azure Functions, Cloudflare Workers, and Compute@Edge.

[00394] In an embodiment, an exemplar may include a processor configured to receive, via an input device, a selection of a particular asset category (step 1103). An input device may include a personal computer, tablet, smartphone, or virtual machine. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing

logic operations. A selection of one or more asset categories may be received and updated as new visualizations are developed based on the current threats to an asset in that category. Asset categories may be selected or deselected by an end user. Once a particular asset category is selected, it may be added to a list of selected asset categories with a list for deselected categories as well. The list of selected asset categories may organized per category or may be compiled to include all current threats in a comprehensive selected asset threat list.

[00395] One exemplary embodiment may further include a processor configured to cause the display to present a list of assets in the selected category that have cyber security risks (step 1105). When an asset in a selected category has a cyber security risk associated with it, the asset may appear in a list of all listed categories having a cyber security risk associated with it, and may be organized by individual category, type of cyber security risk, or some combination or sequential ordering methods based on the information available to the processor and its end user.

[00396] In one of many exemplary embodiments a processor may be configured to retrieve workload component cybersecurity risk information (step 1107A) for each listed asset. When a category of assets is selected, a processor may be configured to order those assets according to available cybersecurity risk information available to said processor, and may organize by type of risk, type of asset, or similar type of sequentialization. Cybersecurity risk information may be determined from historical threats to the workload component, the current active cybersecurity threats to workload components of certain types, or some combination of the same.

[00397] In an embodiment, a processor may be configured to retrieve cloud component cybersecurity risk information (step 1107B) for each listed asset. Cloud component cybersecurity risk information may include identification of the risk based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset. Cybersecurity risk information to a cloud component may be determined from historical threats within the cloud environment 106, current active cybersecurity threats to similar workload components in the cloud environment 106 or the level of risk to assets within the cloud environment 106, as outlined below.

[00398] An exemplary embodiment may include a system configured to cause the display to present a common interface (step 1109) providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information. The common interface may be configured to access component cybersecurity risk information based on the nature of the risk, the likelihood of the risk, or the severity of the risk to the component, including recursive risk – risk that can happen from an attacker getting access to a system and using it as a mean to reach other system. On the same interface, a different interface, or a similar and related interface, a display may present cloud component cybersecurity risk information based

on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset.

[00399] As an exemplary embodiment, the system may be configured to cause the display to present, in the common interface, an interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information (step 1111). With this display presented, an exemplary embodiment may allow for a user to prioritize data from additional sources. The interconnection between workload component cybersecurity risk information and cloud component cybersecurity risk information may generate an interface between them and may be presented as an interconnection between them. The common interface presented may include direct interconnections between the component cybersecurity risk information and the cloud component cybersecurity risk information or links from one set of cybersecurity risk information. For instance, the component cybersecurity risk information may include a link to access the cloud component cybersecurity risk information. Conversely, the cloud component cybersecurity risk information may include a link to access the component cybersecurity risk information.

[00400] In one embodiment, a system may be configured to provide the plurality of asset categories include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. A category may list the currently identified entities of that category type as described below. An account category may include one or more listed accounts of users of the system. An account category may further include listing of assets based on type of account associated with that user. A type of account may include active user account, passive user account, administrator account, maintainer account, a system account, superuser account, or a guest user account.

[00401] A container category may include one or more containers currently listed in the system. A container may involve specific versions of programming language runtimes, libraries required to run software, or another method of packaging applications abstracted from the cloud environment.

[00402] A database category may include one or more databases managed and maintained in relation to the system. Databases may include a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. Databases may be implemented using ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[00403] An image category may contain one or more images based on image type, size, and content. Images may include images of one or more virtual machines displayed. A

managed service category may include one or more managed services listed in the above system, to include at least client or customer-owned systems being managed by a third party entity.

[00404] A messaging service category may include services of users communicating with at least one other user such as Google Cloud Pub and Sub communications, AWS SQS, or similar queues found in Information Systems.

[00405] A monitoring category may include centralized or decentralized methods of security monitoring systems.

[00406] A network category may include one or more systems of establishing a form of communicative connectivity between systems, e.g., TCP-IP services.

[00407] A storage category may list one or more methods of data storage, to include Direct Attached Storage, Network Attached Storage, SSD Flash Drive Arrays, Hybrid Flash Arrays, Hybrid Cloud Storage, Backup Software, Backup Appliances, Cloud Storage, or similar. Storage may include data structures, instructions, or any other data to be contained in a storage medium.

[00408] A user category may include one or more users of a given system along with identifying information relating to its unique or semi-unique identifier, level of authorized access and permissions, and storage volumes accessible by said user.

[00409] An access category may include one or more levels of access permission and a listing of users or devices granted access at a given level of access permission to a system.

[00410] A virtual machine category may list one or more virtual machines accessible to a system and the levels and types of access granted to a given virtual machine.

[00411] A serverless category may list one or more storage volumes stored on a serverless computing system such as Kubeless, Fission, Google App Engine, AWS Lambda, Google Cloud Functions, IBM Cloud Function, IBM Cloud Code Engine, Microsoft Azure Functions, Cloudflare Workers, and Compute@Edge.

[00412] In one of many exemplary embodiments, a system may further include the common interface configured to display information relating to at least one of an asset type, a risk, a region, or an account. A common interface may include the ability to configure and reconfigure the asset type, risk, region, or account to appear in any order and at multiple formats for displaying the information relating to those categories.

[00413] In another embodiment, a system may provide the common interface configured to display description for each listed asset. The display of a description for each listed asset may be listed vertically or horizontally. The display may be displayed as written text, representative graphical figures, or similar method of displaying the aforementioned types of information for an asset.

[00414] In another embodiment, a system may further provide the common interface is configured to display at least one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, or an authentication. A vulnerability may include vulnerability data from a vulnerability database such as NVD, WPVulnDB, US-CERT, Node.js Security Working Group, OVAL – Red Hat, Oracle Linux, Debian, Ubuntu, SUSE, Ruby Advisory Database, JVN, Safety DB(Python), Alpine secdb, PHP Security Advisories Database, Amazon ALAS, RustSec Advisory Database, Red Hat Security Advisories, Microsoft MSRC, KB, Debian Security Bug Tracker, Kubernetes security announcements, Exploit Database, Drupal security advisories, JPCERT. An insecure configuration may include software flaws or misconfigurations, non-encrypted files, improper file or directory permissions, unpatched security flaws in server software, enabled or accessible administrative and debugging functions, administrative account vulnerabilities, SSL certificates and encryption settings not properly configured, or a similar misconfiguration. These misconfigurations may be discovered by performing a scan, e.g., may query devices and systems capable of routing traffic (e.g., load balancer 115, routers, switches, firewalls, and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations. Malware indicated may include Adware, Botnets, Cryptojacking, Malvertising, Ransomware, Remote Administration Tools (RATs), Rootkits, Spyware, Trojans, Virus Malware, Worm Malware, or similar attack vehicles. A neglected asset may include an asset that has been improperly maintained, patched, or similar cybersecurity security measure. A data at risk may include any packet of data that may be exposed to a cybersecurity threat. A lateral movement may include any pathway between assets where a cybersecurity risk can travel from one affected asset to another affected asset. An authentication may include a username, password, one-time password, two-factor authentication, or any other authentication mechanism to gain access to a cloud service provider's system. A common interface may display more than one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, and an authentication separately or concurrently for an asset. A display of said information may be ordered and listed in any order depending on user preference and input.

[00415] An exemplary embodiment may further include a system wherein the common interface is configured to display one or more possible attack vectors reaching the each listed asset. Generating this common interface display may include generating a map of an asset to include recording information such as a region identifier, site identifier, datacenter identifier, physical address, network address, workload name, or any other identifier which may be acquired via an API provided through a cloud service provider's system and demonstrating via a display the myriad attack vectors currently available to an analyzed asset. Display of the one or

more possible attack vectors reaching the each listed asset may be displayed as written text, representative graphical figures, or similar method of displaying the one or more possible attack vectors.

[00416] Another exemplary embodiment may further include a system wherein the common interface is configured to display a recommended mitigation tactic for the each listed asset. A recommended mitigation tactic may be generated based on historical data of the cybersecurity risk, the vulnerability attributed to the listed asset, or the data contained on the each listed asset. Mitigation tactics may be selected based on user input or may be implemented automatically based on the mitigation tactics likelihood of a successful mitigation of the cybersecurity risk. A mitigation tactic may include increased frequency of scanning, heightened access control measures, firewalls and antivirus software patches, increased patch management scheduling, continuous workload monitoring, or similar tactics.

[00417] In another exemplary embodiment is a system wherein the common interface is configured to display one or more workload metrics associated with the each listed asset. Workload metrics may be a statistical representation of the current performance of the workload, the capacity for additional processing of the workload, and the historical performance of the workload based on statistical trend.

[00418] Another exemplary embodiment may include a system wherein the at least one processor is further configured to provide a cybersecurity report for the each listed asset. A cybersecurity report may be generated by scanning system 101 following a complete scan of an analyzed asset, and may include a snapshot of one or more of the number of total cybersecurity threats to a listed asset, number of cybersecurity threats mitigated by recent performance, methods of cybersecurity risk mitigation in the listed asset, rate of cybersecurity threats detected in the listed asset, and possibility of failed cybersecurity mitigation tactics based on the frequency and nature of the historical cybersecurity threats. Data collected to generate a cybersecurity report may include, among many things, operating system packages, installed software applications, libraries, and program language libraries such as Java archives, Python packages, Go modules, Ruby gems, PHP packages, and Node.js modules, other software applications, library versions, software versions, and other identifying characteristics of software and operating systems, lists of users of each system or device (e.g., virtual machines 107A-D), each system's or device's services, password hashes, and application-specific configurations for software/services such as Apache, Nginx, SSH, and other services, bugs or other configuration risks, malware scan results, passwords, scripts, shell history, repositories, or other data that may contain passwords, cloud access keys, SSH keys, or other key/password/access information that provide unchecked access to important resources.

[00419] In another embodiment, a system may include the at least one processor is configured to identify a risk level distribution among the listed assets. The at least one processor

may be configured to list risk level distribution by highest level of risk, lowest level of risk, or largest number of cybersecurity risks attributed among the listed assets. A risk level may be high if the web service is connected to the Internet (e.g., has at least one public port forwarded through a firewall to the web service, is able to be accessed through a load balancer, is able to be accessed via reverse proxy). A risk level may be medium risk if the web service is only accessible internally (e.g., because of firewall configuration). A risk level may be low risk if access to the web server is blocked by a configuration of cloud infrastructure 106.

[00420] Another exemplary embodiment may include a system wherein the at least one processor is configured to receive a search query for a specific risk. A search query may be input by type of risk, name of a risk, or another semi-unique manner of identifying a specific cybersecurity risk. A search may be initiated by end user operating user device 102, by a maintainer or administrator of an analyzed asset, or by a virtual machine associated with operating scanning system 101, A search may be initiated by inputting a number of inputs, to include keywords, phrases, file type, or similar identifiers unique or semi-unique to a given cybersecurity risk. Upon initiating a search, the results generated by the search may be stored in the analyzed asset, transmitted to the operator of a processor running side scanning system 101, or generated for review by an end user operating user device 102. The search and display of a specific risk may be based on historical trends of cybersecurity risks in the listed asset, severity of cybersecurity risk to a listed asset, or a historical search for past cybersecurity threats to a listed asset.

[00421] Another embodiment may include a system wherein the at least one processor is configured to identify one or more assets vulnerable to the specific risk. For instance, a processor operating scanning system 101 may be configured to perform a scan of a set of one or more assets with potential exposure to a given risk, and upon identifying vulnerable pathways for an attack vector to reach a set of identified assets, may generate a list of assets that are potentially or currently exposed to one or more specific risk. Assets vulnerable to a specific risk may be determined to be vulnerable based on the type of the specific risk, the existence of specific risks in adjacent or similar workloads, and related risks that may also cause an asset to be vulnerable based on the specific risk determined.

[00422] In another embodiment is a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing a display to present a plurality of asset categories (step 1101). The method may include the use of a graphical user interface system and may include systems similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows 10 Mobile, PalmOS-Web OS, or Firefox OS. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and

the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of risk inventories. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D.

[00423] One of several embodiments may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising receiving, via an input device, a selection of a particular asset category. A method may include a personal computer, tablet, smartphone, or virtual machine. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations. A selection of one or more asset categories may be received and updated as new visualizations are developed based on the current threats to an asset in that category. Asset categories may be selected or deselected by an end user. Once a particular asset category is selected, it may be added to a list of selected asset categories with a list for deselected categories as well. The list of selected asset categories may organized per category or may be compiled to include all current threats in a comprehensive selected asset threat list.

[00424] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing the display to present a list of assets in the selected category that have cybersecurity risks (step 1105). A display may be graphical, sequential, or multi-dimensional. A list of selected categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. When an asset in a selected category has a cyber security risk associated with it, the asset may appear in a list of all listed categories having a cyber security risk associated with it, and may be organized by individual category, type of cyber security risk, or some combination or sequential ordering methods based on the information available to the processor and its end user.

[00425] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising retrieving workload component cybersecurity risk information 307A for each listed asset. When a category

of assets is selected, a processor may be configured to order those assets according to available cybersecurity risk information available to said processor, and may organize by type of risk, type of asset, or similar type of sequentialization. Cybersecurity risk information may be determined from historical threats to the workload component, the current active cybersecurity threats to workload components of certain types, or some combination of the same.

[00426] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising retrieving cloud component cybersecurity risk information 307B. Cloud component cybersecurity risk information may include identification of the risk based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset. Cybersecurity risk information to a cloud component may be determined from historical threats within the cloud environment 106, current active cybersecurity threats to similar workload components in the cloud environment 106 or the level of risk to assets within the cloud environment 106, as outlined above.

[00427] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing the display to present a common interface 309 providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information. The common interface may be configured to access component cybersecurity risk information based on the nature of the risk, the likelihood of the risk, or the severity of the risk to the component. On the same or a similar and related interface, a display may present cloud component cybersecurity risk information based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset.

[00428] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing the display to present, in the common interface, an interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information. The interconnection between workload component cybersecurity risk information 307A and cloud component cybersecurity risk information 307B may generate an interface between them 309 and may be presented as an interconnection between them 311. The common interface presented may include direct interconnections between the component cybersecurity risk information and the cloud component cybersecurity risk information or links from one set of cybersecurity risk information. For instance, the component cybersecurity risk information may include a link to access the cloud component cybersecurity risk information. Conversely, the cloud component cybersecurity risk information may include a link to access the component cybersecurity risk information.

[00429] In another embodiment, a method may include wherein the plurality of asset categories include at least one of an account category, an authentication category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. A category may list the currently identified entities of that category type as described below. An account category may include one or more listed accounts of users of the system. An account category may further include listing of assets based on type of account associated with that user. A type of account may include active user account, passive user account, administrator account, maintainer account, a system account, superuser account, or a guest user account.

[00430] A container category may include one or more containers currently listed in the system. A container may involve specific versions of programming language runtimes, libraries required to run software, or another method of packaging applications abstracted from the cloud environment.

[00431] A database category may include one or more databases managed and maintained in relation to the system. Databases may include a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. Databases may be implemented using ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[00432] An image category may contain one or more images based on image type, size, and content. Images may include images of one or more virtual machines displayed.

[00433] A managed service category may include one or more managed services listed in the above system, to include at least client or customer-owned systems being managed by a third party entity.

[00434] A messaging service category may include services of users communicating with at least one other user such as Google Cloud Pub and Sub communications, AWS SQS, or similar queues found in Information Systems.

[00435] A monitoring category may include centralized or decentralized methods of security monitoring systems.

[00436] A network category may include one or more systems of establishing a form of communicative connectivity between systems, e.g., TCP-IP services.

[00437] A storage category may list one or more methods of data storage, to include Direct Attached Storage, Network Attached Storage, SSD Flash Drive Arrays, Hybrid Flash Arrays, Hybrid Cloud Storage, Backup Software, Backup Appliances, Cloud Storage, or similar.

Storage may include data structures, instructions, or any other data to be contained in a storage medium.

[00438] A user category may include one or more users of a given system along with identifying information relating to its unique or semi-unique identifier, level of authorized access and permissions, and storage volumes accessible by said user.

[00439] An access category may include one or more levels of access permission and a listing of users or devices granted access at a given level of access permission to a system.

[00440] A virtual machine category may list one or more virtual machines accessible to a system and the levels and types of access granted to a given virtual machine.

[00441] Another embodiment may include a method wherein the common interface is configured to display information relating to at least one of an asset type, a risk, a region, or an account. A common interface may include the ability to configure and reconfigure the asset type, risk, region, or account to appear in any order and at multiple formats for displaying the information relating to those categories.

[00442] Another embodiment may include a method wherein the common interface is configured to display description for each listed asset. The display of a description for each listed asset may be listed vertically or horizontally. The display may be displayed as written text, representative graphical figures, or similar method of displaying the aforementioned types of information for an asset.

[00443] Another embodiment may include a method wherein the common interface is configured to display at least one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, or an authentication. A vulnerability may include vulnerability data from a vulnerability database such as NVD, WPVulnDB, US-CERT, Node.js Security Working Group, OVAL – Red Hat, Oracle Linux, Debian, Ubuntu, SUSE, Ruby Advisory Database, JVN, Safety DB(Python), Alpine secdb, PHP Security Advisories Database, Amazon ALAS, RustSec Advisory Database, Red Hat Security Advisories, Microsoft MSRC, KB, Debian Security Bug Tracker, Kubernetes security announcements, Exploit Database, Drupal security advisories, JPCERT. An insecure configuration may include software flaws or misconfigurations, non-encrypted files, improper file or directory permissions, unpatched security flaws in server software, enabled or accessible administrative and debugging functions, administrative account vulnerabilities, SSL certificates and encryption settings not properly configured, or a similar misconfiguration. These misconfigurations may be discovered by performing a scan, e.g., may query devices and systems capable of routing traffic (e.g., load balancer 115, routers, switches, firewalls, and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations. Malware indicated may include Adware, Botnets, Cryptojacking, Malvertising,

Ransomware, Remote Administration Tools (RATs), Rootkits, Spyware, Trojans, Virus Malware, Worm Malware, or similar attack vehicles. A neglected asset may include an asset that has been improperly maintained, patched, or similar cybersecurity security measure. A data at risk may include any packet of data that may be exposed to a cybersecurity threat. A lateral movement may include any pathway between assets where a cybersecurity risk can travel from one affected asset to another affected asset. An authentication may include a username, password, one-time password, two-factor authentication, or any other authentication mechanism to gain access to a cloud service provider's system. A common interface may display more than one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, and an authentication separately or concurrently for an asset. A display of said information may be ordered and listed in any order depending on user preference and input.

[00444] Another embodiment may include a method wherein the common interface is configured to display one or more possible attack vectors reaching the each listed asset. Generating this common interface display may include generating a map of an asset to include recording information such as a region identifier, site identifier, datacenter identifier, physical address, network address, workload name, or any other identifier which may be acquired via an API provided through a cloud service provider's system and demonstrating via a display the myriad attack vectors currently available to an analyzed asset. Display of the one or more possible attack vectors reaching the each listed asset may be displayed as written text, representative graphical figures, or similar metho of displaying the one or more possible attack vectors.

[00445] Another embodiment may include a method wherein the common interface is configured to display a recommended mitigation tactic for the each listed asset. A recommended mitigation tactic may be generated based on historical data of the cybersecurity risk, the vulnerability attributed to the listed asset, or the data contained on the each listed asset. Mitigation tactics may be selected based on user input or may be implemented automatically based on the mitigation tactics likelihood of a successful mitigation of the cybersecurity risk. A mitigation tactic may include increased frequency of scanning, heightened access control measures, firewalls and antivirus software patches, increased patch management scheduling, continuous workload monitoring, or similar tactics.

[00446] Another embodiment may include a method wherein the common interface is configured to display one or more workload metrics associated with the each listed asset. Workload metrics may be a statistical representation of the current performance of the workload, the capacity for additional processing of the workload, and the historical performance of the workload based on statistical trend.

[00447] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing a display to present a plurality of asset categories 301. A plurality may include at least one asset category to be displayed. Asset categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category.

[00448] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising receiving, via an input device, a selection of a particular asset category (step 1103). An input device may include a personal computer, tablet, smartphone, or virtual machine. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations. A selection of one or more asset categories may be received and updated as new visualizations are developed based on the current threats to an asset in that category. Asset categories may be selected or deselected by an end user. Once a particular asset category is selected, it may be added to a list of selected asset categories with a list for deselected categories as well. The list of selected asset categories may organized per category or may be compiled to include all current threats in a comprehensive selected asset threat list.

[00449] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing the display to present a list of assets in the selected category that have cybersecurity risks (step 1105). A display may be graphical, sequential, or multi-dimensional. A list of selected categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. When an asset in a selected category has a cyber security risk associated with it, the asset may appear in a list of all listed categories having a cyber security risk associated with it, and may be organized by individual category, type of cyber security risk, or

some combination or sequential ordering methods based on the information available to the processor and its end user.

[00450] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising retrieving workload component cybersecurity risk information for each listed asset (step 1107A). When a category of assets is selected, a processor may be configured to order those assets according to available cybersecurity risk information available to said processor, and may organize by type of risk, type of asset, or similar type of sequentialization. Cybersecurity risk information may be determined from historical threats to the workload component, the current active cybersecurity threats to workload components of certain types, or some combination of the same.

[00451] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising retrieving cloud component cybersecurity risk information for each listed asset (step 1107B). Cloud component cybersecurity risk information may include identification of the risk based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset. Cybersecurity risk information to a cloud component may be determined from historical threats within the cloud environment 106, current active cybersecurity threats to similar workload components in the cloud environment 106 or the level of risk to assets within the cloud environment 106, as outlined above.

[00452] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing the display to present a common interface providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information for each listed asset (step 1109). The common interface may be configured to access component cybersecurity risk information based on the nature of the risk, the likelihood of the risk, or the severity of the risk to the component. On the same or a similar and related interface, a display may present cloud component cybersecurity risk information based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset.

[00453] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing the display to present, in the common interface, an

interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information (step 1111). The interconnection between workload component cybersecurity risk information (step 1107A) and cloud component cybersecurity risk information (step 1107B) may generate an interface between them (step 1109) and may be presented as an interconnection between them (step 1111). The common interface presented may include direct interconnections between the component cybersecurity risk information and the cloud component cybersecurity risk information or links from one set of cybersecurity risk information. For instance, the component cybersecurity risk information may include a link to access the cloud component cybersecurity risk information. Conversely, the cloud component cybersecurity risk information may include a link to access the component cybersecurity risk information.

[00454] In some embodiments, a processor may cause a display of one or more risks aggregated in the same visualization. **Fig. 12** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of displaying aggregated cybersecurity risk information, consistent with disclosed embodiments. A visualization may be generated from one or more sources based on scanning system 101. For example, a visual display may include a cloud provider visualization 1201, displaying where the multiple sources of data may be derived from. A cloud service provider may include services such as Cloudflare, Amazon Web Services, Google Cloud, IBM Cloud, Oracle Cloud, Microsoft Azure, or similar. This is further displayed via a communicative connection through an access port 1203 to a data server 1205.

[00455] Data server 1205 may include servers such as 109A-D. Data server 1205 may then display risk information that may be communicatively passed through web server 1207, demonstrating one or more possible sources of the one or more cybersecurity risk information. Data server 1205 may be communicatively connected to a web server 1207 (through access port 1203, which may be the same or a different port from port 1203 between cloud service provider 1201 and data server 1205), which can then be communicatively connected to one or more users via user access points 1209. This may include, among other things, user access that may be verified or validated via a credential 1211. Each of these connections may demonstrate one or more risk paths that may be exploited in a single view to allow for prioritization of threats and analysis of possible users who may exploit those risks and vulnerabilities, as discussed above. This may allow an administrator to prioritize risks to a given storage volume as well as assess lateral storage volumes that may be similarly risked based on the assessed threats in a single image.

[00456] **Fig. 13** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of a flow path for aggregated risk information, consistent with disclosed embodiments. In another embodiment, a processor may be configured to display a flow path of aggregated risk information. In one view, a displayed visualization may include user access

1301, which displays risk information such as service vulnerability 1303 and an insecure private key 1305. These vulnerabilities or risks may be displayed to an administrator via developer server 1307 to an administrator through administrator access 1309. As previously discussed, this will allow an administrator to visualize myriad risks to a storage volume in a single view that will allow an administrator to prioritize risks and analyze possible exploiters of this risk information on lateral systems and storage volumes.

[00457] In another embodiment, risk information may be shown in one or more views. This may involve a risk path similar to the exemplary embodiment of Fig. 13. In an exemplary embodiment, in a single visualization data from multiple sources may be displayed. This may involve identifying risk information such as the origin of the risk and potential exploiters of a risk. In some embodiments, this visualization may be done for a specific risk. In another embodiment, this visualization may be done for a combination of risks.

[00458] Disclosed embodiments may include any one of the following bullet-pointed features alone or in combination with one or more other bullet-pointed features, whether implemented as a method, by at least one processor, and/or stored as executable instructions on non transitory computer readable media:

- establishing a trusted relationship between a source account in a cloud environment and a scanner account;
- using the established trust relationship, utilize at least one cloud provider API to identify workloads in the source account;
- using the at least one cloud provider API to query a geographical location of at least one of the identified workloads;
- receiving an identification of the geographic location;
- using the cloud provider APIs to access block storage volumes of the at least one workload;
- determining a file-system of the at least one workload;
- mounting the block storage volumes on a scanner based on the determined file-system;
- activating a scanner at the geographic location;
- reconstructing from the block storage volumes a state of the workload; and
- assessing the reconstructed state of the workload to extract insights.
- wherein the geographic location includes an identifier of a physical site.
- wherein mounting includes selecting a driver corresponding to the determined file system.
- wherein the at least one processor is further configured to deploy a scanner at the geographical location.

- wherein the identification of the geographic location comprises an identification of a data center, at least one of a data center name, Internet Protocol (IP) address, name of the cloud provider, or a unique identity.
- wherein the reconstructed state of the workload includes at least two of an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content;
- wherein the at least one processor is further configured to update the reconstructed state of the workload based on at least one change to the block storage volumes.
- wherein the insights comprise at least one of a vulnerability associated with the workload or a composition of installed applications associated with the workload.
- wherein to mount the block storage volumes on the scanner, the at least one processor is configured to create a snapshot of the block storage volumes;
- wherein to mount the block storage volumes on the scanner, the at least one processor is configured to mount the snapshot of the block storage volumes on the scanner
- wherein the at least one processor is further configured to encrypt the snapshot of the block storage volumes;
- wherein the at least one processor is further configured to mount the encrypted snapshot of the block storage volumes on the scanner;
- using a cloud provider API, access a block storage volume of a workload maintained in a cloud storage environment;
- identifying an installed software application in the accessed block storage volume;
- analyzing the identified installed software application to determine an associated software version;
- accessing a data structure of known software vulnerabilities for a plurality of versions of software applications;
- performing a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and
- performing at least one of query the cloud provider API to determine network accessibility information related to the workload, identify at least one port on which the vulnerable application is accessible, use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.
- implementing a remedial action in response to the identified one or more vulnerabilities.
- wherein the remedial measure includes transmitting an alert to a device associated with an administrator.

- wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises examining data sources associated with the workload
- wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises determining the network accessibility information based on the examined data sources.
- wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.
- identifying the installed software application comprises extracting data from at least one of operating system packages, libraries, or program language libraries;
- identifying the installed software application comprises identifying the installed software application based on the extracted data;
- identifying a version of the installed software application;
- wherein the identified installed software application includes one or more scripts;
- wherein the data structure includes aggregated vulnerability data;
- wherein the aggregated vulnerability data includes data from one or more third-party vendors;
- wherein the aggregated vulnerability data includes data collected by a scanner;
- wherein the aggregated vulnerability data includes at least one of an advisory, an exploit, a security announcement, or a known bug.
- querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.
- accessing at least one cloud provider API to determine a plurality of entities capable of routing traffic in a virtual cloud environment associated with a target account containing the workload;
- querying the at least one cloud provider API to determine at least one networking configuration of the entities;
- building a graph connecting the plurality of entities based on the networking configuration;
- accessing a data structure identifying services publicly accessible via the internet and capable of serving as an internet proxy;
- integrating the identified services into the graph;

- traversing the graph to identify at least one source originating via the internet and reaching the workload; and
- outputting a risk notification associated with the workload.
- wherein the plurality of entities includes a virtual network appliance.
- wherein the virtual network appliance is at least one of a load balancer, a firewall, a proxy, or a router.
- wherein the networking configuration is at least one of a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration.
- wherein the graph includes a data structure sequentially connecting entities.
- wherein the graph includes directional vectors indicating directions of dataflow.
- wherein building the graph comprises identifying individual entities as nodes
- wherein building the graph comprises connecting the nodes.
- wherein the graph includes port numbers associated with the workload.
- wherein the graph includes a path from the at least one source to the workload.
- utilizing a cloud provider API to access a block storage volume of a workload maintained on a target account in a target system of a cloud storage environment;
- utilizing a scanner at a location of the block storage volume and on a secondary system other than the target system;
- scanning the block storage volume for malicious code, using the secondary system;
- identifying malicious code based on the scan; and
- outputting from the secondary system, a notification of a presence of malicious code in the target system.
- wherein the location of the block storage volume includes at least one of: the target account, a secondary system account, a cloud provider account, or a third party account.
- wherein scanning the block storage volume includes scanning disk-backed memory.
- wherein the disk-backed memory includes at least one of a page file or a cache file.
- wherein the secondary system includes at least one of a virtual machine, a container, or a serverless function.
- wherein the secondary system has an operating system different from an operating system of the target account.
- wherein the malicious code includes a rootkit.
- wherein utilizing a scanner includes suspending an operation of the scanner after the scan of the block storage volume.

- wherein utilizing a scanner includes modifying a pre-utilized scanner at the location of the block storage volume based on information related to the target account to obtain a modified scanner;
- wherein utilizing a scanner includes utilizing the modified scanner.
- identifying assets in a cloud environment;
- identifying risks associated with each of the identified asset;
- identifying relationships between at least some of the identified assets, the relationships including at least one of a trust, a network connectivity, or a mechanism of network proxying;
- receiving an identification of a specific asset under investigation;
- performing a forward analysis of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment;
- performing a backward analysis of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset; and
- outputting a signal to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points.
- wherein the network outside the cloud environment includes the Internet.
- wherein the assets in the cloud environment include at least one of: a virtual machine, a network appliance, a storage appliance, a compute instances, or an engine instance.
- wherein identifying the assets in a cloud environment includes identifying the assets based on at least one of: an identity and access management policy, an organization policy, or an access policy.
- wherein the presentation of the forward and backward paths indicates alternative paths connecting between the specific asset and an upstream asset or a downstream asset.
- wherein the visualization includes a presentation of the alternative paths.
- wherein the presentation of the forward and backward paths indicates port numbers for each pathway.
- wherein the visualization of the entry points indicates at least one entry point at risk.
- monitoring network activities of the assets in a cloud environment.
- detecting detect a potential risk associated with the specific asset based on the monitored network activities.

- detecting a potential risk associated with the specific asset based on a network activity of the specific asset.
- detecting a potential risk associated with the specific asset based on a network activity of an upstream asset of the specific asset.
- detecting a potential risk associated with the specific asset based on a network activity of a downstream asset of the specific asset.
- analyzing a cloud environment to identify a plurality of keys to the compute resources in the cloud environment;
- performing a cryptographic analysis on the plurality of keys to identify a first set of fingerprints that uniquely identify each of the plurality of keys, the first set of fingerprints being non-functional;
- analyzing trust configurations of the compute resources to identify a second set of fingerprints for each of the compute resources; and
- comparing the first set of fingerprints with the second set of fingerprints to match keys with the compute resources without using the keys to access the compute resources.
- wherein the plurality of keys are stored in at least one workload.
- wherein at least one of the first set of fingerprints is not identical to any key of the plurality of keys.
- wherein at least one of the plurality of keys includes at least one of a password, a script containing a password, a private component of a private-public key pair, a cloud key, or an Secure Shell (SSH) key.
- testing validity of at least one of the plurality of keys.
- analyzing a multi-machine interaction in the cloud environment using the first set of fingerprints.
- analyzing the multi-machine interaction includes comparing the first set of fingerprints with the second set of fingerprints.
- analyzing a multi-machine interaction in the cloud environment using the plurality of keys.
- accessing a primary account maintained in a cloud environment;
- receiving information defining a structure of the primary account, wherein the structure includes a plurality of assets, and wherein the information excludes raw data of the primary account;
- deploying, inside the primary account or inside a secondary account for which trust has been established with the primary account, at least one ephemeral scanner configured to scan at least one block storage volume and to output metadata defining the at least one block storage volume, the output excluding raw data of the primary account;

- receiving a transmission of the metadata from the at least one ephemeral scanner, wherein the transmission excludes raw data of the primary account;
- analyzing the received metadata to identify a plurality of cybersecurity vulnerabilities;
- correlating each of the identified plurality of cybersecurity vulnerabilities with one of the plurality of assets; and
- generating a report correlating the plurality of cybersecurity vulnerabilities with the plurality of assets.
- wherein defining the at least one block storage volume includes presenting risk data without sharing consumer data or data that was used to identify the risk data.
- wherein the metadata defining the at least one block storage volume includes at least one of: an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, or a profile configuration.
- receiving a transmission of updated metadata defining the at least one block storage volume in response to at least one change to the at least one block storage volumes.
- wherein the ephemeral scanner is configured to perform vulnerability scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform configuration scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform malware scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform lateral-movement risk analysis of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform sensitive information scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform container scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform keys and password scanning of the at least one block storage volume.
- causing a display to present a plurality of asset categories;
- receiving, via an input device, a selection of a particular asset category;
- causing the display to present a list of assets in the selected category that have cyber security risks;
- for each listed asset, retrieving workload component cybersecurity risk information, and
- for each listed asset. retrieving cloud component cybersecurity risk information;

- for each listed asset, causing the display to present a common interface providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information; and
- for each listed asset, causing the display to present, in the common interface, an interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information.
- wherein the plurality of asset categories include at least one of an account category, an authentication category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category.
- wherein the common interface is configured to display information relating to at least one of an asset type, a risk, a region, or an account.
- wherein the common interface is configured to display description for each listed asset.
- wherein the common interface is configured to display at least one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, or an authentication.
- wherein the common interface is configured to display one or more possible attack vectors reaching the each listed asset.
- wherein the common interface is configured to display a recommended mitigation tactic for the each listed asset.
- wherein the common interface is configured to display one or more workload metrics associated with the each listed asset.
- wherein the at least one processor is further configured to provide a cybersecurity report for the each listed asset.
- wherein the at least one processor is configured to identify a risk level distribution among the listed assets.
- receiving a search query for a specific risk; or
- identifying one or more assets vulnerable to the specific risk.

[00459] Other embodiments will be apparent to those skilled in the art from consideration of the specification and practice of the disclosed embodiments disclosed herein. It is intended that the specification and examples be considered as exemplary only, with a true scope and spirit of the disclosed embodiments being indicated by the following claims.

[00460] Computer programs based on the written description and methods of this specification are within the skill of a software developer. The various programs or program modules can be created using a variety of programming techniques. One or more of such

software sections or modules can be integrated into a computer system, non-transitory computer readable media, or existing software.

[00461] Moreover, while illustrative embodiments have been described herein, the scope includes any and all embodiments having equivalent elements, modifications, omissions, combinations (e.g., of aspects across various embodiments), adaptations or alterations based on the present disclosure. The elements in the claims are to be interpreted broadly based on the language employed in the claims and not limited to examples described in the present specification or during the prosecution of the application. These examples are to be construed as non-exclusive. Further, the steps of the disclosed methods can be modified in any manner, including by reordering steps or inserting or deleting steps. It is intended, therefore, that the specification and examples be considered as exemplary only, with a true scope and spirit being indicated by the following claims and their full scope of equivalents.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**08/19/2022 04:54:10 PM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Dena Bohannon
PATENT CENTER #	60890967	FILING DATE	04/05/2022
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 7

DOCUMENT		PAGES	DESCRIPTION	SIZE (KB)
Amendment.pdf		12	-	52 KB
Amendment-A....pdf	(1-1)	1	Amendment/Request for Reconsideration-After Non-Final Rejection	14 KB
Amendment-DRW.pdf	(2-2)	1	Drawings-only black and white line drawings	13 KB
Amendment-CLM.pdf	(3-8)	6	Claims	29 KB
Amendment-REM.pdf	(9-12)	4	Applicant Arguments/Remarks Made in an Amendment	28 KB

ReplacementDwg.pdf	1	Drawings-only black and white line drawings	812 KB
SubSpec_MarkedUp.pdf	101	Specification	624 KB
SubSpec_CleanCopy.pdf	101	Specification	626 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
Amendment.pdf	A5C5BCB29AA66FF9B44643A09A1FEF9AB853B73004819738C209BCBAD23E3DA2C5AE8A511945D828343ED3EDC216DAE21D79B51EA97C6E2A5C6A69DD80B01ECE
Amendment-A....pdf	AABEF8130F0C083272B93609E7AC3EDDC0D961E7EF77A345F125D1388FF74F733439176E332D7A8C8C61A4AEA50F540866ED1F55C2B0B8E14BE388C6219D151F
Amendment-DRW.pdf	639DAF947C9EA7DD66DA229BB67DCEC699B1571BB0B779C7B8E0DC643A4B1319188B20B98B96079F51F94E0EFBB1CFE90B81FAC7D2714E1C70C6028FB268012E
Amendment-CLM.pdf	917D7A6F0348A88B7AC2E4C2BE6449A20A978F9C8E1F3D82FE8BBA1CDFE79EAD9ABAEE0F5F0AC33D23482FBB110A215DBD3DAB2BF7352AC1168F30682DA6F5D3
Amendment-REM.pdf	76991F283D0E8891D999DBE3ED19089E0E853152D6E8DB3B2C3B4613259600E7C66697B61A1EF07874F7446D9CDCE1708A5596B0796EBC1B4514C557F8E8C0C7
ReplacementDwg.pdf	943D720A2B7DB5472605A42A6C8BADF6C4BCED006E6D7309D64FD0CB48234CB05BF927C127843B8F5C55EEE67B16B2D80100A825D51B44B40EC83715D4546089
SubSpec_MarkedUp.pdf	019B51BB58E7DD12E6923A95D466A66855F9D53F4BF1C16178A321D877249FD2E0C3557AD0438A1DEB34AD984D570FA6E2D27E1CAB3FB0A495A404325C717D87

SubSpec_CleanCopy.pdf	77B115F2BF1E1379F8362A350D28A699FCC823088D462AF434 4014C9C092BA6F6A7339A12529E12ACEECEBD68521C3A55F1 9584BF717266E3218599F9607846F

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Docket Number: 15635.0004-00304

SYSTEMS AND METHODS FOR SIDE SCANNING

PRIORITY

[001] This application claims the benefit of U.S. Provisional Application No. 63/180,048 filed on April 26, 2021, the contents of which are hereby incorporated by reference.

TECHNICAL FIELD

[002] Disclosed embodiments include new systems and methods for securing cloud infrastructure to help meet compliance mandates, without disrupting business operations in live environments.

BACKGROUND

[003] Modern organizations typically depend on cloud infrastructure for data storage and processing. Data storage and processing nodes, among other things, power today's electronic infrastructure. In contrast to early network-centric days, an organization may have little in the way of "on-prem" (on-premises) systems and may run its entire business through systems having shared resources. Such an infrastructure may be the entire inventory of cloud assets for an organization, including running and stopped workloads of all types: virtual machines, containers, storage objects, load balancers, IAM (Identity and Access Management) configurations, and more. Organizations are searching for effective ways to scan their entire cloud estate to look for risks stemming from vulnerabilities, misconfigurations, malware, lateral movement risk, weak and leaked passwords, and improperly secured PII.

[004] The infrastructure of these organizations are plainly different than '90s-style physical servers running on bare metal. Existing agents and scanners from the on-prem days have been simply reused in cloud environments without reconfiguration. Before the cloud, it was required to secure physical hosts. That meant spending time installing multiple security agents—one for each server. IP addresses were assigned to physical assets and they seldom changed. Even then, however, agent integration was tedious and coverage rarely reached 100% of assets. In a cloud environment, scaling up and down is a frequent occurrence—possibly thousands of times per hour across multiple clouds—and all within a CI/CD pipeline that builds an organization's infrastructure. There are containers and virtual machines to contend with, and agents carry high operational costs.

[005] This causes problems of inefficiency and overuse of resources, including excess processor cycles on regular scans. In turn, such problems and inefficiencies can lead to a higher monetary cost for hosted resources as some cloud providers charge per processing cycle.

[006] Relying on agents for security visibility in the cloud also has drawbacks. Visibility is limited to only those assets that are already known and accessible to an agent, requiring that such assets are capable of having the agent installed and maintained in the first place. The assets also must have ongoing network connectivity to the backend, something that makes the

development cycle inefficient as developers tend not to be inclined to deploy agents on virtual machines, in containers, and in serverless configurations—let alone dealing with their never-ending maintenance—thus leading to insecure implementations.

[007] An authenticated scan allows for direct host access using remote protocols such as SSH or RDP. The scanner uses a privileged account to log in and determine how secure each host is from an inside vantage point. While authenticated scans can successfully discover potential vulnerabilities, they are limited as they require a privileged account on each scanned host. Furthermore, scans use significant system resources during the test procedures and require opening ports that by themselves pose a security risk.

[008] An unauthenticated scan can only examine publicly visible information and is not able to provide detailed information about assets. It essentially acts as a friendly attacker. An unauthenticated scan can easily miss identifying some assets and vulnerabilities, making it much less effective. For example, a website at [[https://mydomain.com/email_campaign]] mydomain.com/email_campaign not linked from a main website will not be scanned unless the scanner is manually configured. This can leave organizations exposed to vulnerabilities in areas where the scanner cannot reach.

[009] While unauthenticated scanners act like an attacker, they often get stuck where a real attacker would not. For example, a CAPTCHA can easily prevent automatic mechanisms (such as scanners) from registering. But such techniques would not have any effect on a human attacker.

[0010] Cloud security posture managers (CSPMs) are scanning tools developed specifically for the cloud. Rather than going inside the machine, a CSPM analyzes the cloud configuration itself for weaknesses. CSPMs are used to discover, assess, and solve cloud misconfigurations but provide shallow coverage where cloud security is concerned because CSPMs have difficulty detecting critical risks such as vulnerabilities, malware, and misconfigurations within the workloads themselves. Organizations choosing to combine agent-based solutions with a CSPM end up getting flooded with separate alerts that lack context which results in alert fatigue on behalf of security analysts.

[0011] Therefore, there is a need for improved security scanning systems that solve the problem with existing systems and are more adapted for use in cloud infrastructures.

[0012] The foregoing background is for illustrative purposes and is not intended as a discussion of the scope of the prior art or an admission of prior art.

SUMMARY

[0013] This summary is provided for the convenience of the reader to present a basic understanding of a few exemplary embodiments of this disclosure, and does not wholly define the breadth of the disclosure. This summary is not an extensive overview of all contemplated embodiments and is intended to neither identify key or critical elements of all embodiments nor

to delineate the scope of any or all aspects. Its sole purpose is to present some features of one or more embodiments in a simplified form as a prelude to the more detailed description presented later. For convenience, the term "embodiments" may be used herein to refer to one or more features or combinations of features described herein. A description associated with an embodiment does not necessarily apply to all embodiments.

[0014] Embodiments of the present disclosure may deliver instant-on, workload level visibility across cloud infrastructure assets without running opcodes on the target scanned workloads, helping organizations to: (1) Detect risks such as vulnerabilities, malware, misconfigurations, lateral movement risk, and unsecured sensitive data; (2) See cloud inventory at every layer: infrastructure, operating system, applications, and data; (3) Discover and see previously missed assets.

[0015] Embodiments of the present disclosure may further use an out-of-band process to reach cloud workloads through the runtime storage layer, combining this with information gathered from cloud provider APIs, including but not limited to K8S APIs.

DRAWINGS

[0016] The drawings and figures contained herein are a part of this specification, illustrate several embodiments consistent with the disclosure and, together with the description, serve to explain principles of the disclosure.

[0017] **Fig. 1** is a schematic block diagram illustrating an exemplary embodiment of a network including computerized systems, consistent with disclosed embodiments.

[0018] **Fig. 2A** is a schematic block diagram illustrating an exemplary embodiment of a process for integration, scanning, assessment, analyzing/reporting, and change reviewing, consistent with disclosed embodiments.

[0019] **Fig. 2B** is a schematic block diagram illustrating an exemplary embodiment of a process for integration, consistent with disclosed embodiments.

[0020] **Fig. 2C** is a schematic block diagram illustrating an exemplary embodiment of a process for scanning/mapping, consistent with disclosed embodiments.

[0021] **Fig. 2D** is a schematic block diagram illustrating an exemplary embodiment of a process for assessment, consistent with disclosed embodiments.

[0022] **Fig. 3** is a block diagram of method for insight extraction, consistent with disclosed embodiments.

[0023] **Fig. 4** depicts a cybersecurity system performing a side scanning function to protect against potential vulnerabilities, consistent with disclosed embodiments.

[0024] **Fig. 5** of the disclosed embodiments describes a method of operating a cybersecurity system performing a side scanning function to protect against potential vulnerabilities, consistent with disclosed embodiments.

[0025] **Fig. 6** is a block diagram of method for risk prioritization, consistent with disclosed embodiments.

[0026] **Fig. 7** is a block diagram illustrating an exemplary process of cyber security scanning for a cloud environment, consistent with the disclosed embodiments.

[0027] **Fig. 8** is a schematic block diagram illustrating an exemplary embodiment of a system for performing visualization of forward and backward facing threats, consistent with the disclosed embodiments.

[0028] **Fig. 9** is a block diagram illustrating an exemplary process of matching keys with compute resources in a cloud environment, consistent with the disclosed embodiments.

[0029] **Fig. 10** is a block diagram of a method for deployment of ephemeral scanners, consistent with disclosed embodiments.

[0030] **Fig. 11** represents a schematic block diagram illustrating an exemplary embodiment of a method for providing a dashboard aggregating risk information, consistent with disclosed embodiments.

[0031] **Fig. 12** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of displaying aggregated cybersecurity risk information, consistent with disclosed embodiments.

[0032] **Fig. 13** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of a flow path for aggregated risk information, consistent with disclosed embodiments.

DETAILED DESCRIPTION

[0033] Exemplary embodiments are described with reference to the accompanying drawings. The figures are not necessarily drawn to scale. While examples and features of disclosed principles are described herein, modifications, adaptations, and other implementations are possible without departing from the spirit and scope of the disclosed embodiments. For example, with this detailed description provides a few examples, these implementations are provided as examples only and are not restrictive of the claim concepts that follow or any of the descriptions herein. Also, the words “comprising,” “having,” “containing,” and “including,” and other similar forms are intended to be equivalent in meaning and be open ended in that an item or items following any one of these words is not meant to be an exhaustive listing of such item or items or meant to be limited to only the listed item or items. It should also be noted that as used herein and in the appended claims, the singular forms “a,” “an,” and “the” include plural references unless the context clearly dictates otherwise.

[0034] In the following description, various working examples are provided for illustrative purposes. However, it is to be understood that the present disclosure may be practiced without one or more of these details.

[0035] It is intended that one or more aspects of any mechanism may be combined with one or more aspect of any other mechanisms, and such combinations are within the scope of this disclosure.

[0036] Various embodiments are described herein with reference to a system, method, device, or computer readable medium. It is intended that the disclosure of one is a disclosure of all. For example, it is to be understood that disclosure of a computer readable medium described herein also constitutes a disclosure of methods implemented by the computer readable medium, and systems and devices for implementing those methods, via for example, at least one processor. It is to be understood that this form of disclosure is for ease of discussion only, and one or more aspects of one embodiment herein may be combined with one or more aspects of other embodiments herein, within the intended scope of this disclosure.

[0037] Embodiments described herein may refer to a non-transitory computer readable medium containing instructions that when executed by at least one processor, cause the at least one processor to perform a method. Non-transitory computer readable medium may include any medium capable of storing data in any memory in a way that may be read by any computing device with a processor to carry out methods or any other instructions stored in the memory. The non-transitory computer readable medium may be implemented as hardware, firmware, software, or any combination thereof. Moreover, the software may preferably be implemented as an application program tangibly embodied on a program storage unit or computer readable medium consisting of parts, or of certain devices and/or a combination of devices. The application program may be uploaded to, and executed by, a machine having any suitable architecture. Preferably, the machine may be implemented on a computer platform having hardware such as one or more central processing units ("CPUs"), a memory, and input/output interfaces. The computer platform may also include an operating system and microinstruction code. The various processes and functions described in this disclosure may be either part of the microinstruction code or part of the application program, or any combination thereof, which may be executed by a CPU, whether or not such a computer or processor is explicitly shown. In addition, various other peripheral units may be connected to the computer platform such as an additional data storage unit and a printing unit. Furthermore, a non-transitory computer readable medium may be any computer readable medium except for a transitory propagating signal.

[0038] Memory employed herein may include a Random Access Memory (RAM), a Read-Only Memory (ROM), a hard disk, an optical disk, a magnetic medium, a flash memory, other permanent, fixed, volatile or non-volatile memory, or any other mechanism capable of storing instructions. The memory may include one or more separate storage devices collocated or disbursed, capable of storing data structures, instructions, or any other data. The memory may further include a memory portion containing instructions for the processor to execute. The

memory may also be used as a working scratch pad for the processors or as a temporary storage.

[0039] Some embodiments may involve at least one processor. A processor may be any physical device or group of devices having electric circuitry that performs a logic operation on input or inputs. For example, the at least one processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations. The instructions executed by at least one processor may, for example, be pre-loaded into a memory integrated with or embedded into the controller or may be stored in a separate memory.

[0040] In some embodiments, the at least one processor may include more than one processor. Each processor may have a similar construction, or the processors may be of differing constructions that are electrically connected or disconnected from each other. For example, the processors may be separate circuits or integrated in a single circuit. When more than one processor is used, the processors may be configured to operate independently or collaboratively. The processors may be coupled electrically, magnetically, optically, acoustically, mechanically or by other means that permit them to interact.

[0041] Consistent with the present disclosure, disclosed embodiments may involve a network. A network may constitute any type of physical or wireless computer networking arrangement used to exchange data. For example, a network may be the Internet, a private data network, a virtual private network using a public network, a Wi-Fi network, a LAN or WAN network, and/or other suitable connections that may enable information exchange among various components of the system. In some embodiments, a network may include one or more physical links used to exchange data, such as Ethernet, coaxial cables, twisted pair cables, fiber optics, or any other suitable physical medium for exchanging data. A network may also include a public switched telephone network ("PSTN") and/or a wireless cellular network. A network may be a secured network or unsecured network. In other embodiments, one or more components of the system may communicate directly through a dedicated communication network. Direct communications may use any suitable technologies, including, for example, BLUETOOTH™, BLUETOOTH LE™ (BLE), Wi-Fi, near field communications (NFC), or other suitable communication methods that provide a medium for exchanging data and/or information between separate entities.

[0042] Certain embodiments disclosed herein may also include a computing device for cloud cybersecurity, the computing device may include processing circuitry communicatively connected to a network interface and to a memory, wherein the memory contains instructions to be executed. The computing devices may be devices such as mobile devices, desktops,

laptops, tablets, or any other devices capable of processing data. Such computing devices may include a display such as an LED display, augmented reality (AR), virtual reality (VR) display.

[0043] "Software" as used herein refers broadly to any type of instructions, whether referred to as software, firmware, middleware, microcode, hardware description language, or otherwise. Instructions may include code (e.g., in source code format, binary code format, executable code format, or any other suitable format of code). The instructions, when executed by the one or more processors, may cause the processing system to perform the various functions described in further detail herein.

[0044] The one or more processors may be implemented with any combination of general-purpose microprocessors, microcontrollers, digital signal processors (DSPs), field programmable gate array (FPGAs), programmable logic devices (PLDs), controllers, state machines, gated logic, discrete hardware components, dedicated hardware finite state machines, or any other suitable entities that can perform calculations or other manipulations of information.

[0045] Aspects of this disclosure may provide technical solutions to challenges associated with cloud cybersecurity. Disclosed embodiments include methods, systems, devices, and computer-readable media. For ease of discussion, a system is described below with the understanding that the disclosed details may equally apply to methods, devices, and computer-readable media.

[0046] Embodiments of the present disclosure include technology referred to as "SideScanning." In contrast to some existing systems and techniques, these embodiments may provide a distinct advantage because the technology does not necessarily require entering into each workload to inspect data. Rather, some embodiments use an out-of-band process to reach cloud workloads through a runtime storage layer, combining this with metadata gathered from API provided through a cloud service provider's system, thus providing visibility of cloud environments both at a low level and with context, without the requirement for an agent or network scanner.

[0047] **Fig. 1** is a schematic block diagram 100 illustrating an exemplary embodiment of a network including computerized systems, consistent with the disclosed embodiments. Diagram 100 includes user device 102, network 105, and cloud infrastructure 106.

[0048] Cloud infrastructure 106 includes scanning system 101, databases 103A-103D, virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115. While particular numbers and arrangements of devices, systems, and connections, are depicted in exemplary Fig. 1, in some embodiments, each of the devices, systems, or connections may be omitted, duplicated, or modified. For example, in some embodiments, databases 109A-109D may exist as only a single database; in other embodiments, cloud infrastructure 106 may exist as one or more distinct or combined

infrastructures (e.g., operated by the same or different cloud services). In some embodiments, scanning system 101 and/or databases 103A-103D may be part of cloud infrastructure 106 (and may be connected to the various other systems and devices in cloud infrastructure 106); in other embodiments, scanning system 101 and/or databases 103A-103D may be separate from cloud infrastructure 106 (e.g., connected to the systems and devices in cloud infrastructure 106 through network 105).

[0049] Scanning system 101, in some embodiments, may include one or more computer systems. Each of the one or more computer systems may include memory storing instructions and at least one CPU configured to execute those instructions to perform operations as discussed herein. In some embodiments, the instructions cause the CPU to perform scanning operations. In some embodiments, scanning system 101 may perform a scanning operation on one or more workloads (e.g., systems, devices, resources, etc.) in cloud infrastructure 106.

[0050] User device 102, in some embodiments, may include a mechanism operated by a user to control scanning system 101. For example, in some embodiments, user device 102 may be any of a personal computer, a server, a thin client, a tablet, a personal digital assistant, a smartphone, a kiosk, or any other mechanism enabling data input. User device 102 may be operated to instantiate functionality, access data, or otherwise interact with scanning system 101 via network 105, as described herein.

[0051] Databases 103A-103D include data stores for use by scanning system 101. In some embodiments, one or more of databases 103A-103D may be implemented as a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. In some embodiments, one or more of databases 103A-103D may be implemented as flat file stores, data stores, or other non-database storage systems. In some embodiments, databases 103A-103D may be implemented using one or more of ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[0052] Network 105 may be implemented as one or more interconnected data networks. For example, network 105 may include one or more of any type of network (including infrastructure) that provides communications, exchanges information, and/or facilitates the exchange of information, such as the Internet, a Local Area Network, a near field communication (NFC) network, or other suitable connection(s) that enables the sending and receiving of information between the components of system 100. Network 105 may be implemented using wireless connections, wired connections, or both. In some embodiments, one or more components of system 100 can communicate through network 105. In some embodiments, one or more components of system 100 may communicate directly through one or more dedicated communication links. While particular devices and systems are shown as connected to network

105, in some embodiments, more or fewer devices and systems may be connected to network 105.

[0053] Cloud infrastructure 106 may be implemented as a set of devices and systems offered by a single cloud service provider. For example, cloud infrastructure 106 may comprise devices and systems that are part of Amazon Web Services, Microsoft Azure, Google Cloud Platform, IBM Cloud, Alibaba Cloud, or any other cloud platform provider. In some embodiments, one or more of the devices and systems in cloud infrastructure may require authentication or other identity validation for access. For example, to access virtual machine 107A, a user may be required to enter a password or provide a key. Systems (e.g., scanning system 101 or user device 102) may administer or interact with cloud infrastructure 106 using a cloud service provider's system (not pictured).

[0054] Virtual machines 107A-107D may include one or more devices and systems that implement a virtualized/emulated version of a computer. A virtual machine may be implemented as an emulated version of a computer - including an operating system, memory, storage, graphics processing - such that it can be indistinguishable from a standard (non-virtual) machine to a running program. A computer system, referred to as a "host," may operate virtual machines 107A-107D, referred to as "guests," by dividing the resources of the host between the virtual machines such that each virtual machine is isolated from one another. This means that in some embodiments, one virtual machine, and the operating system(s) and application(s) running thereon, is only able to access the resources that are allocated to that virtual machine and cannot access resources allocated to other virtual machines. For example, if a host has 32 gigabytes of random access memory (RAM), and is hosting three virtual machines, the host may segment 8 gigabytes of RAM to each virtual machine such that each virtual machine may only access data in that 8 gigabytes of RAM and not any of the other 24 gigabytes. Examples of commercial virtual machine software and services include VMWare Workstation, VMWare Server, VMWare ESXi, VirtualBox, Parallels Desktop, Parallels RAS, Amazon Machine Image, Amazon ECS, Kubernetes, Microsoft Hyper-V, and Xen.

[0055] Databases 109A-109D may include data stores for use by devices and systems in cloud infrastructure 106. In some embodiments, one or more of databases 109A-109D may be implemented as a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. In some embodiments, one or more of databases 109A-109D may be implemented as flat file stores, data stores, or other non-database storage systems. In some embodiments, databases 109A-109D may be implemented using one or more of ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances. Databases 109A-109D may store data usable by devices or systems in cloud infrastructure 106. The data, in some embodiments,

may include e-commerce data (e.g., shipments, orders, inventory), media data (e.g., pictures, movies, streaming data), financial data (e.g., banking data, investment data), or other data.

[0056] Storage 111A-111D may include storage systems for use by devices and systems in cloud infrastructure 106. In some embodiments, one or more of storage 111A-111D may be implemented as a hard drive, a RAID array, flash memory, optical storage, or any other kind of storage. Each of 111A-111D may include one or more filesystems (e.g., Amazon Elastic File System, GlusterFS, Google File System, Hadoop Distributed File System, OpenZFS, S3, Elastic Block Storage).

[0057] In some embodiments, systems and devices of cloud infrastructure 106 may use databases 109A-109D to store data that is accessed frequently (where, for example, access is required within a few milliseconds), and may use storage 111A-111D to store data that is accessed less frequently (where, for example, access is required within a few minutes or hours).

[0058] Keystores 113A-113D may include systems storing keys for accessing data and functionality. For example, to access certain data or systems, a system may require the use of passwords or keys in keystores 113A-113D for authentication. The data and functionality that the keys grant access to may be part of cloud infrastructure 106 or may be separate from cloud infrastructure 106. For example, keystores 113A-113D may include systems that store public and private keys (e.g., for use via SSH), may store passwords (e.g., login information for websites or programs), may store tokens (e.g., one-time passcodes), or the like. In some embodiments, keystores 113A-113D may be implemented as one or more of Amazon Web Services KMS, Azure Key Vault, or Google KMS.

[0059] Load balancer 115 may include one or more systems that balance incoming requests between the different systems and devices of cloud infrastructure 106. For example, load balancer 115 may be configured to determine usage (e.g., processor load, used storage capacity) of systems or devices in cloud infrastructure 106 to assist in determining where to route an incoming request from network 105 to store data, perform processing, or retrieve data.

[0060] Load balancer 115 may be configured to receive an incoming request from user device 102. Upon receipt of the request, load balancer 115 may consult a data store (part of or separate from load balancer 115; not pictured) to determine usage or forecasted usage of various systems or devices in cloud infrastructure 106, and may forward the request to the systems or devices having the lowest usage or forecasted usage.

[0061] **Fig. 2A** is a schematic block diagram illustrating an exemplary embodiment of a process 200 for integration, scanning, assessment, and review, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2A are executed using software and hardware of scanning system 101. In some embodiments, the steps in Fig. 2A may be performed in an order other than those depicted in Fig. 2A. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step

of Fig. 2A may be used to provide context or other information for use in another step of Fig. 2A. In other embodiments, the steps in Fig. 2A may be executed by other devices.

[0062] Process 200 begins with step 201. In step 201, scanning system 101 may execute a process of integration. The integration process may be performed by scanning system 101 with cloud infrastructure 106. In some embodiments, the integration process includes creating a connection between an account on scanning system 101 and an account on cloud infrastructure 106. The process of integration in step 201 may, in some embodiments, be implemented as described below with respect to Fig. 2B.

[0063] In step 203, scanning system 101 may execute a process of scanning/mapping. The scanning/mapping process may be performed by scanning system 101 with cloud infrastructure 106. In some embodiments, the scanning and mapping process may include analyzing data relating to cloud infrastructure 106 using scanning system 101, by reading cloud infrastructure 106 through the connection made in step 201 and generate a "map" (e.g., a data structure or data collection) representing systems and devices in cloud infrastructure 106. The process of scanning and mapping in step 203 may, in some embodiments, be implemented as described below with respect to Fig. 2C.

[0064] In step 205, scanning system 101 may execute a process of assessing. The assessing process may be performed by scanning system 101. In some embodiments, the assessing process may include reviewing vulnerabilities, infrastructure, interconnections, data, and other information, using scanning system 101. The process of assessing in step 205 may, in some embodiments, be implemented as described below with respect to Fig. 2D.

[0065] In step 207, scanning system 101 may execute a process of analyzing/reporting. The analyzing/reporting may be performed by scanning system 101. In some embodiments, the change reviewing process comprises scanning system 101 scanning cloud infrastructure 106 again to determine the differences between an earlier observed snapshot of cloud infrastructure 106 and the current state thereof. In some embodiments, scanning system 101 may analyze information (e.g., steps 203 and/or 205) to generate reports. The reports may list what each vulnerability is, where it is located, and its priority. In this way, security engineers and DevOps teams may be able to easily assess how to best allocate their time and attention.

[0066] In some embodiments, analyzing/reporting in step 207 may include scanning system 101 combining conclusions from different environmental perspectives (e.g., metadata) into a single model. For example, scanning system 101 may map the running services on cloud infrastructure 106 and consider collected vulnerability data. Scanning system 101, in some embodiments, may generate a visualization of the map for review, listing where the vulnerabilities are on a two-dimensional graphical representation of cloud infrastructure 106.

[0067] In some embodiments, scanning system 101 may indicate in the map whether an asset (e.g., a system or device) is Internet-facing and easily accessible to attackers, exposed only to

internal assets or if it is private altogether (and may be less critical). For example, based on the contextual map, scanning system 101 may perform a "forward" analysis of the specific asset under identification to identify at least one possible Internet-originating attack vector to the asset. Alternatively or additionally, scanning system 101 may perform a "backward" analysis of the specific asset to identify exposure risk to assets downstream of the specific asset, wherein the downstream exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset. Both backward and forward analysis can be recursively, for one or more hops, Scanning system 101 may present these analyses to a user via graphical user interface or expose them via an API.

[0068] As one example, scanning system 101 may determine a vulnerability in a web service. In some embodiments, scanning system 101 may score the vulnerability as: high risk if the web service is connected to the Internet (e.g., has at least one public port forwarded through a firewall to the web service, is able to be accessed through a load balancer, is able to be accessed via reverse proxy); medium risk if the web service is only accessible internally (e.g., because of firewall configuration); and low risk if access to the web server is blocked by a configuration of cloud infrastructure 106. As another example, if a machine is stopped (e.g., turned off, not running, not connected), a vulnerability, but one that is less likely to be exploited because the machine isn't running. This affects its risk score and other mitigating factors.

[0069] In step 209, scanning system 101 may execute a process of change reviewing. The change reviewing may be performed by scanning system 101. In some embodiments, the change reviewing process may include scanning system 101 scanning cloud infrastructure 106 again to determine the differences between an earlier observed snapshot of cloud infrastructure 106 and the current state thereof.

[0070] In some embodiments, scanning system 101 may monitor cloud change, or "delta," logs (e.g., Cloud event logs such as AWS CloudTrail, monitoring a network-related change in state, a trust-related change in state, or a disk configuration-related change in state in at least one of the primary asset group or a secondary asset group) and contextualize associated risks discovered within a customer cloud account. In some embodiments, scanning system 101 may generate a new map of assets based on a review of all systems or devices in cloud infrastructure 106 (e.g., as described above with respect to step 203), and compare the new map to the map generated earlier in step 203 to determine which assets have changed. In some embodiments, this comparison may require comparing data, metadata, network connections, software configurations, firewall configurations, or any other aspects related to cloud infrastructure 106, including, for example, determining new assets that are created or assets for which their configuration has changed.

[0071] After step 209, as depicted in Fig. 2A, process 200 may return to step 205. Scanning system 101 may then perform step 205 again. In some embodiments, steps 205, and 207 may

be performed only on the devices or systems that have experienced a change that was detected by step 209. In some embodiments, process 200 may return to step 203 after step 209.

[0072] **Fig. 2B** is a schematic block diagram illustrating an exemplary embodiment of a process 210 for integration, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2B are executed using software and hardware of scanning system 101.Integration, in some embodiments, is a process for creating a trust relationship between scanning system 101 and cloud infrastructure 106.

[0073] In some embodiments, the steps in Fig. 2B may be performed in an order other than those depicted in Fig. 2B. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step of Fig. 2B may be used to provide context or other information for use in another step of Fig. 2B. In other embodiments, the steps in Fig. 2B may be executed by other devices.

[0074] Process 210 may begin with step 211 to initiate a connection to cloud infrastructure 106. In step 211, scanning system 101 may send a message to user device 102 instructing a user to authenticate, or log in, to a cloud service provider's system operating cloud infrastructure 106. For example, the user may use a username, password, one-time password, two-factor authentication, or any other authentication mechanism to gain access to a cloud service provider's system.

[0075] Concurrently with or after the first message, scanning system 101 may send a second message to user device 102, instructing the user to generate a role. The second message may include instructions for the user to follow to generate the role. In step 213, a user may provide (e.g., via a keyboard at user device 102) a role definition to the cloud service provider's system. In some embodiments, the role definition includes read-only permissions and permissions to read a block storage layer (containing block storage volumes). In some embodiments, scanning system 101 provides a role formation template (e.g., an Amazon Web Services CloudFormation Template) for use with cloud infrastructure 106 to create the necessary role. In step 213, the user may utilize user device 102, for example, by copying and pasting a URL of the template, downloading and uploading the template to the cloud service provider's system, or selecting the template from a list of templates.

[0076] In step 215, the cloud service provider's system may generate a string (e.g., a "key" or "resource name") for use by scanning system 101. In some embodiments, this string may be used to enable access by scanning system 101 to the workload of cloud infrastructure as permitted by the generated role. A user using user device 102 may copy the string and paste it into a user interface presented by scanning system 101 on user device 102. Other aspects of transmitting this string to scanning system 101 are possible as well.

[0077] In step 217, scanning system 101 determines that it is able to access cloud infrastructure 106. For example, scanning system 101 may attempt to perform a command such as authenticating to cloud infrastructure 106 using the string received in step 215 or attempting to perform a command using cloud infrastructure (e.g., requesting a listing of files stored on storage 111A). Once scanning system 101 determines that it is able to access cloud infrastructure 106, the process may return to Fig. 2A and step 203.

[0078] **Fig. 2C** is a schematic block diagram illustrating an exemplary embodiment of a process 220 for scanning/mapping, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2C are executed using software and hardware of scanning system 101. In some embodiments, the steps in Fig. 2C may be performed in an order other than those depicted in Fig. 2C. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step of Fig. 2C may be used to provide context or other information for use in another step of Fig. 2C. In other embodiments, the steps in Fig. 2C may be executed by other devices.

[0079] Process 220 may begin with step 221 where scanning system 101 initiates a process to access keys, such as those keys stored in keystores 113A-113D. Scanning system 101 may utilize the string (e.g., a key or resource name) received in process 210 to authenticate in order to retrieve keys stored in keystores 113A-113D. Once authenticated, scanning system 101 may access one, some, or all keys stored in keystores 113A-113D. In some embodiments, step 221 may comprise utilizing pre-established trust relationships, such as AWS trust policies, instead of creating a new relationship.

[0080] In step 223, scanning system 101 may generate a “snapshot” of devices and systems in cloud infrastructure 106. In some embodiments, generating a snapshot may include reading devices and systems in cloud infrastructure, such as storage 111A-111D, databases 109A-109D, and virtual machines 107A-107D, and copying the information read from those devices and systems to storage at scanning system 101. Generating a snapshot, in some embodiments, may include recording a reference count to data blocks in one or more of storage 111A-111D, databases 109A-109D, and virtual machines 107A-107D, and copying each block them like a copy and write operation. In some embodiments, scanning system 101 may generate an Elastic Block Storage snapshot in step 223.

[0081] In some embodiments, scanning system 101 uses one or more keys retrieved from keystores 113A-113D to encrypt “snapshots” of cloud infrastructure.

[0082] In step 225, scanning system 101 may apply “tags” to the snapshot. This process may include, in some embodiments, adding information to the snapshot to identify the snapshot as being associated with scanning system 101. Scanning system 101, in some embodiments, may be configured to delete only snapshots with the associated tags.

[0083] In step 227, scanning system 101 may generate a map of cloud infrastructure 106. In some embodiments, this map may be in a form of graph: a plurality of interconnected vectors connecting the plurality of systems and devices, based on the networking configuration. Scanning system 101, or other devices, may traverse the map to identify vectors originating in the Internet and reaching the devices and systems in cloud infrastructure 106. In some embodiments, generating the map may include enumerating Internet-accessible services that are capable of serving as an Internet proxy. In some embodiments, a user may control scanning system 101 (e.g., via user device 102) to display the map on a graphical user interface.

[0084] In some embodiments, generating the map may include enumerating properties of all assets, including: virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, load balancer 115, log files or databases, API gateway resources, API gateway REST APIs, Autoscaling groups, CloudTrail logs, CloudFront services, volumes, snapshots, VPCs, subnets, route tables, network ACLs, VPC endpoints, NAT gateways, ELB and ALB, ECR repositories, ECS clusters, services, and tasks, EKS, S3 bucket and Glacier storage, SNS topics, IAM roles, policies, groups, users, KMS keys, and Lambda functions. In some embodiments, generating the map in step 227 may further include analyzing devices or systems for a subset of risk related situations, including determinations of compromise situations (e.g., where an attacker has already gained access), imminent compromise situations (e.g., a known attack vector exists and can be used, such as a data store that is exposed to the public Internet without authentication), hazardous situations (e.g., a serious security implication, but no full attack vector exists), or informational situations (e.g., when storage 111A-111D has a limited amount of free space, or an unexploitable vulnerability exists).

[0085] Generating the map in step 227 may include recording information such a region identifier, site identifier, datacenter identifier, physical address, network address, workload name, or any other identifier which may be acquired via an API provided through a cloud service provider's system.

[0086] In step 229, scanning system 101 may provide alerts, e.g., to user device 102, indicating any situations found during the process of generating the map. For example, scanning system 101 may send one or more of an email, a popup alert, a text message, or other notification to user device 102.

[0087] After performing step 229, the process may return to Fig. 2A and step 205.

[0088] **Fig. 2D** is a schematic block diagram illustrating an exemplary embodiment of a process 230 for assessing, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2D are executed using software and hardware of scanning system 101. In other embodiments, the steps in Fig. 2D may be executed by other devices. In some embodiments, scanning system 101 may execute the steps in Fig. 2D against a snapshot created in step 223 of Fig. 2C and stored at scanning system 101. In some embodiments, the steps in Fig. 2D may

be performed in an order other than those depicted in Fig. 2D. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step of Fig. 2D may be used to provide context or other information for use in another step of Fig. 2D.

[0089] Process 230 begins with step 231. In step 231, scanning system 101 may perform a step of vulnerability scanning. In some embodiments, step 231 comprises extracting everything in the snapshot, including operating system packages, installed software applications, libraries, and program language libraries such as Java archives, Python packages, Go modules, Ruby gems, PHP packages, and Node.js modules, or other software applications.

[0090] In some embodiments, step 231 may determine library versions, software versions, and other identifying characteristics of software and operating systems in the snapshot. Scanning system 101 may then try to match them to known vulnerabilities stored in a vulnerability database (e.g., one of databases 103A-103D). The vulnerability database, in some embodiments, may include vulnerability data from: NVD, WPVulnDB, US-CERT, Node.js Security Working Group, OVAL – Red Hat, Oracle Linux, Debian, Ubuntu, SUSE, Ruby Advisory Database, JVN, Safety DB(Python), Alpine secdb, PHP Security Advisories Database, Amazon ALAS, RustSec Advisory Database, Red Hat Security Advisories, Microsoft MSRC, KB, Debian Security Bug Tracker, Kubernetes security announcements, Exploit Database, Drupal security advisories, JPCERT. The vulnerability database may also include other vulnerability data (including, e.g., manually-added vulnerability data or vulnerability sources not listed above).

[0091] In step 233, scanning system 101 may perform a step of configuration scanning. In step 233, scanning system 101 may gather configuration information—such as a list of users of each system or device (e.g., virtual machines 107A-D), each system's or device's services, password hashes, and application-specific configurations for software/services such as Apache, Nginx, SSH, and other services. In some embodiments, scanning system 101 may perform a first analysis on all information collected in step 233 to remove sensitive information (e.g., social security numbers, passwords, birthdates) before proceeding to review the configuration information. In some embodiments, step 233 may comprise verifying adherence of the systems or devices in cloud infrastructure 106 to standards or benchmarks established by an external entity, such as the Center for Internet Security.

[0092] In some embodiments, scanning system 101 may perform a benchmarking process to detect misconfigurations of any services based on the information gathered in step 233. For example, scanning system 101 may determine the software version of each service and examine it against known vulnerabilities (e.g., stored in database 103A).

[0093] In some embodiments, scanning system 101 may determine bugs or other configuration risks that might only be exploitable from internal machines, because such bugs and risks can facilitate an attacker's lateral movement.

[0094] In some embodiments, scanning system 101 may evaluate network misconfigurations and their implications. Scanning system 101 may query devices and systems capable of routing traffic (e.g., load balancer 115, routers, switches, firewalls, and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations.

[0095] In step 235, scanning system 101 may perform a step of malware scanning. In some embodiments, scanning system 101 may perform malware scanning across all filesystems in the snapshot (e.g., gathered from virtual machines 107A-107D or storage 111A-111D). Scanning system 101 may use multiple malware scanning software solutions to perform a malware scan against the filesystems, including one sourced from another vendor, such as bucketAV, Trend Micro Cloud One, Sophos Cloud Optix, Crowdstrike Falcon CWP, or others. In some embodiments, malware scanning in step 235 comprises utilizing signatures, heuristics, or sandboxing capabilities to deduce whether there is an infection on the machine.

[0096] In step 237, scanning system 101 may perform a step of lateral movement scanning. An attacker who establishes a network foothold usually attempts to move laterally from one resource to another in search of rich targets such as valuable data. Stolen passwords and keys unlock access to servers, files, and privileged accounts. In some embodiments, scanning system 101 may gather keys from each scanned system or device (e.g., virtual machines 107A-107D or storage 111A-111D). In some embodiments, scanning system 101 searches for passwords, scripts, shell history, repositories, or other data that may contain passwords, cloud access keys, SSH keys, or other key/password/access information that provide unchecked access to important resources. In some embodiments, scanning system 101 searches for such keys/passwords/access information and calculates a "hash" (a mathematical fingerprint) of each string. Scanning system 101 then attempts to match the hashed strings to hashes of strings that that are stored on different systems or devices. This will be used to detect the potential lateral movement between assets.

[0097] In step 239, scanning system 101 may perform a step of key/password scanning. As one example situation, suppose there is a weak or unprotected password stored (in plain text) in storage 111A. For example, if a personal email account has been compromised, the passwords may be known about in advance. Scanning system 101 may search the snapshot for similar usernames or login names, and, either using known dictionaries or the account owner's previously leaked passwords (stored in, e.g., database 103A), may attempt to login to one or more systems or devices in cloud infrastructure 106, and may record the result thereof.

[0098] In some embodiments, scanning system 101 may perform a "fuzzy search" on any usernames found in a password database (e.g., database 103A) to determine existing targets for password testing. In some embodiments, the fuzzy search uses the Damerau-Levenshtein edit distance algorithm.

[0099] As one example, scanning system 101 may determine that a leaked username includes the email address johndoe@acme.com. Scanning system 101 may try to match passwords from john.doe@acme.net, johndoe@gmail.com, and other variations. Scanning system 101 may perform a similar process against leaked passwords (e.g., if a leaked password is "Victory@19," scanning system may attempt to log in using "Victory@20."

[00100] In step 241, scanning system 101 may perform a step of sensitive information scanning. In some embodiments, scanning system 101 may search the snapshot for sensitive information, such as personally identifiable information (PII), Social Security numbers, healthcare information, or credit card numbers. In some embodiments, scanning system 101 may search data repository history as well. This is because it is not uncommon for an entire production environment repository to be cloned, with no one remembering the copy contains sensitive information. In some situations, detecting sensitive data not secured is critical in adherence to data privacy regulations.

[00101] To be certain that such alerts do not constitute false positives, in some embodiments, scanning system 101 may perform statistical scans against the data. For example, it is possible for a random number to resemble a Social Security number, yet it is extremely unlikely for the majority of a file with thousands of numbers to be valid Social Security Number by pure chance.

[00102] In step 243, scanning system 101 may perform a step of container scanning. In some embodiments, scanning system 101 may apply one or more of the preceding steps of Fig. 2D against containerized environments. In some embodiments, in order to do so, scanning system 101 reconstructs a container runtime layered file system (LFS) before recursively running one or more of steps 231-241 on the reconstructed file system.

[00103] Scanning system 101 may read network information from the snapshot in order to determine which services within which containers are exposed externally and within which ports they are accessible. For example, scanning system 101 may identify a port on which a vulnerable application is accessible based on known software vulnerabilities for a versions of software application. Scanning system 101 may query network accessibility information via an API provided through a cloud service provider's system and may use it to identify specific vulnerabilities susceptible to attack.

[00104] After performing step 243 on one or more containers, the process may return to Fig. 2A and step 207 (as discussed above).

[00105] **Foundational Techniques**

[00106] Aspects of this disclosure may include establishing a trusted relationship between a source account in a cloud environment and a scanner account. A trusted relationship, as used herein, may refer to a secure communication channel between at least two accounts (e.g., domains). The secure communications channel may include, for example, an administration link, communication link, a connection or communication network configured with security protocols, or any other secure relationship between two networked entities.. A trusted relationship between two accounts may enable user accounts and global groups to be used in a domain other than the domain where the accounts are defined. A source account, as used herein, may refer to a domain, a location on a network server, or anything used to access system resources. A cloud environment, as used herein, may refer to a platform implemented on, hosted on, and/or accessing servers that may be reached via the Internet or other shared network. A scanner account, as used herein, may refer to any type of account associated with a scanner. A scanner, as used herein, may refer to a device for examining, reading, or monitoring electronically accessible information. Fig. 1 illustrates one example of a scanner, which is depicted in an exemplary manner as scanning system 101.

[00107] By way of example, establishing a trusted relationship between a source account in a cloud environment and a scanner account may include an integration process (creating a connection between an account on scanning system 101 and an account on cloud infrastructure 106) discussed with reference to Fig. 2A. By way of one example, scanning system 101 of Fig. 1 may create a trusted relationship between a source account in a cloud environment and a scanner account.

[00108] Aspects of this disclosure may include using the established trust relationship, utilizing at least one cloud provider API to identify workloads in the source account. A cloud provider API, as used herein, may refer to any application program interface that allows an end user to interact with a cloud provider's service. A cloud provider, may include any entity through whom information is accessible over the Internet or other shared network. Workloads, as used herein, may refer to systems, devices, or resources in a network (or available via a network) such as a cloud infrastructure. By way of example, Fig. 1 illustrates examples of a workload: virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115. In some embodiments, scanning system 101 of Fig. 1 may perform a scanning operation on workload (e.g., systems, devices, resources, etc.) in cloud infrastructure 106.

[00109] In some embodiments, scanning system 101 of Fig. 1 may use an API to detect workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115 in the source account.

[00110] Aspects of this disclosure may involve using the at least one cloud provider API to query a geographical location of at least one of the identified workloads. Querying, as used

herein, may refer to a request for data or information from a resource. A geographical location, as used herein, may refer to the data center where this workload is served, a position on Earth, a location in a network, an address of a server or other hardware, or any other information capable of identifying a locus, region, or position. Internet geolocation involves software capable of deducing a geographic position of a device connected to the Internet. For example, the device's IP address may be used to determine the country, city, or ZIP code, determining its geographical location. By way of example, scanning system 101 of Fig. 1 may query any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 for their geographical location using a cloud provider API.

[00111] In some embodiments, the geographic location may include an identifier of a physical site. An identifier, as used herein, may refer to anything that allows for recognition of an object, location, or resource. A physical site, as used herein, may refer to an actual location. By way of example, scanning system 101 of Fig. 1 may query any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 for an identifier of their physical site (e.g., address, postal code, or any other physical descriptor) using a cloud provider API.

[00112] In some embodiments, scanning system 101 of Fig. 1 may use an out-of-band process to reach cloud workloads through a runtime storage layer, in combination with information gathered from cloud provider APIs.

[00113] Aspects of this disclosure may include receive an identification of the geographic location. An identification of the geographic location, as used herein, may refer to a recognition, association, or classification of the geographic location. By way of example, scanning system 101 of Fig. 1 may receive a geographical location, after querying for it, by interrogating one or more workloads, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, or keystores 113A-113D of cloud infrastructure 106. For example, the scanning system 101 of Fig. 1 may receive address, postal code, or any other physical descriptor of any workloads in cloud infrastructure 106. In some embodiments, scanning system 101 of Fig. 1 may receive a listing locating the workloads of cloud infrastructure 106 on a two-dimensional graphical representation.

[00114] In some embodiments, the identification of the geographic location may include an identification of a data center, at least one of a data center name, an Internet Protocol (IP) address, a name of the cloud provider, or ~~a unique identity.~~ a unique identity. A data center, as used herein, may refer to a facility that centralizes an organization's shared IT operations and equipment for the purposes of storing, processing, and/or disseminating data and applications. A data center, for example, may include a dedicated space within a building, or a group of buildings used to house computer systems and associated components, such as

telecommunications and storage systems. An identification of a data center, as used herein, may refer to a recognition, association, or classification of a data center. A data center name, as used herein, may refer to a series of letters or numbers, a word, or set of words or numbers, an address, or a code identifying the data center. An Internet Protocol (IP) address, as used herein, may refer to a unique string of characters that identifies a computer using the Internet Protocol to communicate over a network. A cloud provider, as used herein, may refer to a company that offers any components of cloud computing (e.g., infrastructure as a service (IaaS), software as a service (SaaS) or platform as a service (PaaS)). A unique identity, as used herein, may refer to a distinctive numeric or alphanumeric string that is associated with a single entity within a given system.

[00115] By way of example, scanning system 101 of Fig. 1 may query for and receive a geographical location such as a data center name/city and state, data center address, IP address / specific cloud identifier of the workload, name of cloud provider (e.g., Amazon Web Services), or a specific identifier of information of any of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00116] In some embodiments, scanning system 101 of Fig. 1 may receive a listing indicating locations of the workloads of cloud infrastructure 106 on a two-dimensional graphical representation.

[00117] In some other embodiments, scanning system 101 may receive additional information (in addition to geographic location information) such as whether the workload is Internet-facing and easily accessible to attackers, or if it is private (and may be less critical).

[00118] Aspects of this disclosure may include using the cloud provider APIs to access block storage volumes of the at least one workload. Block storage volumes, as used herein, may refer to data files maintained on Storage Area Networks (SANs) or in cloud-based storage environments. Each storage volume may act as an individual hard drive configured by a storage administrator. Accessing block storage volumes, as used herein, may refer to obtaining, examining, or retrieving data.

[00119] As described with relation to process 210 of Fig. 2B (beginning with step 211 to initiate a connection to cloud infrastructure 106), scanning system 101 may send a second message to user device 102, instructing the user to generate a role. A role may refer to a type associated with a node that may be assigned or defined. The second message may include instructions for the user to follow to generate the role. In step 213, a user may provide (e.g., via a keyboard at user device 102) a role definition to the cloud service provider's system. In some embodiments, the role definition includes read-only permissions and permissions to read a block storage layer (containing block storage volumes). In some embodiments, scanning system 101 provides a role formation template (e.g., an Amazon Web Services CloudFormation

Template) for use with cloud infrastructure 106 to create the necessary role. In step 213, the user may utilize user device 102, for example, by copying and pasting a URL of the template, downloading and uploading the template to the cloud service provider's system, or selecting the template from a list of templates.

[00120] By way of example, scanning system 101 of Fig. 1 may access the block storage volume of any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00121] Aspects of this disclosure may include determining a file-system of the at least one workload. A file-system, as used herein, may refer to a method and/or a data structure that the operating system uses to control how data is stored and retrieved.

[00122] By way of example, scanning system 101 of Fig. 1 may identify the type of file-system of any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106. Some file systems may include NTFS, FAT32, ext3, exFAT, HFS, HFS+, HPFC, UFS, ZFS.

[00123] Aspects of this disclosure may include mounting the block storage volumes on a scanner based on the determined file-system. Mounting the block storage volumes, as used herein, may refer to a process by which the operating system makes files and directories (such as block storage volumes) on a storage device available for users to access via the computer's file system. A scanner, as used herein, may refer to a device for examining, reading, or monitoring electronically accessible information. By way of example, Fig. 1 illustrates one example of a scanner, scanning system 101 of Fig. 1. In some embodiments, mounting may include selecting a driver corresponding to the determined file system. In some embodiments, to mount the block storage volumes on the scanner, may include creating a snapshot of the block storage volumes; and mounting the snapshot of the block storage volumes on the scanner. A snapshot of the block storage volumes, as used herein, may refer to a current state or earlier state of the block storage volumes.

[00124] In some embodiments, the at least one processor may be further configured to encrypt the snapshot of the block storage volumes and mount the encrypted snapshot of the block storage volumes on the scanner. Encrypting a snapshot, as used herein, may refer to a process of encoding information (such as a snapshot). This process may convert the original representation of the information, known as plaintext, into an alternative form known as ciphertext.

[00125] In some embodiments, the scanner may use a privileged account to log in and determine how secure each host is from an inside vantage point. While authenticated scans can successfully discover potential vulnerabilities, they may be limited to the extent they require a privileged account on each scanned host. Furthermore, scans use significant system resources during the test procedures and require opening ports that by themselves pose a security risk.

[00126] Aspects of this disclosure may include activating a scanner at the geographic location. Activating a scanner, as used herein, may refer to starting up or initiating a device for examining, reading, or monitoring data. By way of example, scanning system 101 of Fig. 1 may start up a scanner (a different scanner than scanning system 101) at the geographic location identified by scanning system 101.

[00127] Aspects of this disclosure may include reconstructing from the block storage volumes a state of the workload. Reconstructing, as used herein, may refer to rebuilding or reforming a state of the workload. A state of the workload, as used herein, may refer to a condition of a workload at a certain time. By way of example, scanning system 101 of Fig. 1 may rebuild the block storage volumes as an earlier version of any of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00128] In some embodiments, the reconstructed state of the workload may include at least two of an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content. An indication of an installed application, as used herein, may refer to an identifier of a software. A version of an installed application, as used herein, may refer to a particular form, variant, edition, or revision of a piece or pieces of software. An operating system configuration, as used herein, may refer to a manner in which components are arranged to make up an operating system or a computer system. An application configuration, as used herein, may refer to a manner in which components are arranged to make up an application or a computer system. A profile configuration, as used herein, may refer to a manner in which components are arranged to make up a profile or a computer system. A log, as used herein, may refer to a detailed list of an application information, system performance, or user activities. Database content, as used herein, may refer to any collection of data, or information, that is specially organized for rapid search and retrieval by a computer. In some embodiments, scanning system 101 of Fig. 1 may reconstruct any workload, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 from the block storage volumes to an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content.

[00129] Aspects of this disclosure may include assessing the reconstructed state of the workload to extract insights. Assessing the reconstructed state of the workload, as used herein, may refer viewing, utilizing, or evaluating the reconstructed state of the workload. Insights, as used herein, may refer to information or knowledge attained based on an analysis of some data. Extracting insights, as used herein, may refer to pulling information or data about, for example, a vulnerability associated with the workload or a composition of installed applications associated

with the workload. By way of example, scanning system 101 of Fig. 1 may evaluate the reconstructed state of any of workloads, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 to pull valuable data and information.

[00130] In some embodiments, insights may include at least one of a vulnerability associated with the workload or a composition of installed applications associated with the workload. A vulnerability, as used herein, may refer to any weakness within an organization's information systems, internal controls, or system processes that can be exploited. A composition of installed applications associated with the workload, as used herein, may refer to a structure or group of accessible computer programs. By way of example, scanning system 101 of Fig. 1 may evaluate the reconstructed state of any workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 to pull valuable data and information regarding a vulnerability with the workload.

[00131] In some embodiments, a scanner may be deployed at the geographical location. Deploying a scanner, as used herein, may refer to installing, activating, utilizing, or bringing a scanner into effective action . By way of example, Fig. 1 illustrates one example of a scanner, scanning system 101 of Fig. 1 that is deployed.

[00132] In some embodiments, scanning system 101 of Fig. 1 may update the reconstructed state of the workload based on at least one change to the block storage volumes . Updating the reconstructed state of the workload, as used herein, may refer to overhauling or refurbishing the reconstructed state of the workload. A change to the block storage volumes, as used herein, may refer to any shift, alteration, or modification to the block storage volumes.

[00133] **Fig. 3** illustrates a block diagram of method 300 for insight extraction, consistent with disclosed embodiments. In some embodiments, the method may include ten (or more or less) steps:

[00134] Block 302: Establish a trusted relationship between a source account in a cloud environment and a scanner account. In some embodiments, establishing a trusted relationship between a source account in a cloud environment and a scanner account may include an integration process (creating a connection between an account on scanning system 101 and an account on cloud infrastructure 106) discussed with reference to Fig. 2A. By way of one example, scanning system 101 of Fig. 1 may create a trusted relationship between a source account in a cloud environment and a scanner account.

[00135] Block 304: Using the established trust relationship, utilize at least one cloud provider API to identify workloads in the source account. In some embodiments, scanning system 101 of Fig. 1 may use an API to detect workloads such as virtual machines 107A-107D,

databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115 in the source account.

[00136] Block 306: Use the at least one cloud provider API to query a geographical location of at least one of the identified workloads. In some embodiments, scanning system 101 of Fig. 1 may query any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 for their geographical location using a cloud provider API.

[00137] Block 308: Receive an identification of the geographic location. In some embodiments, scanning system 101 may indicate in the map additional information regarding the workload (e.g., a system or device). For example, scanning system 101 may provide: an identification of a data center (e.g., a name or number associated with the data center where the workload is located), at least one of a data center name (e.g., a name or address associated with the data center where the workload is located), Internet Protocol (IP) address, name of the cloud provider, or a unique identity (e.g., any other information related to the workload that may be used).

[00138] Block 310: Use the cloud provider APIs to access block storage volumes of the at least one workload. In some embodiments, scanning system 101 of Fig. 1 may access the block storage volume of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106. In some embodiments, accessing the block storage may include taking the mean of generating snapshots.

[00139] Block 312: Determine a file-system of the at least one workload. In some embodiments, scanning system 101 of Fig. 1 may identify the type of file-system any of workloads (virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106).

[00140] Block 314: Mount the block storage volumes on a scanner based on the determined file-system. In some embodiments, to mount the block storage volumes on the scanner, may include instructing the operating system to map a directory structure logically to a physical storage device. A storage volume may be mounted after it is attached and formatted for use by a server's operating system. Mounting may include creating a snapshot of the block storage..

[00141] Block 316: Activate a scanner at the geographic location. In some embodiments, scanning system 101 of Fig. 1 may start up a scanner (a different scanner than scanning system 101) at the geographic location identified by scanning system 101.

[00142] Block 318: Reconstruct from the block storage volumes a state of the workload. In some embodiments, scanning system 101 of Fig. 1 may rebuild the block storage volumes to

an earlier version of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00143] Block 320: Assess the reconstructed state of the workload to extract insights. In some embodiments, scanning system 101 of Fig. 1 may evaluate the reconstructed state of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 to derive valuable data and information.

[00144] **Vulnerability Management Techniques**

[00145] **Fig. 4** depicts a cybersecurity system performing a side scanning function to protect against potential vulnerabilities. Referring to Fig. 4, a processor 400 configured to identify in a cloud environment one or more block storage volumes 401A-C communicatively connected to a cloud provider API where the block storage volumes 401A-C may be contained. When a block storage volume, e.g., 401A is identified, the processor 400 may be configured to perform the function of identifying the software installed ~~403A~~ 403A-C in or relating to block storage volume 401. Upon identifying installed software ~~403A~~ 403A-C, the processor may be further configured to identify the version of software ~~405A~~ 405A-C operating in the installed software ~~403A~~ 403A-C.. Based on the installed software ~~403A~~ 403A-C and the version of the software ~~405A~~ 405A-C, the processor may be configured to identify a listing of known vulnerabilities ~~407~~ 407A-C that may relate to the installed software ~~403A~~ 403A-C and the version installed ~~405A~~ 405A-C. When the listing of known vulnerabilities 407 is obtained, the system may list the known vulnerabilities ~~409A~~ 409A-C for that software version ~~411A~~ 411A-C, which can be communicated to the processor 400 of the end user. This information may be processed and displayed to an end user using processor 400.

[00146] **Fig. 5** of the disclosed embodiments describes a method of operating a cybersecurity system performing a side scanning function to protect against potential vulnerabilities. Referring to Fig. 5, a processor unit 500 may be configured to perform the method . When initiated either manually or automatically, processor unit 500 may be communicatively connected with a cloud provider API 501 to initiate a cybersecurity function as disclosed above. Contained digitally on a memory device and accessed by a cloud provider API 501 is a series of block storage volumes 503. Processor unit 500 may be capable of performing a vertical and horizontal scan for network access or security information of a block storage volume 503 to detect potential vulnerabilities to the block storage volume 503 or any processing unit 500 designed to access said block storage volume 503.

[00147] Based on the accessed block storage volume 500, processor unit 500 with the disclosed embodiments of a cybersecurity system may perform an identification of the type of software installed 505 connected with block storage volume 503. Following identification of the type of software installed 505, the processing unit may then identify an installed version 507 of

said installed software 505 for a version comprised of a unique identifier based on a combination of letters, numbers, or similar unique identifiers.

[00148] Upon recognition of an installed software version 507 of the installed software 505, processing unit 500 with the disclosed embodiments of the disclosed scanning system 101 may provide a list of known vulnerabilities 509 for review by said scanning system 101 and/or its end user and maintainer of said block storage volumes 503. Based on the list of known vulnerabilities 509, the disclosed scanning system 101 or its end user may identify one or more ports of accessibility to said block storage volume 503 that may be accessed by a known, associated vulnerability 513 from listed vulnerabilities 509 to determine an avenue for potential vulnerability 513 to access and infiltrate block storage volume 503.

[00149] In some embodiments, a cyber security scanning system for a cloud environment 101 may include a processor 500 to operate said system. This processor 500 may include central processing units and other similar computer-enabling equipment for processing and executing commands based on the information inputted to said system. The processor 500 may be communicatively connected to a computer network or series of networks to accomplish said cyber security function.

[00150] As an example embodiment, a processor unit 500 may be configured to use a cloud-provider API that may communicate with one or more specified computer-readable media across a digital network. This can be accomplished through internet protocols, internet control message protocols, transmission control protocol, or user datagram protocol. Cloud provider API 501 may be one of several forms of middleware, interface, middle layer, or other systems of interfacing applications. A processor 300 may be one or more computer processing units, central processing unit, server, microcomputer, mainframe, and any other manifestation of digital computing.

[00151] Further to one of several possible embodiments, a cloud-provider API 501 may be configured to access a block-storage volume 503 of a workload maintained in a cloud-storage environment. This may be accomplished through a system of computer-readable media communicatively connected. Said block-storage volume 303 may be contained on a Storage Area Network (SAN) or similar cloud-based memory storage environment. The block storage volume 503 may be contained in smaller storage volumes with an associated identifier unique to that portion of said block storage volume 503. In some embodiments, the block-storage volume 503 of a workload may have multiple paths for the storage volume to be reaggregated and retrieved quickly.

[00152] Among several embodiments, a scanning system 101 may comprise a system for identifying an installed software application in the accessed block-storage volume 503. This identification of installed software may be accomplished by accessing installed software 505 files through signature verification, root license, or authorized user lists. The installed software

505 may be located and identified within applications such as file storage, database storage, and virtual machine file system volumes. The identification of said installed software application 507 may be processed, analyzed, and communicated to the scanning system 101 for processing, cataloging, and protection through encryption and various methods of layered cyber defense.

[00153] Further, the scanning system 101 described herein may include functionality to analyze installed software applications to determine the associated software version. The software application version 507 may identify the software version based on unique version name, unique version number, and may be based on unique states of the currently installed computer software 505.

[00154] One of many embodiments disclosed above may include the scanning system 101 having the ability to access a data structure of known software vulnerabilities 509 for a plurality of versions of software applications. The known software vulnerabilities 509 may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software. The scanning system 101 may be able to access and identify software vulnerabilities for mitigation, rectification, correction, and fortification.

[00155] In one embodiment, a cybersecurity system may also perform scanning according to scanning system 101 by performing a lookup of the identified installed software version 507 in the data structure to identify known vulnerabilities 509. This function can be performed by the scanning system 101 according to Fig. 1 performing a query of the installed software 505 for unique version number or designator and comparing to, amongst many things, a set of likely or potential vulnerabilities to that software version for potential deficiencies or cybersecurity threats known or suspected to similar software types and versions. This query may be performed according to a predetermined set of values, to include previously identified unique version numbers or designators that may contain the known list of previously identified vulnerabilities.

[00156] Among many embodiments, embodiments of the disclosed scanning system 101 may query the cloud provider API 501 to determine network accessibility information 511 related to the workload. In order to accomplish this query of the cloud provider API 301, the scanning system 101 may involve index of search results and display of said search results, followed by processing and grouping search results. Network accessibility information 511 may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications,

services, capabilities, and specific processes such as virtual machines, databases, containers, or Hadoop nodes, among others.

[00157] If the system detects a vulnerable application 513, one embodiment may identify one or more port on which said vulnerable application is accessible. In one of several embodiments, the scanning system 101 may detect a vulnerable application in one or more computation process. In another embodiment, the cybersecurity system may perform a network accessibility query in a separate process. Further, a disclosed embodiment may perform these separate functions in subsequent and sequential steps of the same process. A person having skill would understand that an authorized user or an authorized scanning system 101 can perform these functions concurrently and subsequently by an authorized user or an authorized cybersecurity system while performing the same function as the disclosed embodiment.

[00158] Upon gathering network accessibility information and the identified port to identify one or more vulnerabilities susceptible to attack from outside the workload, a disclosed embodiment would have the functionality to perform processes to gather, display, and mitigate a discovered vulnerability in order to minimize the likelihood and effectiveness of a cyber threat outside of and attempting to access a workload through a known or previously encountered type of cyber threat. This functionality may include collecting and organizing the vulnerabilities according to type or category of vulnerability, displaying the gathered data for an end user or maintainer, and implementing security features automatically or manually by a user or maintainer such as security patches, password or passcode changes and suggestions for users to do the same, and malicious code eradication.

[00159] As one of several possible embodiments, the scanning system 101 may also, upon identification of one or more vulnerabilities, implement remedial actions via one or more processors. Remedial actions may include, among other things, notification to an end user of an identified threat, compensation through a revised security code to mitigate the potential threat, publication of the identified threat and vulnerability in a log or record of detected vulnerabilities, and communication of the sensed vulnerability and threat to a server operator or maintainer to fortify the protections of workloads existing on similar environments.

[00160] In a similar embodiment to above, the remedial measure may include transmission of an alert to a device associated with an administrator. The alert may be, amongst others, written, auditory, and visual for processing and use by an administrator of said scanning system 101. Said administrator may take action based on the received alert, to include eliminating the cyber threat through mitigation measures, change in cybersecurity posture, or removing the workload from the cybersecurity threat environment.

[00161] In another embodiment, a query of the cloud provider API 501 to determine network accessibility information related to the workload may be performed by at least one processor 500 configured to examine data sources associated with the workload. The data

examined may include user data, system processing data, accessibility data, clock cycles, storage input/output, or similar data processors. A query of the cloud provider API 501 may be automated or manually-initiated. Based on the said query, the network accessibility information 511 related to the workload may change based on the data sources associated with said workload.

[00162] In the above processor and similar embodiments, further configuration may include a process to determine network accessibility information 511 based on the examined data sources. The examined data sources may include various cloud-based workloads, internet protocols, transmission communication protocols, or other methods and systems of memory and data storage.

[00163] As an exemplary embodiment, network accessibility information 511 includes at least one of: data from an external data source, cloud provider information, or at least one network capture log. These embodiments of an external data source may include data from the operating environment of the cloud-based environment, an external operating system for a computer processing unit 500, or other similar computer readable media. Cloud provider information may further include information that may identify the network accessibility information 511 vertically or horizontally to fully describe the associated workload. Network capture logs may be automated or manually updated to include possible vulnerabilities 513 and threats to the cloud-based storage medium.

[00164] Further, the disclosed embodiments may include an installed software application 505, with the at least one processor 500 configured to extract data from at least one of operating system packages, libraries, or program language libraries. This data may be extracted through a system query, random access algorithm, or similar automated process. Operating system packages may include systems operable on Microsoft, Apple, Linux, and similar operating systems. Libraries may consist of a series of files, folders, and databases of information stored on one of any indexed data repositories. A program language library may contain several of an exemplary program languages including but not limited to Javascript, Swift, Scala, Go, Python, Elm, Ruby, C#, C++ and other similar sources of software code.

[00165] As another exemplary embodiment, a scanning system 101 may also include a processor 500 configured to identify installed software application 505 based on the extracted data. The processor 500 may perform only this function or this function among many to accomplish the layered cybersecurity defense described herein this disclosed cloud-based security environment. The identification of installed software by said processor may include identifying the software by unique version number or designator, recognized source code, metadata associated with the installed software application files, or similar software-identifying information.

[00166] One embodiment may include an additional function wherein the installed software application 505 that has been identified includes one or more scripts. These scripts may be processed through various computer readable languages to include Javascript, C#, C++ and other forms of computer code.

[00167] One embodiment contemplated by the disclosed scanning system 101 may also include data structure includes aggregated vulnerability data 309. This aggregated vulnerability data 509 may be compiled by an end user or maintainer from within the cloud-based environment of the current data structure as well as compilation from similar data sets and aggregation of common threats to data structures likely to experience similar vulnerabilities. This aggregation of vulnerability data 509 may be contained within the data structure and it may be collectively aggregated to provide for a more robust and layered cybersecurity defense posture.

[00168] In an embodiment of the disclosed scanning system 101, the aggregated vulnerability data 509 may include data from one or more third-party vendors. These vendors may include operators of the cloud-based server environment, providers of networking and internet communication, methods of layered authentication, and other similar providers of services directly related and in communication with the cloud-based cybersecurity system.

[00169] As an additional exemplary embodiment, the aggregated vulnerability data 309 may include data collected by a scanner. This scanner may involve use of continuous or periodic monitoring of the workload. The scanner may perform security screenings of the various workloads vertically or horizontally to identify network identification information, port accessibility, and associated vulnerabilities. Any scan performed may be communicated to the scanning system 101 that may be responsible for performing and logging the results of the scan and may be able to initiate follow-on processes and protocols to protect the data contained in the workload that is the subject of scanning.

[00170] An embodiment of the disclosed scanning system 101 may also include aggregated vulnerability data 509 that may include at least one of an advisory, an exploit, a security announcement, or a known bug. An advisory may include notification to a system maintainer or user of the potential vulnerability, may log notice of the advisory, and may recommend possible user or maintainer actions to potentially address said advisory. An exploit may further consist of an automated system response designed to take advantage of the sensed vulnerability data. The exploit can be further reflected in the aggregated vulnerability data and protocols can be written into the cybersecurity infrastructure to prevent said exploit from gaining access and permissions to unauthorized areas of the workload storage environment.

[00171] A person having ordinary skill in the art would appreciate the above described embodiments are among many potential embodiments, to include a method of operating a

scanning system 101 similar to the one described above. A disclosed embodiment contemplates this method to be accomplished through manual user operation, automated computer processes, or similar manners of operation. These manners of operation and those contemplated similar to them would allow the scanning system 101 described and disclosed to execute its operations as the system above describes.

[00172] One of several embodiments of the disclosed method of operating a scanning system 101 may include a cyber security scanning system for a cloud environment 101 may include a processor 500 to operate said system. This processor may include central processing units and other similar computer-enabling equipment for processing and executing commands based on the information inputted to said system. The processor may be communicatively connected to a computer network or series of networks to accomplish said cyber security function.

[00173] As an example embodiment, a processor unit 500 may be configured to use a cloud-provider API 501 that may communicate with one or more specified computer-readable media across a digital network. This can be accomplished through internet protocols, internet control message protocols, transmission control protocol, or user datagram protocol. Cloud provider API 501 may be one of several forms of middleware, interface, middle layer, or other systems of interfacing applications. A processor may be one or more computer processing units, central processing unit, server, microcomputer, mainframe, and any other manifestation of digital computing.

[00174] Further to one of several possible embodiments, a cloud-provider API 501 may be configured to access a block-storage volume 503 of a workload maintained in a cloud-storage environment. This may be accomplished through a system of computer-readable media communicatively connected. Said block-storage volume 503 may be contained on a Storage Area Network (SAN) or similar cloud-based memory storage environment. The block storage volume 503 may be contained in smaller storage volumes with an associated identifier unique to that portion of said block storage volume 503. In some embodiments, the block-storage volume 503 of a workload may have multiple paths for the storage volume to be reaggregated and retrieved quickly.

[00175] Among several embodiments, a method of operating a scanning system 101 may comprise a system for identifying an installed software application 505 in the accessed block-storage volume 503. This identification of installed software 505 may be accomplished by accessing installed software files through signature verification, root license, or authorized user lists. The installed software 505 may be located and identified within applications such as file storage, database storage, and virtual machine file system volumes. The identification of said installed software application 505 may be processed, analyzed, and communicated to the

method of operating a scanning system 101 for processing, cataloging, and protection through encryption and various methods of layered cyber defense.

[00176] Further, a method of operating a scanning system 101 described herein may include functionality to analyze installed software applications 505 to determine the associated software version 507. The software application version 507 may identify the software version based on unique version name, unique version number, and may be based on unique states of the currently installed computer software.

[00177] One of many embodiments disclosed above may include the method of operating a scanning system 101 having the ability to access a data structure of known software vulnerabilities 509 for a plurality of versions of software applications 507. The known software vulnerabilities 509 may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software. The cyber security system may be able to access and identify software vulnerabilities 309 for mitigation, rectification, correction, and fortification.

[00178] In one embodiment, a method of operating a scanning system 101 may also perform scanning according to scanning system 101 by performing a lookup of the identified installed software version 507 in the data structure to identify known vulnerabilities 309. This function can be performed by the scanning system 101 according to Fig. 1 performing a query of the installed software 505 for unique version number 507 or designator and comparing to, amongst many things, a set of likely or potential vulnerabilities to that software version for potential deficiencies or cybersecurity threats known or suspected to similar software types and versions. This query may be performed according to a predetermined set of values, to include previously identified unique version numbers or designators that may contain the known list of previously identified vulnerabilities.

[00179] Among many embodiments, embodiments of the disclosed method of operating a scanning system 101 may query the cloud provider API 501 to determine network accessibility information 511 related to the workload. In order to accomplish this query of the cloud provider API 501, the scanning system 101 may involve index of search results and display of said search results, followed by processing and grouping search results. Network accessibility information 511 may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, or Hadoop nodes, among others.

[00180] If the system detects a vulnerable application, one embodiment may identify one or more port on which said vulnerable application is accessible. In one of several embodiments, the scanning system 101 may detect a vulnerable application in one or more computation process. In another embodiment, the scanning system 101 may perform a network accessibility query in a separate process. Further, a disclosed embodiment may perform these separate functions in subsequent and sequential steps of the same process. A person having skill would understand that an authorized user or an authorized cybersecurity system may perform these functions concurrently and subsequently while performing the same function as the disclosed embodiment.

[00181] Upon gathering network accessibility information 511 and the identified port 513 to identify one or more vulnerabilities susceptible to attack from outside the workload, a disclosed embodiment would have the functionality to perform processes to gather, display, and mitigate a discovered vulnerability in order to minimize the likelihood and effectiveness of a cyber threat outside of and attempting to access a workload through a known or previously encountered type of cyber threat. This functionality may include collecting and organizing the vulnerabilities according to type or category of vulnerability, displaying the gathered data for an end user or maintainer, and implementing security features automatically or manually by a user or maintainer such as security patches, password or passcode changes and suggestions for users to do the same, and malicious code eradication.

[00182] A person having ordinary skill in the art would similarly understand that the above disclosed system can be disclosed using a suitable non-transitory computer-readable medium performing each of the disclosed functions. The disclosed embodiment may include a non-transitory computer readable medium with a scanning system 101 similar to the one described above on a non-transitory computer-readable medium. A disclosed embodiment contemplates this system to be accomplished through a medium that may contain a central processing unit, virtual machine, or a similar non-transitory medium. These manners of operation and those contemplated similar to them would allow the scanning system 101 described and disclosed to execute its operations as the system above describes.

[00183] One of several embodiments of the disclosed non-transitory computer readable medium with a scanning system 101 may include a cyber security scanning system for a cloud environment 101 may include a processor 500 to operate said system. This processor 500 may include central processing units and other similar computer-enabling equipment for processing and executing commands based on the information inputted to said system. The processor 500 may be communicatively connected to a computer network or series of networks to accomplish said cyber security function.

[00184] As an example embodiment, a processor unit 300 may be configured to use a cloud-provider API 501 that may communicate with one or more specified computer-readable

media across a digital network. This can be accomplished through internet protocols, internet control message protocols, transmission control protocol, or user datagram protocol. Cloud provider API 501 may be one of several forms of middleware, interface, middle layer, or other systems of interfacing applications. A processor 500 may be one or more computer processing units, central processing unit, server, microcomputer, mainframe, and any other manifestation of digital computing.

[00185] Further to one of several possible embodiments, a cloud-provider API 501 may be configured to access a block-storage volume 503 of a workload maintained in a cloud-storage environment. This may be accomplished through a system of computer-readable media communicatively connected. Said block-storage volume 503 may be contained on a Storage Area Network (SAN) or similar cloud-based memory storage environment. The block storage volume 503 may be contained in smaller storage volumes with an associated identifier unique to that portion of said block storage volume 503. In some embodiments, the block-storage volume 503 of a workload may have multiple paths for the storage volume to be reaggregated and retrieved quickly.

[00186] Among several embodiments, a non-transitory computer readable medium with a scanning system 101 may comprise a system for identifying an installed software application 505 in the accessed block-storage volume 503. This identification of installed software 505 may be accomplished by accessing installed software files through signature verification, root license, or authorized user lists. The installed software 505 may be located and identified within applications such as file storage, database storage, and virtual machine file system volumes. The identification of said installed software application 505 may be processed, analyzed, and communicated to the method of operating a scanning system 101 for processing, cataloging, and protection through encryption and various methods of layered cyber defense.

[00187] Further, a non-transitory computer readable medium with a scanning system 101 described herein may include functionality to analyze installed software applications 505 to determine the associated software version 507. The software application version 507 may identify the software version based on unique version name, unique version number, and may be based on unique states of the currently installed computer software.

[00188] One of many embodiments disclosed above may include the non-transitory computer readable medium with a scanning system 101 having the ability to access a data structure of known software vulnerabilities 509 for a plurality of versions of software applications 507. The known software vulnerabilities 509 may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords,

and previously infected software. The scanning system 101 may be able to access and identify software vulnerabilities for mitigation, rectification, correction, and fortification.

[00189] In one embodiment, a non-transitory computer readable medium with a scanning system 101 may also perform scanning according to scanning system 101 by performing a lookup of the identified installed software version in the data structure to identify known vulnerabilities 509. This function can be performed by the scanning system 101 according to Fig. 1 performing a query of the installed software for unique version number 307or designator and comparing to, amongst many things, a set of likely or potential vulnerabilities 309 to that software version 307 for potential deficiencies or cybersecurity threats known or suspected to similar software types and versions. This query may be performed according to a predetermined set of values, to include previously identified unique version numbers or designators that may contain the known list of previously identified vulnerabilities 509.

[00190] Among many embodiments, embodiments of the disclosed non-transitory computer readable medium with a scanning system 101 may query the cloud provider API 501 to determine network accessibility information 511 related to the workload. In order to accomplish this query of the cloud provider API 501, the scanning system 101 may involve index of search results and display of said search results, followed by processing and grouping search results. Network accessibility information 511 may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, or Hadoop nodes, among others.

[00191] If the system detects a vulnerable application, one embodiment may identify one or more ports 515 on which said vulnerable application is accessible. In one of several embodiments, the scanning system 101 may detect a vulnerable application in one or more computation processes. In another embodiment, the scanning system 101 may perform a network accessibility query in a separate process. Further, a disclosed embodiment may perform these separate functions in subsequent and sequential steps of the same process. A person having skill would understand that an authorized user or an authorized cybersecurity system can perform these functions concurrently and subsequently while performing the same function as the disclosed embodiment.

[00192] Upon gathering network accessibility information 511 and the identified port 515 to identify one or more vulnerabilities susceptible to attack from outside the workload, a disclosed embodiment would have the functionality to perform processes to gather, display, and mitigate a discovered vulnerability in order to minimize the likelihood and effectiveness of a cyber threat outside of and attempting to access a workload through a known or previously encountered type of cyber threat. This functionality may include collecting and organizing the

vulnerabilities according to type or category of vulnerability, displaying the gathered data for an end user or maintainer, and implementing security features automatically or manually by a user or maintainer such as security patches, password or passcode changes and suggestions for users to do the same, and malicious code eradication.

[00193] Another disclosed embodiment may include a method of using a cloud provider API 501, accessing block storage volume 503 of a workload maintained in a cloud storage environment. The method may include manual user operation, automated system operation, systematic and random operating parameters for the system to provide its cybersecurity and similar security functions. The accessed block storage volume may be contained as a collection of block units organized together, may be a series of individual blocks that can be reorganized to form a new storage volume, and may have the ability to be disaggregated and reaggregated as necessary to accomplish its storage and cybersecurity functions.

[00194] The disclosed method may further comprise a system that may analyze the identified installed software application 505 to determine an associated software version 507. In analyzing an installed software application 305, the disclosed embodiment may include a method of querying the installed software application to access a software version 507 that may include a series of letters, numbers, and other identifying characters to differentiate the version of software currently operative and its associated identifying characteristics and capabilities.

[00195] A method of the disclosed may also include accessing a data structure of known software vulnerabilities 509 for a plurality of versions of software applications 507. The disclosed method may determine a data structure of known software vulnerabilities 509 of one among many operative versions of software installed in the monitored storage system as compared to historical data of similar storage systems.

[00196] The disclosed method as described herein may further include performing a lookup of the identified installed software version 507 in the data structure to identify known vulnerabilities 509. The lookup of identified installed software 507 may be previously indexed based on unique version identifier, and the indexed information may include known vulnerability information 509, network accessibility information 511, network protocols, and administrator identifier information. Based on the said indexed known vulnerabilities 509, the scanning system 101 disclosed may take actions to mitigate a cybersecurity threat and reinforce cybersecurity defenses at the type of cybersecurity threat, as previously discussed.

[00197] Further to the above, a disclosed method may also include querying the cloud provider API 501 to determine network accessibility information 511 related to the workload. A query of the cloud provider API 501 to determine network accessibility information 511 related to the workload may be automatically or manually initiated. Network accessibility information 511 similar to the above embodiment may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads

associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, and Hadoop nodes, among others.

[00198] A disclosed method of one embodiment also may provide a method of identifying at least one port 515 on which the vulnerable application is accessible. The identification of one port 515 on which the vulnerable application is accessible may be for the purpose of identifying a cyber weakness, and it may include the ability to change the port currently accessible to mitigate the said cyber weakness.

[00199] Yet another disclosed embodiment of a disclosed method may include using the network accessibility information 511 and the identified at least one port 515 to identify one or more vulnerabilities 513 susceptible to attack from outside the workload. The vulnerability 513 identified as susceptible to attack from outside the workload may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software.

[00200] A method of the disclosed embodiment may further include implementing a remedial action in response to the identified one or more vulnerabilities 513. Said remedial action may include, among other things, notification to an end user of an identified threat, compensation through a cybersecurity patch, publish of the identified threat and vulnerability in a log or record of detected vulnerabilities, and communication of the sensed vulnerability and threat to a server operator and maintainer to fortify the protections of workloads existing on similar environments.

[00201] The disclosed method may also include wherein the remedial measure includes transmitting an alert to a device associated with an administrator. Said alert may be, amongst others, written, auditory, and visual for processing and use by an administrator of said cybersecurity system. Said alert may further be logged and catalogued for future identification of known threats and vulnerabilities to similar software application versions.

[00202] A disclosed method of the present embodiment may also provide querying the cloud provider API 501 to determine the network accessibility information 311 related to the workload and examining data sources associated with the workload. A query of the cloud provider API 501 to determine network accessibility information 511 related to the workload may be automatically or manually initiated. Network accessibility information 511 similar to the above embodiment may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, and Hadoop nodes, among others.

[00203] The method of the above disclosed embodiment may further determine network accessibility information 511 by determining the network accessibility information 511 based on the examined data sources. The data examined may include user data, system processing data, accessibility data, clock cycles, storage input/output, and similar processes. A query of the cloud provider API 501 may be automated or manually-initiated. Based on the said query, the network accessibility information 511 related to the workload may change based on the data sources associated with said workload.

[00204] In the method of one disclosed embodiment, network accessibility information 511 may include at least one of data from an external data source, cloud provider information, or at least one network capture log. These embodiments may include data from the operating environment of the cloud-based environment, an external operating system for a computer processing unit, or other similar computer readable media. Network capture logs may be automated or manually updated to include possible vulnerabilities and threats to the cloud-based storage medium.

[00205] A method of the disclosed embodiment may also include identifying the installed software application 507 by extracting data from at least one of OS packages, libraries, or program language libraries. This data may be extracted through a system query, random access algorithm, or similar automated process. Operating system packages may include systems operable on Microsoft, Apple, Linux, and similar operating systems. Libraries may consist of a series of files, folders, and databases of information stored on one of any indexed data repositories. A program language library may contain several of an exemplary program languages including but not limited to Javascript, Swift, Scala, Go, Python, Elm, Ruby, C#, C++ and other similar sources of software code.

[00206] Another method of the disclosed embodiment may identify the installed software application 507 based on the extracted data. The method of the processor 500 may perform only this function or this function among many to accomplish the layered cybersecurity defense described herein this disclosed cloud-based security environment.

[00207] A further disclosed method of the present embodiment may include at least one processor 500 further configured to identify a version 507 of the installed software application 505. The method of the processor may perform only this function or this function among many to accomplish the layered cybersecurity defense described herein this disclosed cloud-based security environment.

[00208] **Prioritization Techniques**

[00209] Fig. 6 is a block diagram of method 600 for risk prioritization, consistent with disclosed embodiments.

[00210] Aspects of this disclosure may include a cloud-based cybersecurity system for assessing internet exposure of a cloud-based workload with embodiments configured to access

at least one cloud provider API to determine a plurality of entities capable of routing and/or filtering traffic in a virtual cloud environment associated with a target account containing the workload. Routing traffic, as used herein, may refer to the process of selecting a path for traffic (e.g., cells, blocks, frames, packets, calls, messages, or other units of data) in a network or between or across multiple networks. The cloud-based cybersecurity system, in some embodiments, may comprise at least one processor configured to assess internet exposure as described herein.

[00211] A cloud provider API, as used herein, may refer to any application program interface that allows the end user to interact with a cloud provider's service. Workloads, as used herein, may refer to devices in a network. By way of example, workloads may include systems, devices, resources in a cloud infrastructure. By way of example, Fig. 1 illustrates examples of a workload, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, or load balancer 115. In some embodiments, scanning system 101 of Fig. 1 may perform a scanning operation on any workload (e.g., systems, devices, resources, etc.) in cloud infrastructure 106.

[00212] In some embodiments, scanning system 101 of Fig. 1 may use an API to detect workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115 in the source account. In some embodiments, workloads may exist as virtual machines, while in other embodiments, workloads may exist as discrete, physical devices. For example, in some embodiments, a layer 3 routing system could be implemented as a physical device (e.g., a router or switch) or a virtual device (e.g., a routing system instantiated as a virtual device on a computer).

[00213] Scanning system 101 may query devices and systems capable of routing and filtering traffic (e.g., load balancer 115, routers, switches, firewalls, security groups, API Gateways and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations. These devices can be cloud native and / or unmanaged (e.g. NGINX proxy)

[00214] A cloud environment, as used herein, may refer to a platform implemented on, hosted on, and/or accessing servers that are accessed over the Internet. A scanner account, as used herein, may refer to any type of account associated with a scanner. A scanner, as used herein, may refer to a device for examining, reading, or monitoring something. By way of example, Fig. 1 illustrates one example of a scanner, scanning system 101 of Fig. 1. A target account containing the workload, as used herein, may refer to a location on a network server selected as an aim of connection, for example. In some embodiments, scanning system 101 of Fig. 1 may determine entities capable of routing traffic.

[00215] In some embodiments, the plurality of entities includes a virtual network appliance. A virtual network appliance, as used herein, may refer to a machine focused on virtualizing network functionality. In some embodiments, a typical network virtual appliance may include functionality directed to various layer four to seven. That is, such appliances may implement functionality associated with OSI model layers 4-7 (Transport, Session, Presentation, and Application). Such functionality may include, for example, a firewall, WAN optimizer, application delivery controllers, routers, load balancers, IDS/IPS, proxies, or SD-WAN edge device. In some embodiments, scanning system 101 of Fig. 1 may determine a virtual network appliance capable of routing traffic.

[00216] In some embodiments, the virtual network appliance is at least one of a load balancer, a firewall, a proxy, or a router. A load balancer, as used herein, may refer to any device that acts as a reverse proxy and distributes network or application traffic across a number of servers. In some embodiments, load balancer 115 may comprise one or more systems that balance incoming requests between the different systems and devices of cloud infrastructure 106. For example, load balancer 115 may be configured to determine usage (e.g., processor load, used storage capacity) of systems or devices in cloud infrastructure 106 to assist in determining where to route an incoming request from network 105 to store data, perform processing, or retrieve data.

[00217] Load balancer 115 may be configured to receive an incoming request from user device 102. Upon receipt of the request, load balancer 115 may consult a data store (part of or separate from load balancer 115; not pictured) to determine usage or forecasted usage of various systems or devices in cloud infrastructure 106, and may forward the request to the systems or devices having the lowest usage or forecasted usage. A firewall, as used herein, may refer to a device or system that monitors incoming and outgoing network traffic and permits or blocks data packets based on a set of security rules. A proxy, as used herein, may refer to a device or system that translates traffic between networks or protocols. In some embodiments, a proxy may be implemented as an intermediary a device or system separating end-user devices from destinations devices (e.g., web servers) by routing traffic through them. Proxies (or proxy servers) may be configured to provide varying levels of functionality, security, and privacy depending on use case, needs, or company policy. A router, as used herein, may refer to a device or system that connects a local network to the internet.

[00218] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to query the at least one cloud provider API to determine at least one networking configuration of the entities. Querying, as used herein, may refer to a request for data or information from a resource. A cloud provider API, as used herein, may refer to any application program interface that allows an end user to interact with a cloud provider's service. A cloud provider may include any entity through whom information is accessible over

the Internet or other shared network. At least one networking configuration of the entities, as used herein, may refer to a manner in which components are arranged to make up an operating system, networking system, or a computer system. In some embodiments, scanning system 101 of Fig. 1 may send a request to determine at least one networking configuration of the entities.

[00219] In some embodiments, the networking configuration is at least one of a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration.

[00220] A routing configuration, as used herein, may refer to a manner in which components are arranged to make up a routing system. For example, a routing configuration may provide for dynamic or static "routes" between devices on a network or on networks.

[00221] A proxy configuration, as used herein, may refer to a manner in which components are arranged to make up a proxy system. For example, a proxy configuration may provide for a configuration of how a client can route traffic through a central point (e.g., a proxy server), in order to relay multiple clients' requests from a single point on a network.

[00222] A load balancing configuration, as used herein, may refer to a manner in which components are arranged to make up a load balancing system. For example, a loan balancing configuration may define how resources such as uplinks or distributed applications may be utilized by devices on the same (or another) network, in order to balance usage of multiple redundant uplinks or distributed application nodes.

[00223] A firewall configuration, as used herein, may refer to a manner in which components are arranged to make up a firewall system. For example, a firewall configuration may comprise rules that determine what traffic may be forwarded, which devices are permitted to communicate with one another, or what protocols may be used for communication between devices.

[00224] A VPN (virtual private network) configuration, as used herein, may refer to a manner in which components are arranged to make up a VPN system. A VPN may refer to a type of network that extends a private network across a public network and enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network. For example, a VPN configuration may provide rules or other information by which data is sent or received over a single, encrypted route, such that intermediate points cannot determine the content, source, or destination of the data.

[00225] In some embodiments, scanning system 101 of Fig. 1 may send a request to determine at least a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration of the entities.

[00226] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to build a graph connecting the plurality of entities based on the networking configuration. A graph connecting the plurality of entities, as used herein, may

refer to a diagram showing the relation between variable quantities (e.g., entities). In some embodiments, the graph may include any data structures that includes the connections from one entity to another. In some embodiments, scanning system 101 of Fig. 1 may construct a graph or diagram linking the plurality of entities based on the networking configuration. In some embodiments, graph includes a data structure sequentially connecting entities. A data structure sequentially connecting entities, as used herein, may refer to an organization, management, and/or storage format that enables efficient access and modification of data such as entities.

[00227] In some embodiments, a graph may include directional vectors indicating directions of dataflow. Directional vectors, as used herein, may refer to a quantity having direction as well as magnitude (indicating, for example, what kinds of traffic are allowed to flow, what ports are open, what protocols are allowed), especially as determining the position of one point in space relative to another. Directions of dataflow, as used herein, may refer to the movement of data through a system comprised of software, hardware or a combination of both along a particular course. In some embodiments, scanning system 101 of Fig. 1 may construct a graph including a data structure connecting entities sequentially or a graph including directional vectors showing the direction of data flow.

[00228] In some embodiments, to build the graph, the at least one processor may be configured to identify individual entities as nodes and connect the nodes. Nodes, as used herein, may refer to a device in a network (e.g., devices in cloud infrastructure 100). Connecting nodes, as used herein, may refer to linking or bonding devices or data points in a larger network. In some embodiments, the graph includes port numbers associated with the workload. Port numbers, as used herein, may refer to a way to identify a specific process to which an internet or other network message is to be forwarded when it arrives at a server and/or which services are running on a server. In some embodiments, the graph includes a path from the at least one source to the workload. A path from the at least one source to the workload, as used herein, may refer to a string of characters used to uniquely identify a location in a directory structure.

[00229] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to access a data structure identifying services publicly accessible via the Internet and capable of serving as an internet proxy. Services capable of serving as an internet proxy, as used herein, may refer to a system or router that provides a gateway between users and the internet (e.g., network 105). In some embodiments, scanning system 101 of Fig. 1 may access a configuration detecting services over the internet and capable of serving as an internet proxy.

[00230] Aspects of this disclosure may include embodiments configured to integrate the identified services into the graph. Integrating identified services into a graph, as used herein, may refer to the act of bringing together smaller components (the identified services) into a

single system that functions as one (the graph). For example, scanning system 101 may integrate information on services available in each node into the data structure, associated with each node in the data structure.

[00231] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to traverse the graph to identify at least one source originating via the Internet and reaching the workload. Traversing the graph, as used herein, may refer to a device, such as scanning system 101, accessing the systems represented by the nodes in the graph, to determine which systems are accessible and how. For example, scanning system 101 may analyze the graph to detect sources coming from the internet and reaching the workload, by accessing a first system, and traversing between systems in a depth-first, breadth-first, or other manner, until a path to an external network (e.g., the Internet) is reached.

[00232] In some embodiments, the at least one source may be a potential malicious source. In some embodiments, the source may include a code in any part of a software system or script that is intended to cause undesired effects, security breaches or damage to a system. (In some embodiments, the Internet at large may be considered to be a potentially malicious source.)

[00233] Aspects of this disclosure may include embodiments configured to output a risk notification associated with the workload. Outputting a risk notification, as used herein, may refer to sending out data such as a message indicating a possibility of danger. In some embodiments, scanning system 101 of Fig. 1 may produce a notification indicating a risk that is related with the workload. Examples of such risk notifications may include, for example, an electronic mail message, a visual alert, an audio alert, an audio-visual alert,

[00234] In some embodiments, the risk notification includes one or more resolution recommendations. Resolution recommendations, as used herein, may refer to a suggestion or proposal as to the best course of action in view of the identified risk.

[00235] **Fig. 6** illustrates a block diagram of method 600 performed by a processor (e.g., a processor associated with scanning system 101), consistent with disclosed embodiments. In some embodiments, the processor may operate on instructions stored in a non-transitory computer readable medium. In some embodiments, the method may include seven (or more or less) steps:

[00236] Block 602: Access at least one cloud provider API to determine a plurality of entities capable of routing traffic in a virtual cloud environment associated with a target account containing the workload. In some embodiments, scanning system 101 of Fig. 1 may determine entities capable of routing traffic, consistent with the above-disclosed embodiments.

[00237] Block 604: Query the at least one cloud provider API to determine at least one networking configuration of the entities. In some embodiments, scanning system 101 of Fig. 1

may send a request to determine at least one networking configuration of the entities, consistent with the above-disclosed embodiments.

[00238] Block 606: Build a graph connecting the plurality of entities based on the networking configuration. In some embodiments, scanning system 101 of Fig. 1 may construct a graph or diagram linking the plurality of entities based on the networking configuration, consistent with the above-disclosed embodiments.

[00239] Block 608: Access a data structure identifying services publicly accessible via the Internet and capable of serving as an internet proxy. In some embodiments, scanning system 101 of Fig. 1 may access a configuration detecting services over the internet and capable of serving as an internet proxy, consistent with the above-disclosed embodiments.

[00240] Block 610: Integrate the identified services into the graph. In some embodiments, scanning system 101 of Fig. 1 may incorporate the services into the graph, consistent with the above-disclosed embodiments.

[00241] Block 612: Traverse the graph to identify at least one source originating via the Internet and reaching the workload. In some embodiments, scanning system 101 of Fig. 1 may analyze the graph to detect sources coming from the internet and reaching the workload, consistent with the above-disclosed embodiments.

[00242] Block 614: Output a risk notification associated with the workload. In some embodiments, scanning system 101 of Fig. 1 may produce a notification indicating a risk that is related with the workload, consistent with the above-disclosed embodiments.

[00243] **Techniques for Malware Detection Using Secondary System**

[00244] Aspects of this disclosure may provide a technical solution to the challenging technical problem of malware detection on a primary system using a secondary system other than the primary system. In existing technologies, malware detection system may be run on a primary system that itself may be susceptible to malware. The malware may detect existence of the malware detection system and deceive it to avoid being detected. To resolve such a technical problem, a secondary system may be provided to host the malware detection system, which may isolate the malware detection system from potential exposure or influence from the malware that may reside on the primary system. The disclosed technical solutions may increase success rate of malware detection and thus enhance security level of the primary system.

[00245] A cyber security system, as used herein, may refer to a system including any combination of software and hardware for enhancing security of a device, a platform, or another system in a network environment. By way of example, the cyber security may be implemented as a system including scanning system 101 in Fig. 1. As another example, scanning system 101 may include the disclosed cyber security system as a subset. A cloud environment, as used herein, may refer to a computing environment running on a cloud. By way of example, the cloud environment may include or be part of cloud infrastructure 106 in Fig. 1.

[00246] Consistent with disclosed embodiments, at least one processor may be configured to utilize a cloud provider API to access a block storage volume of a workload maintained on a target account in a target system of a cloud storage environment. Utilizing, as used herein, may refer to an operation of using, deploying, enacting, enabling, activating, allocating, invoking, calling, or any operation of putting a thing into use in a computer environment. An API refers to an application programming interface herein. A cloud provider in this disclosure may refer to a cloud service provider (e.g., a service provider of cloud computing, cloud storage, or any services provided to users on demand over a network). A cloud provider API, as used herein, may refer to an API provided, prepared, enabled, activated, programmed, or written by a cloud provider. A block storage volume in this disclosure may refer to a storage volume in a block storage scheme. In some embodiments, a block storage volume may be connected, disconnected, or reconnected to a system (e.g., a physical computer, a virtual machine, a network interface, a cloud, or any combination of hardware or software modules) without interfering (e.g., shutting down, halting, rebooting, or any manner of interrupting) the operation status of the system or other running tasks on the system. For example, a block storage volume may be implemented as a virtual disk. In some embodiments, a block storage volume can be associated with an account of a user of the system, the system that it is connected to, or both.

[00247] Block storage, as used herein, may refer to a data storage scheme or model in which data is saved to storage media in fixed-sized raw data chunks (referred to as “blocks”). The raw data blocks may contain no metadata. Each block may be associated with a unique address as metadata assigned to the block. Storage blocks may be controlled by an operating system (OS) and may be accessed by a protocol (e.g., iSCSI, Fibre Channel, or Fibre Channel over Ethernet).

[00248] A storage volume, as used herein, may refer to an identifiable unit (e.g., a physical unit or a logical, virtual unit) of data storage. The storage volume may be mounted to a device via an operation system and be configured with a specific file system (e.g., New Technology File System) assigned a system-unique name or number that identifies the storage volume. The storage volume may represent a named, logical area of storage that enables users and applications to access data on the underlying device. By way of example, a storage volume may be a logical disk that represents a named, logical area of a physical storage device (e.g., a hard disk drive, solid-state drive, compact disc read-only memory, digital video disk, floppy disk, or any other type of storage device). In some embodiments, a logical storage volume may span multiple physical storage devices (e.g., hard disks) and appear as a single, contiguous storage area that works like a physical volume. Storage volumes may be flexibly configured, such as being expanded, contracted, mirrored, stripped, or adapted to support multiple disks (e.g., redundant array of independent disks).

[00249] A workload, as used herein, may refer to a specific application, service, capability, or a specific amount of work that can be run on a cloud resource, system, or infrastructure. By way of example, a workload may be a virtual machine, a database, a container, a Hadoop node, an application, a storage object, a load balancer, or an IAM (Identity and Access Management) configuration. A cloud storage environment in this disclosure may refer to a computing environment of a cloud computing model (referred to as "cloud storage") that stores data on the Internet through a cloud computing provider that manages and operates data storage as a service. A cloud storage environment may be managed and controlled by an operating system. The operation system may be susceptible to malware and security vulnerabilities, and may become a primary system (or referred to as a "target system" herein) that needs malware detection. By way of example, the cloud storage environment may include or be part of cloud infrastructure 106 in Fig. 1.

[00250] A target system of a cloud storage environment in this disclosure may refer to an operating system or a computing system with an installed operating system, which manages and controls the cloud storage environment or a subsystem of the target system. The target system may maintain one or more accounts for users of the cloud storage environment. A target account of the target system in this disclosure may refer to an account of a user of a cloud infrastructure (e.g., a cloud environment or a cloud storage environment).

[00251] By way of example, to utilize the cloud provider API to access the block storage volume (e.g., existing in storage 111A-111D in Fig. 1) of the workload, the at least one processor (e.g., a processor in scanning system 101 of Fig. 1) may communicate with the cloud storage environment (e.g., cloud infrastructure 106 in Fig. 1) via sending and receiving data (e.g., data packets) over a network (e.g., network 105 in Fig. 1), and enable one or more APIs provided by the cloud provider for accessing the block storage volume. For example, the communication and API enabling may be performed through the process of integration as described in step 201 of Fig. 2. In some embodiments, the at least one processor may create a connection (e.g., by logging in) between a first account on scanning system 101 and a second account on cloud infrastructure 106, then cause the second account to authorize access privileges to the first account via the cloud provider API for accessing storage 111A-111D in cloud infrastructure 106.

[00252] Consistent with disclosed embodiments, the at least one processor may be configured to utilize a scanner at a location of the block storage volume and on a secondary system other than the target system. A scanner, as used herein, may refer to an application, a program, a service, a process, a thread, a function, or any executable codes for performing a scanning process on a computer system to obtain information (e.g., related to structures, vulnerabilities, security issues, or any information of the computer system).

[00253] A location of the block storage volume in this disclosure may refer to a system (e.g., a physical computer, a virtual machine, a network interface, a cloud, a data center, or any combination of hardware or software modules) or an identifier associated with the system that the block storage volume is connected to. In some embodiments, the location of the block storage volume may include at least one of the target account, a secondary system account, a cloud provider account, or a third party account. By way of example, the secondary system account may be an account connected with the target account (e.g., connected by an integration process similar to step 201 in Fig. 2A). The cloud provider account may be an account connected to at least one of the target account or the secondary system account. The third party account may be an account hold by a third party and connected to at least one of the target account, the secondary system account, or the cloud provider account.

[00254] In some embodiments, the at least one processor may determine the location of the block storage volume based on at least one of the target account, a secondary system account, a cloud provider account, or a third party account. By way of example, to determine the location of the block storage volume, the at least one processor may consult a lookup table, a list, or a database that stores a relationship record between an account (e.g., a target account, a secondary system account, a cloud provider account, or a third party account) and a system (e.g., a physical computer, a virtual machine, a network interface, a cloud, or any combination of hardware or software modules) that the block storage volume is connected to. Based on the relationship record, the at least one processor may identify the system as the location using the account as a key.

[00255] A secondary system, as used herein, may refer to a computer system running at an environment or conditions not affected by another system (referred to as a "primary system" or "target system" herein). In some embodiments, the secondary system may have an operating system different from an operating system of the target account. By way of example, the target account may have a WINDOWS® operating system, and the secondary system may have a LINUX® operating system.

[00256] In some embodiments, the secondary system includes at least one of a virtual machine (e.g., any of virtual machines 107A-107D in Fig. 1), a container, or a serverless function. A container, as used herein, may refer to an OS-level virtualization instance assigned with resources by a non-virtual computer operating system (OS). A non-virtual computer operating system (OS) may create multiple isolated user space instances that may function like real computers from the point of view of programs running in them. A computer program running on a non-virtual operating system can access all available resources (e.g., connected devices, files, folders, network shares, CPU power, or any other software or hardware capabilities) of the computer. In contrast, a computer program running inside of a container can only access the resources assigned to the container.

[00257] A serverless function, as used herein, may refer to a computer function hosted in a cloud environment that may allocate resources on demand to perform a function (e.g., scanning) for a user of the cloud environment. The user of the serverless function may be free from concerns of capacity planning, configuration, management, maintenance, fault tolerance, or scaling of containers, virtual machines, or physical servers. When a serverless function is not running, no computing resources may be allocated to it (e.g., using no provisioned server, thus named "serverless"). By way of example, a serverless function may be implemented as any combination of Lambda functions, event sources, and other computing resources. A serverless function may be invoked by the user of a cloud infrastructure (e.g., a cloud environment or a cloud environment).

[00258] By way of example, to utilize the scanner at the location of the block storage volume (e.g., existing in storage 111A-111D in Fig. 1), the at least one processor (e.g., a processor in scanning system 101 of Fig. 1) may communicate with the secondary system (e.g., one or more of virtual machines 107A-107D in Fig. 1) other than the target system (e.g., the main system operating and managing storage 111A-111D) and activate the secondary system for scanning. For example, to activate the secondary system, the at least one processor may transmit a scanner (e.g., in the form of a computer program or executable codes) to the secondary system, and the secondary system may allocate computing resources (e.g., CPU powers, memory space, network ports, or any other software or hardware resources) in preparation of running the scanner. The scanner may be programmed to be stored at the location of the block storage volume or to start the scanning at the location of the block storage volume. For example, the block storage volume may be connected to a particular system in cloud infrastructure 106 in Fig. 1, and the scanner may be stored at that system (e.g., on storage connected to the system). When the scanner begins scanning, the scanner may start the scanning at that system before scanning other systems.

[00259] In some embodiments, to utilize the scanner, the at least one processor may suspend an operation of the scanner after the scan of the block storage volume. Suspending, as used herein, may refer to an operation of pausing, stopping, ceasing, freezing, halting, hanging, terminating, killing, or any manner or temporarily or permanently setting aside an ongoing process or computer program. By way of example, after the scanner completes scanning the block storage volume, the at least one processor may detect the completion by an active manner (e.g., by periodically polling the secondary system for a status of the scanning) or a passive manner (e.g., by receiving a message from the secondary system indicating the completion of the scanning). The at least one process may then send data to the secondary system, the data including a signal or computer-executable codes for suspending the operation of the scanner. After receiving the data, the secondary system may suspend the scanner and release the computing resources previously allocated to the scanner.

[00260] In some embodiments, to utilize the scanner, the at least one processor may modify a pre-utilized scanner at the location of the block storage volume based on information related to the target account to obtain a modified scanner, and then utilize the modified scanner. A pre-utilized scanner in this disclosure may refer to a scanner that has been previously utilized (e.g., on the same secondary system) and not destroyed. In some embodiments, the information related to the target account may include data or parameters indicating a status or a change of the target account, such as, for example, newly created files, deleted files, changed files, or any data indicating a state of the target account. For example, the pre-utilized scanner at the block storage volume may use a configuration file. The configuration file may include data or parameters indicating a previous status (e.g., list of existing files, public network ports, Internet-accessible data channels, or any operation state parameters) of the target account. The information related to the target account may include data indicating a current status of the target system. The current status of the target system may be different from the previous status of the target system (e.g., including changed files, changed public network ports, changed Internet-accessible data channels, or any changed operation state parameters). The at least one processor may modify the pre-utilized scanner by modifying the configuration file to reflect the current status of the target system. Then, the at least one processor may utilize the modified scanner.

[00261] Consistent with disclosed embodiments, the at least one processor may be configured to scan the block storage volume for malicious code, using the secondary system. Scanning, as used herein, may refer to an operation of traversing, checking, inspecting, reading, examining, searching, or any manner of looking up. Malicious code, as used herein, may refer to an application, a computer program, a service, a process, a thread, a function, a script, or any type of executable codes that may be executed on a computer without authorization and with a malicious or harmful intent (e.g., manipulating the computer into dangerous behaviors, writing changes or add-ons to existing computer programs or files, stealing computer files, copying sensitive information without authorization, damaging the computer itself, disabling one or more functions of the computer, or attacking another computer using the computer as a disguise). For example, malicious code may include attack scripts, computer viruses, computer worms, Trojan horses, backdoors, or any other malicious executable content. In some embodiments, the malicious code may include a rootkit. A rootkit in this disclosure may refer to any type of computer software designed to enable access to a computer or an area of its software without authorization.

[00262] By way of example, to scan the block storage volume (e.g., existing in storage 111A-111D in Fig. 1) for malicious code, the secondary system (e.g., one or more of virtual machines 107A-107D in Fig. 1) may run the utilized scanner with allocated computing resources to look up databases (e.g., a blacklist) of known malicious files, look up databases storing

patterns of malicious files and identify the patterns, look up databases of cryptographic hashes that identify malicious files, run potential malicious code in a sandbox (e.g., a confined and isolated environment for running programs without causing changes outside the confined and isolated environment), determine whether there are scanned files that are not in a database (e.g., a whitelist) of known good files, detect changes over time in the scanned environment, or any combination thereof.

[00263] In some embodiments, scanning the block storage volume may include scanning disk-backed memory. A disk-backed memory in this disclosure may refer to a storage space implemented on a disk (e.g., a hard disk) and functioning like a volatile memory from the perspective of a running program. By way of example, scanning the block storage volume may include scanning files stored in the disk-backed memory. In some embodiments, the disk-backed memory may include at least one of a page file or a cache file. A page file, as used herein, may refer to a file containing one or more pages (also referred to as "memory pages" or "virtual pages"), each of the one or more pages being a fixed-length contiguous block of virtual memory and described by a single entry in a page table. A page file may be stored on a disk (e.g., a hard disk) for memory management of a virtual memory used by an operating system. A cache file, as used herein, may refer to a file storing data so that future requests for the stored data can be served faster. The data stored in a cache file may be the result of a computation or a copy of the data stored elsewhere.

[00264] Consistent with disclosed embodiments, the at least one processor may be configured to identify malicious code based on the scan. By way of example, if the secondary system scans the block storage volume by consulting databases of known malicious files and finds a scanned file matches a record in the databases, the at least one processor may identify the matched scanned file as the malicious code. In another example, if the secondary system scans the block storage volume by consulting databases storing patterns of malicious files and identifies a pattern of a scanned file matches one or more of the stored patterns, the at least one processor may identify the scanned file as the malicious code. In yet another example, if the secondary system scan the block storage volume by consulting databases of cryptographic hashes that identify malicious files and finds that a cryptographic hash of a scanned file matches one or more of the cryptographic hashes that identify malicious files, the at least one processor may identify the scanned file as the malicious code.

[00265] In yet another example, if the secondary system scans the block storage volume by running potential malicious code in a sandbox and finds that running a scanned file causes malicious or harmful changes or behavior (e.g., an attempt to modify a system setting, an attempt to access files outside of a set of directories, an attempt to perform a network mapping process), the at least one processor may identify the scanned file as the malicious code. In yet another example, if the secondary system scans the block storage volume to determine whether

there are scanned files that are not in a database of known good files and finds that a scanned file is not in the database of known good files, the at least one processor may identify the scanned file as the malicious code. As yet another example, if the secondary system scans the block storage volume to detect changes over time in the scanned environment and finding that a scanned file causes a malicious or harmful change in the scanned environment over time, the at least one processor may identify the scanned file as the malicious code. It should be noted that the identification of the malicious code based on the scan may be implemented in various manners and is not limited to the examples described herein.

[00266] Consistent with disclosed embodiments, the at least one processor may be configured to output from the secondary system, a notification of a presence of malicious code in the target system. In some embodiments, the notification may be a pop-up window, a pop-pup dialog, a flashing symbol, an email, a text message, a mobile application notice, a phone call, a warning sound, a tactile feedback (e.g., a vibration), or any type of communications. By way of example, the secondary system may trigger a communication channel between the secondary system and the target system and feed one or more parameters to the communication channel, in which the one or more parameters represent the presence of the malicious code. The secondary system may then send data representing the notification via the triggered communication channel including the one or more parameters to the target system. After receiving the data, the target system may parse the data and identify the one or more parameters. The target system may then display the notification (e.g., as a pop-up window on a screen) that present the presence of the malicious code.

[00267] Consistent with disclosed embodiments, the at least one processor may be further configured to identify a source of the malicious code. The notification of the presence of malicious code outputted from the secondary system in the target system may include information related to the identified source of the malicious code. In some embodiments, the source of the malicious code may include identity of a developer or distributor of the malicious code, the landing location (e.g., a network port or a vulnerable access point) where the malicious code is imported to the target system, source codes of the malicious code, or any information or data indicating the origination of the malicious code. The information related to the identified source of the malicious code may include a text, a picture, a symbol, a video, a link, a voice, or any type of data for presenting and describing the identified source of the malicious code.

[00268] By way of example, **Fig. 7** is a block diagram illustrating an exemplary process 300 of cyber security scanning for a cloud environment, consistent with the disclosed embodiments. While the block diagram may be described below in connection with certain implementation embodiments presented in other figures, those implementations are provided for illustrative purposes only, and are not intended to serve as a limitation on the block diagram.

For example, the steps in Fig. 7 may be executed in any order, steps may be duplicated, or steps may be omitted. In some embodiments, the process 300 may be performed by at least one processor (e.g., a CPU) of a computing device or system (e.g., scanning system 101 in Fig. 1) to perform operations or functions described herein, and may be described hereinafter with reference to Figs. 1-2D by way of example. In some embodiments, some aspects of the process 300 may be implemented as software (e.g., program codes or instructions) that are stored in a memory or a non-transitory computer-readable medium. In some embodiments, some aspects of the process 300 may be implemented as hardware (e.g., a specific-purpose circuit). In some embodiments, the process 700 may be implemented as a combination of software and hardware.

[00269] Fig. 7 includes process blocks 702-710. At block 702, at least one processor may utilize a cloud provider API to access a block storage volume of a workload maintained on a target account in a target system of a cloud storage environment.

[00270] At block 704, the at least one processor may utilize a scanner at a location of the block storage volume and on a secondary system other than the target system. In some embodiments, the secondary system may include at least one of a virtual machine, a container, or a serverless function. In some embodiments, the secondary system may have an operating system different from an operating system of the target account.

[00271] In some embodiments, to utilize the scanner, the at least one processor may suspend an operation of the scanner after the scan of the block storage volume. In some embodiments, to utilize the scanner, the at least one processor may modify a pre-utilized scanner at the location of the block storage volume based on information related to the target account to obtain a modified scanner, and the utilize the modified scanner.

[00272] At block 706, the at least one processor may scan the block storage volume for malicious code, using the secondary system. In some embodiments, to scan the block storage volume, the at least one processor may scan disk-backed memory. For example, the disk-backed memory includes at least one of a page file or a cache file. In some embodiments, the malicious code includes a rootkit.

[00273] At block 708, the at least one processor may identify malicious code based on the scan. At block 710, the at least one processor may output from the secondary system, a notification of a presence of malicious code in the target system.

[00274] Consistent with disclosed embodiments, besides blocks 702-710, the at least one processor may further determine the location of the block storage volume based on at least one of the target account, a secondary system account, a cloud provider account, or a third party account.

[00275] Consistent with disclosed embodiments, besides blocks 302-310, the at least one processor may further identify a source of the malicious code. The notification outputted in block 710 may include information related to the identified source of the malicious code.

[00276] **Forward and Rearward Facing Attack Vector Visualization Techniques**

[00277] **Fig. 8** is a schematic block diagram illustrating an exemplary embodiment of a system for performing visualization of forward and backward facing threats.

[00278] In one of many embodiments, the system may include a graphical user interface system for providing forward and backward facing attack vector visualizations. A graphical user interface system may include systems similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows 10 Mobile, PalmOS-Web OS, or Firefox OS. Forward facing attack vectors may be those external entities that may access an analyzed asset. Backward facing attack vectors may include the possible impacts to an analyzed asset and those impacts' possible effect. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of attack vectors.

[00279] In an embodiment, the system may comprise at least one processor configured to identify assets 801 in a cloud environment. The at least one processor may include a processor such as user device 102 of Fig. 1. The at least one processor may include a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service.

[00280] Another embodiment contemplated may consist of at least one processor configured to identify risks 803 associated with each of the identified asset. The identified risks 803 associated with each of the identified assets may be listed sequentially based on probability of risk, severity of risk, or similar method of ordering the risks. The identified risks 303 associated with an identified asset may be scored based on the level of risk to the asset and may be categorized as a high risk, medium risk, or low risk. Once the level of risk can be determined by scanning system 101, the level of risk can be conveyed and displayed to a user device 102 with a representative graphical display.

[00281] An exemplar embodiment may additionally comprise at least one processor configured to identify relationships 805 between at least some of the identified assets, the relationships including at least one of a trust 807A, a network connectivity 807B, or a mechanism of network proxying 807C. A trust 807A may establish a relationship and may

involve cryptography, digital signatures, electronic certificates, or similar method of establishing a trusting relationship. A network connectivity 807B may include wireless or wired connections and may be established once a relationship of trust 807A is established. A mechanism of network proxying 807C can be accommodated once trust 807A and network connectivity 807B are established through any number of connected servers to accomplish proxying.

[00282] A further embodiment may also comprise a processor configured to receive an identification 809 of a specific asset under investigation. The specific asset under investigation may be a physical or virtual machine that has software operating a specific version that is vulnerable to any number of associated risks. The identification 809 of a specific asset under investigation may involve an IP address, universally unique identifier, or similar system to uniquely or semi-uniquely identify the specific asset under investigation.

[00283] A system among several embodiments may include a processor configured to perform a forward analysis 811A of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment. A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. The analysis performed by scanning system 101 may determine the origin of the forward threat, the nature of the threat to the analyzed system, and the seriousness of the threat originating from the Internet. For example, scanning system 101 may identify at least one possible attack vector to the workload currently being analyzed. The network outside of the cloud environment may identify new or existing threats and attack vectors currently threatening the associated system and workload.

[00284] An exemplary embodiment may further comprise a processor configured to perform a backward analysis 811B of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset. A backward analysis 811B of the associated machine may look downstream to related workloads and systems connected to the analyzed systems. A scan downstream may involve, among several embodiments, the downstream exposure risk including an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset. The at least one exposure risk may include one or more entry point to access the exposed asset and any related machines potentially vulnerable to attack from the analyzed and exposed asset.

[00285] Another embodiment may comprise a processor configured to output a signal 813 to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points. The display of a forward or

backward path to a specific asset may be one of several visual displays such as linear, graphical, or through computer-generated images. The visualization of a plurality of entry points and lateral movement risks with the plurality of entry points may include visualization of the plurality of paths may sequence the paths based on likelihood of access via that path or the severity of potential threat in a path of access to an asset.

[00286] An example embodiment may comprise a system wherein the network outside the cloud environment includes the Internet. The network outside the cloud environment may include access to the Internet in order for the system to access a plurality of analyzed and potentially vulnerable assets.

[00287] Another embodiment may comprise a system wherein the assets in the cloud environment include at least one of a virtual machine, a network appliance, a storage appliance, a compute instance, or an engine instance. A virtual machine may include an emulated version of a computer - including an operating system, memory, storage, graphics processing - such that it can be indistinguishable from a standard (non-virtual) machine to a running program. A network appliance may include a plurality of servers connected through the Internet in order to establish a system and network of stored assets and workloads. A storage appliance may be comprised by a server designed to store workloads and assets to be analyzed. A compute instance may involve input and processing of data entries that cause the analyzed asset to be accessed and run. An engine instance may include a search algorithm run to analyze and search the analyzed workloads or virtual machines to determine the software and data stored on the said virtual machine or stored instance.

[00288] One among many embodiments may comprise a system wherein identifying the assets in a cloud environment includes identifying the assets based on at least one of an identity and access management policy, an organization policy, or an access policy. An identity and access management policy may comprise a software tool or similar method of establishing identity of users attempting to access a system to verify identities of users and control access to an associated workload or system functionality. An organization policy may include a policy applied to all users by the controlling organization of a workload. An access policy may determine the authorized users of an associated system.

[00289] One possible embodiment may comprise a system wherein the presentation of the forward and backward paths indicates alternative paths connecting between the specific asset and an upstream asset or a downstream asset. The indication will be via a visualization showing specific alternative paths connecting specific assets or it may visualize a plurality of alternative paths connecting specific assets. Alternative paths connecting between the specific asset and an upstream asset or a downstream asset may involve a direct connection, such as via the Internet, or an indirect connection via one or more series of servers acting as proxy servers between the analyzed asset and the upstream or downstream asset.

[00290] An exemplary embodiment may further comprise a system wherein the visualization includes a presentation of the alternative paths. The alternative paths may be presented visually through a graphical representation, sequential list of alternative paths, or a computer-generated image that represents the plurality of alternative paths.

[00291] Another embodiment may comprise a system wherein the presentation of the forward and backward paths indicates port numbers for each pathway. The port numbers may be indicated by a series of alphanumeric characters (e.g., “HTTP” or “80”) or other identifier.

[00292] One among several exemplary embodiments may further comprise a system wherein the visualization of the entry points indicates at least one entry point of risk. The visualization may indicate one or more points of entry to a given analyzed system. An entry point of risk may involve a port of access to the workload, and may involve a one or more entry points visualized for use by user device 102.

[00293] Another exemplary embodiment may comprise a system wherein the at least one processor is further configured to monitor network activities of the assets in a cloud environment. The processor may monitor continuously, intermittently, or as directed by an authorized user to direct monitoring of the analyzed network activities of the assets. The cloud environment may be, among several embodiments, the cloud infrastructure 106.

[00294] An embodiment contemplated may comprise a system further configured to detect a potential risk associated with the specific asset based on a network activity of the specific asset. The potential risk may be associated to a specific asset based on historical threats to the specific asset or a list of possible threats to the specific asset based on its current network activities.

[00295] Another embodiment may comprise a system further configured to detect a potential risk associated with the specific asset based on a network activity of an upstream asset of the specific asset. The potential risk associated with the specific asset may be identified by scanning system 101 and determine whether a potential risk can be classified as high risk, medium risk, or low risk. The potential risk may be identified based on the specific asset, the network activity performed by the specific asset, or the specific asset type or related software of the upstream asset.

[00296] One of many embodiments may comprise a system further configured to detect a potential risk associated with the specific asset based on a network activity of a downstream asset of the specific asset. The potential risk may be identified based on the specific asset, the network activity performed by the specific asset, or the specific asset type or related software of the upstream asset. For example, scanning system 101 may perform a scan on an asset being analyzed, and based on the one or more entry points of risk, may determine if the downstream asset of the specific asset poses a potential risk to the analyzed system.

[00297] Another exemplary embodiment may comprise a method for generating a graphical user interface for providing forward and backward facing attack vector visualizations. The method may include a method of generating a graphical user interface similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows 10 Mobile, PalmOS-Web OS, or Firefox OS. A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. Backward analysis 811B may include the possible impacts to an analyzed asset and those impacts' possible effect. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of attack vectors.

[00298] One exemplar embodiment may further comprise a method of identifying assets 801 in a cloud environment. A method may include at least one processor that may include a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service.

[00299] One of several embodiments may comprise a method of identifying risks 803 associated with each of the identified assets 801. The method of identifying risks 803 associated with each of the identified assets 801 may be listed sequentially based on probability of risk, severity of risk, or similar method of ordering the risks. The identified risks 803 associated with an identified asset may be scored based on the level of risk to the asset and may be categorized as a high risk, medium risk, or low risk. Once the level of risk can be determined by scanning system 101, the level of risk can be conveyed and displayed to a user device 102 with a representative graphical display.

[00300] Another embodiment contemplated may comprise a method for identifying relationships 305 between at least some of the identified assets 801, the relationships including at least one of a trust 807A, a network connectivity 807B, or a mechanism of network proxying 807C. A trust 807A may establish a relationship and may involve cryptography, digital signatures, electronic certificates, or similar method of establishing a trusting relationship. A network connectivity 807B may include wireless or wired connections and may be established once a relationship of trust 807A is established. A mechanism of network proxying 807C can be accommodated once trust 807A and network connectivity 807B are established through any number of connected servers to accomplish proxying.

[00301] An embodiment may further comprise a method of receiving an identification of a specific asset 809 under investigation. The specific asset under investigation may be a physical

or virtual machine that has software operating a specific version that is vulnerable to any number of associated risks 803. The identification 809 of a specific asset under investigation may involve an IP address, universally unique identifier, or similar system to uniquely or semi-uniquely identify the specific asset under investigation.

[00302] An embodiment may also comprise a method of performing a forward analysis 811A of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment. A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. A forward analysis may involve analysis of the specific asset under identification to identify at least one possible Internet-originating attack vector to the asset. The analysis performed by scanning system 101 may determine the origin of the forward threat, the nature of the threat to the analyzed system, and the seriousness of the threat originating from the Internet. For example, scanning system 101 may identify at least one possible attack vector to the workload currently being analyzed. The network outside of the cloud environment may identify new or existing threats and attack vectors currently threatening the associated system and workload.

[00303] One embodiment may comprise a method of performing a backward analysis 811B of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset. Backward analysis 811B may include the possible impacts to an analyzed asset and those impacts' possible effect. A scan downstream may involve, among several embodiments, the downstream exposure risk including an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset. The at least one exposure risk may include one or more entry point to access the exposed asset and any related machines potentially vulnerable to attack from the analyzed and exposed asset.

[00304] An embodiment may comprise a method of outputting a signal 813 to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points. The display of a forward or backward path to a specific asset may be one of several visual displays such as linear, graphical, or through computer-generated images. The visualization of a plurality of entry points and lateral movement risks with the plurality of entry points may include visualization of the plurality of paths may sequence the paths based on likelihood of access via that path or the severity of potential threat in a path of access to an asset.

[00305] Another embodiment may comprise a method wherein the network outside the cloud environment includes the Internet. The network outside the cloud environment may include access to the Internet in order for the system to access a plurality of analyzed and potentially vulnerable assets.

[00306] One of several embodiments may comprise a method wherein the assets in the cloud environment include at least one of a virtual machine, a network appliance, a storage appliance, a compute instance, or an engine instance. A virtual machine may include an emulated version of a computer - including an operating system, memory, storage, graphics processing - such that it can be indistinguishable from a standard (non-virtual) machine to a running program. A network appliance may include a plurality of servers connected through the Internet in order to establish a system and network of stored assets and workloads. A storage appliance may be comprised by a server designed to store workloads and assets to be analyzed. A compute instance may involve input and processing of data entries that cause the analyzed asset to be accessed and run. An engine instance may include a search algorithm run to analyze and search the analyzed workloads or virtual machines to determine the software and data stored on the said virtual machine or stored instance.

[00307] An embodiment may further comprise a method wherein identifying assets in a cloud environment includes identifying the assets based on at least one of an identity and access management policy, an organization policy, or an access policy. An identity and access management policy may comprise a software tool to verify identities of users and control access to an associated workload or system functionality. An organization policy may include a policy applied to all users by the controlling organization of a workload. An access policy may determine the authorized users of an associated system.

[00308] An embodiment may also comprise a method wherein the presentation of the forward and backward paths indicates alternative paths connecting between the specific asset and an upstream or a downstream asset. The indication may be via a visualization showing specific alternative paths connecting specific assets or it may visualize a plurality of alternative paths connecting specific assets.

[00309] Another exemplar embodiment may comprise a method wherein the visualization includes a presentation of the alternative paths. The alternative paths may be presented visually through a graphical representation, sequential list of alternative paths, or a computer-generated image that represents the plurality of alternative paths.

[00310] Another embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations.

[00311] One embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor 800, are configured to

perform operations identifying assets 801 in a cloud environment. the instructions stored on a non-transitory computer-readable medium may include at least one processor 800 that may include a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service.

[00312] An embodiment may further comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor 300 to perform operations comprising identifying risks 803 associated with each of the identified asset. The instructions performed of identifying risks 803 associated with each of the identified assets 301 may be listed sequentially based on probability of risk, severity of risk, or similar method of ordering the risks.

[00313] Another embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising identifying relationships 805 between at least some of the identified assets 301, the relationships including at least one of a trust 807A, network connectivity 807B, or a mechanism of network proxying 807C.

[00314] An exemplar embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising receiving an identification of a specific asset 809 under investigation. The instructions may include receiving information identifying the specific asset 809. The specific asset 809 under investigation may be a physical or virtual machine that has software operating a specific version that is vulnerable to any number of associated risks 803.

[00315] One among many embodiments may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising performing a forward analysis 811A of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment. . A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. And the scanning system 101 may identify at least one possible attack vector to the workload currently being analyzed. The network outside of the cloud environment may identify new or existing threats and attack vectors currently threatening the associated system and workload.

[00316] An embodiment may further comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising performing a backward

analysis 811B of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset. Backward analysis 811B may include the possible impacts to an analyzed asset and those impacts' possible effect. The at least one exposure risk may include one or more entry point to access the exposed asset and any related machines potentially vulnerable to attack from the analyzed and exposed asset.

[00317] An embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising outputting a signal 813 to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points. The display of a forward or backward path to a specific asset may be one of several visual displays such as linear, graphical, or through computer-generated images. The visualization of a plurality of entry points and lateral movement risks with the plurality of entry points may include visualization of the plurality of paths may sequence the paths based on likelihood of access via that path or the severity of potential threat in a path of access to an asset.

[00318] **Passive Key Identification Techniques**

[00319] Aspects of this disclosure may provide a technical solution to the challenging technical problem of identifying access keys (e.g., passwords, Secure Shell (SSH) keys, or cloud keys) to compute resources (e.g., machines, containers, storages, or any hardware or software component) in a cloud environment without using the access keys to access the compute resources. In existing technologies, scanning systems may identify access keys to compute resources and may need to verify the identified keys by actually accessing the compute resources using the access keys. In such cases, the actual accesses may generate logs or records (e.g., for successful or failed accesses) that can constitute activity patterns similar to an attacker to the system. Such activity patterns may distract a system administrator (e.g., an individual or a computer program) to misidentify the scanning system as an attacker. Further, actual accesses may create loads on both the scanning system and the compute resources and may reduce the computing resources and power for other tasks. To resolve such a technical problem, a cryptographic analysis may be performed to identify a first set of fingerprints of the identified keys, and trust configurations of the compute resources may be analyzed to identify a second set of fingerprints of the compute resources. The first and second set of fingerprints may be compared to match keys with the compute resources without using the keys to access the compute resources. By doing so, the scanning system may perform its

function without raising false alarms to the system administrator, while reducing computing costs to the cloud environment.

[00320] A cyber security system, as used herein, may refer to a system including any combination of software and hardware for enhancing security of a device, a platform, or another system in a network environment. By way of example, the cyber security may be implemented as a system including scanning system 101 in Fig. 1. As another example, scanning system 101 may include the disclosed cyber security system as a subset. A compute resource, as used herein, may refer to a database, a virtual machine, a storage, a keystore, a scanning system, a load balancer, a server, a computer, a container, or any physical or virtual component integrated in or communicatively connected to a cloud infrastructure. By way of example, the compute resource may be scanning system 101, any of databases 103A-103D, any of virtual machines 107A-107D, any of databases 109A-109D, any of storage 111A-111D, any of keystores 113A-113D, or load balancer 115 in cloud infrastructure 106 in Fig. 1.

[00321] A key to access a compute resource, as used herein, may refer to computerized data that includes a credential for granting permission to an accessor to access the compute resource, or a service or function provided by the compute resource. By way of example, the key may include at least one of a password, a remote access key, a script containing a password, a cloud key (e.g., a credential file for accessing a cloud service, such as an AWS credential file), an Secure Shell (SSH) key, an AWS key, a private component of a private-public key pair, or any type of credential that provide checked or unchecked access to the compute resource. Matching a key to a compute resource, as used herein, may refer to confirming, verifying, validating, or any operation or process of determining that the key is valid for an accessor (e.g., another compute resource) to access the compute resource. For example, to match a key to a compute resource, an accessor may test by using the key to access the compute resource and determines that the key is matched to the compute resource if such access is successful. As another example, to match a key to a compute resource, an accessor may perform methods or algorithms to verify that the key is matched to the compute resource without using the key to access the compute resource.

[00322] Consistent with disclosed embodiments, at least one processor may be configured to analyze a cloud environment to identify a plurality of keys to the compute resources in the cloud environment. In some embodiments, the at least one processor may analyze the cloud environment to identify the plurality of keys to the compute resources in the cloud environment in any of steps 231, 237, or 239 in FIG. 2D. In some embodiments, the at least one processor may analyze the cloud environment to identify the plurality of keys to the compute resources in the cloud environment in any step described in association with FIGS. 2A-2D. In some embodiments, at least one of the plurality of keys may include at least one of a password (e.g., a combination of alphanumeric characters and symbols), a script (e.g., a

PYTHON® script or a shell script) containing a password, a cloud key, or an Secure Shell (SSH) key. For example, a cloud key may be a key for accessing a cloud resource.

[00323] A cloud environment, as used herein, may refer to a computing environment running on a cloud. By way of example, the cloud environment may include or be part of cloud infrastructure 106 in Fig. 1. Analyzing a cloud environment, as used herein, may refer to an operation or a process of separating or dividing the cloud environment into parts or components (e.g., physical components or logic components), then determining nature and relationship of the parts or components based on data associated with the parts or components.

[00324] By way of example, if the cloud environment is cloud infrastructure 106 in Fig. 1, to analyze cloud infrastructure 106, the at least one processor may access cloud infrastructure 106 (e.g., via network 105 or via internal connections in cloud infrastructure 106) and obtain a list of compute resources in cloud infrastructure 106 (e.g., scanning system 101, any of databases 103A-103D, any of virtual machines 107A-107D, any of databases 109A-109D, any of storage 111A-111D, any of keystores 113A-113D, or load balancer 115). For example, the at least one processor may obtain the list of compute resources by reading a configuration file from a database (e.g., one of databases 109A-109D), or visit available compute resources one after another through communicative connections between the compute resources to generate the list of compute resources. After obtaining the list of compute resources in cloud infrastructure 106, the at least one processor may determine the nature of the compute resources, such as their types, numbers, functions, or any other characteristics or features. The at least one processor may further determine the relationship between the compute resources, such as communicative connections, access privileges, data input/output directions, or controls between the compute resources.

[00325] Identifying a key to a compute resource, as used herein, may refer to an operation or a process of locating, recognizing, or any operation or process or analyzing computerized data or information to determine that the computerized data or information is a key to the compute resources. By way of example, the at least one processor may analyze cloud infrastructure 106 in Fig. 1 to recognize a list of compute resources in cloud infrastructure 106 (e.g., scanning system 101, any of databases 103A-103D, any of virtual machines 107A-107D, any of databases 109A-109D, any of storage 111A-111D, any of keystores 113A-113D, or load balancer 115). Then, the at least one processor may read computerized data stored in a compute resource and identify that some of the computerized data are one or more keys to other compute resources in cloud infrastructure 106. For example, to recognize the keys, the at least one processor may read and compare the computerized data with records stored in keystores 113A-113D, and determine that the computerized data is a key if it matches a record in any of keystores 113A-113D. As another example, the at least one processor may read the computerized data and check its syntaxes, text string patterns, file formats, file properties,

encryption manners, library versions, software versions, or any other characteristics or features of the computerized data, and determine that the computerized data is a key if its checked characteristics or features fit a predetermined pattern of a key or fit an entry in a dictionary of keys. For example, the dictionary of keys may be stored in a keystore (e.g., any of keystores 113A-113D).

[00326] In some embodiments, the plurality of keys may be stored in at least one workload. A workload, for example, may refer to a specific application, service, capability, or a specific amount of work that can be run on a cloud resource, system, or infrastructure. By way of example, a workload may be a virtual machine, a database server, a container, a Hadoop node, an application, a storage server, a load balancer, or an IAM (Identity and Access Management) configuration. By way of example, with reference to FIG. 1, cloud infrastructure 106 may include workloads such as scanning system 101, databases 103A-103D, virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115.

[00327] Consistent with disclosed embodiments, at least one processor may be configured to perform a cryptographic analysis on the plurality of keys to identify a first set of fingerprints that uniquely identify each of the plurality of keys. The first set of fingerprints may be non-functional. In some embodiments, the at least one processor may perform the cryptographic analysis in step 237 of FIG. 2D. In some embodiments, the at least one processor may perform the cryptographic analysis in any step described in association with FIGS. 2A-2D.

[00328] A cryptographic analysis on data, as used herein, may refer to an operation or process of identifying ciphertext, ciphers and cryptosystems contained in the data to discover hidden aspects of the data that may improve or weaken security of the data or compute resources associated with the data. In some embodiments, to perform the cryptographic analysis on the plurality of keys, the at least one processor may perform an algorithm on the plurality of keys to convert the plurality of keys into another form of data. For example, the at least one processor may perform a hashing process on the plurality of keys (e.g., by inputting values of the keys to a hash function) to obtain corresponding hash values of respective keys.

[00329] A fingerprint of a key, as used herein, may refer to non-functional data (e.g., a numeric value, an alphanumeric string, or any combination of letters, numbers, or symbols) generated based on the key and may uniquely identify the key. By way of example, a fingerprint may be generated by performing a hashing process on a key or a part of the key. The hashing process may include, for example, an MD5 algorithm, an SHA-1 algorithm, an SHA-2 algorithm, an SHA-3 algorithm, a RIPEMD-160 algorithm, a BLAKE2 algorithm, a BLAKE3 algorithm, or any type of cryptographic hash algorithms. In such cases, the fingerprint may be a hash value of the key, a part (e.g., a truncated part) of the hash value of the key, a hash value of a part of the key, or a part (e.g., a truncated part) of the hash value of the part of the key. The hash value

may be unique (e.g., non-overlap with any hash value of any other key) and non-functional (e.g., unable to be used as a key to access a compute resource). To identify the first set of fingerprints, the at least one processor may perform the hashing process on the plurality of keys to obtain a plurality of hash values, and search the cloud environment to check whether there exists any compute resource that stores the plurality of hash values.

[00330] In some embodiments, at least one of the first set of fingerprints may be non-identical to any key of the plurality of keys. For example, if the first set of fingerprints are hash values, they may be non-identical to the plurality of keys.

[00331] Consistent with disclosed embodiments, at least one processor may be configured to analyze trust configurations of the compute resources to identify a second set of fingerprints for each of the compute resources. In some embodiments, the at least one processor may analyze the trust configurations in step 233 of FIG. 2D. In some embodiments, the at least one processor may analyze the trust configurations in any step described in association with FIGS. 2A-2D.

[00332] A trust configuration, as used herein, may refer to data (e.g., a stored file, a database entry, a value stored in a data structure, or any computerized information) that stores a pre-established trust relationship between at least two compute resources. For example, a trust configuration may be a file storing a trust policy (e.g., a public key infrastructure (PKI), a digital signature system, or an AWS trust policy). Analyzing a trust configuration, as used herein, may refer to an operation or a process of separating or dividing compute resources associated with the trust configuration into units or groups, then determining nature and relationship of the units or groups based on relationship data associated with the units or groups. For example, the relationship data may include flags, indicators, or any type of value that indicates a first compute resource has an access privilege to a second compute resource.

[00333] A fingerprint for a compute resource, as used herein, may refer to non-functional data (e.g., a numeric value, an alphanumeric string, or any combination of letters, numbers, or symbols) generated based on a key for accessing the compute resource and may uniquely identify the key. By way of example, a fingerprint for a compute resource may be generated by applying a cryptographic hash function (e.g., an SHA-1 or SHA-2 function) to a key for accessing the compute resource to obtain a sequence of bytes (e.g., a hash value or a modified hash value), and the sequence of bytes may be the fingerprint for the compute resource. The key may be uniquely associated with the compute resource.

[00334] For example, to identify the second set of fingerprints, the at least one processor may analyze a trust configuration file (e.g., a file storing fingerprints and their associated data) by reading the byte stream of the trust configuration file to determine whether any sequence of bytes that fits to a predetermined pattern (e.g., an alphanumeric string pattern). The sequences

of bytes that fit to the predetermined pattern may be determined as the second set of fingerprints.

[00335] Consistent with disclosed embodiments, at least one processor may be configured to compare the first set of fingerprints with the second set of fingerprints to match keys with the compute resources without using the keys to access the compute resources. In some embodiments, the at least one processor may compare the first set of fingerprints with the second set of fingerprints in any step described in association with FIGS. 2A-2D. Comparing a first fingerprint and a second fingerprint, as used herein, may refer to performing a textual, numerical, or semantic comparison of the first fingerprint and the second fingerprint, or performing an algorithmic conversion of at least one of the first fingerprint or the second fingerprint to further perform a textual, numerical, or semantic comparison of the conversion results. By way of example, a first compute resource may store a key (e.g., a public key) for accessing a second compute resource. The at least one processor may analyze the cloud environment to identify the key from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. A third compute resource (e.g., may be the same as or different from the second compute resource) may store in its trust configurations a second fingerprint of the key. The second fingerprint may be generated by a different processor at a different time. The at least one processor may then analyze the trust configurations of the third compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other. If the first fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource. As can be seen, after comparison, if the first set of fingerprints are matched with the second set of fingerprints, keys may be matched with the compute resources, and the at least one processor needs not use the keys to access the compute resources to confirm that the keys are matched to the compute resources.

[00336] Consistent with disclosed embodiments, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the first set of fingerprints. A multi-machine interaction, as used herein, may refer to data input/output, operation control, status polling, scanning, or any type of an interaction between a plurality of machines in a cloud environment. Analyzing a multi-machine interaction, as used herein, may refer to an operation or a process of separating or dividing machines in the cloud environment into units or groups, then determining nature and relationship of the units or groups based on relationship data associated with the units or groups. For example, the relationship data may be the first set of fingerprints.

[00337] By way of example, to analyze the multi-machine interaction, the at least one processor may compare the first set of fingerprints with the second set of fingerprints. If a first fingerprint of the first set of fingerprints matches with a second fingerprint of the second set of fingerprints, the at least one processor may determine that a first compute resource associated with the first fingerprint may have an inter-machine interaction (e.g., having an ability to access the other) with a second compute resource associated with the second fingerprint. By way of example, a first compute resource may store a key (e.g., a private key) for accessing a second compute resource. The at least one processor may analyze the cloud environment to identify the key from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. A third compute resource (e.g., may be the same as or different from the second compute resource) may store in its trust configurations a second fingerprint of the key. The at least one processor may then analyze the trust configurations of the third compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other (e.g., being the same). If the first fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource, and may further determine an inter-machine interaction in which the first compute resource has ability to access the second compute resource.

[00338] Consistent with disclosed embodiments, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the plurality of keys. For example, the at least one processor may use the plurality of keys as the relationship data for determining nature and relationship of compute resources in the cloud environment. By way of example, to analyze the multi-machine interaction, the at least one processor may compare a first key for accessing a first one of the compute resources and a second key for accessing a second one of the compute resources. If the first key matches with (e.g., is identical to) the second key, the at least one processor may determine that the first one of the compute resources may have an interaction (e.g., having an ability to access the other) with the second one of the compute resources.

[00339] By way of example, **Fig. 9** is a block diagram illustrating an exemplary process 900 of matching keys with compute resources in a cloud environment, consistent with the disclosed embodiments. While the block diagram may be described below in connection with certain implementation embodiments presented in other figures, those implementations are provided for illustrative purposes only, and are not intended to serve as a limitation on the block diagram. For example, the steps in Fig. 9 may be executed in any order, steps may be duplicated, or steps may be omitted. In some embodiments, the process 900 may be performed by at least one processor (e.g., a CPU) of a computing device or system (e.g., scanning system

101 in Fig. 1) to perform operations or functions described herein, and may be described hereinafter with reference to Figs. 1-2D by way of example. In some embodiments, some aspects of the process 900 may be implemented as software (e.g., program codes or instructions) that are stored in a memory or a non-transitory computer-readable medium. In some embodiments, some aspects of the process 300 may be implemented as hardware (e.g., a specific-purpose circuit). In some embodiments, the process 300 may be implemented as a combination of software and hardware.

[00340] Fig. 9 includes process blocks 902-908. At block 902, at least one processor may analyze a cloud environment to identify a plurality of keys to the compute resources in the cloud environment. In some embodiments, the plurality of keys may be stored in at least one workload. In some embodiments, at least one of the plurality of keys may include at least one of a password (e.g., a combination of alphanumeric characters and symbols), a script (e.g., a PYTHON® script or a shell script) containing a password, a cloud key, or an Secure Shell (SSH) key.

[00341] At block 904, the at least one processor may perform a cryptographic analysis on the plurality of keys to identify a first set of fingerprints that uniquely identify each of the plurality of keys. The first set of fingerprints may, in some embodiments, be non-functional. In some embodiments, at least one of the first set of fingerprints may be non-identical to the key.

[00342] At block 906, the at least one processor may analyze trust configurations of the compute resources to identify a second set of fingerprints for each of the compute resources. As discussed above, to identify the second set of fingerprints, in some embodiments, the at least one processor may analyze a trust configuration file (e.g., a file storing fingerprints and their associated compute resources) by reading the byte stream of the trust configuration file to determine whether any sequence of bytes that fits to a predetermined pattern (e.g., an alphanumeric string pattern). The sequences of bytes that fit to the predetermined pattern may be determined as the second set of fingerprints.

[00343] At block 908, the at least one processor may compare the first set of fingerprints with the second set of fingerprints to match keys with the compute resources without using the keys to access the compute resources. As discussed above, in some embodiments, a first compute resource may store a key (e.g., a private component of a private-public key pair) to a second compute resource, and the second compute resource may store in its trust configurations a second fingerprint of the key. The at least one processor may analyze the cloud environment to identify the key from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. The at least one processor may then analyze the trust configurations of the second compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other (e.g., being identical). If the first

fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource.

[00344] Consistent with disclosed embodiments, besides blocks 902-908, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the first set of fingerprints. By way of example, to analyze the multi-machine interaction, the at least one processor may compare the first set of fingerprints with the second set of fingerprints. As discussed above, in some embodiments, a first compute resource may store a key (e.g., a private key) for accessing a second compute resource. The at least one processor may analyze the cloud environment to identify the private component from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. A third compute resource (e.g., may be the same as or different from the second compute resource) may store in its trust configurations a second fingerprint of the key. The at least one processor may then analyze the trust configurations of the third compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other (e.g., being the same). If the first fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource, and may further determine an inter-machine interaction in which the first compute resource has ability to access the second compute resource.

[00345] Consistent with disclosed embodiments, besides blocks 902-908, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the plurality of keys. By way of example, to analyze the multi-machine interaction, the at least one processor may compare a first key for accessing a first one of the compute resources and a second key for accessing a second one of the compute resources.

[00346] **Hybrid Ephemeral Scanner Techniques**

[00347] Aspects of this disclosure may include accessing a primary account maintained in a cloud environment. A primary account as used herein, may refer to a principal or main identity created for a person in a computer or computing system. A primary account may also be created for machine entities, such as service accounts for running programs, system accounts for storing system files and processes, and root and administrator accounts for system administration. A cloud environment, as used herein, may refer to a platform implemented on, hosted on, and/or accessing servers that are accessed over the Internet. An example of a cloud environment is cloud infrastructure 106 in Fig. 1. In some embodiments, scanning system 101 of Fig. 1 may gain access to an account within cloud infrastructure 106.

[00348] Aspects of this disclosure may include receiving information defining a structure of the primary account, wherein the structure includes a plurality of assets.

Information as used herein, may refer to data received. Structure of the primary account as used herein, may refer to a set up or of the primary account. A plurality of assets as used herein, may refer to one or more data, devices, or components within an organization's system. For example, the plurality of assets may include any of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, load balancer 115, log files or databases, API gateway resources, API gateway REST APIs, Autoscaling groups, CloudTrail logs, CloudFront services, volumes, snapshots, VPCs, subnets, route tables, network ACLs, VPC endpoints, NAT gateways, ELB and ALB, ECR repositories, ECS clusters, services, and tasks, EKS, S3 bucket and Glacier storage, SNS topics, IAM roles, policies, groups, users, KMS keys, and Lambda functions.

[00349] In some embodiments, scanning system 101 of Fig. 1 may receive information relating to the structure of an account. In such an embodiment, the structure of the account may include virtual machines 107A-D and databases 109A-109D, for example. In some embodiments, the information may be acquired from another device, received via a one way communication, received in response to a request, retrieved from a storage device, or generated.

[00350] In some embodiments, the information may exclude raw data of the primary account. Raw data as used herein, may refer to a collection of data as gathered before it has been processed, cleaned, or analyzed. For example, raw data may include usage data, passwords, and cache.

[00351] Aspects of this disclosure may include deploying, inside the primary account, at least one ephemeral scanner configured to scan at least one block storage volume and to output metadata defining the at least one block storage volume. In some embodiments, the output may exclude raw data of the primary account. In some embodiments, the ephemeral scanner may be deployed inside a secondary account for which trust has been established with a primary account. For example, a secondary account on another system may have an existing trust relationship (such as a cloud trust policy) with a first account. The ephemeral scanner may be operated inside of that secondary account to scan block storage associated with the first account. Each account may be hosted on the same system or on different systems.

[00352] Further, in some embodiments, the at least one ephemeral scanner may be configured to periodically scan at least one block storage volume. For example, the scanner may be configured to scan at least one block storage volume once per hour, once per day, once per week, or more or less often.

[00353] Further, in some embodiments, the at least one ephemeral scanner may be periodically deployed to scan at least one block storage volume. For example, the scanner may be deployed for scanning at least one block storage volume once per hour, once per day, once per week, or more or less often.

[00354] Outputting metadata as used herein, may refer to producing a set of data that describes and gives information about other data. In some embodiments, scanning system 101 of Fig. 1 may scan a block storage volume and output data related to it (e.g., vulnerability information, configuration information, malware information, risk analysis information, and sensitive information). Periodically scanning as used herein, may refer to conducting a scan occurring or recurring at regular intervals. In some embodiments, the ephemeral scanner is configured to perform vulnerability scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for security vulnerabilities on the device. In some embodiments, the ephemeral scanner is configured to perform configuration scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for configuration information related to the device. In some embodiments, the ephemeral scanner may scan the block storage volume in order to look for security issues. In some embodiments, the ephemeral scanner is configured to perform malware scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for malware information related to the device. As discussed above with respect to Fig. 2D, in step 235, scanning system 101 may perform a step of malware scanning. In some embodiments, scanning system 101 may perform malware scanning across all filesystems in the snapshot (e.g., gathered from virtual machines 107A-107D or storage 111A-111D). Scanning system 101 may use multiple malware scanning software solutions to perform a malware scan against the filesystems, including one sourced from another vendor, such as bucketAV, Trend Micro Cloud One, Sophos Cloud Optix, Crowdstrike Falcon CWP, or others. In some embodiments, malware scanning in step 235 comprises utilizing signatures, heuristics, or sandboxing capabilities to deduce whether there is an infection on the machine.

[00355] In some embodiments, the ephemeral scanner is configured to perform lateral-movement risk analysis of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for lateral-movement risk information related to the device. For example, in some embodiments, scanning system 101 may perform a “backward” analysis of the specific asset to identify exposure risk to assets downstream of the specific asset, wherein the downstream exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset.

[00356] Further, as discussed above with respect to Fig. 2D, in step 237, scanning system 101 may perform a step of lateral movement scanning. An attacker who establishes a network foothold usually attempts to move laterally from one resource to another in search of rich targets such as valuable data. Stolen passwords and keys unlock access to servers, files, and privileged accounts. In some embodiments, scanning system 101 may gather keys from

each scanned system or device (e.g., virtual machines 107A-107D or storage 111A-111D). In some embodiments, scanning system 101 searches for passwords, scripts, shell history, repositories, or other data that may contain passwords, cloud access keys, SSH keys, or other key/password/access information that provide unchecked access to important resources. In some embodiments, scanning system 101 searches for such keys/passwords/access information and calculates a "hash" (a mathematical fingerprint) of each string. Scanning system 101 then attempts to match the hashed strings to hashes of strings that that are stored on different systems or devices. This will be used to detect the potential lateral movement between assets.

[00357] In some embodiments, the ephemeral scanner is configured to perform sensitive information scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for sensitive information related to the device. As discussed above with respect to Fig. 2D, in step 241, scanning system 101 may perform a step of sensitive information scanning. In some embodiments, scanning system 101 may search the snapshot for sensitive information, such as personally identifiable information (PII), Social Security numbers, healthcare information, or credit card numbers. In some embodiments, scanning system 101 may search data repository history as well. This is because it is not uncommon for an entire production environment repository to be cloned, with no one remembering the copy contains sensitive information. In some situations, detecting sensitive data not secured is critical in adherence to data privacy regulations.

[00358] In some embodiments, the ephemeral scanner is configured to perform container scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for container information related to the device. As discussed with respect to Fig. 2D, in step 243, scanning system 101 may perform a step of container scanning. In some embodiments, scanning system 101 may apply one or more of the preceding steps of Fig. 2D against containerized environments. In some embodiments, in order to do so, scanning system 101 reconstructs a container runtime layered file system (LFS) before recursively running one or more of steps 231-241 on the reconstructed file system.

[00359] In some embodiments, the ephemeral scanner is configured to perform keys and password scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for keys and password information related to the device. As discussed above with respect to Fig. 2D, in step 239, scanning system 101 may perform a step of key/password scanning. As one example situation, suppose there is a weak or unprotected password stored (in plain text) in storage 111A. For example, if a personal email account has been compromised, the passwords may be known about in advance. Scanning system 101 may search the snapshot for similar usernames or login names, and, either using known dictionaries or the account owner's previously leaked

passwords (stored in, e.g., database 103A), may attempt to login to one or more systems or devices in cloud infrastructure 106, and may record the result thereof.

[00360] In some embodiments, defining the at least one block storage volume includes presenting risk data without sharing consumer data or data that was used to identify the risk data. Risk data as used herein, may refer to information relating to causing harm. Consumer data as used herein, may refer to a user's personal information or any information trail user's leave behind as a result of their computer or Internet use. Data used to identify the risk as used herein, may refer to information for determining possible issues. In some embodiments, scanning system 101 of Fig. 1 may present information without sharing a user's personal information or any information trail user's leave behind as a result of their computer or Internet use.

[00361] In some embodiments, the metadata defining the at least one block storage volume includes at least one of: an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, or a profile configuration. An indication of an installed application as used herein, may refer to a sign or signal of a stored program. A version of an installed application as used herein, may refer to an older or newer form of a stored program. An operating system configuration as used herein, may refer to one or more computer system settings that have been set by default automatically or manually by a given program or the user. An application configuration as used herein, may refer to one or more computer program settings that have been set by default automatically or manually by a given program or the user. A profile configuration as used herein, may refer to one or more computer program file settings that have been set by default automatically or manually by a given program or the user.

[00362] Aspects of this disclosure may include receiving a transmission of the metadata from the at least one ephemeral scanner. In some embodiments, the transmission may exclude raw data of the primary account. A transmission of metadata as used herein, may refer to movement of a set of data that describes and gives information about other data. In some embodiments, scanning system 101 of Fig. 1 may receive data excluding any raw data (e.g., data as gathered before it has been processed, cleaned or analyzed).

[00363] Aspects of this disclosure may include analyzing the received metadata to identify a plurality of cybersecurity vulnerabilities. A plurality of cybersecurity vulnerabilities as used herein, may refer to a weakness in a system. For example, a cybersecurity vulnerability may be exploited by cybercriminals to gain unauthorized access to a computer system. In some embodiments, scanning system 101 of Fig. 1 may analyze the received data in order to identify any weakness in a system (e.g., to avoid cybercriminals gaining unauthorized access to a computer system). In some embodiments, vulnerability information may include information stored in a system.

[00364] Aspects of this disclosure may include correlating each of the identified plurality of cybersecurity vulnerabilities with one of the plurality of assets. In some embodiments, scanning system 101 of Fig. 1 may connect a security risk with an asset in the cloud environment. In such embodiments, the connection or correlation of vulnerability to asset may allow the system to address the vulnerability.

[00365] Aspects of this disclosure may include generating a report correlating the plurality of cybersecurity vulnerabilities with the plurality of assets. A report as used herein, may refer to a document containing information. For example, a report may notify the administrator of a website or application about a problem such as a security issue or vulnerability in the system that should be addressed. In some embodiments, scanning system 101 of Fig. 1 may create a report providing information related to the vulnerability and asset. In such an embodiment, the report may allow for the vulnerability to be addressed and resolved.

[00366] In such embodiments, the system may allow scanners (e.g., ephemeral scanners, scanning system 101) to run inside an account, including an Amazon Web Services, Azure, or GCP account. In this mode, a cloud cybersecurity service may generate ephemeral scanners inside an account that performs the same actions as a SAAS node but are logically hosted inside the account.

[00367] In some embodiments, the at least one processor is further configured to receive a transmission of updated metadata defining the at least one block storage volume in response to at least one change to the at least one block storage volumes. Updated metadata as used herein, may refer to a renewed set of data that describes and gives information about other data.

[00368] **Fig. 10** is a block diagram of a method 1000 for deployment of ephemeral scanners, consistent with disclosed embodiments. In some embodiments, the method may include seven (or more or less) steps:

[00369] Block 1002: Access a primary account maintained in a cloud environment. In some embodiments, scanning system 101 of Fig. 1 may gain access to an account within cloud infrastructure 106 using a password stored in a database, for example.

[00370] Block 1004: Receiving information defining a structure of the primary account, wherein the structure includes a plurality of assets. In some embodiments, the information may exclude raw data of the primary account. In some embodiments, scanning system 101 of Fig. 1 may receive data relating to the structure of an account. In such an embodiment, the structure of the account may include virtual machines 107A-D and databases 109A-109D, for example.

[00371] Block 1006: Deploying, inside the primary account, at least one ephemeral scanner configured to periodically scan at least one block storage volume and to output metadata defining the at least one block storage volume. In some embodiments, the output may exclude raw data of the primary account. In some embodiments, scanning system 101 of Fig. 1

may scan a block storage volume and output data related to it (e.g., vulnerability information, configuration information, malware information, risk analysis information, and sensitive information).

[00372] Block 1008: Receiving a transmission of the metadata from the at least one ephemeral scanner. In some embodiments, the transmission may exclude raw data of the primary account. In some embodiments, scanning system 101 of Fig. 1 may receive data excluding any raw data (e.g., data as gathered before it has been processed, cleaned or analyzed).

[00373] Block 1010: Analyzing the received metadata to identify a plurality of cybersecurity vulnerabilities. For example, a cybersecurity vulnerability may be exploited by cybercriminals to gain unauthorized access to a computer system. In some embodiments, scanning system 101 of Fig. 1 may analyze the received data in order to identify any weakness in a system (e.g., to avoid cybercriminals gaining unauthorized access to a computer system).

[00374] Block 1012: Correlating each of the identified plurality of cybersecurity vulnerabilities with one of the plurality of assets. In some embodiments, scanning system 101 of Fig. 1 may connect a security risk with an asset in the cloud environment. In such embodiments, the connection or correlation of vulnerability to asset may allow the system to address the vulnerability.

[00375] Block 1014: Generating a report correlating the plurality of cybersecurity vulnerabilities with the plurality of assets. For example, a report may notify the administrator of a website or application about a problem such as a security issue or vulnerability in the system that should be addressed. In some embodiments, scanning system 101 of Fig. 1 may create a report providing information related to the vulnerability and asset. In such an embodiment, the report may allow for the vulnerability to be addressed and resolved.

[00376] **Risk Information Aggregation Techniques**

[00377] **Fig. 11** represents a schematic block diagram 1100 illustrating an exemplary embodiment of a method for providing a dashboard aggregating risk information.

[00378] Assets operating in a cloud environment face myriad cybersecurity risks that vary in type and nature of threat with increasing regularity. In order to combine the varied and multiple cybersecurity risks an asset may encounter in a cloud environment into a single view to allow for faster and more flexible processing, prioritization and mitigation, a system and method of aggregating cybersecurity risks into a single dashboard is needed. Aggregation of multiple cybersecurity risks into a single view for an administrator is needed. In one of many embodiments, an exemplary embodiment may include a graphical user interface system for providing comprehensive cloud environment risk inventory visualization. A graphical user interface system may include systems similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows

10 Mobile, PalmOS-Web OS, or Firefox OS. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of risk inventories. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D.

[00379] An exemplary embodiment may also include at least one processor. The at least one processor may be part of a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations.

[00380] In an embodiment may include a processor configured to cause a display to present a plurality of asset categories (step 1101). A display may be graphical, sequential, or multi-dimensional. A plurality may include at least one asset category to be displayed. Asset categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category.

[00381] An account category may further include listing of assets based on type of account associated with that user. A type of account may include active user account, passive user account, administrator account, maintainer account, a system account, superuser account, or a guest user account.

[00382] A container category may include one or more containers currently listed in the system. A container may involve specific versions of programming language runtimes, libraries required to run software, or another method of packaging applications abstracted from the cloud environment.

[00383] A database category may include one or more databases managed and maintained in relation to the system. Databases may include a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database.

Databases may be implemented using ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[00384] An image category may contain one or more images based on image type, size, and content. Images may include images of one or more virtual machines.

[00385] A managed service category may include one or more managed services listed in the above system, to include at least client or customer-owned systems being managed by a third party entity.

[00386] A messaging service category may include systems/programs communicating with other systems/programs using services such as Google Cloud Pub and Sub communications, AWS SQS, or similar queues found in Information Systems.

[00387] A monitoring category may include centralized or decentralized methods of security monitoring systems.

[00388] A network category may include one or more systems of establishing a form of communicative connectivity between systems, e.g., TCP-IP services.

[00389] A storage category may list one or more methods of data storage, to include Direct Attached Storage, Network Attached Storage, SSD Flash Drive Arrays, Hybrid Flash Arrays, Hybrid Cloud Storage, Backup Software, Backup Appliances, Cloud Storage, or similar. Storage may include data structures, instructions, or any other data to be contained in a storage medium.

[00390] A user category may include one or more users of a given system along with identifying information relating to its unique or semi-unique identifier, level of authorized access and permissions, and storage volumes accessible by said user.

[00391] An access category may include one or more levels of access permission and a listing of users or devices granted access at a given level of access permission to a system.

[00392] A virtual machine category may list one or more virtual machines accessible to a system and the levels and types of access granted to a given virtual machine.

[00393] A serverless category may list one or more storage volumes stored on a serverless computing system such as Kubeless, Fission, Google App Engine, AWS Lambda, Google Cloud Functions, IBM Cloud Function, IBM Cloud Code Engine, Microsoft Azure Functions, Cloudflare Workers, and Compute@Edge.

[00394] In an embodiment, an exemplar may include a processor configured to receive, via an input device, a selection of a particular asset category (step 1103). An input device may include a personal computer, tablet, smartphone, or virtual machine. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing

logic operations. A selection of one or more asset categories may be received and updated as new visualizations are developed based on the current threats to an asset in that category. Asset categories may be selected or deselected by an end user. Once a particular asset category is selected, it may be added to a list of selected asset categories with a list for deselected categories as well. The list of selected asset categories may organized per category or may be compiled to include all current threats in a comprehensive selected asset threat list.

[00395] One exemplary embodiment may further include a processor configured to cause the display to present a list of assets in the selected category that have cyber security risks (step 1105). When an asset in a selected category has a cyber security risk associated with it, the asset may appear in a list of all listed categories having a cyber security risk associated with it, and may be organized by individual category, type of cyber security risk, or some combination or sequential ordering methods based on the information available to the processor and its end user.

[00396] In one of many exemplary embodiments a processor may be configured to retrieve workload component cybersecurity risk information (step 1107A) for each listed asset. When a category of assets is selected, a processor may be configured to order those assets according to available cybersecurity risk information available to said processor, and may organize by type of risk, type of asset, or similar type of sequentialization. Cybersecurity risk information may be determined from historical threats to the workload component, the current active cybersecurity threats to workload components of certain types, or some combination of the same.

[00397] In an embodiment, a processor may be configured to retrieve cloud component cybersecurity risk information (step 1107B) for each listed asset. Cloud component cybersecurity risk information may include identification of the risk based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset. Cybersecurity risk information to a cloud component may be determined from historical threats within the cloud environment 106, current active cybersecurity threats to similar workload components in the cloud environment 106 or the level of risk to assets within the cloud environment 106, as outlined below.

[00398] An exemplary embodiment may include a system configured to cause the display to present a common interface (step 1109) providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information. The common interface may be configured to access component cybersecurity risk information based on the nature of the risk, the likelihood of the risk, or the severity of the risk to the component, including recursive risk – risk that can happen from an attacker getting access to a system and using it as a mean to reach other system. On the same interface, a different interface, or a similar and related interface, a display may present cloud component cybersecurity risk information based

on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset.

[00399] As an exemplary embodiment, the system may be configured to cause the display to present, in the common interface, an interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information (step 1111). With this display presented, an exemplary embodiment may allow for a user to prioritize data from additional sources. The interconnection between workload component cybersecurity risk information and cloud component cybersecurity risk information may generate an interface between them and may be presented as an interconnection between them. The common interface presented may include direct interconnections between the component cybersecurity risk information and the cloud component cybersecurity risk information or links from one set of cybersecurity risk information. For instance, the component cybersecurity risk information may include a link to access the cloud component cybersecurity risk information. Conversely, the cloud component cybersecurity risk information may include a link to access the component cybersecurity risk information.

[00400] In one embodiment, a system may be configured to provide the plurality of asset categories include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. A category may list the currently identified entities of that category type as described below. An account category may include one or more listed accounts of users of the system. An account category may further include listing of assets based on type of account associated with that user. A type of account may include active user account, passive user account, administrator account, maintainer account, a system account, superuser account, or a guest user account.

[00401] A container category may include one or more containers currently listed in the system. A container may involve specific versions of programming language runtimes, libraries required to run software, or another method of packaging applications abstracted from the cloud environment.

[00402] A database category may include one or more databases managed and maintained in relation to the system. Databases may include a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. Databases may be implemented using ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[00403] An image category may contain one or more images based on image type, size, and content. Images may include images of one or more virtual machines displayed. A

managed service category may include one or more managed services listed in the above system, to include at least client or customer-owned systems being managed by a third party entity.

[00404] A messaging service category may include services of users communicating with at least one other user such as Google Cloud Pub and Sub communications, AWS SQS, or similar queues found in Information Systems.

[00405] A monitoring category may include centralized or decentralized methods of security monitoring systems.

[00406] A network category may include one or more systems of establishing a form of communicative connectivity between systems, e.g., TCP-IP services.

[00407] A storage category may list one or more methods of data storage, to include Direct Attached Storage, Network Attached Storage, SSD Flash Drive Arrays, Hybrid Flash Arrays, Hybrid Cloud Storage, Backup Software, Backup Appliances, Cloud Storage, or similar. Storage may include data structures, instructions, or any other data to be contained in a storage medium.

[00408] A user category may include one or more users of a given system along with identifying information relating to its unique or semi-unique identifier, level of authorized access and permissions, and storage volumes accessible by said user.

[00409] An access category may include one or more levels of access permission and a listing of users or devices granted access at a given level of access permission to a system.

[00410] A virtual machine category may list one or more virtual machines accessible to a system and the levels and types of access granted to a given virtual machine.

[00411] A serverless category may list one or more storage volumes stored on a serverless computing system such as Kubeless, Fission, Google App Engine, AWS Lambda, Google Cloud Functions, IBM Cloud Function, IBM Cloud Code Engine, Microsoft Azure Functions, Cloudflare Workers, and Compute@Edge.

[00412] In one of many exemplary embodiments, a system may further include the common interface configured to display information relating to at least one of an asset type, a risk, a region, or an account. A common interface may include the ability to configure and reconfigure the asset type, risk, region, or account to appear in any order and at multiple formats for displaying the information relating to those categories.

[00413] In another embodiment, a system may provide the common interface configured to display description for each listed asset. The display of a description for each listed asset may be listed vertically or horizontally. The display may be displayed as written text, representative graphical figures, or similar method of displaying the aforementioned types of information for an asset.

[00414] In another embodiment, a system may further provide the common interface is configured to display at least one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, or an authentication. A vulnerability may include vulnerability data from a vulnerability database such as NVD, WPVulnDB, US-CERT, Node.js Security Working Group, OVAL – Red Hat, Oracle Linux, Debian, Ubuntu, SUSE, Ruby Advisory Database, JVN, Safety DB(Python), Alpine secdb, PHP Security Advisories Database, Amazon ALAS, RustSec Advisory Database, Red Hat Security Advisories, Microsoft MSRC, KB, Debian Security Bug Tracker, Kubernetes security announcements, Exploit Database, Drupal security advisories, JPCERT. An insecure configuration may include software flaws or misconfigurations, non-encrypted files, improper file or directory permissions, unpatched security flaws in server software, enabled or accessible administrative and debugging functions, administrative account vulnerabilities, SSL certificates and encryption settings not properly configured, or a similar misconfiguration. These misconfigurations may be discovered by performing a scan, e.g., may query devices and systems capable of routing traffic (e.g., load balancer 115, routers, switches, firewalls, and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations. Malware indicated may include Adware, Botnets, Cryptojacking, Malvertising, Ransomware, Remote Administration Tools (RATs), Rootkits, Spyware, Trojans, Virus Malware, Worm Malware, or similar attack vehicles. A neglected asset may include an asset that has been improperly maintained, patched, or similar cybersecurity security measure. A data at risk may include any packet of data that may be exposed to a cybersecurity threat. A lateral movement may include any pathway between assets where a cybersecurity risk can travel from one affected asset to another affected asset. An authentication may include a username, password, one-time password, two-factor authentication, or any other authentication mechanism to gain access to a cloud service provider's system. A common interface may display more than one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, and an authentication separately or concurrently for an asset. A display of said information may be ordered and listed in any order depending on user preference and input.

[00415] An exemplary embodiment may further include a system wherein the common interface is configured to display one or more possible attack vectors reaching the each listed asset. Generating this common interface display may include generating a map of an asset to include recording information such as a region identifier, site identifier, datacenter identifier, physical address, network address, workload name, or any other identifier which may be acquired via an API provided through a cloud service provider's system and demonstrating via a display the myriad attack vectors currently available to an analyzed asset. Display of the one or

more possible attack vectors reaching the each listed asset may be displayed as written text, representative graphical figures, or similar method of displaying the one or more possible attack vectors.

[00416] Another exemplary embodiment may further include a system wherein the common interface is configured to display a recommended mitigation tactic for the each listed asset. A recommended mitigation tactic may be generated based on historical data of the cybersecurity risk, the vulnerability attributed to the listed asset, or the data contained on the each listed asset. Mitigation tactics may be selected based on user input or may be implemented automatically based on the mitigation tactics likelihood of a successful mitigation of the cybersecurity risk. A mitigation tactic may include increased frequency of scanning, heightened access control measures, firewalls and antivirus software patches, increased patch management scheduling, continuous workload monitoring, or similar tactics.

[00417] In another exemplary embodiment is a system wherein the common interface is configured to display one or more workload metrics associated with the each listed asset. Workload metrics may be a statistical representation of the current performance of the workload, the capacity for additional processing of the workload, and the historical performance of the workload based on statistical trend.

[00418] Another exemplary embodiment may include a system wherein the at least one processor is further configured to provide a cybersecurity report for the each listed asset. A cybersecurity report may be generated by scanning system 101 following a complete scan of an analyzed asset, and may include a snapshot of one or more of the number of total cybersecurity threats to a listed asset, number of cybersecurity threats mitigated by recent performance, methods of cybersecurity risk mitigation in the listed asset, rate of cybersecurity threats detected in the listed asset, and possibility of failed cybersecurity mitigation tactics based on the frequency and nature of the historical cybersecurity threats. Data collected to generate a cybersecurity report may include, among many things, operating system packages, installed software applications, libraries, and program language libraries such as Java archives, Python packages, Go modules, Ruby gems, PHP packages, and Node.js modules, other software applications, library versions, software versions, and other identifying characteristics of software and operating systems, lists of users of each system or device (e.g., virtual machines 107A-D), each system's or device's services, password hashes, and application-specific configurations for software/services such as Apache, Nginx, SSH, and other services, bugs or other configuration risks, malware scan results, passwords, scripts, shell history, repositories, or other data that may contain passwords, cloud access keys, SSH keys, or other key/password/access information that provide unchecked access to important resources.

[00419] In another embodiment, a system may include the at least one processor is configured to identify a risk level distribution among the listed assets. The at least one processor

may be configured to list risk level distribution by highest level of risk, lowest level of risk, or largest number of cybersecurity risks attributed among the listed assets. A risk level may be high if the web service is connected to the Internet (e.g., has at least one public port forwarded through a firewall to the web service, is able to be accessed through a load balancer, is able to be accessed via reverse proxy). A risk level may be medium risk if the web service is only accessible internally (e.g., because of firewall configuration). A risk level may be low risk if access to the web server is blocked by a configuration of cloud infrastructure 106.

[00420] Another exemplary embodiment may include a system wherein the at least one processor is configured to receive a search query for a specific risk. A search query may be input by type of risk, name of a risk, or another semi-unique manner of identifying a specific cybersecurity risk. A search may be initiated by end user operating user device 102, by a maintainer or administrator of an analyzed asset, or by a virtual machine associated with operating scanning system 101, A search may be initiated by inputting a number of inputs, to include keywords, phrases, file type, or similar identifiers unique or semi-unique to a given cybersecurity risk. Upon initiating a search, the results generated by the search may be stored in the analyzed asset, transmitted to the operator of a processor running side scanning system 101, or generated for review by an end user operating user device 102. The search and display of a specific risk may be based on historical trends of cybersecurity risks in the listed asset, severity of cybersecurity risk to a listed asset, or a historical search for past cybersecurity threats to a listed asset.

[00421] Another embodiment may include a system wherein the at least one processor is configured to identify one or more assets vulnerable to the specific risk. For instance, a processor operating scanning system 101 may be configured to perform a scan of a set of one or more assets with potential exposure to a given risk, and upon identifying vulnerable pathways for an attack vector to reach a set of identified assets, may generate a list of assets that are potentially or currently exposed to one or more specific risk. Assets vulnerable to a specific risk may be determined to be vulnerable based on the type of the specific risk, the existence of specific risks in adjacent or similar workloads, and related risks that may also cause an asset to be vulnerable based on the specific risk determined.

[00422] In another embodiment is a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing a display to present a plurality of asset categories (step 1101). The method may include the use of a graphical user interface system and may include systems similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows 10 Mobile, PalmOS-Web OS, or Firefox OS. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and

the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of risk inventories. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D.

[00423] One of several embodiments may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising receiving, via an input device, a selection of a particular asset category. A method may include a personal computer, tablet, smartphone, or virtual machine. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations. A selection of one or more asset categories may be received and updated as new visualizations are developed based on the current threats to an asset in that category. Asset categories may be selected or deselected by an end user. Once a particular asset category is selected, it may be added to a list of selected asset categories with a list for deselected categories as well. The list of selected asset categories may organized per category or may be compiled to include all current threats in a comprehensive selected asset threat list.

[00424] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing the display to present a list of assets in the selected category that have cybersecurity risks (step 1105). A display may be graphical, sequential, or multi-dimensional. A list of selected categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. When an asset in a selected category has a cyber security risk associated with it, the asset may appear in a list of all listed categories having a cyber security risk associated with it, and may be organized by individual category, type of cyber security risk, or some combination or sequential ordering methods based on the information available to the processor and its end user.

[00425] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising retrieving workload component cybersecurity risk information 307A for each listed asset. When a category

of assets is selected, a processor may be configured to order those assets according to available cybersecurity risk information available to said processor, and may organize by type of risk, type of asset, or similar type of sequentialization. Cybersecurity risk information may be determined from historical threats to the workload component, the current active cybersecurity threats to workload components of certain types, or some combination of the same.

[00426] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising retrieving cloud component cybersecurity risk information 307B. Cloud component cybersecurity risk information may include identification of the risk based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset. Cybersecurity risk information to a cloud component may be determined from historical threats within the cloud environment 106, current active cybersecurity threats to similar workload components in the cloud environment 106 or the level of risk to assets within the cloud environment 106, as outlined above.

[00427] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing the display to present a common interface 309 providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information. The common interface may be configured to access component cybersecurity risk information based on the nature of the risk, the likelihood of the risk, or the severity of the risk to the component. On the same or a similar and related interface, a display may present cloud component cybersecurity risk information based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset.

[00428] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing the display to present, in the common interface, an interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information. The interconnection between workload component cybersecurity risk information 307A and cloud component cybersecurity risk information 307B may generate an interface between them 309 and may be presented as an interconnection between them 311. The common interface presented may include direct interconnections between the component cybersecurity risk information and the cloud component cybersecurity risk information or links from one set of cybersecurity risk information. For instance, the component cybersecurity risk information may include a link to access the cloud component cybersecurity risk information. Conversely, the cloud component cybersecurity risk information may include a link to access the component cybersecurity risk information.

[00429] In another embodiment, a method may include wherein the plurality of asset categories include at least one of an account category, an authentication category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. A category may list the currently identified entities of that category type as described below. An account category may include one or more listed accounts of users of the system. An account category may further include listing of assets based on type of account associated with that user. A type of account may include active user account, passive user account, administrator account, maintainer account, a system account, superuser account, or a guest user account.

[00430] A container category may include one or more containers currently listed in the system. A container may involve specific versions of programming language runtimes, libraries required to run software, or another method of packaging applications abstracted from the cloud environment.

[00431] A database category may include one or more databases managed and maintained in relation to the system. Databases may include a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. Databases may be implemented using ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[00432] An image category may contain one or more images based on image type, size, and content. Images may include images of one or more virtual machines displayed.

[00433] A managed service category may include one or more managed services listed in the above system, to include at least client or customer-owned systems being managed by a third party entity.

[00434] A messaging service category may include services of users communicating with at least one other user such as Google Cloud Pub and Sub communications, AWS SQS, or similar queues found in Information Systems.

[00435] A monitoring category may include centralized or decentralized methods of security monitoring systems.

[00436] A network category may include one or more systems of establishing a form of communicative connectivity between systems, e.g., TCP-IP services.

[00437] A storage category may list one or more methods of data storage, to include Direct Attached Storage, Network Attached Storage, SSD Flash Drive Arrays, Hybrid Flash Arrays, Hybrid Cloud Storage, Backup Software, Backup Appliances, Cloud Storage, or similar.

Storage may include data structures, instructions, or any other data to be contained in a storage medium.

[00438] A user category may include one or more users of a given system along with identifying information relating to its unique or semi-unique identifier, level of authorized access and permissions, and storage volumes accessible by said user.

[00439] An access category may include one or more levels of access permission and a listing of users or devices granted access at a given level of access permission to a system.

[00440] A virtual machine category may list one or more virtual machines accessible to a system and the levels and types of access granted to a given virtual machine.

[00441] Another embodiment may include a method wherein the common interface is configured to display information relating to at least one of an asset type, a risk, a region, or an account. A common interface may include the ability to configure and reconfigure the asset type, risk, region, or account to appear in any order and at multiple formats for displaying the information relating to those categories.

[00442] Another embodiment may include a method wherein the common interface is configured to display description for each listed asset. The display of a description for each listed asset may be listed vertically or horizontally. The display may be displayed as written text, representative graphical figures, or similar method of displaying the aforementioned types of information for an asset.

[00443] Another embodiment may include a method wherein the common interface is configured to display at least one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, or an authentication. A vulnerability may include vulnerability data from a vulnerability database such as NVD, WPVulnDB, US-CERT, Node.js Security Working Group, OVAL – Red Hat, Oracle Linux, Debian, Ubuntu, SUSE, Ruby Advisory Database, JVN, Safety DB(Python), Alpine secdb, PHP Security Advisories Database, Amazon ALAS, RustSec Advisory Database, Red Hat Security Advisories, Microsoft MSRC, KB, Debian Security Bug Tracker, Kubernetes security announcements, Exploit Database, Drupal security advisories, JPCERT. An insecure configuration may include software flaws or misconfigurations, non-encrypted files, improper file or directory permissions, unpatched security flaws in server software, enabled or accessible administrative and debugging functions, administrative account vulnerabilities, SSL certificates and encryption settings not properly configured, or a similar misconfiguration. These misconfigurations may be discovered by performing a scan, e.g., may query devices and systems capable of routing traffic (e.g., load balancer 115, routers, switches, firewalls, and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations. Malware indicated may include Adware, Botnets, Cryptojacking, Malvertising,

Ransomware, Remote Administration Tools (RATs), Rootkits, Spyware, Trojans, Virus Malware, Worm Malware, or similar attack vehicles. A neglected asset may include an asset that has been improperly maintained, patched, or similar cybersecurity security measure. A data at risk may include any packet of data that may be exposed to a cybersecurity threat. A lateral movement may include any pathway between assets where a cybersecurity risk can travel from one affected asset to another affected asset. An authentication may include a username, password, one-time password, two-factor authentication, or any other authentication mechanism to gain access to a cloud service provider's system. A common interface may display more than one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, and an authentication separately or concurrently for an asset. A display of said information may be ordered and listed in any order depending on user preference and input.

[00444] Another embodiment may include a method wherein the common interface is configured to display one or more possible attack vectors reaching the each listed asset. Generating this common interface display may include generating a map of an asset to include recording information such as a region identifier, site identifier, datacenter identifier, physical address, network address, workload name, or any other identifier which may be acquired via an API provided through a cloud service provider's system and demonstrating via a display the myriad attack vectors currently available to an analyzed asset. Display of the one or more possible attack vectors reaching the each listed asset may be displayed as written text, representative graphical figures, or similar metho of displaying the one or more possible attack vectors.

[00445] Another embodiment may include a method wherein the common interface is configured to display a recommended mitigation tactic for the each listed asset. A recommended mitigation tactic may be generated based on historical data of the cybersecurity risk, the vulnerability attributed to the listed asset, or the data contained on the each listed asset. Mitigation tactics may be selected based on user input or may be implemented automatically based on the mitigation tactics likelihood of a successful mitigation of the cybersecurity risk. A mitigation tactic may include increased frequency of scanning, heightened access control measures, firewalls and antivirus software patches, increased patch management scheduling, continuous workload monitoring, or similar tactics.

[00446] Another embodiment may include a method wherein the common interface is configured to display one or more workload metrics associated with the each listed asset. Workload metrics may be a statistical representation of the current performance of the workload, the capacity for additional processing of the workload, and the historical performance of the workload based on statistical trend.

[00447] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing a display to present a plurality of asset categories 301. A plurality may include at least one asset category to be displayed. Asset categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category.

[00448] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising receiving, via an input device, a selection of a particular asset category (step 1103). An input device may include a personal computer, tablet, smartphone, or virtual machine. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations. A selection of one or more asset categories may be received and updated as new visualizations are developed based on the current threats to an asset in that category. Asset categories may be selected or deselected by an end user. Once a particular asset category is selected, it may be added to a list of selected asset categories with a list for deselected categories as well. The list of selected asset categories may organized per category or may be compiled to include all current threats in a comprehensive selected asset threat list.

[00449] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing the display to present a list of assets in the selected category that have cybersecurity risks (step 1105). A display may be graphical, sequential, or multi-dimensional. A list of selected categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. When an asset in a selected category has a cyber security risk associated with it, the asset may appear in a list of all listed categories having a cyber security risk associated with it, and may be organized by individual category, type of cyber security risk, or

some combination or sequential ordering methods based on the information available to the processor and its end user.

[00450] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising retrieving workload component cybersecurity risk information for each listed asset (step 1107A). When a category of assets is selected, a processor may be configured to order those assets according to available cybersecurity risk information available to said processor, and may organize by type of risk, type of asset, or similar type of sequentialization. Cybersecurity risk information may be determined from historical threats to the workload component, the current active cybersecurity threats to workload components of certain types, or some combination of the same.

[00451] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising retrieving cloud component cybersecurity risk information for each listed asset (step 1107B). Cloud component cybersecurity risk information may include identification of the risk based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset. Cybersecurity risk information to a cloud component may be determined from historical threats within the cloud environment 106, current active cybersecurity threats to similar workload components in the cloud environment 106 or the level of risk to assets within the cloud environment 106, as outlined above.

[00452] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing the display to present a common interface providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information for each listed asset (step 1109). The common interface may be configured to access component cybersecurity risk information based on the nature of the risk, the likelihood of the risk, or the severity of the risk to the component. On the same or a similar and related interface, a display may present cloud component cybersecurity risk information based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset.

[00453] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing the display to present, in the common interface, an

interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information (step 1111). The interconnection between workload component cybersecurity risk information (step 1107A) and cloud component cybersecurity risk information (step 1107B) may generate an interface between them (step 1109) and may be presented as an interconnection between them (step 1111). The common interface presented may include direct interconnections between the component cybersecurity risk information and the cloud component cybersecurity risk information or links from one set of cybersecurity risk information. For instance, the component cybersecurity risk information may include a link to access the cloud component cybersecurity risk information. Conversely, the cloud component cybersecurity risk information may include a link to access the component cybersecurity risk information.

[00454] In some embodiments, a processor may cause a display of one or more risks aggregated in the same visualization. **Fig. 12** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of displaying aggregated cybersecurity risk information, consistent with disclosed embodiments. A visualization may be generated from one or more sources based on scanning system 101. For example, a visual display may include a cloud provider visualization 1201, displaying where the multiple sources of data may be derived from. A cloud service provider may include services such as Cloudflare, Amazon Web Services, Google Cloud, IBM Cloud, Oracle Cloud, Microsoft Azure, or similar. This is further displayed via a communicative connection through an access port 1203 to a data server 1205.

[00455] Data server 1205 may include servers such as 109A-D. Data server 1205 may then display risk information that may be communicatively passed through web server 1207, demonstrating one or more possible sources of the one or more cybersecurity risk information. Data server 1205 may be communicatively connected to a web server 1207 (through access port 1203, which may be the same or a different port from port 1203 between cloud service provider 1201 and data server 1205), which can then be communicatively connected to one or more users via user access points 1209. This may include, among other things, user access that may be verified or validated via a credential 1211. Each of these connections may demonstrate one or more risk paths that may be exploited in a single view to allow for prioritization of threats and analysis of possible users who may exploit those risks and vulnerabilities, as discussed above. This may allow an administrator to prioritize risks to a given storage volume as well as assess lateral storage volumes that may be similarly risked based on the assessed threats in a single image.

[00456] **Fig. 13** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of a flow path for aggregated risk information, consistent with disclosed embodiments. In another embodiment, a processor may be configured to display a flow path of aggregated risk information. In one view, a displayed visualization may include user access

1301, which displays risk information such as service vulnerability 1303 and an insecure private key ~~1303~~ 1305. These vulnerabilities or risks may be displayed to an administrator via developer server 1307 to an administrator through administrator access 1309. As previously discussed, this will allow an administrator to visualize myriad risks to a storage volume in a single view that will allow an administrator to prioritize risks and analyze possible exploiters of this risk information on lateral systems and storage volumes.

[00457] In another embodiment, risk information may be shown in one or more views. This may involve a risk path similar to the exemplary embodiment of Fig. 13. In an exemplary embodiment, in a single visualization data from multiple sources may be displayed. This may involve identifying risk information such as the origin of the risk and potential exploiters of a risk. In some embodiments, this visualization may be done for a specific risk. In another embodiment, this visualization may be done for a combination of risks.

[00458] Disclosed embodiments may include any one of the following bullet-pointed features alone or in combination with one or more other bullet-pointed features, whether implemented as a method, by at least one processor, and/or stored as executable instructions on non transitory computer readable media:

- establishing a trusted relationship between a source account in a cloud environment and a scanner account;
- using the established trust relationship, utilize at least one cloud provider API to identify workloads in the source account;
- using the at least one cloud provider API to query a geographical location of at least one of the identified workloads;
- receiving an identification of the geographic location;
- using the cloud provider APIs to access block storage volumes of the at least one workload;
- determining a file-system of the at least one workload;
- mounting the block storage volumes on a scanner based on the determined file-system;
- activating a scanner at the geographic location;
- reconstructing from the block storage volumes a state of the workload; and
- assessing the reconstructed state of the workload to extract insights.
- wherein the geographic location includes an identifier of a physical site.
- wherein mounting includes selecting a driver corresponding to the determined file system.
- wherein the at least one processor is further configured to deploy a scanner at the geographical location.

- wherein the identification of the geographic location comprises an identification of a data center, at least one of a data center name, Internet Protocol (IP) address, name of the cloud provider, or a unique identity.
- wherein the reconstructed state of the workload includes at least two of an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content;
- wherein the at least one processor is further configured to update the reconstructed state of the workload based on at least one change to the block storage volumes.
- wherein the insights comprise at least one of a vulnerability associated with the workload or a composition of installed applications associated with the workload.
- wherein to mount the block storage volumes on the scanner, the at least one processor is configured to create a snapshot of the block storage volumes;
- wherein to mount the block storage volumes on the scanner, the at least one processor is configured to mount the snapshot of the block storage volumes on the scanner
- wherein the at least one processor is further configured to encrypt the snapshot of the block storage volumes;
- wherein the at least one processor is further configured to mount the encrypted snapshot of the block storage volumes on the scanner;
- using a cloud provider API, access a block storage volume of a workload maintained in a cloud storage environment;
- identifying an installed software application in the accessed block storage volume;
- analyzing the identified installed software application to determine an associated software version;
- accessing a data structure of known software vulnerabilities for a plurality of versions of software applications;
- performing a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and
- performing at least one of query the cloud provider API to determine network accessibility information related to the workload, identify at least one port on which the vulnerable application is accessible, use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.
- implementing a remedial action in response to the identified one or more vulnerabilities.
- wherein the remedial measure includes transmitting an alert to a device associated with an administrator.

- wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises examining data sources associated with the workload
- wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises determining the network accessibility information based on the examined data sources.
- wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.
- identifying the installed software application comprises extracting data from at least one of operating system packages, libraries, or program language libraries;
- identifying the installed software application comprises identifying the installed software application based on the extracted data;
- identifying a version of the installed software application;
- wherein the identified installed software application includes one or more scripts;
- wherein the data structure includes aggregated vulnerability data;
- wherein the aggregated vulnerability data includes data from one or more third-party vendors;
- wherein the aggregated vulnerability data includes data collected by a scanner;
- wherein the aggregated vulnerability data includes at least one of an advisory, an exploit, a security announcement, or a known bug.
- querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.
- accessing at least one cloud provider API to determine a plurality of entities capable of routing traffic in a virtual cloud environment associated with a target account containing the workload;
- querying the at least one cloud provider API to determine at least one networking configuration of the entities;
- building a graph connecting the plurality of entities based on the networking configuration;
- accessing a data structure identifying services publicly accessible via the internet and capable of serving as an internet proxy;
- integrating the identified services into the graph;

- traversing the graph to identify at least one source originating via the internet and reaching the workload; and
- outputting a risk notification associated with the workload.
- wherein the plurality of entities includes a virtual network appliance.
- wherein the virtual network appliance is at least one of a load balancer, a firewall, a proxy, or a router.
- wherein the networking configuration is at least one of a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration.
- wherein the graph includes a data structure sequentially connecting entities.
- wherein the graph includes directional vectors indicating directions of dataflow.
- wherein building the graph comprises identifying individual entities as nodes
- wherein building the graph comprises connecting the nodes.
- wherein the graph includes port numbers associated with the workload.
- wherein the graph includes a path from the at least one source to the workload.
- utilizing a cloud provider API to access a block storage volume of a workload maintained on a target account in a target system of a cloud storage environment;
- utilizing a scanner at a location of the block storage volume and on a secondary system other than the target system;
- scanning the block storage volume for malicious code, using the secondary system;
- identifying malicious code based on the scan; and
- outputting from the secondary system, a notification of a presence of malicious code in the target system.
- wherein the location of the block storage volume includes at least one of: the target account, a secondary system account, a cloud provider account, or a third party account.
- wherein scanning the block storage volume includes scanning disk-backed memory.
- wherein the disk-backed memory includes at least one of a page file or a cache file.
- wherein the secondary system includes at least one of a virtual machine, a container, or a serverless function.
- wherein the secondary system has an operating system different from an operating system of the target account.
- wherein the malicious code includes a rootkit.
- wherein utilizing a scanner includes suspending an operation of the scanner after the scan of the block storage volume.

- wherein utilizing a scanner includes modifying a pre-utilized scanner at the location of the block storage volume based on information related to the target account to obtain a modified scanner;
- wherein utilizing a scanner includes utilizing the modified scanner.
- identifying assets in a cloud environment;
- identifying risks associated with each of the identified asset;
- identifying relationships between at least some of the identified assets, the relationships including at least one of a trust, a network connectivity, or a mechanism of network proxying;
- receiving an identification of a specific asset under investigation;
- performing a forward analysis of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment;
- performing a backward analysis of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset; and
- outputting a signal to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points.
- wherein the network outside the cloud environment includes the Internet.
- wherein the assets in the cloud environment include at least one of: a virtual machine, a network appliance, a storage appliance, a compute instances, or an engine instance.
- wherein identifying the assets in a cloud environment includes identifying the assets based on at least one of: an identity and access management policy, an organization policy, or an access policy.
- wherein the presentation of the forward and backward paths indicates alternative paths connecting between the specific asset and an upstream asset or a downstream asset.
- wherein the visualization includes a presentation of the alternative paths.
- wherein the presentation of the forward and backward paths indicates port numbers for each pathway.
- wherein the visualization of the entry points indicates at least one entry point at risk.
- monitoring network activities of the assets in a cloud environment.
- detecting detect a potential risk associated with the specific asset based on the monitored network activities.

- detecting a potential risk associated with the specific asset based on a network activity of the specific asset.
- detecting a potential risk associated with the specific asset based on a network activity of an upstream asset of the specific asset.
- detecting a potential risk associated with the specific asset based on a network activity of a downstream asset of the specific asset.
- analyzing a cloud environment to identify a plurality of keys to the compute resources in the cloud environment;
- performing a cryptographic analysis on the plurality of keys to identify a first set of fingerprints that uniquely identify each of the plurality of keys, the first set of fingerprints being non-functional;
- analyzing trust configurations of the compute resources to identify a second set of fingerprints for each of the compute resources; and
- comparing the first set of fingerprints with the second set of fingerprints to match keys with the compute resources without using the keys to access the compute resources.
- wherein the plurality of keys are stored in at least one workload.
- wherein at least one of the first set of fingerprints is not identical to any key of the plurality of keys.
- wherein at least one of the plurality of keys includes at least one of a password, a script containing a password, a private component of a private-public key pair, a cloud key, or an Secure Shell (SSH) key.
- testing validity of at least one of the plurality of keys.
- analyzing a multi-machine interaction in the cloud environment using the first set of fingerprints.
- analyzing the multi-machine interaction includes comparing the first set of fingerprints with the second set of fingerprints.
- analyzing a multi-machine interaction in the cloud environment using the plurality of keys.
- accessing a primary account maintained in a cloud environment;
- receiving information defining a structure of the primary account, wherein the structure includes a plurality of assets, and wherein the information excludes raw data of the primary account;
- deploying, inside the primary account or inside a secondary account for which trust has been established with the primary account, at least one ephemeral scanner configured to scan at least one block storage volume and to output metadata defining the at least one block storage volume, the output excluding raw data of the primary account;

- receiving a transmission of the metadata from the at least one ephemeral scanner, wherein the transmission excludes raw data of the primary account;
- analyzing the received metadata to identify a plurality of cybersecurity vulnerabilities;
- correlating each of the identified plurality of cybersecurity vulnerabilities with one of the plurality of assets; and
- generating a report correlating the plurality of cybersecurity vulnerabilities with the plurality of assets.
- wherein defining the at least one block storage volume includes presenting risk data without sharing consumer data or data that was used to identify the risk data.
- wherein the metadata defining the at least one block storage volume includes at least one of: an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, or a profile configuration.
- receiving a transmission of updated metadata defining the at least one block storage volume in response to at least one change to the at least one block storage volumes.
- wherein the ephemeral scanner is configured to perform vulnerability scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform configuration scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform malware scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform lateral-movement risk analysis of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform sensitive information scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform container scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform keys and password scanning of the at least one block storage volume.
- causing a display to present a plurality of asset categories;
- receiving, via an input device, a selection of a particular asset category;
- causing the display to present a list of assets in the selected category that have cyber security risks;
- for each listed asset, retrieving workload component cybersecurity risk information, and
- for each listed asset. retrieving cloud component cybersecurity risk information;

- for each listed asset, causing the display to present a common interface providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information; and
- for each listed asset, causing the display to present, in the common interface, an interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information.
- wherein the plurality of asset categories include at least one of an account category, an authentication category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category.
- wherein the common interface is configured to display information relating to at least one of an asset type, a risk, a region, or an account.
- wherein the common interface is configured to display description for each listed asset.
- wherein the common interface is configured to display at least one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, or an authentication.
- wherein the common interface is configured to display one or more possible attack vectors reaching the each listed asset.
- wherein the common interface is configured to display a recommended mitigation tactic for the each listed asset.
- wherein the common interface is configured to display one or more workload metrics associated with the each listed asset.
- wherein the at least one processor is further configured to provide a cybersecurity report for the each listed asset.
- wherein the at least one processor is configured to identify a risk level distribution among the listed assets.
- receiving a search query for a specific risk; or
- identifying one or more assets vulnerable to the specific risk.

[00459] Other embodiments will be apparent to those skilled in the art from consideration of the specification and practice of the disclosed embodiments disclosed herein. It is intended that the specification and examples be considered as exemplary only, with a true scope and spirit of the disclosed embodiments being indicated by the following claims.

[00460] Computer programs based on the written description and methods of this specification are within the skill of a software developer. The various programs or program modules can be created using a variety of programming techniques. One or more of such

software sections or modules can be integrated into a computer system, non-transitory computer readable media, or existing software.

[00461] Moreover, while illustrative embodiments have been described herein, the scope includes any and all embodiments having equivalent elements, modifications, omissions, combinations (e.g., of aspects across various embodiments), adaptations or alterations based on the present disclosure. The elements in the claims are to be interpreted broadly based on the language employed in the claims and not limited to examples described in the present specification or during the prosecution of the application. These examples are to be construed as non-exclusive. Further, the steps of the disclosed methods can be modified in any manner, including by reordering steps or inserting or deleting steps. It is intended, therefore, that the specification and examples be considered as exemplary only, with a true scope and spirit being indicated by the following claims and their full scope of equivalents.

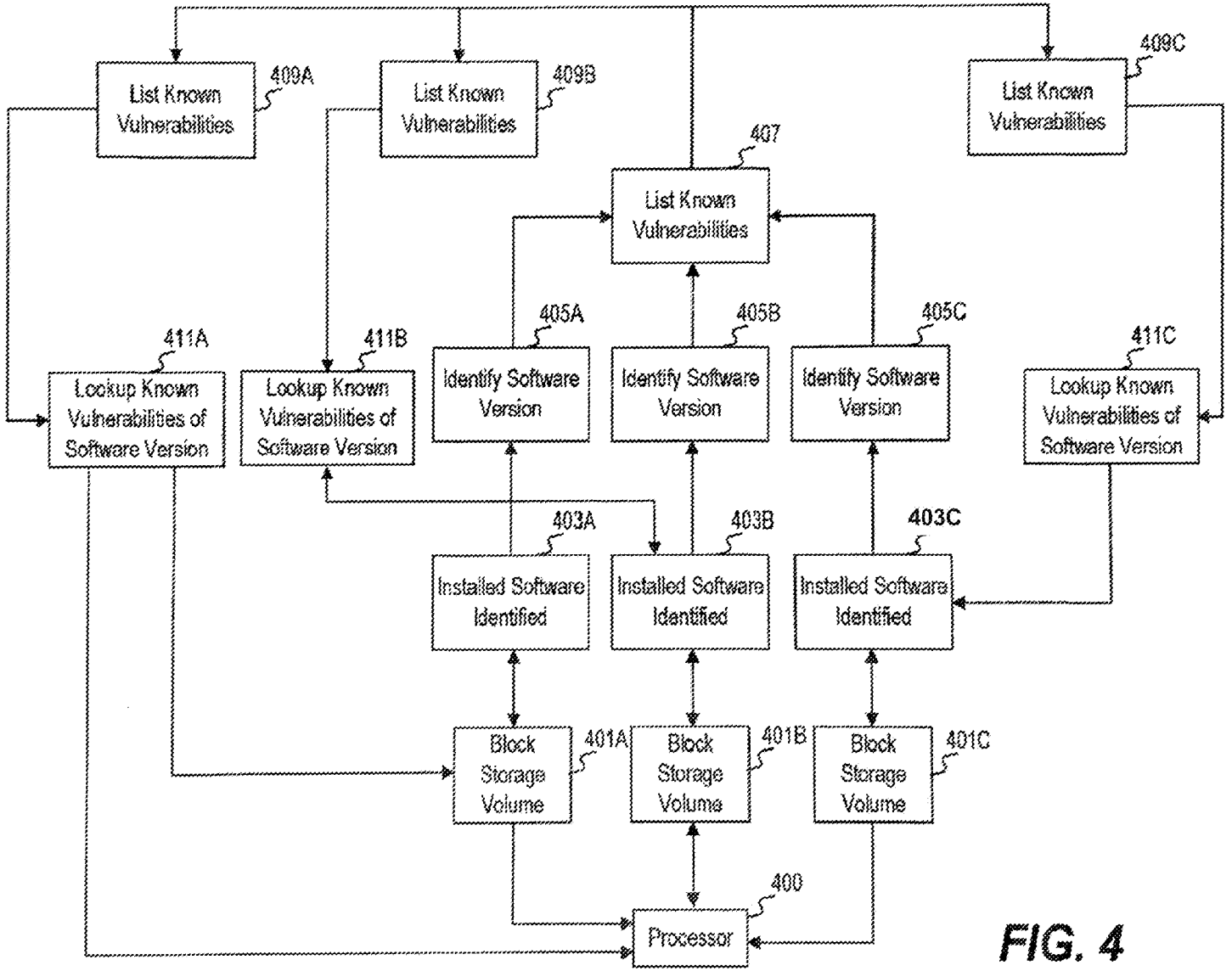

List Known Vulnerabilities
409A
List Known Vulnerabilities
409B
List Known Vulnerabilities
409C
407
List Known Vulnerabilities
405A
405B
405C
411A
411B
411C
Identify Software Version
Identify Software Version
Identify Software Version
Lookup Known Vulnerabilities of Software Version
Lookup Known Vulnerabilities of Software Version
Lookup Known Vulnerabilities of Software Version
403A
403B
403C
Installed Software Identified
Installed Software Identified
Installed Software Identified
Block Storage Volume
401A
Block Storage Volume
401B
Block Storage Volume
401C
400
Processor

FIG. 4

PTO/SB/06 (09-11)
Approved for use through 1/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 17/657,972	Filing Date 04/05/2022	☐ To be Mailed

ENTITY: ☑ LARGE ☐ SMALL ☐ MICRO

APPLICATION AS FILED - PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA		RATE ($)	FEE ($)
☐ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A		N/A	
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A		N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A		N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*		x $100 =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*		x $480 =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $310 ($155 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).				
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))					
* If the difference in column 1 is less than zero, enter "0" in column 2.				TOTAL	

APPLICATION AS AMENDED - PART II

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT	08/19/2022	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 20	Minus	** 20	= 0		x $100 =	0
	Independent (37 CFR 1.16(h))	* 3	Minus	*** 3	= 0		x $480 =	0
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	0

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	*	Minus	**	=		x $0 =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=		x $0 =	
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.	ADMIN
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".	/ERNEST D DAVIS/
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".	
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.	

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

PTO/SB/08a (01-10)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	17657972
Filing Date	2022-04-05
First Named Inventor	Avi SHUA
Art Unit	2431
Examiner Name	Sarah Su
Attorney Docket Number	15635.0005-00000

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1					

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20200137097	A1	2020-04-30	Gil ZIMMERMAN et al.	
	2	20190281115	A1	2019-09-12	Douglas SWALLOW et al.	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	17657972
	Filing Date	2022-04-05
	First Named Inventor	Avi SHUA
	Art Unit	2431
	Examiner Name	Sarah Su
	Attorney Docket Number	15635.0005-00000

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1	International Search Report and Written Opinion for PCT International Application No. PCT/IB2022/52670 filed March 23, 2022; pgs. 1-14; mailed July 26, 2022.	

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	17657972
Filing Date	2022-04-05
First Named Inventor	Avi SHUA
Art Unit	2431
Examiner Name	Sarah Su
Attorney Docket Number	15635.0005-00000

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

☒ That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Christopher C. Johns/	Date (YYYY-MM-DD)	2022-08-11
Name/Print	Christopher C. Johns	Registration Number	68664

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

PATENT COOPERATION TREATY

From the INTERNATIONAL SEARCHING AUTHORITY

To: CHRISTOPHER C. JOHNS
FINNEGAN, HENDERSON, FARABOW,
GARRETT, & DUNNER L.L.P.
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

PCT

NOTIFICATION OF TRANSMITTAL OF
THE INTERNATIONAL SEARCH REPORT AND
THE WRITTEN OPINION OF THE INTERNATIONAL
SEARCHING AUTHORITY, OR THE DECLARATION

(PCT Rule 44.1)

Date of mailing *(day/month/year)*	JUL 26 2022
Applicant's or agent's file reference 15635-0004-00304	**FOR FURTHER ACTION** See paragraphs 1 and 4 below
International application No. PCT/IB 22/52670	International filing date *(day/month/year)* 23 March 2022 (23.03.2022)
Applicant ORCA SECURITY	

1. ☒ The applicant is hereby notified that the international search report and the written opinion of the International Searching Authority have been established and are transmitted herewith.

   Filing of amendments and statement under Article 19:
   The applicant is entitled, if he so wishes, to amend the claims of the international application (see Rule 46):

   When? The time limit for filing such amendments is normally two months from the date of transmittal of the international search report.

   How? Directly to the International Bureau preferably through ePCT, or on paper to:
   The International Bureau of WIPO, 34, chemin des Colombettes, 1211 Geneva 20, Switzerland

   For more detailed instructions, see the *PCT Applicant's Guide*, International Phase, paragraphs 9.004 – 9.011.

2. ☐ The applicant is hereby notified that no international search report will be established and that the declaration under Article 17(2)(a) to that effect and the written opinion of the International Searching Authority are transmitted herewith.

3. ☐ **With regard to any protest** against payment of (an) additional fee(s) under Rule 40.2, the applicant is notified that:
   - ☐ the protest together with the decision thereon has been transmitted to the International Bureau together with any request to forward the texts of both the protest and the decision thereon to the designated Offices.
   - ☐ no decision has been made yet on the protest; the applicant will be notified as soon as a decision is made.

4. **Reminders**

   The applicant may **submit comments on an informal basis on the written opinion of the International Searching Authority** to the International Bureau. These comments will be made available to the public after international publication. The International Bureau will send a copy of such comments to all designated Offices unless an international preliminary examination report has been or is to be established.

   Shortly after the expiration of **18 months from the priority date, the international application will be published** by the International Bureau. If the applicant wishes to avoid or postpone publication, a notice of withdrawal of the international application, or of the priority claim, must reach the International Bureau before the completion of the technical preparations for international publication (Rules 90*bis*.1 and 90*bis*.3).

   Within **19 months** from the priority date, but only in respect of some designated Offices, a demand for international preliminary examination must be filed if the applicant wishes to postpone the entry into the national phase **until 30 months** from the priority date (in some Offices even later); otherwise, the applicant must, **within 20 months** from the priority date, perform the prescribed acts for **entry into the national phase** before those designated Offices. In respect of other designated Offices, the time limit of **30 months** (or later) will apply even if no demand is filed within 19 months. For details about the applicable time limits, Office by Office, see www.wipo.int/pct/en/texts/time_limits.html and the *PCT Applicant's Guide*, National Chapters.

   Within **22 months from the priority date, the applicant may request that a supplementary international search be carried out** by a different International Searching Authority that offers this service (Rule 45*bis*.1). The procedure for requesting supplementary international search is described in the *PCT Applicant's Guide*, International Phase, paragraphs 8.006-8.032.

Name and mailing address of the ISA/US Mail Stop PCT, Attn: ISA/US Commissioner for Patents P.O. Box 1450, Alexandria, Virginia 22313-1450 Facsimile No. 571-273-8300	Authorized officer Kari Rodriquez Telephone No. PCT Helpdesk: 571-272-4300

0991

Form PCT/ISA/220 (revised January 2020)

PATENT COOPERATION TREATY

PCT

INTERNATIONAL SEARCH REPORT

(PCT Article 18 and Rules 43 and 44)

Applicant's or agent's file reference	FOR FURTHER ACTION
15635-0004-00304	see Form PCT/ISA/220 as well as, where applicable, item 5 below.

International application No.	International filing date *(day/month/year)*	(Earliest) Priority Date *(day/month/year)*
PCT/IB 22/52670	23 March 2022 (23.03.2022)	26 April 2021 (26.04.2021)

Applicant
ORCA SECURITY

This international search report has been prepared by this International Searching Authority and is transmitted to the applicant according to Article 18. A copy is being transmitted to the International Bureau.

This international search report consists of a total of 5 sheets.

☐ It is also accompanied by a copy of each prior art document cited in this report.

1. **Basis of the report**

   a. With regard to the **language**, the international search was carried out on the basis of:

   ☒ the international application in the language in which it was filed.

   ☐ a translation of the international application into _____ which is the language of a translation furnished for the purposes of international search (Rules 12.3(a) and 23.1(b)).

   b. ☐ This international search report has been established taking into account the **rectification of an obvious mistake** authorized by or notified to this Authority under Rule 91 (Rule 43.6*bis*(a)).

   c. ☐ With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, see Box No. I.

2. ☐ **Certain claims were found unsearchable** (see Box No. II).

3. ☒ **Unity of invention is lacking** (see Box No. III).

4. With regard to the **title**,

   ☒ the text is approved as submitted by the applicant.

   ☐ the text has been established by this Authority to read as follows:

5. With regard to the **abstract**,

   ☒ the text is approved as submitted by the applicant.

   ☐ the text has been established, according to Rule 38.2, by this Authority as it appears in Box No. IV. The applicant may, within one month from the date of mailing of this international search report, submit comments to this Authority.

6. With regard to the **drawings**,

   a. the figure of the **drawings** to be published with the abstract is Figure No. 1

   ☒ as suggested by the applicant.

   ☐ as selected by this Authority, because the applicant failed to suggest a figure.

   ☐ as selected by this Authority, because this figure better characterizes the invention.

   b. ☐ none of the figures is to be published with the abstract.

0992

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IB 22/52670

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:
This application contains the following inventions or groups of inventions which are not so linked as to form a single general inventive concept under PCT Rule 13.1.

Group I: Claims 1-20, directed to a system, method and medium for analysis of cloud workloads.

Group II: Claims 21-40, directed to a system, method and medium for analysis of installed cloud software to determine security vulnerabilities.

Group III: Claims 41-60, directed to a system, method and medium for detecting the network structure of a cloud system and performing a risk analysis.

Group IV: Claims 61-80, directed to a system, method and medium for scanning cloud storage for malicious code.

(PLEASE SEE EXTRA SHEET)

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying additional fees, this Authority did not invite payment of additional fees.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:
1-20

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.

☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.

☐ No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IB 22/52670

A. CLASSIFICATION OF SUBJECT MATTER

IPC - G06F 9/50 (2022.01)

CPC - G06F 9/50, H04L 63/00, G06F 2009/45591, G06F 16/128

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
See Search History document

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
See Search History document

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
See Search History document

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2020/0137097 A1 (CISCO TECHNOLOGY, INC.), 30 April 2020 (30.04.2020), entire document, especially Abstract; para [0012], [0098], [0109]-[0110], [0125], [0131], [0157], [0262], [0272], [0355], [0514], [0550]	1-20
Y	US 2019/0281115 A1 (COMMVAULT SYSTEMS, INC.), 12 September 2019 (12.09.2019), entire document, especially Abstract; para [0005], [0086], [0143], [0150], [0266], [0310]	1-20

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"D" document cited by the applicant in the international application
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
02 July 2022 (02.07.2022)	JUL 26 2022

Name and mailing address of the ISA/US	Authorized officer
Mail Stop PCT, Attn: ISA/US, Commissioner for Patents P.O. Box 1450, Alexandria, Virginia 22313-1450 Facsimile No. 571-273-8300	Kari Rodriquez Telephone No. PCT Helpdesk: 571-272-4300

0994

Continued from Box III - Observations where unity of invention is lacking:

Group V: Claims 81-100, directed to a system, method and medium for performing forward and backward analysis of cloud assets to detect and visualize attack vectors.

Group VI: Claims 101-118, directed to a system, method and medium for performing cryptographic analysis on keys in a cloud environment to match relationships between keys and their relevant resources.

Group VII: Claims 119-138, directed to a system, method and medium for scanning account data in a cloud environment in order to detect security vulnerabilities.

Group VIII: Claims 139-159, directed to a system, method and medium for displaying asset categories in a cloud environment and displaying interconnections between workload risks and component risks.

The groups of inventions listed above do not relate to a single general inventive concept under PCT Rule 13.1 because, under PCT Rule 13.2, they lack the same or corresponding special technical features for the following reasons:

The special technical feature of the Group I invention is identifying a geographical location of a workload, activating a scanner at the geographical location, reconstructing a state of the workload, and assessing the reconstructed state of the workload to extract insights, not required by any of Groups II-VIII.

The special technical feature of the Group II invention is identifying installed software, determining an associated software version, performing a lookup to determine software vulnerabilities, and identifying said vulnerabilities and the ports affected, not required by Group I or II-VIII.

The special technical feature of the Group III invention is determining cloud entities capable of routing network traffic, querying a cloud API to determine a network configuration, building a network configuration graph, identifying network services, integrating those services into a the graph, traversing the graph, and identifying and outputting network risks, not required by Group I-II or IV-VIII.

The special technical feature of the Group IV invention is accessing block storage, scanning block storage for malicious code, identifying malicious code, and notifying a user of said malicious code, not required by Group I-III or V-VIII.

The special technical feature of the Group V invention is identifying cloud assets, identifying risks related to said assets, performing forward and backward analysis on specifically identified assets under investigation, and outputting a visualization of risks identified in forward and backward analysis, not required by Group I-IV or VI-VIII.

The special technical feature of the Group VI invention is analyzing a cloud environment to identify cryptographic keys, analysing said keys to identify fingerprints that uniquely identify said keys, identifying a second set of fingerprints related to a resource, and comparing the sets of fingerprints to determine matching relationships between keys and resources without using the keys, not required by Group I-V or VII-VIII.

The special technical feature of the Group VII invention is accessing a primary account in a cloud environment, receiving information on assets related to the account excluding raw data, deploying an ephemeral scanner to scan block storage and output metadata defining the volume, receiving and analyzing the metadata to determine vulnerabilities, correlating the vulnerabilities with assets, and outputting a report correlating the assets with the vulnerabilities, not required by Group I-VI or VIII.

The special technical feature of the Group VIII invention is a graphical interface displaying cloud asset categories, receiving a selection of a category, displaying a list of assets with risks in a selected category, retrieving and displaying cybersecurity risks related to workload and cloud components, and displaying interconnections between workload risks and component risks, not required by Group I-VII.

Groups I-VIII share a technical feature, being related to analysis and reporting in a cloud environment.

However, these shared technical features fail to represent a contribution over the prior art of US 2020/0244692 A1 to Orca Security LTD. (hereinafter 'Orca'), which discloses the analysis of cloud assets for security vulnerabilities ("In some example embodiments, the security system 140 may further interface with the cloud computing platform 110 and external systems 170. The external systems may include intelligence systems, security information and event management (SIEM) systems, and mitigation tools. The external intelligence systems may include common vulnerabilities and exposures (CVE RTM.) databases, reputation services, security systems (providing feeds on discovered threats), and so on. The information provided by the intelligence systems may detect certain known vulnerabilities identified in, for example, a CVE database," para [0030]).

Groups I-IV all share the technical feature of a cloud provider API.

However, these shared technical features fail to represent a contribution over the prior art of Orca, which teaches the use of a cloud API ("In an embodiment, the interface between the client environment 130 and the security system 140 can be realized using APIs or services provided by the cloud computing platform 11," para [0030]).

-- continued on next extra sheet --

-- Continued from previous sheet --

Groups I, IV and VII all share the technical features of a scanner that scans block storage volumes.

However, these shared technical features fail to represent a contribution over the prior art of US 2020/0137097 A1 to Cisco Technology, Inc., which discloses the scanning of cloud storage for malware and other security vulnerabilities ("Account compromise features (e.g., UBA/threat protection) for developers may include UBA, configuration monitoring, such as for AWS accounts (e.g., by CloudTrail and/or AWS Config), UBA for machine access and UBA for accesses to a code repository. UBA for machine access may include ssh activity (e.g., by CloudWatch). A code repository to which the services of the CSF 100 may be applied may be a Github code repository and the like. Features related to data breaches (e.g., as determined by Cloud DLP) for developers may include Github configuration monitoring and scanning of Github logs (e.g., S3/CloudWatch logs) such as for secrets or other sensitive data. Compliance features for developers may include scanning of storage, such as S3 buckets, RDS DB tables and DynamoDB objects. Cloud malware features for developers may include the capabilities of the AFW 300, such as for access keys or tokens, such as AWS access keys or tokens, as well as use of the AFW 300 for developer resources like Github, Slack, JIRA and the like, as well as scanning of code for included packages," para [0591]).

Groups I-II and VII all share the technical feature of identifying installed software.

However, these shared technical features fail to represent a contribution over the prior art of Orca, which teaches the identification and risk analysis of installed software and versions ("In an embodiment, the security system 140 is configured to detect whether there is vulnerable code executed by the VMs 119-1 and 119-2. In an embodiment, the VM 119-2 being analyzed is shut down, being, therefore, at rest. The VM 119-1 may be running or paused. In an embodiment, to detect vulnerabilities existing in the VM 119-2, the security system 140 is configured to match installed application lists, with their respective versions, to a known list of vulnerable applications. Further, the security system 140 may be configured to match the application files, either directly, using binary comparison, or by computing a cryptographic hash against database of files in vulnerable applications," para [0042]).

Thus, the inventions listed as Groups I-VIII lack unity of invention because they do not share a same or corresponding special technical feature providing a contribution over the prior art.

PATENT COOPERATION TREATY

From the
INTERNATIONAL SEARCHING AUTHORITY

To: CHRISTOPHER C. JOHNS
FINNEGAN, HENDERSON, FARABOW,
GARRETT, & DUNNER L.L.P.
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

PCT

WRITTEN OPINION OF THE
INTERNATIONAL SEARCHING AUTHORITY

(PCT Rule 43*bis*.1)

Date of mailing *(day/month/year)*	JUL 26 2022
Applicant's or agent's file reference 15635-0004-00304	FOR FURTHER ACTION See paragraph 2 below

International application No.	International filing date *(day/month/year)*	Priority date *(day/month/year)*
PCT/IB 22/52670	23 March 2022 (23.03.2022)	26 April 2021 (26.04.2021)

International Patent Classification (IPC) or both national classification and IPC

IPC - G06F 9/50 (2022.01)

CPC - G06F 9/50, H04L 63/00, G06F 2009/45591, G06F 16/128

Applicant ORCA SECURITY

1. This opinion contains indications relating to the following items:

- [x] Box No. I Basis of the opinion
- [ ] Box No. II Priority
- [ ] Box No. III Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
- [x] Box No. IV Lack of unity of invention
- [x] Box No. V Reasoned statement under Rule 43*bis*.1(a)(i) with regard to novelty, inventive step and industrial applicability; citations and explanations supporting such statement
- [ ] Box No. VI Certain documents cited
- [ ] Box No. VII Certain defects in the international application
- [ ] Box No. VIII Certain observations on the international application

2. FURTHER ACTION

If a demand for international preliminary examination is made, this opinion will be considered to be a written opinion of the International Preliminary Examining Authority ("IPEA") except that this does not apply where the applicant chooses an Authority other than this one to be the IPEA and the chosen IPEA has notified the International Bureau under Rule 66.1*bis*(b) that written opinions of this International Searching Authority will not be so considered.

If this opinion is, as provided above, considered to be a written opinion of the IPEA, the applicant is invited to submit to the IPEA a written reply together, where appropriate, with amendments, before the expiration of 3 months from the date of mailing of Form PCT/ISA/220 or before the expiration of 22 months from the priority date, whichever expires later.

For further options, see Form PCT/ISA/220.

Name and mailing address of the ISA/US	Date of completion of this opinion	Authorized officer
Mail Stop PCT, Attn: ISA/US Commissioner for Patents P.O. Box 1450, Alexandria, Virginia 22313-1450 Facsimile No. 571-273-8300	03 July 2022 (03.07.2022)	Kari Rodriquez PCT Help Desk Telephone No. 571-272-4300

0997

WRITTEN OPINION OF THE
INTERNATIONAL SEARCHING AUTHORITY

International application No.
PCT/IB 22/52670

Box No. I Basis of this opinion

1. With regard to the **language**, this opinion has been established on the basis of:

   ☒ the international application in the language in which it was filed.

   ☐ a translation of the international application into _____ which is the language of a translation furnished for the purposes of international search (Rules 12.3(a) and 23.1(b)).

2. ☐ This opinion has been established taking into account the **rectification of an obvious mistake** authorized by or notified to this Authority under Rule 91 (Rule 43*bis*.1(b)).

3. ☐ With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, this opinion has been established on the basis of a sequence listing:

   a. ☐ forming part of the international application as filed:

      ☐ in the form of an Annex C/ST.25 text file.

      ☐ on paper or in the form of an image file.

   b. ☐ furnished together with the international application under PCT Rule 13*ter*.1(a) for the purposes of international search only in the form of an Annex C/ST.25 text file.

   c. ☐ furnished subsequent to the international filing date for the purposes of international search only:

      ☐ in the form of an Annex C/ST.25 text file (Rule 13*ter*.1(a)).

      ☐ on paper or in the form of an image file (Rule 13*ter*.1(b) and Administrative Instructions, Section 713).

4. ☐ In addition, in the case that more than one version or copy of a sequence listing has been filed or furnished, the required statements that the information in the subsequent or additional copies is identical to that forming part of the application as filed or does not go beyond the application as filed, as appropriate, were furnished.

5. Additional comments:

WRITTEN OPINION OF THE
INTERNATIONAL SEARCHING AUTHORITY

International application No.
PCT/IB 22/52670

Box No. IV Lack of unity of invention

1. ☒ In response to the invitation (Form PCT/ISA/206) to pay additional fees the applicant has, within the applicable time limit:

   ☐ paid additional fees.

   ☐ paid additional fees under protest and, where applicable, the protest fee.

   ☐ paid additional fees under protest but the applicable protest fee was not paid.

   ☒ not paid additional fees.

2. ☐ This Authority found that the requirement of unity of invention is not complied with and chose not to invite the applicant to pay additional fees.

3. This Authority considers that the requirement of unity of invention in accordance with Rule 13.1, 13.2 and 13.3 is

   ☐ complied with.

   ☒ not complied with for the following reasons:

This application contains the following inventions or groups of inventions which are not so linked as to form a single general inventive concept under PCT Rule 13.1.

Group I: Claims 1-20, directed to a system, method and medium for analysis of cloud workloads.

Group II: Claims 21-40, directed to a system, method and medium for analysis of installed cloud software to determine security vulnerabilities.

Group III: Claims 41-60, directed to a system, method and medium for detecting the network structure of a cloud system and performing a risk analysis.

Group IV: Claims 61-80, directed to a system, method and medium for scanning cloud storage for malicious code.

Group V: Claims 81-100, directed to a system, method and medium for performing forward and backward analysis of cloud assets to detect and visualize attack vectors.

Group VI: Claims 101-118, directed to a system, method and medium for performing cryptographic analysis on keys in a cloud environment to match relationships between keys and their relevant resources.

Group VII: Claims 119-138, directed to a system, method and medium for scanning account data in a cloud environment in order to detect security vulnerabilities.

Group VIII: Claims 139-159, directed to a system, method and medium for displaying asset categories in a cloud environment and displaying interconnections between workload risks and component risks.

The groups of inventions listed above do not relate to a single general inventive concept under PCT Rule 13.1 because, under PCT Rule 13.2, they lack the same or corresponding special technical features for the following reasons:

The special technical feature of the Group I invention is identifying a geographical location of a workload, activating a scanner at the geographical location, reconstructing a state of the workload, and assessing the reconstructed state of the workload to extract insights, not required by any of Groups II-VIII.

The special technical feature of the Group II invention is identifying installed software, determining an associated software version, performing a lookup to determine software vulnerabilities, and identifying said vulnerabilities and the ports affected, not required by Group I or II-VIII.

The special technical feature of the Group III invention is determining cloud entities capable of routing network traffic, querying a cloud API to determine a network configuration, building a network configuration graph, identifying network services, integrating those services into a the graph, traversing the graph, and identifying and outputting network risks, not required by Group I-II or IV-VIII.

(PLEASE SEE SUPPLEMENTAL BOX)

4. Consequently, this opinion has been established in respect of the following parts of the international application:

   ☐ all parts.

   ☒ the parts relating to claims Nos. 1-20

WRITTEN OPINION OF THE
INTERNATIONAL SEARCHING AUTHORITY

International application No.
PCT/IB 22/52670

Box No. V **Reasoned statement under Rule 43*bis*.1(a)(i) with regard to novelty, inventive step and industrial applicability; citations and explanations supporting such statement**

1. Statement

Novelty (N)	Claims	1-20	YES
	Claims	None	NO
Inventive step (IS)	Claims	None	YES
	Claims	1-20	NO
Industrial applicability (IA)	Claims	1-20	YES
	Claims	None	NO

2. Citations and explanations:

(PLEASE SEE SUPPLEMENTAL BOX)

WRITTEN OPINION OF THE
INTERNATIONAL SEARCHING AUTHORITY

International application No.
PCT/IB 22/52670

Supplemental Box

In case the space in any of the preceding boxes is not sufficient.
Continuation of:
Box IV.3 - Lack of unity of invention:

The special technical feature of the Group IV invention is accessing block storage, scanning block storage for malicious code, identifying malicious code, and notifying a user of said malicious code, not required by Group I-III or V-VIII.

The special technical feature of the Group V invention is identifying cloud assets, identifying risks related to said assets, performing forward and backward analysis on specifically identified assets under investigation, and outputting a visualization of risks identified in forward and backward analysis, not required by Group I-IV or VI-VIII.

The special technical feature of the Group VI invention is analyzing a cloud environment to identify cryptographic keys, analysing said keys to identify fingerprints that uniquely identify said keys, identifying a second set of fingerprints related to a resource, and comparing the sets of fingerprints to determine matching relationships between keys and resources without using the keys, not required by Group I-V or VII-VIII.

The special technical feature of the Group VII invention is accessing a primary account in a cloud environment, receiving information on assets related to the account excluding raw data, deploying an ephemeral scanner to scan block storage and output metadata defining the volume, receiving and analyzing the metadata to determine vulnerabilities, correlating the vulnerabilities with assets, and outputting a report correlating the assets with the vulnerabilities, not required by Group I-VI or VIII.

The special technical feature of the Group VIII invention is a graphical interface displaying cloud asset categories, receiving a selection of a category, displaying a list of assets with risks in a selected category, retrieving and displaying cybersecurity risks related to workload and cloud components, and displaying interconnections between workload risks and component risks, not required by Group I-VII.

Groups I-VIII share a technical feature, being related to analysis and reporting in a cloud environment.

However, these shared technical features fail to represent a contribution over the prior art of US 2020/0244692 A1 to Orca Security LTD. (hereinafter 'Orca'), which discloses the analysis of cloud assets for security vulnerabilities ("In some example embodiments, the security system 140 may further interface with the cloud computing platform 110 and external systems 170. The external systems may include intelligence systems, security information and event management (SIEM) systems, and mitigation tools. The external intelligence systems may include common vulnerabilities and exposures (CVE.RTM.) databases, reputation services, security systems (providing feeds on discovered threats), and so on. The information provided by the intelligence systems may detect certain known vulnerabilities identified in, for example, a CVE database," para [0030]).

Groups I-IV all share the technical feature of a cloud provider API.

However, these shared technical features fail to represent a contribution over the prior art of Orca, which teaches the use of a cloud API ("In an embodiment, the interface between the client environment 130 and the security system 140 can be realized using APIs or services provided by the cloud computing platform 11," para [0030]).

Groups I, IV and VII all share the technical features of a scanner that scans block storage volumes.

However, these shared technical features fail to represent a contribution over the prior art of US 2020/0137097 A1 to Cisco Technology, Inc., which discloses the scanning of cloud storage for malware and other security vulnerabilities ("Account compromise features (e.g., UBA/threat protection) for developers may include UBA, configuration monitoring, such as for AWS accounts (e.g., by CloudTrail and/or AWS Config), UBA for machine access and UBA for accesses to a code repository. UBA for machine access may include ssh activity (e.g., by CloudWatch). A code repository to which the services of the CSF 100 may be applied may be a Github code repository and the like. Features related to data breaches (e.g., as determined by Cloud DLP) for developers may include Github configuration monitoring and scanning of Github logs (e.g., S3/CloudWatch logs) such as for secrets or other sensitive data. Compliance features for developers may include scanning of storage, such as S3 buckets, RDS DB tables and DynamoDB objects. Cloud malware features for developers may include the capabilities of the AFW 300, such as for access keys or tokens, such as AWS access keys or tokens, as well as use of the AFW 300 for developer resources like Github, Slack, JIRA and the like, as well as scanning of code for included packages," para [0591]).

Groups I-II and VII all share the technical feature of identifying installed software.

However, these shared technical features fail to represent a contribution over the prior art of Orca, which teaches the identification and risk analysis of installed software and versions ("In an embodiment, the security system 140 is configured to detect whether there is vulnerable code executed by the VMs 119-1 and 119-2. In an embodiment, the VM 119-2 being analyzed is shut down, being, therefore, at rest. The VM 119-1 may be running or paused. In an embodiment, to detect vulnerabilities existing in the VM 119-2, the security system 140 is configured to match installed application lists, with their respective versions, to a known list of vulnerable applications. Further, the security system 140 may be configured to match the application files, either directly, using binary comparison, or by computing a cryptographic hash against database of files in vulnerable applications," para [0042]).

Thus, the inventions listed as Groups I-VIII lack unity of invention because they do not share a same or corresponding special technical feature providing a contribution over the prior art.

WRITTEN OPINION OF THE
INTERNATIONAL SEARCHING AUTHORITY

International application No.
PCT/IB 22/52670

Supplemental Box

In case the space in any of the preceding boxes is not sufficient.
Continuation of:
Box V.2 - Citations and explanations:

Claims 1-20 lack an inventive step under PCT Article 33(3) as being obvious over US 2020/0137097 A1 to CISCO TECHNOLOGY, INC. (hereinafter 'Cisco') in view of US 2019/0281115 A1 to COMMVAULT SYSTEMS, INC. (hereinafter 'Commvault').

As to claim 1, Cisco teaches a system for inspecting data, the system comprising: at least one processor configured to: establish a trusted relationship between a source account in a cloud environment and a scanner account ("The cloud security fabric may integrate a number of significant security components and other functional components that enable effective security of sensitive data in enterprises that have users who are using cloud resources. Components or modules of the architecture may include, without limitation, ones for information protection and ones for threat management, including, without limitation, ones for content inspection and analysis (including metadata); contextual analysis; user behavior analysis and modeling (e.g., for identifying suspicious logins from geographies, access patterns and other information obtained from behavioral modeling); global policy creation and policy automation (such as enabling management of work flows and implementing various rules, such as enabled by a rules engine, about taking action in the enterprise environment (including any cloud) with automated actions taking place in response to the policy engine); central configuration management (for security-related items); centralized auditing; incident management; federated searching; selective encryption (where a user (not just the enterprise as a whole) may self-select an additional measure of control over items of data); and security and user behavior analytics," para [0012]); using the established trust relationship, utilize at least one cloud provider API to identify workloads in the source account ("one or more of the developer APIs 102 (including cyber dev APIs 140) can enable a user of the CSF 100, such as a security application developer or IT administrator, to scan and access information collected from the various cloud resources, including SaaS applications 190 and applications being used in PaaS and IaaS environments 136," para [0098]); use the at least one cloud provider API to query a geographical location of at least one of the identified workloads ("To enable executive threat visualization, the following capabilities may be enabled, with the following benefits. In embodiments, an activities heat map widget may comprise a dashboard representation of recent activities (e.g., over the last 30 days) by location on a geographical map, with links that provide the ability to obtain additional information on activities (such as by linking to an activities page). This may include various filters, such as by city, by country, by application, by user, or the like," para [0157]); receive an identification of the geographic location (para [0157]; "Referring again to FIG. 1, in embodiments, the methods and systems disclosed herein may include one or more modules, associated with or as part of a cloud security fabric, for providing behavior analysis and modeling 114 (e.g., for identifying suspicious logins from geographies, access patterns and other information obtained from behavioral modeling)," para [0355]); use the cloud provider APIs to access block storage volumes of the at least one workload ("Infrastructure as a Service (IaaS) refers to online services, including cloud services, that may abstract a user from the detail of various infrastructure elements, like physical computing resources, location, data partitioning, scaling, security, backup and the like. As an example, a hypervisor, such as Xen, Oracle VirtualBox, KVM, VMware ESX/ESXi, or Hyper-V may run one or more virtual machines as guests, and pools of hypervisors within a cloud operational system may support large numbers of virtual machines, enabling the ability to scale services up and down according to customers' varying requirements. IaaS clouds may offer additional resources such as a virtual-machine disk-image library, raw block storage," para [0550]); determine a file-system of the at least one workload ("The extent and complexity of data collection for the CSF 100 varies. A comprehensive approach may involve scanning the entire file structure of an enterprise to obtain an extensive picture of the users, data, accounts, applications and the like. When one scans for all of that material, it is a very structured, highly intensive service. One enumerates the users, and for all the users, one scans and knows their files. One can also scan for file metadata, apply rules on the metadata, and pull down content for scanning. An alternative to full scanning of a file system is using a web hook for the domain to obtain selected information. While easier, that is still a structured process," para [0109]; "With a relatively simple layer of field mapping (e.g., mapping a particular user to a particular account and mapping a set of events to event types) one can provide considerable insight into security risks, threats and behavior. For example, one can provide basic frequency analysis, rules and anomaly detection around login events, which is an easier starting point than trying to understand an entire structure of a file system. Thus, processing events, such as user behavior modeling, allows one to obtain value from the CSF 100 without fully scanning a file structure," para [0110]); reconstruct from the block storage volumes a state of the workload ("Additional opportunities for the application of cyber intelligence based on UBA may include community intelligence (including intelligence relating to threats that affect multiple enterprises, including across different applications and platforms), secure configuration management, identity management integration, and real time session management. Real time session management may include reconstructing user sessions and allowing users to manage a session," para [0131]); and assess the reconstructed state of the workload to extract insights ("The UBA platform 500 seeks to extract actionable security insights from user behavior (and in some cases, application or machine behavior, in which case UBA is sometimes referred to as User and Entity Behavioral Analysis, or "UEBA"," para [0125]; para [0131]). Cisco fails to explicitly teach such a system further comprising: at least one processor configured to: mount the block storage volumes on a scanner based on the determined file-system; activate a scanner at the geographic location. However, Commvault teaches such a system further comprising: at least one processor configured to: mount the block storage volumes on a scanner based on the determined file-system ("During a backup or other secondary copy operation, the client computing device 202 mounts the designated NFS path and writes data to that NFS path. The NFS path may be obtained from NFS path data 215 stored locally at the client computing device 202, and which may be a copy of or otherwise derived from NFS path data 219 stored in the secondary storage subsystem 218," para [0266]); activate a scanner at the geographic location ("In some embodiments, the network storage system 302 may be managed by the same entity as the primary storage environment 117. Thus, in some embodiments, although the network storage system 302 may be remote from portions of the primary storage environment 117, the network storage system 302 may share a network with the client computing device 102. In some such embodiments, determining the number of scanning threads to instantiate may be based at least in part on the internal network configuration for the primary storage environment 117. Further, the number of scanning threads to instantiate may be based at least in part on the location of the network storage system 302 within the primary storage environment 117 relative to the location of the client computing device 102 and elements of the secondary storage environment 118 involved in the backup process," para [0310]). It would have been obvious to one of ordinary skill in the art to combine the cloud environment scanning system of Cisco with the support for locally mounting storage volumes and deploying scanners at specific geographic locations of Commvault, because such systems and methods allow for mounting data and performing scans at designated physical device locations (Commvault: para [0266], [0310]). Furthermore, both Cisco and Commvault are directed to systems and methods for scanning cloud storage systems to perform security and other operations.

1002

(PLEASE SEE NEXT SUPPLEMENTAL BOX)

WRITTEN OPINION OF THE
INTERNATIONAL SEARCHING AUTHORITY

International application No.
PCT/IB 22/52670

Supplemental Box

In case the space in any of the preceding boxes is not sufficient.
Continuation of:
Box V.2 - Citations and explanations:

As to claim 2, the system of claim 1 is discussed above. Further, Cisco teaches such a system wherein the geographic location includes an identifier of a physical site ("Data enrichment for a compromised account use case may include geoIP data enrichment to extract location information and user lookup data enrichment to get full user account details," para [0272]).

As to claim 3, the system of claim 1 is discussed above. Further, Commvault teaches such a system wherein mounting includes selecting a driver corresponding to the determined file system ("One system to perform differential backups and/or de-duplication may include installing a data agent or filter driver on a system that is to be backed up. For example, a computer system included as part of a primary storage environment may have a filter driver that is located between the operating system of the computer system and the hardware of the computer system. This filter driver can monitor read and write commands, which can enable the filter driver to facilitate performing differential backups and/or de-duplication by monitoring the status of files on the portion of the primary storage environment that includes the computer system," para [0005]).

As to claim 4, the system of claim 1 is discussed above. Further, Commvault teaches such a system wherein the at least one processor is further configured to deploy a scanner at the geographical location (para [0310]).

As to claim 5, the system of claim 1 is discussed above. Further, Cisco teaches such a system wherein the identification of the geographic location comprises an identification of a data center, at least one of a data center name, Internet Protocol (IP) address, name of the cloud provider, or a unique identity (para [0272]).

As to claim 6, the system of claim 1 is discussed above. Further, Cisco teaches such a system wherein the reconstructed state of the workload includes at least two of an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content (para [0131]; "User behavior analytics happens by collecting and analyzing event log data from various sources. The event log data can be roughly divided to these categories. Categories may include a login category, an activity category, a clickstream category, a resource access category, a resource sharing category, an admin activity category, a configuration category, a user management category and an application and API category. A login category may include but is not limited to login, failed login and logout events by a user or an application (program/machine). An activity category may include a full log of all activity performed by a user or an application (program/machine)," para [0262]).

As to claim 7, the system of claim 1 is discussed above. Further, Commvault teaches such a system wherein the at least one processor is further configured to update the reconstructed state of the workload based on at least one change to the block storage volumes ("Any of the above types of backup operations can be at the volume level, file level, or block level. Volume level backup operations generally involve copying of a data volume (e.g., a logical disk or partition) as a whole. In a file-level backup, information management system 100 generally tracks changes to individual files and includes copies of files in the backup copy. For block-level backups, files are broken into constituent blocks, and changes are tracked at the block level. Upon restore, system 100 reassembles the blocks into files in a transparent fashion," para [0150]).

As to claim 8, the system of claim 1 is discussed above. Further, Cisco teaches such a system wherein the insights comprise at least one of a vulnerability associated with the workload or a composition of installed applications associated with the workload ("FIG. 40 shows an example of a report 4002 that can be generated using the information collected in the application index 2912 once applications are identified and risk ratings 2930 are performed for each of them. The report 4002 shows the top 5 most risky users that have installed and are using cloud applications, with the users identified by unique IP address. The report shows a breakdown of the applications used by each user by high, medium and low risk according to the risk rating 2930 for each of those applications," para [0514]).

As to claim 9, the system of claim 1 is discussed above. Further, Commvault teaches such a system wherein to mount the block storage volumes on the scanner, the at least one processor is configured to: create a snapshot of the block storage volumes ("snapshot operations," para [0143]); and mount the snapshot of the block storage volumes on the scanner (para [0143], [0266]).

As to claim 10, the system of claim 9 is discussed above. Further, Commvault teaches such a system wherein the at least one processor is further configured to: encrypt the snapshot of the block storage volumes ("store a secondary copy including a transformed and/or supplemented representation of a primary data object and/or metadata that is different from the original format, e.g., in a compressed, encrypted, deduplicated, or other modified format," para [0086]); and mount the encrypted snapshot of the block storage volumes on the scanner (para [0086], [0143], [0266]).

(PLEASE SEE NEXT SUPPLEMENTAL BOX)

WRITTEN OPINION OF THE
INTERNATIONAL SEARCHING AUTHORITY

International application No.
PCT/IB 22/52670

Supplemental Box

In case the space in any of the preceding boxes is not sufficient.
Continuation of:
Box V.2 - Citations and explanations:

As to claim 11, Cisco teaches a method for inspecting data, comprising: establishing a trusted relationship between a source account in a cloud environment and a scanner account (para [0012]); using the established trust relationship, utilizing at least one cloud provider API to identify workloads in the source account (para [0098]); using the at least one cloud provider API to query a geographical location of at least one of the identified workloads (para [0157]); receiving an identification of the geographic location (para [0157], [0355]); using the cloud provider APIs to access block storage volumes of the at least one workload (para [0550]); determining a file-system of the at least one workload (para [0109]-[0110]); reconstructing from the block storage volumes a state of the workload (para [0131]); and assessing the reconstructed state of the workload to extract insights (para [0125], [0131]). Cisco fails to explicitly teach such a method further comprising: mounting the block storage volumes on a scanner based on the determined file- system; activating a scanner at the geographic location. However, Commvault teaches such a method further comprising: mounting the block storage volumes on a scanner based on the determined file- system (para [0266]); activating a scanner at the geographic location (para [0310]). It would have been obvious to one of ordinary skill in the art to combine the cloud environment scanning system of Cisco with the support for locally mounting storage volumes and deploying scanners at specific geographic locations of Commvault, because such systems and methods allow for mounting data and performing scans at designated physical device locations (Commvault: para [0266], [0310]). Furthermore, both Cisco and Commvault are directed to systems and methods for scanning cloud storage systems to perform security and other operations.

As to claim 12, the method of claim 11 is discussed above. Further, Cisco teaches such a method wherein the geographic location includes an identifier of a physical site (para [0272]).

As to claim 13, the method of claim 11 is discussed above. Further, Commvault teaches such a method wherein mounting includes selecting a driver corresponding to the determined file system (para [0005]).

As to claim 14, the method of claim 11 is discussed above. Further, Commvault teaches such a method wherein the at least one processor is further configured to deploy a scanner at the geographical location (para [0310]).

As to claim 15, the method of claim 11 is discussed above. Further, Cisco teaches such a method wherein the identification of the geographic location comprises at least one of an identification of a data center, at least one of a data center name, Internet Protocol (IP) address, name of the cloud provider, or a unique identity (para [0272]).

As to claim 16, the method of claim 11 is discussed above. Further, Cisco teaches such a method wherein the reconstructed state of the workload includes at least two of an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content (para [0131], [0262]).

As to claim 17, the method of claim 11 is discussed above. Further, Commvault teaches such a method further comprising updating the reconstructed state of the workload based on at least one change to the block storage volumes (para [0150]).

As to claim 18, the method of claim 11 is discussed above. Further, Cisco teaches such a method wherein the insights comprise at least one of a vulnerability associated with the workload or a composition of installed applications associated with the workload (para [0514]).

As to claim 19, the method of claim 11 is discussed above. Further, Commvault teaches such a method wherein mounting the block storage volumes on the scanner comprising: creating a snapshot of the block storage volumes (para [0143]); and mounting the snapshot of the block storage volumes on the scanner (para [0143], [0266].

As to claim 20, Cisco teaches a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising: establishing a trusted relationship between a source account in a cloud environment and a scanner account (para [0012]); using the established trust relationship, utilizing at least one cloud provider API to identify workloads in the source account (para [0098]); using the at least one cloud provider API to query a geographical location of at least one of the identified workloads (para [0157]); receiving an identification of the geographic location (para [0157], [0355]); using the cloud provider APIs to access block storage volumes of the at least one workload (para [0550]); determining a file-system of the at least one workload (para [0109]-[0110]); reconstructing from the block storage volumes a state of the workload (para [0131]); and assessing the reconstructed state of the workload to extract insights (para [0125], [0131]). Cisco fails to explicitly teach such a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations further comprising: mounting the block storage volumes on a scanner based on the determined filesystem; activating a scanner at the geographic location. However, Commvault teaches such a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations further comprising: mounting the block storage volumes on a scanner based on the determined filesystem (para [0266]); activating a scanner at the geographic location (para [0310]). It would have been obvious to one of ordinary skill in the art to combine the cloud environment scanning system of Cisco with the support for locally mounting storage volumes and deploying scanners at specific geographic locations of Commvault, because such systems and methods allow for mounting data and performing scans at designated physical device locations (Commvault: para [0266], [0310]). Furthermore, both Cisco and Commvault are directed to systems and methods for scanning cloud storage systems to perform security and other operations.

Claims 1-20 have industrial applicability as defined by PCT Article 33(4) because the subject matter can be made or used in industry.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**08/11/2022 04:09:55 PM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Wanda Whittington
PATENT CENTER #	60865613	FILING DATE	04/05/2022
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 2

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
15635-0005-00000IDS-08-11-2022.pdf	4	Information Disclosure Statement (IDS) Form (SB08)	598 KB
15635-0004-00304-PCTIB2252670-8-11-2022.pdf	14	Non Patent Literature	1398 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
15635-0005-00000IDS-08-11-	3A216D70A8169D919A8BADE281A0E2426CC11AD67CB5C6675

2022.pdf	11A693E3F5160D84EBBD669FA9582E3C744D14C0AD16EF335 10230457FBC89AB686F40F1C1EC7F1
15635-0004-00304- PCTIB2252670-8-11-2022.pdf	4029DEA591AACBE91DC433CB7B40F369EDD63036F93243064 A14BCA45C4F24BB30CB6901B290E23121A59773B49E37C78E CE4E2A138EDD59BCE4376F621DAAB4

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING or 371(c) DATE	GRP ART UNIT	FIL FEE REC'D	ATTY.DOCKET.NO	TOT CLAIMS	IND CLAIMS
17/657,972	04/05/2022	2431	2240	15635.0005-00000	20	3

CONFIRMATION NO. 2547

CORRECTED FILING RECEIPT

22852
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413



Date Mailed: 08/08/2022

Receipt is acknowledged of this non-provisional utility patent application. The application will be taken up for examination in due course. Applicant will be notified as to the results of the examination. Any correspondence concerning the application must include the following identification information: the U.S. APPLICATION NUMBER, FILING DATE, NAME OF FIRST INVENTOR, and TITLE OF INVENTION. Fees transmitted by check or draft are subject to collection.

Please verify the accuracy of the data presented on this receipt. If an error is noted on this Filing Receipt, please submit a written request for a corrected Filing Receipt, including a properly marked-up ADS showing the changes with strike-through for deletions and underlining for additions. If you received a "Notice to File Missing Parts" or other Notice requiring a response for this application, please submit any request for correction to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections provided that the request is grantable.

Inventor(s)

Avi SHUA, Tel Aviv-Yafo, ISRAEL;

Applicant(s)

Orca Security LTD., Tel Aviv-Yafo, ISRAEL;

Power of Attorney: None

Domestic Priority data as claimed by applicant

This application is a CON of PCT/IB2022/052670 03/23/2022
which claims benefit of 63/180,048 04/26/2021

Foreign Applications for which priority is claimed (You may be eligible to benefit from the **Patent Prosecution Highway** program at the USPTO. Please see http://www.uspto.gov for more information.) - None.
Foreign application information must be provided in an Application Data Sheet in order to constitute a claim to foreign priority. See 37 CFR 1.55 and 1.76.

Permission to Access Application via Priority Document Exchange: Yes

Permission to Access Search Results: Yes

Applicant may provide or rescind an authorization for access using Form PTO/SB/39 or Form PTO/SB/69 as appropriate.

If Required, Foreign Filing License Granted: 04/08/2022

The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is **US 17/657,972**

Projected Publication Date: 10/27/2022

Non-Publication Request: No

Early Publication Request: No
Title

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Preliminary Class

726

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications: No

PROTECTING YOUR INVENTION OUTSIDE THE UNITED STATES

Since the rights granted by a U.S. patent extend only throughout the territory of the United States and have no effect in a foreign country, an inventor who wishes patent protection in another country must apply for a patent in a specific country or in regional patent offices. Applicants may wish to consider the filing of an international application under the Patent Cooperation Treaty (PCT). An international (PCT) application generally has the same effect as a regular national patent application in each PCT-member country. The PCT process **simplifies** the filing of patent applications on the same invention in member countries, but **does not result** in a grant of "an international patent" and does not eliminate the need of applicants to file additional documents and fees in countries where patent protection is desired.

Almost every country has its own patent law, and a person desiring a patent in a particular country must make an application for patent in that country in accordance with its particular laws. Since the laws of many countries differ in various respects from the patent law of the United States, applicants are advised to seek guidance from specific foreign countries to ensure that patent rights are not lost prematurely.

Applicants also are advised that in the case of inventions made in the United States, the Director of the USPTO must issue a license before applicants can apply for a patent in a foreign country. The filing of a U.S. patent application serves as a request for a foreign filing license. The application's filing receipt contains further information and guidance as to the status of applicant's license for foreign filing.

Applicants may wish to consult the USPTO booklet, "General Information Concerning Patents" (specifically, the section entitled "Treaties and Foreign Patents") for more information on timeframes and deadlines for filing foreign patent applications. The guide is available either by contacting the USPTO Contact Center at 800-786-9199, or it can be viewed on the USPTO website at http://www.uspto.gov/web/offices/pac/doc/general/index.html.

For information on preventing theft of your intellectual property (patents, trademarks and copyrights), you may wish to consult the U.S. Government website, http://www.stopfakes.gov. Part of a Department of Commerce initiative, this website includes self-help "toolkits" giving innovators guidance on how to protect intellectual property in specific countries such as China, Korea and Mexico. For questions regarding patent enforcement issues, applicants may call the U.S. Government hotline at 1-866-999-HALT (1-866-999-4258).

LICENSE FOR FOREIGN FILING UNDER

Title 35, United States Code, Section 184

Title 37, Code of Federal Regulations, 5.11 & 5.15

GRANTED

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Bureau of Industry and Security, Department of Commerce (15 CFR parts 730-774); the Office of Foreign AssetsControl, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

NOT GRANTED

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).

SelectUSA

The United States represents the largest, most dynamic marketplace in the world and is an unparalleled location for business investment, innovation, and commercialization of new technologies. The U.S. offers tremendous resources and advantages for those who invest and manufacture goods here. Through SelectUSA, our nation works to promote and facilitate business investment. SelectUSA provides information assistance to the international investor community; serves as an ombudsman for existing and potential investors; advocates on behalf of U.S. cities, states, and regions competing for global investment; and counsels U.S. economic development organizations on investment attraction best practices. To learn more about why the United States is the best country in the world to develop technology, manufacture products, deliver services, and grow your business, visit http://www.SelectUSA.gov or call +1-202-482-6800.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371(C) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
17/657,972	04/05/2022	Avi SHUA	15635.0005-00000

CONFIRMATION NO. 2547

IMPROPER CFR REQUEST

22852
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

OC000000135542285

Date Mailed: 08/08/2022

RESPONSE TO REQUEST FOR CORRECTED FILING RECEIPT

Power of Attorney, Claims, Fees, System Limitations, and Miscellaneous

In response to your request for a corrected Filing Receipt, the Office is unable to comply with your request because:

- The ADS submitted on 07/29/2022 attempts to change the applicant but cannot be entered. Any request to change the applicant once the applicant has been specified must include (1) an application data sheet specifying the new applicant in the Applicant Information section, and (2) a statement under 37 CFR 3.73(c) (USPTO Form PTO/AIA/96 or an equivalent) to show chain of title to the new applicant. The application data sheet must contain markings to show the information that is being changed, with underlining for additions and strike-through or brackets for deletions. See 37 CFR 1.76(c)(2).

In order to make changes to the information of record, an ADS must be properly signed and properly marked up relative to the current information of record.

Proper signature: The ADS must be signed with a handwritten signature or proper S-signature by:

- A patent practitioner, with the practitioner's registration number accompanying the signature (e.g., immediately below or adjacent to the signature), or
- The applicant, if the applicant is an individual other than the inventor(s) and no power of attorney has been appointed, or
- All of the inventors, if no other applicant has been established and no power of attorney has been appointed.

A proper S-signature consists of only letters and/or Arabic numerals, with appropriate spaces and commas, periods, apostrophes, or hyphens for punctuation contained between a first single forward slash mark before, and a second single forward slash mark after, the S-signature.

Proper markings: The ADS must identify the changes being made with underlining for insertions and strike-through or brackets for text removed. No other markings or indications are acceptable. Where an ADS providing corrected or updated information does not contain all of the sections of the ADS, the entire section in which changes are being made must be included in the ADS. Information of record can generally be found on the latest filing receipt.

Questions about the contents of this notice and the requirements it sets forth should be directed to the Office of Data Management, Application Assistance Unit, at **(571) 272-4000** or **(571) 272-4200** or **1-888-786-0101**.

/ttran/

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING or 371(c) DATE	GRP ART UNIT	FIL FEE REC'D	ATTY.DOCKET.NO	TOT CLAIMS	IND CLAIMS
17/657,972	04/05/2022		2240	15635.0005-00000	20	3

CONFIRMATION NO. 2547

FILING RECEIPT

22852
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

OC000000133065336

Date Mailed: 04/11/2022

Receipt is acknowledged of this non-provisional utility patent application. The application will be taken up for examination in due course. Applicant will be notified as to the results of the examination. Any correspondence concerning the application must include the following identification information: the U.S. APPLICATION NUMBER, FILING DATE, NAME OF FIRST INVENTOR, and TITLE OF INVENTION. Fees transmitted by check or draft are subject to collection.

Please verify the accuracy of the data presented on this receipt. If an error is noted on this Filing Receipt, please submit a written request for a corrected Filing Receipt, including a properly marked-up ADS showing the changes with strike-through for deletions and underlining for additions. If you received a "Notice to File Missing Parts" or other Notice requiring a response for this application, please submit any request for correction to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections provided that the request is grantable.

Inventor(s)

Avi SHUA, Tel Aviv-Yafo, ISRAEL;

Applicant(s)

~~Orca Security,~~ Tel Aviv-Yafo, ISRAEL; **Orca Security LTD.**

Power of Attorney: None

Domestic Priority data as claimed by applicant

This application is a CON of PCT/IB2022/052670 03/23/2022
which claims benefit of 63/180,048 04/26/2021

Foreign Applications for which priority is claimed (You may be eligible to benefit from the **Patent Prosecution Highway** program at the USPTO. Please see http://www.uspto.gov for more information.) - None.
Foreign application information must be provided in an Application Data Sheet in order to constitute a claim to foreign priority. See 37 CFR 1.55 and 1.76.

Permission to Access Application via Priority Document Exchange: Yes

Permission to Access Search Results: Yes

Applicant may provide or rescind an authorization for access using Form PTO/SB/39 or Form PTO/SB/69 as appropriate.

If Required, Foreign Filing License Granted: 04/08/2022

The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is **US 17/657,972**

Projected Publication Date: 10/27/2022

Non-Publication Request: No

Early Publication Request: No
Title

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Preliminary Class

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications: No

PROTECTING YOUR INVENTION OUTSIDE THE UNITED STATES

Since the rights granted by a U.S. patent extend only throughout the territory of the United States and have no effect in a foreign country, an inventor who wishes patent protection in another country must apply for a patent in a specific country or in regional patent offices. Applicants may wish to consider the filing of an international application under the Patent Cooperation Treaty (PCT). An international (PCT) application generally has the same effect as a regular national patent application in each PCT-member country. The PCT process **simplifies** the filing of patent applications on the same invention in member countries, but **does not result** in a grant of "an international patent" and does not eliminate the need of applicants to file additional documents and fees in countries where patent protection is desired.

Almost every country has its own patent law, and a person desiring a patent in a particular country must make an application for patent in that country in accordance with its particular laws. Since the laws of many countries differ in various respects from the patent law of the United States, applicants are advised to seek guidance from specific foreign countries to ensure that patent rights are not lost prematurely.

Applicants also are advised that in the case of inventions made in the United States, the Director of the USPTO must issue a license before applicants can apply for a patent in a foreign country. The filing of a U.S. patent application serves as a request for a foreign filing license. The application's filing receipt contains further information and guidance as to the status of applicant's license for foreign filing.

Applicants may wish to consult the USPTO booklet, "General Information Concerning Patents" (specifically, the section entitled "Treaties and Foreign Patents") for more information on timeframes and deadlines for filing foreign patent applications. The guide is available either by contacting the USPTO Contact Center at 800-786-9199, or it can be viewed on the USPTO website at http://www.uspto.gov/web/offices/pac/doc/general/index.html.

For information on preventing theft of your intellectual property (patents, trademarks and copyrights), you may wish to consult the U.S. Government website, http://www.stopfakes.gov. Part of a Department of Commerce initiative, this website includes self-help "toolkits" giving innovators guidance on how to protect intellectual property in specific countries such as China, Korea and Mexico. For questions regarding patent enforcement issues, applicants may call the U.S. Government hotline at 1-866-999-HALT (1-866-999-4258).

LICENSE FOR FOREIGN FILING UNDER

Title 35, United States Code, Section 184

Title 37, Code of Federal Regulations, 5.11 & 5.15

GRANTED

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Bureau of Industry and Security, Department of Commerce (15 CFR parts 730-774); the Office of Foreign AssetsControl, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

NOT GRANTED

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).

SelectUSA

The United States represents the largest, most dynamic marketplace in the world and is an unparalleled location for business investment, innovation, and commercialization of new technologies. The U.S. offers tremendous resources and advantages for those who invest and manufacture goods here. Through SelectUSA, our nation works to promote and facilitate business investment. SelectUSA provides information assistance to the international investor community; serves as an ombudsman for existing and potential investors; advocates on behalf of U.S. cities, states, and regions competing for global investment; and counsels U.S. economic development organizations on investment attraction best practices. To learn more about why the United States is the best country in the world to develop technology, manufacture products, deliver services, and grow your business, visit http://www.SelectUSA.gov or call +1-202-482-6800.

PATENT
Attorney Docket No. 15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Avi SHUA	)	Group Art Unit: 2431
	)	
Application No.: 17/657,972	)	Examiner: SARAH SU
	)	
Filed: April 5, 2022	)	Confirmation No.: 2547
	)	
For: VULNERABILITY MANAGEMENT	)	
	)	
	)	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Commissioner:

SUBMISSION OF CORRECTED APPLICATION DATA SHEET AND

REQUEST FOR UPDATED FILING RECEIPT

Applicant submits herewith a Corrected Application Data Sheet ("Corrected ADS") to update the Assignee's name. "Orca Security" should read -- Orca Security Ltd.--. A corrected Filing Receipt is enclosed noting this change.

If any fees are necessary for the submission of this Corrected ADS, please charge said fees to Deposit Account No. 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: July 29, 2022

By: /Christopher C. Johns/
Christopher C. Johns
Reg. No. 68,664
(202) 408-4000

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	17/657,972
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

The application data sheet is part of the provisional or nonprovisional application for which it is being submitted. The following form contains the bibliographic data arranged in a format specified by the United States Patent and Trademark Office as outlined in 37 CFR 1.76.
This document may be completed electronically and submitted to the Office in electronic format using the Electronic Filing System (EFS) or the document may be printed and included in a paper filed application.

Secrecy Order 37 CFR 5.2:

☐ Portions or all of the application associated with this Application Data Sheet may fall under a Secrecy Order pursuant to 37 CFR 5.2 (Paper filers only. Applications that fall under Secrecy Order may not be filed electronically.)

Inventor Information:

Inventor 1 Remove

Legal Name

Prefix	Given Name	Middle Name	Family Name	Suffix
	Avi		SHUA	

Residence Information (Select One) ○ US Residency ◉ Non US Residency ○ Active US Military Service

City	Tel Aviv-Yafo	Country of Residence	IL

Mailing Address of Inventor:

Address 1	~~c/o Orca Security~~ c/o Orca Security Ltd.		
Address 2	HaMasger St 66		
City	Tel Aviv-Yafo	**State/Province**	
Postal Code	6721714	**Country**	IL

All Inventors Must Be Listed - Additional Inventor Information blocks may be generated within this form by selecting the **Add** button. Add

Correspondence Information:

Enter either Customer Number or complete the Correspondence Information section below.
For further information see 37 CFR 1.33(a).

☐ **An Address is being provided for the correspondence Information of this application.**

Customer Number	22852		
Email Address		Add Email	Remove Email

Application Information:

Title of the Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES		
Attorney Docket Number	15635.0005-00000	**Small Entity Status Claimed**	☐
Application Type	Nonprovisional		
Subject Matter	Utility		
Total Number of Drawing Sheets (if any)	16	**Suggested Figure for Publication (if any)**	4

CORRECTED ADS

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	17/657,972
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

Filing By Reference:

Only complete this section when filing an application by reference under 35 U.S.C. 111(c) and 37 CFR 1.57(a). Do not complete this section if application papers including a specification and any drawings are being filed. Any domestic benefit or foreign priority information must be provided in the appropriate section(s) below (i.e., "Domestic Benefit/National Stage Information" and "Foreign Priority Information").

For the purposes of a filing date under 37 CFR 1.53(b), the description and any drawings of the present application are replaced by this reference to the previously filed application, subject to conditions and requirements of 37 CFR 1.57(a).

Application number of the previously filed application	Filing date (YYYY-MM-DD)	Intellectual Property Authority or Country

Publication Information:

☐ Request Early Publication (Fee required at time of Request 37 CFR 1.219)

☐ **Request Not to Publish.** I hereby request that the attached application not be published under 35 U.S.C. 122(b) and certify that the invention disclosed in the attached application **has not and will not** be the subject of an application filed in another country, or under a multilateral international agreement, that requires publication at eighteen months after filing.

Representative Information:

Representative information should be provided for all practitioners having a power of attorney in the application. Providing this information in the Application Data Sheet does not constitute a power of attorney in the application (see 37 CFR 1.32).
Either enter Customer Number or complete the Representative Name section below. If both sections are completed the customer Number will be used for the Representative Information during processing.

Please Select One:	◉ Customer Number	○ US Patent Practitioner	○ Limited Recognition (37 CFR 11.9)
Customer Number	22852		

Domestic Benefit/National Stage Information:

This section allows for the applicant to either claim benefit under 35 U.S.C. 119(e), 120, 121, 365(c), or 386(c) or indicate National Stage entry from a PCT application. Providing benefit claim information in the Application Data Sheet constitutes the specific reference required by 35 U.S.C. 119(e) or 120, and 37 CFR 1.78.
When referring to the current application, please leave the "Application Number" field blank.

Prior Application Status	Pending		Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)
	Continuation of	PCT/IB2022/052670	2022-03-23

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	17/657,972
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

Prior Application Status	Pending		Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)
PCT/IB2022/052670	Claims benefit of provisional	63180048	2021-04-26
Additional Domestic Benefit/National Stage Data may be generated within this form by selecting the **Add** button.			

Foreign Priority Information:

This section allows for the applicant to claim priority to a foreign application. Providing this information in the application data sheet constitutes the claim for priority as required by 35 U.S.C. 119(b) and 37 CFR 1.55. When priority is claimed to a foreign application that is eligible for retrieval under the priority document exchange program (PDX)[i] the information will be used by the Office to automatically attempt retrieval pursuant to 37 CFR 1.55(i)(1) and (2). Under the PDX program, applicant bears the ultimate responsibility for ensuring that a copy of the foreign application is received by the Office from the participating foreign intellectual property office, or a certified copy of the foreign priority application is filed, within the time period specified in 37 CFR 1.55(g)(1).

			Remove
Application Number	Country[i]	Filing Date (YYYY-MM-DD)	Access Code[i] (if applicable)
Additional Foreign Priority Data may be generated within this form by selecting the **Add** button.			

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications

☐ This application (1) claims priority to or the benefit of an application filed before March 16, 2013 and (2) also contains, or contained at any time, a claim to a claimed invention that has an effective filing date on or after March 16, 2013.
NOTE: By providing this statement under 37 CFR 1.55 or 1.78, this application, with a filing date on or after March 16, 2013, will be examined under the first inventor to file provisions of the AIA.

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	17/657,972
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

Authorization or Opt-Out of Authorization to Permit Access:

When this Application Data Sheet is properly signed and filed with the application, applicant has provided written authority to permit a participating foreign intellectual property (IP) office access to the instant application-as-filed (see paragraph A in subsection 1 below) and the European Patent Office (EPO) access to any search results from the instant application (see paragraph B in subsection 1 below).

Should applicant choose not to provide an authorization identified in subsection 1 below, applicant **must opt-out** of the authorization by checking the corresponding box A or B or both in subsection 2 below.

NOTE: This section of the Application Data Sheet is **ONLY** reviewed and processed with the **INITIAL** filing of an application. After the initial filing of an application, an Application Data Sheet cannot be used to provide or rescind authorization for access by a foreign IP office(s). Instead, Form PTO/SB/39 or PTO/SB/69 must be used as appropriate.

1. Authorization to Permit Access by a Foreign Intellectual Property Office(s)

A. Priority Document Exchange (PDX) - Unless box A in subsection 2 (opt-out of authorization) is checked, the undersigned hereby **grants the USPTO authority** to provide the European Patent Office (EPO), the Japan Patent Office (JPO), the Korean Intellectual Property Office (KIPO), the State Intellectual Property Office of the People's Republic of China (SIPO), the World Intellectual Property Organization (WIPO), and any other foreign intellectual property office participating with the USPTO in a bilateral or multilateral priority document exchange agreement in which a foreign application claiming priority to the instant patent application is filed, access to: (1) the instant patent application-as-filed and its related bibliographic data, (2) any foreign or domestic application to which priority or benefit is claimed by the instant application and its related bibliographic data, and (3) the date of filing of this Authorization. See 37 CFR 1.14(h)(1).

B. Search Results from U.S. Application to EPO - Unless box B in subsection 2 (opt-out of authorization) is checked, the undersigned hereby **grants the USPTO authority** to provide the EPO access to the bibliographic data and search results from the instant patent application when a European patent application claiming priority to the instant patent application is filed. See 37 CFR 1.14(h)(2).

The applicant is reminded that the EPO's Rule 141(1) EPC (European Patent Convention) requires applicants to submit a copy of search results from the instant application without delay in a European patent application that claims priority to the instant application.

2. Opt-Out of Authorizations to Permit Access by a Foreign Intellectual Property Office(s)

☐ A. Applicant **DOES NOT** authorize the USPTO to permit a participating foreign IP office access to the instant application-as-filed. If this box is checked, the USPTO will not be providing a participating foreign IP office with any documents and information identified in subsection 1A above.

☐ B. Applicant **DOES NOT** authorize the USPTO to transmit to the EPO any search results from the instant patent application. If this box is checked, the USPTO will not be providing the EPO with search results from the instant application.

NOTE: Once the application has published or is otherwise publicly available, the USPTO may provide access to the application in accordance with 37 CFR 1.14.

CORRECTED ADS

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	17/657,972
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

Applicant Information:

Providing assignment information in this section does not substitute for compliance with any requirement of part 3 of Title 37 of CFR to have an assignment recorded by the Office.

Applicant 1

If the applicant is the inventor (or the remaining joint inventor or inventors under 37 CFR 1.45), this section should not be completed. The information to be provided in this section is the name and address of the legal representative who is the applicant under 37 CFR 1.43; or the name and address of the assignee, person to whom the inventor is under an obligation to assign the invention, or person who otherwise shows sufficient proprietary interest in the matter who is the applicant under 37 CFR 1.46. If the applicant is an applicant under 37 CFR 1.46 (assignee, person to whom the inventor is obligated to assign, or person who otherwise shows sufficient proprietary interest) together with one or more joint inventors, then the joint inventor or inventors who are also the applicant should be identified in this section.

Clear

◉ Assignee	○ Legal Representative under 35 U.S.C. 117	○ Joint Inventor
○ Person to whom the inventor is obligated to assign.	○ Person who shows sufficient proprietary interest	

If applicant is the legal representative, indicate the authority to file the patent application, the inventor is:

Name of the Deceased or Legally Incapacitated Inventor:

If the Applicant is an Organization check here. ☒

Organization Name	Orca Security LTD.		
Mailing Address Information For Applicant:			
Address 1	HaMasger St 66		
Address 2			
City	Tel Aviv-Yafo	**State/Province**	
Country	IL	Postal Code	6721714
Phone Number		Fax Number	
Email Address			

Additional Applicant Data may be generated within this form by selecting the Add button.

Assignee Information including Non-Applicant Assignee Information:

Providing assignment information in this section does not substitute for compliance with any requirement of part 3 of Title 37 of CFR to have an assignment recorded by the Office.

CORRECTED ADS

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	17/657,972
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

Assignee 1

Complete this section if assignee information, including non-applicant assignee information, is desired to be included on the patent application publication. An assignee-applicant identified in the "Applicant Information" section will appear on the patent application publication as an applicant. For an assignee-applicant, complete this section only if identification as an assignee is also desired on the patent application publication.

If the Assignee or Non-Applicant Assignee is an Organization check here. ☐

Prefix	**Given Name**	Middle Name	**Family Name**	Suffix

Mailing Address Information For Assignee including Non-Applicant Assignee:

Address 1			
Address 2			
City		**State/Province**	
Country		Postal Code	
Phone Number		Fax Number	
Email Address			

Additional Assignee or Non-Applicant Assignee Data may be generated within this form by selecting the Add button.

Signature:

NOTE: This Application Data Sheet must be signed in accordance with 37 CFR 1.33(b). **However, if this Application Data Sheet is submitted with the INITIAL filing of the application and either box A or B is not checked in subsection 2 of the "Authorization or Opt-Out of Authorization to Permit Access" section, then this form must also be signed in accordance with 37 CFR 1.14(c).**

This Application Data Sheet **must** be signed by a patent practitioner if one or more of the applicants is a **juristic entity** (e.g., corporation or association). If the applicant is two or more joint inventors, this form must be signed by a patent practitioner, **all** joint inventors who are the applicant, or one or more joint inventor-applicants who have been given power of attorney (e.g., see USPTO Form PTO/AIA/81) on behalf of **all** joint inventor-applicants.

See 37 CFR 1.4(d) for the manner of making signatures and certifications.

Signature	/Christopher C. Johns/			Date (YYYY-MM-DD)	2022-07-29
First Name	Christopher	Last Name	Johns	Registration Number	68664

Additional Signature may be generated within this form by selecting the Add button.

CORRECTED ADS

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	17/657,972
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

This collection of information is required by 37 CFR 1.76. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 23 minutes to complete, including gathering, preparing, and submitting the completed application data sheet form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1 The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these records.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3 A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent CooperationTreaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**07/29/2022 03:25:53 PM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Wanda Whittington
PATENT CENTER #	60825483	FILING DATE	04/05/2022
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 3

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
15635-0005-Corrected ADS-as-filed.pdf	8	Application Data Sheet	165 KB
Warning: This is not a USPTO supplied ADS fillable form. Data in the form cannot be automatically loaded to other USPTO systems.			
15635-0005SubmissionofCorrectedADS-as-filed.pdf	1	Transmittal Letter	13 KB
15635-0005-00000Filing Receipt-4-11-2022-15635.0005-00000-Marked-up-as-filed.pdf	3	Request for Corrected Filing Receipt	219 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
15635-0005-Corrected ADS-as-filed.pdf	093DF80CD8410487D8B799DE931646776704AB835891EF14E78CFFED3DAC0474174F96300BE003619A2AB5646A188EF72C079F350E7AB8F923886BCADFA9FC97
15635-0005SubmissionofCorrectedADS-as-filed.pdf	FAC449F44513708C82B83825F4653D734EB5F1C64E54DA4D97355FB1DBC2E24124D052FFAF4A3FD89FC386A6E303D905C5E1DAE770261D8916AD6B171035E1FA
15635-0005-00000Filing Receipt-4-11-2022-15635.0005-00000-Marked-up-as-filed.pdf	C11E5D321D2AE68342ED77861B34281C5BF808EEE5D915A2060D21BA594FA8E7A102E6407CBA3AA405E555F507093FC6AF5AE8D934DBDB9B0D03693285FF6948

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
17/657,972	04/05/2022	Avi SHUA	15635.0005-00000	2547

22852 7590 06/21/2022

FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

EXAMINER
SU, SARAH

ART UNIT	PAPER NUMBER
2431	

NOTIFICATION DATE	DELIVERY MODE
06/21/2022	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

regional-desk@finnegan.com

Office Action Summary	Application No. 17/657,972	Applicant(s) SHUA, Avi	
	Examiner SARAH SU	Art Unit 2431	AIA (FITF) Status Yes

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTHS FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☑ Responsive to communication(s) filed on 5 April 2022.

☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.

2a)☐ This action is **FINAL.** 2b) ☑ This action is non-final.

3)☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

4)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims*

5) ☑ Claim(s) 21-40 is/are pending in the application.

5a) Of the above claim(s) _____ is/are withdrawn from consideration.

6) ☐ Claim(s) _____ is/are allowed.

7) ☑ Claim(s) 21-40 is/are rejected.

8) ☑ Claim(s) 23,31-32 and 35 is/are objected to.

9) ☐ Claim(s) _____ are subject to restriction and/or election requirement

* If any claims have been determined allowable, you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see http://www.uspto.gov/patents/init_events/pph/index.jsp or send an inquiry to **PPHfeedback@uspto.gov.**

Application Papers

10)☑ The specification is objected to by the Examiner.

11)☑ The drawing(s) filed on 4/5/22 is/are: a)☐ accepted or b)☑ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

Certified copies:

a)☐ All b)☐ Some** c)☐ None of the:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

** See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☑ Notice of References Cited (PTO-892)

2) ☐ Information Disclosure Statement(s) (PTO/SB/08a and/or PTO/SB/08b) Paper No(s)/Mail Date _____.

3) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date _____.

4) ☐ Other: _____.

DETAILED ACTION

Notice of Pre-AIA or AIA Status

1. The present application, filed on or after March 16, 2013, is being examined under the first inventor to file provisions of the AIA.

2. Preliminary Amendment, received on 5 April 2022, has been entered into record. In this amendment, claims 1-20 and 41-159 have been canceled.

3. Claims 21-40 are presented for examination.

Priority

4. The claim for priority from US Provisional 63/180,048 filed 26 April 2021 and is a continuation of PCT/IB2022/052670 filed on 23 March 2022 is duly noted.

Specification

5. The disclosure is objected to because of the following informalities:

a. it contains an embedded hyperlink and/or other form of browser-executable code. Applicant is required to delete the embedded hyperlink and/or other form of browser-executable code; references to websites should be limited to the top-level domain name without any prefix such as http:// or other browser-executable code. See MPEP § 608.01. See [008], line 4.

b. In [00114], line 3: “a unique identity..” should read –a unique identity.–

Appropriate correction is required.

Claim Objections

6. Claims 23, 31, 32, and 35 are objected to because of the following informalities:

a. In claim 23, line 1: "the remedial measure" should read –the remedial action–;

b. It is believed that claims 31 and 32 should be dependent on claim 29 and have been treated as such for the remainder of this Office Action.

c. In claim 35, line 1: "the remedial measure" should read –the remedial action–;

Appropriate correction is required.

Drawings

7. The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because they include the following reference character(s) not mentioned in the description: 403B, 405B, 405C, 409B, 409C, 411B, 411C (Figure 4); 1305 (Figure 13).

Corrected drawing sheets in compliance with 37 CFR 1.121(d), or amendment to the specification to add the reference character(s) in the description in compliance with 37 CFR 1.121(b) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Claim Rejections - 35 USC § 102

8. In the event the determination of the status of the application as subject to AIA 35 U.S.C. 102 and 103 (or as subject to pre-AIA 35 U.S.C. 102 and 103) is incorrect, any correction of the statutory basis for the rejection will not be considered a new ground of rejection if the prior art relied upon, and the rationale supporting the rejection, would be the same under either status.

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (a)(1) the claimed invention was patented, described in a printed publication, or in public use, on sale, or otherwise available to the public before the effective filing date of the claimed invention.

10. Claim(s) 21, 22, 26, 27, 29, 30, 32-34, and 38-40 is/are rejected under 35 U.S.C. 102(a)(1) as being anticipated by Dye et al. (US 2020/0389483 A1 and Dye hereinafter).

As to claims 21, 33, and 40, Dye discloses a system and method for computer security vulnerability assessment, the system and method having:

at least one processor configured to: using a cloud provider API, access a block storage volume of a workload maintained in a cloud storage environment (0062, lines 6-13)**;**

identify an installed software application in the accessed block storage volume (0024, lines 1-11)**;**

analyze the identified installed software application to determine an associated software version (0024, lines 11-17)**;**

access a data structure of known software vulnerabilities for a plurality of versions of software applications (0004, lines 16-19; 0021, lines 3-11)**;**

perform a lookup of the identified installed software version in the data structure to identify known vulnerabilities (0004, lines 19-26)**;**

perform at least one of: query the cloud provider API to determine network accessibility information related to the workload; identify at least one port on which the vulnerable application is accessible; and use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload (0030, lines 7-16; 0062, lines 2-13).

As to claims 22 and 34, Dye discloses:

wherein the at least one processor is further configured to implement a remedial action in response to the identified one or more vulnerabilities (0046, lines 22-29).

As to claims 26 and 38, Dye discloses:

extract data from at least one of operating system packages, libraries, or program language libraries (0018, lines 1-9)**; and**

identify the installed software application based on the extracted data (0018, lines 1-18).

As to claims 27 and 39, Dye discloses:

identify a version of the installed software application (0024, lines 11-17).

As to claim 29, Dye discloses:

wherein the data structure includes aggregated vulnerability data (0004, lines 16-19; 0021, lines 3-11).

As to claim 30, Dye discloses:

wherein the aggregated vulnerability data includes data from one or more third-party vendors (0025, lines 1-15)**.**

As to claim 32, Dye discloses:

wherein the aggregated vulnerability data includes at least one of an advisory, an exploit, a security announcement, or a known bug (0017, lines 21-26)**.**

Claim Rejections - 35 USC § 103

11. The following is a quotation of 35 U.S.C. 103 which forms the basis for all obviousness rejections set forth in this Office action:

> A patent for a claimed invention may not be obtained, notwithstanding that the claimed invention is not identically disclosed as set forth in section 102, if the differences between the claimed invention and the prior art are such that the claimed invention as a whole would have been obvious before the effective filing date of the claimed invention to a person having ordinary skill in the art to which the claimed invention pertains. Patentability shall not be negated by the manner in which the invention was made.

12. Claim(s) 23, 31, and 35 is/are rejected under 35 U.S.C. 103 as being unpatentable over Dye as applied to claims 22 and 34 above, and further in view of Jones et al. (US 2018/0063182 A1 and Jones hereinafter).

As to claims 23 and 35, Dye fails to specifically disclose:

wherein the remedial measure includes transmitting an alert to a device associated with an administrator.

Nonetheless, this feature is well known in the art and would have been an obvious modification of the teachings disclosed by Dye, as taught by Jones.

Jones discloses a system and method for identifying and mapping sensitive data on an enterprise, the system and method having:

wherein the remedial measure includes transmitting an alert to a device associated with an administrator (0040, lines 7-12)**.**

Given the teaching of Jones, a person having ordinary skill in the art before the effective filing date of the claimed invention would have readily recognized the desirability and advantages of modifying the teachings of Dye with the teachings of Jones by transmitting an alert. Jones recites motivation by disclosing that an alert is transmitted in order to remediate a potential threat as soon as possible to ensure the least amount of damage or compromise to the enterprise (0040, lines 7-12). It is obvious that the teachings of Jones would have improved the teachings of Dye by transmitting an alert in order to minimize damage and protect the enterprise.

As to claim 31, Dye fails to specifically disclose:

wherein the aggregated vulnerability data includes data collected by a scanner.

Nonetheless, this feature is well known in the art and would have been an obvious modification of the teachings disclosed by Dye, as taught by Jones.

Jones discloses:

wherein the aggregated vulnerability data includes data collected by a scanner (0082, lines 2-6; 0090, lines 5-7)**.**

Given the teaching of Jones, a person having ordinary skill in the art before the effective filing date of the claimed invention would have readily recognized the desirability and advantages of modifying the teachings of Dye with the teachings of Jones by using data collected by a scanner. Please refer to the motivation recited above with respect to claims 23 and 35 as to why it is obvious to apply the teachings of Jones to the teachings of Dye.

13. Claim(s) 24, 25, 36, and 37 is/are rejected under 35 U.S.C. 103 as being unpatentable over Dye as applied to claims 21 and 33 above, and further in view of Joffe (US Patent 9,356,942 B1 and Joffe hereinafter).

As to claims 24 and 36, Dye fails to specifically disclose:

examining data sources associated with the workload; and

determining the network accessibility information based on the examined data sources.

Nonetheless, this feature is well known in the art and would have been an obvious modification of the teachings disclosed by Dye, as taught by Joffe.

Joffe discloses a system and method for detecting network compromise, the system and method having:

examining data sources associated with the workload (col. 10, lines 8-13)**; and**

determining the network accessibility information based on the examined data sources (col. 10, lines 8-13)**.**

Given the teaching of Joffe, a person having ordinary skill in the art before the effective filing date of the claimed invention would have readily recognized the desirability and advantages of modifying the teachings of Dye with the teachings of Joffe by determining network accessibility. Joffe recites motivation by disclosing that network accessibility information such as log information is collected in order to perform an analysis to determine if a security event is imminent or has occurred, thus providing security to the system (col. 10, lines 8-13, 53-67). It is obvious that the teachings of Joffe would have improved the teachings of Dye by determining network accessibility information in order to provide security.

As to claims 25 and 37, Dye fails to specifically disclose:

wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.

Nonetheless, this feature is well known in the art and would have been an obvious modification of the teachings disclosed by Dye, as taught by Joffe.

Joffe discloses a system and method for detecting network compromise, the system and method having:

wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log (col. 10, lines 8-13)**.**

Given the teaching of Joffe, a person having ordinary skill in the art before the effective filing date of the claimed invention would have readily recognized the desirability and advantages of modifying the teachings of Dye with the teachings of Joffe by determining network accessibility information including network capture log data. Please refer to the motivation recited above with respect to claims 24 and 36 as to why it is obvious to apply the teachings off Joffe to the teachings of Dye.

14. Claim(s) 28 is/are rejected under 35 U.S.C. 103 as being unpatentable over Dye as applied to claim 21 above, and further in view of Liu et al. (US Patent 9,413,774 B1 and Liu hereinafter).

As to claims 28, Dye fails to specifically disclose:

wherein the identified installed software application includes one or more scripts.

Nonetheless, this feature is well known in the art and would have been an obvious modification of the teachings disclosed by Dye, as taught by Liu.

Liu discloses a system and method for dynamic malware analysis of a URL, the system and method having:

wherein the identified installed software application includes one or more scripts (col. 5, lines 58-62)**.**

Given the teaching of Liu, a person having ordinary skill in the art before the effective filing date of the claimed invention would have readily recognized the desirability and advantages of modifying the teachings of Dye with the teachings of Liu by using scripts. Liu recites motivation by disclosing that scripts are executable content that can be malware or facilitate the delivery of malware and thus may be included as installed software application which must be analyzed (col. 5, lines 58-62). It is obvious that the teachings of Liu would have improved the teachings of Dye by including a script as the installed software application since scripts also can be malware or facilitate the delivery of malware and must also be analyzed.

Prior Art Made of Record

15. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

a. Alabdulhadi (US 2020/0344256 A1) discloses a system and method for online system identification for data reliability enhancement.

b. Czarny et al. (US Patent 9,749,349 B1) discloses a system and method for computer security vulnerability assessment.

c. Lotem et al. (US Patent 8,272,061 B1) discloses a system and method for evaluating a network.

d. Lotem et al. (US Patent 9,569,201 B1) discloses a system and method for translating product banners.

e. Shakarian et al. (US 2022/0141247 A1) discloses a system and method for identifying, reporting, and analyzing threats and vulnerabilities associated with remote network devices.

f. Sundaram et al. (US 2021/0409438 A1) discloses a system and method for automated protection against vulnerability and notification to service provider.

g. Yampolskiy et al. (US Patent 9,294,498 B1) discloses a system and method for online portal for improving cybersecurity risk scores.

Conclusion

16. Any inquiry concerning this communication or earlier communications from the examiner should be directed to SARAH SU whose telephone number is (571)270-3835. The examiner can normally be reached 7:30 AM - 4:00 PM.

Examiner interviews are available via telephone, in-person, and video conferencing using a USPTO supplied web-based collaboration tool. To schedule an interview, applicant is encouraged to use the USPTO Automated Interview Request (AIR) at http://www.uspto.gov/interviewpractice.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lynn Feild can be reached on 571-272-2092. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of published or unpublished applications may be obtained from Patent Center. Unpublished application information in Patent Center is available to registered users. To file and manage patent submissions in Patent Center, visit: https://patentcenter.uspto.gov. Visit https://www.uspto.gov/patents/apply/patent-center for more information about Patent Center and https://www.uspto.gov/patents/docx for information about filing in DOCX format. For additional questions, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/SARAH SU/
Primary Examiner, Art Unit 2431

Notice of References Cited	Application/Control No. 17/657,972	Applicant(s)/Patent Under Reexamination SHUA, Avi	
	Examiner SARAH SU	Art Unit 2431	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-9749349-B1	08-2017	Czarny; Benjamin	H04L63/1433	1/1
*	B	US-9569201-B1	02-2017	Lotem; Amnon	G06F8/70	1/1
*	C	US-9294498-B1	03-2016	Yampolskiy; Aleksandr	H04L63/1425	1/1
*	D	US-20210409438-A1	12-2021	Sundaram; Sudha	H04L63/0876	1/1
*	E	US-20200389483-A1	12-2020	Dye; Frank	H04L63/1425	1/1
*	F	US-8272061-B1	09-2012	Lotem; Amnon	H04L63/1416	709/224
*	G	US-20220141247-A1	05-2022	Shakarian; Paulo	G06F8/75	726/25
*	H	US-20200344256-A1	10-2020	Alabdulhadi; Nawwaf S.	H04L63/1433	1/1
*	I	US-9356942-B1	05-2016	Joffe; Rodney L.	H04L63/1408	1/1
*	J	US-9413774-B1	08-2016	Liu; Jiangxia	G06F21/566	1/1
	K					
	L					
	M					

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	Art Unit
	SARAH SU	2431

CPC - Searched*		
Symbol	Date	Examiner
H04L 63/1433, 1441, 14$2	06/14/2022	SS

CPC Combination Sets - Searched*		
Symbol	Date	Examiner

US Classification - Searched*			
Class	Subclass	Date	Examiner

* See search history printout included with this form or the SEARCH NOTES box below to determine the scope of the search.

Search Notes		
Search Notes	Date	Examiner
PLUS Search	06/14/2022	SS
Inventor Search	06/14/2022	SS
IP.com Search	06/14/2022	SS
PE2E Search (see attached)	06/14/2022	SS
H04L 63/1433, 1441, 14$2 (see search history notes to determine scope)	06/14/2022	SS

Interference Search			
US Class/CPC Symbol	US Subclass/CPC Group	Date	Examiner

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	Art Unit
	SARAH SU	2431

✓	Rejected	-	Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

CLAIMS

☐ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

CLAIM		DATE								
Final	Original	06/14/2022								
	1	-								
	2	-								
	3	-								
	4	-								
	5	-								
	6	-								
	7	-								
	8	-								
	9	-								
	10	-								
	11	-								
	12	-								
	13	-								
	14	-								
	15	-								
	16	-								
	17	-								
	18	-								
	19	-								
	20	-								
	21	✓								
	22	✓								
	23	✓								
	24	✓								
	25	✓								
	26	✓								
	27	✓								
	28	✓								
	29	✓								
	30	✓								
	31	✓								
	32	✓								
	33	✓								
	34	✓								
	35	✓								
	36	✓								
	37	✓								
	38	✓								
	39	✓								
	40	✓								
	41	-								
	42	-								

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	**Art Unit**
	SARAH SU	2431

CLAIM		DATE								
Final	Original	06/14/2022								
	43	-								
	44	-								
	45	-								
	46	-								
	47	-								
	48	-								
	49	-								
	50	-								
	51	-								
	52	-								
	53	-								
	54	-								
	55	-								
	56	-								
	57	-								
	58	-								
	59	-								
	60	-								
	61	-								
	62	-								
	63	-								
	64	-								
	65	-								
	66	-								
	67	-								
	68	-								
	69	-								
	70	-								
	71	-								
	72	-								
	73	-								
	74	-								
	75	-								
	76	-								
	77	-								
	78	-								
	79	-								
	80	-								
	81	-								
	82	-								
	83	-								
	84	-								
	85	-								
	86	-								
	87	-								
	88	-								
	89	-								
	90	-								
	91	-								
	92	-								
	93	-								
	94	-								

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	**Art Unit**
	SARAH SU	2431

CLAIM		DATE								
Final	Original	06/14/2022								
	95	-								
	96	-								
	97	-								
	98	-								
	99	-								
	100	-								
	101	-								
	102	-								
	103	-								
	104	-								
	105	-								
	106	-								
	107	-								
	108	-								
	109	-								
	110	-								
	111	-								
	112	-								
	113	-								
	114	-								
	115	-								
	116	-								
	117	-								
	118	-								
	119	-								
	120	-								
	121	-								
	122	-								
	123	-								
	124	-								
	125	-								
	126	-								
	127	-								
	128	-								
	129	-								
	130	-								
	131	-								
	132	-								
	133	-								
	134	-								
	135	-								
	136	-								
	137	-								
	138	-								
	139	-								
	140	-								
	141	-								
	142	-								
	143	-								
	144	-								
	145	-								
	146	-								

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	**Art Unit**
	SARAH SU	2431

CLAIM		DATE								
Final	Original	06/14/2022								
	147	-								
	148	-								
	149	-								
	150	-								
	151	-								
	152	-								
	153	-								
	154	-								
	155	-								
	156	-								
	157	-								
	158	-								
	159	-								

PE2E SEARCH - Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
L1	9	((("SHUA") near3 ("Avi"))).INV.	(US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT)	ADJ	ON	ON	2022/06/09 01:43 PM
L2	7	L1 AND version	(US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT)	ADJ	ON	ON	2022/06/09 01:43 PM
L3	0	((("Orca") near3 ("Security"))).AS,AANM.	(USPAT)	ADJ	ON	ON	2022/06/09 01:44 PM
L4	182419	(H04L63/1433 OR H04L63/1441 OR H04L63/14$2).cpc.	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/09 01:53 PM
L5	2266	L4 AND (software OR program OR app OR application) WITH version WITH (vulnerab$5 OR risk OR threat)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/09 01:53 PM
L6	170	L4 AND (software OR program OR app OR application) WITH version WITH (vulnerab$5 OR risk OR threat) SAME cloud	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO,	ADJ	ON	ON	2022/06/09 01:54 PM

			RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L7	50	("10539275" "20170314745" "9109162" "9646093" "20050089749" "20150130346" "20150269204" "4591689" "5186249" "7392163" "20010046558" "20050106574" "20110003808" "5774249" "5999117" "6749768" "6749894" "8098282" "8868379" "10444489" "20020137625" "20040001966" "20040069969" "20050106712" "20060212502" "20080069735" "20090015670" "20100069660" "20110151019" "20120147382" "20130261976" "20140322337" "20160215211" "20170003497" "3554073" "3554085" "3554088" "3554091" "3554093" "3554095" "3554098" "3554099" "3554120" "3554135" "3554137" "3554147" "3554153" "3554155" "3554165" "3554167").pn.	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:33 AM
L8	5	L7 AND (vulnerab$5 OR threat OR risk)	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:34 AM
L9	4	L7 AND cloud	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:34 AM
L10	50	("4770410" "5637144" "6145625" "4246184" "4269744" "4286481" "4301570" "4302939" "4436326" "4448396" "4551623" "4555258" "4796926" "4806203" "4836893" "4921707" "5188692" "5202667" "5250146" "5406914"	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:35 AM

		"5494546" "6036446" "4243070" "4257482" "4269611" "4272640" "4273374" "4281805" "4288558" "4299530" "4310334" "4320333" "4324078" "4324921" "4330553" "4334908" "4336606" "4338499" "4338737" "4347679" "4352957" "4355357" "4359761" "4360187" "4362072" "4365112" "4366353" "4367229" "4375510" "4381127").pn.					
L11	3	L10 AND (vulnerab$5 OR threat OR risk)	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:35 AM
L12	0	L10 AND cloud	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:35 AM
L13	584	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/13 09:37 AM
L14	355	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version WITH (software OR app OR application OR software OR program OR code)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/13 09:37 AM
L15	335	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU,	ADJ	ON	ON	2022/06/13 09:37 AM

		WITH (software OR app OR application OR software OR program)	CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L16	239	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version near2 (software OR app OR application OR software OR program)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/13 09:38 AM
L17	376	L4 AND (query$3 OR request$3) WITH cloud WITH source	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 08:46 AM
L18	47	L4 AND (query$3 OR request$3) WITH cloud WITH source WITH access$5 WITH network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN,	ADJ	ON	ON	2022/06/14 09:12 AM

			WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L19	1219	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH access$5 WITH network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 09:16 AM
L22	420	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH access$7 near3 network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 09:17 AM
L23	75	L4 AND (examin$5 OR analyz$3 OR analysis) near3 source WITH access$7 near3 network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 09:27 AM
L24	44	L4 AND (examin$5 OR analyz$3 OR analysis) near3 source WITH access$7 WITH network AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB,	ADJ	ON	ON	2022/06/14 09:28 AM

			HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L26	768	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH log AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 09:54 AM
L27	171	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH log near3 network AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 09:54 AM
L28	1094	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT;	ADJ	ON	ON	2022/06/14 10:39 AM

			IBM_TDB)				
L29	181	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 10:39 AM
L30	69	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH scan$4	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 10:40 AM
L31	4	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH scan$4 WITH (add$3 OR creat$3)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 10:41 AM
L32	38	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV,	ADJ	ON	ON	2022/06/14 10:42 AM

		owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH (add$3 OR creat$3)	MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L33	1815	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 12:05 PM
L34	701	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 12:06 PM
L35	95	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script SAME cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 12:06 PM

PE2E SEARCH - Search History (Interference)

There are no Interference searches to show.

Bibliographic Data

Application No: 17/657,972

Foreign Priority claimed: ○ Yes ◉ No

35 USC 119 (a-d) conditions met: ☐ Yes ☑ No ☐ Met After Allowance

Verified and Acknowledged: /SARAH SU/ Examiner's Signature | Initials

Title: SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

FILING or 371(c) DATE	CLASS	GROUP ART UNIT	ATTORNEY DOCKET NO.
04/05/2022	726	2431	15635.0005-00000
RULE			

APPLICANTS

Orca Security, Tel Aviv-Yafo, ISRAEL

INVENTORS

Avi SHUA, Tel Aviv-Yafo, ISRAEL

CONTINUING DATA

This application is a CON of PCT/IB2022/052670 03/23/2022

PCT/IB2022/052670 has PRO of 63180048 04/26/2021

FOREIGN APPLICATIONS

IF REQUIRED, FOREIGN LICENSE GRANTED**

04/08/2022

STATE OR COUNTRY

ISRAEL

ADDRESS

FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER
LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413
UNITED STATES

FILING FEE RECEIVED

$6,580

Main Concept Text

==========

21. (Original) A cyber security system for a cloud environment, the system comprising: at least one processor configured to: using a cloud provider API, access a block storage volume of a workload maintained in a cloud storage environment; identify an installed software application in the accessed block storage volume; analyze the identified installed software application to determine an associated software version; access a data structure of known software vulnerabilities for a plurality of versions of software applications; perform a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and perform at least one of: query the cloud provider API to determine network accessibility information related to the workload; identify at least one port on which the vulnerable application is accessible; and use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload. 22. (Original) The system of claim 21, wherein the at least one processor is further configured to implement a remedial action in response to the identified one or more vulnerabilities. 23. (Original) The system of claim 22, wherein the remedial measure includes transmitting an alert to a device associated with an administrator. 24. (Original) The system of claim 21, wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises examining data sources associated with the workload; and determining the network accessibility information based on the examined data sources. 25. (Original) The system of claim 21, wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log. 26. (Original) The system of claim 21, wherein to identify the installed software application, the at least one processor is configured to: extract data from at least one of operating system packages, libraries, or program language libraries; and identify the installed software application based on the extracted data. 27.(Original) The system of claim 21, wherein the at least one processor is further configured to identify a version of the installed software application. 28. (Original)The system of claim 21, wherein the identified installed software application includes one or more scripts. 29. (Original) The system of claim 21, wherein the data structure includes aggregated vulnerability data. 30. (Original) The system of claim 29, wherein the aggregated vulnerability data includes data from one or more third-party vendors. 31. (Original)The system of claim 28, wherein the aggregated vulnerability data includes data collected by a scanner. 32. (Original) The system of claim 28, wherein the aggregated vulnerability data includes at least one of an advisory, an exploit, a security announcement, or a known bug. 33. (Original) A method, comprising: using a cloud provider API, accessing a block storage volume of a workload maintained in a cloud storage environment; identifying an installed software application in the accessed block storage volume; analyzing the identified installed software application to determine an associated software version; accessing a data structure of known software vulnerabilities for a plurality of versions of software applications; performing a lookup of the identified installed software version in the data

structure to identify known vulnerabilities; and perform at least one of the following: query the cloud provider API to determine network accessibility information related to the workload; identify at least one port on which the vulnerable application is accessible; or use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload. 34. (Original) The method of claim 33, further comprising implementing a remedial action in response to the identified one or more vulnerabilities. 35. (Original) The method of claim 34, wherein the remedial measure includes transmitting an alert to a device associated with an administrator. 36. (Original) The method of claim 33, wherein querying the cloud provider API to determine the network accessibility information related to the workload further comprises: examining data sources associated with the workload; and determining the network accessibility information based on the examined data sources. 37. (Original) The method of claim 33, wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log. 38. (Original) The method of claim 33, wherein identify the installed software application comprises: extracting data from at least one of OS packages, libraries, or program language libraries; and identifying the installed software application based on the extracted data. 39. (Original) The method of claim 38, wherein the at least one processor is further configured to identify a version of the installed software application. 40. (Original) A non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising: using a cloud provider API, accessing a block storage volume of a workload maintained in a cloud storage environment; identifying an installed software application in the accessed block storage volume; analyzing the identified installed software application to determine an associated software version; accessing a data structure of known software vulnerabilities for a plurality of versions of software applications; performing a lookup of the identified installed software version in the data structure to identify known vulnerabilities; perform at least one of the following: query the cloud provider API to determine network accessibility information related to the workload; identify at least one port on which the vulnerable application is accessible; and use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.

(No Concept Modifiers)
==========

Filters (4)
==========
Authors/Inventors excluded: "SHUA AVI"

Publication Date: To 2021-04-26

Classic Syntax limited to: (software or program) with version with (vulnerab* or threat or risk)

Classic Syntax limited to: cloud with (api or interface or "application programming interface")

Options
=======
De-dup: Simple Family Numbers (PAT only)
Cut-off: None
Sort: Relevance

Content: Patents + NPL (190)
============================
US Patents
US Designs
US Applications
EPO Patents
EPO Applications
China Patents
China Applications
Japan Patents
Japan Applications
Korea Patents
Korea Applications
WIPO Applications
Argentina Patents
Argentina Applications
Brazil Patents
Brazil Applications
Canada Patents
Canada Applications
Chile Patents
Chile Applications
Colombia Applications
Costa Rica Applications
Cuba Patents
Cuba Applications
Dominican Republic Applications
Ecuador Patents
Ecuador Applications
El Salvador Applications
Guatemala Applications
Honduras Applications

Mexico Patents
Mexico Applications
Nicaragua Patents
Panama Applications
Peru Applications
Trinidad & Tobago Patents
Uruguay Applications
Austria Patents
Austria Applications
Belarus Patents
Belgium Patents
Belgium Applications
Bosnia & Herzegovina Patents
Bosnia & Herzegovina Applications
Bulgaria Patents
Bulgaria Applications
Croatia Patents
Croatia Applications
Czech Republic Patents
Czech Republic Applications
Czechoslovakia Patents
Czechoslovakia Applications
Denmark Patents
Denmark Applications
Estonia Patents
Estonia Applications
EUIPO Patents
Finland Patents
Finland Applications
France Patents
France Applications
Germany Patents
Germany Applications
Great Britain Patents
Great Britain Applications
Greece Patents
Greece Applications
Hungary Patents
Hungary Applications
Iceland Patents
Iceland Applications
Ireland Patents
Ireland Applications
Italy Patents
Italy Applications
Latvia Patents
Latvia Applications
Lithuania Patents

Lithuania Applications
Luxembourg Patents
Malta Patents
Monaco Patents
Montenegro Patents
Montenegro Applications
Netherlands Patents
Netherlands Applications
Norway Patents
Norway Applications
Poland Patents
Poland Applications
Portugal Patents
Portugal Applications
Republic of Moldova Patents
Republic of Moldova Applications
Romania Patents
Romania Applications
San Marino Patents
San Marino Applications
Serbia Patents
Serbia Applications
Slovakia Patents
Slovakia Applications
Slovenia Patents
Spain Patents
Spain Applications
Sweden Patents
Sweden Applications
Switzerland Patents
Switzerland Applications
Ukraine Patents
Yugoslavia/Serbia and Montenegro Patents
Yugoslavia/Serbia and Montenegro Applications
Armenia Patents
Australia Patents
Australia Applications
Cyprus Patents
Gulf Cooperation Council Patents
Hong Kong Patents
India Patents
India Applications
Indonesia Patents
Indonesia Applications
Israel Applications
Jordan Patents
Jordan Applications
Kyrgyzstan Patents

Macao Applications
Malaysia Patents
Mongolia Patents
New Zealand Patents
Philippines Patents
Philippines Applications
Saudi Arabia Patents
Saudi Arabia Applications
Singapore Patents
Singapore Applications
Taiwan Patents
Taiwan Applications
Tajikistan Patents
Tajikistan Applications
Thailand Patents
Thailand Applications
Uzbekistan Patents
Vietnam Patents
Algeria Patents
ARIPO Patents
ARIPO Applications
Egypt Patents
Kenya Patents
Malawi Patents
Morocco Patents
Morocco Applications
OAPI Patents
South Africa Patents
Tunisia Applications
Zambia Patents
Zimbabwe Patents
EAPO Patents
EAPO Applications
Georgia Patents
Georgia Applications
Kazakhstan Patents
Kazakhstan Applications
Russia Patents
Russia Applications
Turkey Patents
Turkey Applications
IEEE Xplore Publications: IEEE Periodicals
IEEE Xplore Publications: IEEE Conferences
IEEE Xplore Publications: IEEE Standards
IEEE Xplore Publications: IEEE Early Access
IEEE Xplore Publications: SMPTE Periodicals
IEEE Xplore Publications: SMPTE Conferences
IEEE Xplore Publications: SMPTE Standards

IEEE Xplore Publications: MIT Press eBooks
IEEE Xplore Publications: Wiley-IEEE eBooks
IEEE Xplore Publications: IBM Periodicals
IEEE Xplore Publications: URSI Periodicals
IEEE Xplore Publications: VDE Conferences
IEEE Xplore Publications: Periodicals from China
IP.com Prior Art Database: The IP.com Journal
IP.com Prior Art Database: Internet Society RFC
IP.com Prior Art Database: IBM TDB Archive
IP.com Prior Art Database: Legacy Journals
IP.com Prior Art Database: Software Patent Institute
OnePetro.org: Periodicals at OnePetro.org
OnePetro.org: Conferences at OnePetro.org
Other Literature: IBM Redbooks
Other Literature: PubMed Central
Other Literature: arXiv.org

PLUS Search Results for S/N 17657972, Searched Thu Jun 09 13:47:48 EDT 2022

The Patent Linguistics Utility System (PLUS) is a USPTO automated search system for U.S. Patents from 1971 to the present PLUS is a query-by-example search system which produces a list of patents that are most closely related linguistically to the application searched. This search was prepared by the staff of the Scientific and Technical Information Center, SIRA.

4770410 99
5637144 99
6145625 99
4246184 87
4269744 87
4286481 87
4301570 87
4302939 87
4436326 87
4448396 87
4551623 87
4555258 87
4796926 87
4806203 87
4836893 87
4921707 87
5188692 87
5202667 87
5250146 87
5406914 87
5494546 87
6036446 87
4243070 75
4257482 75
4269611 75
4272640 75
4273374 75
4281805 75
4288558 75
4299530 75
4310334 75
4320333 75
4324078 75
4324921 75
4330553 75
4334908 75
4336606 75
4338499 75
4338737 75
4347679 75
4352957 75
4355357 75
4359761 75
4360187 75
4362072 75
4365112 75
4366353 75
4367229 75
4375510 75
4381127 75

PLUS Search Results for S/N 17657972, Searched Thu Jun 09 13:48:10 EDT 2022

The Patent Linguistics Utility System (PLUS) is a USPTO automated search system for U.S. Patents from 1971 to the present PLUS is a query-by-example search system which produces a list of patents that are most closely related linguistically to the application searched. This search was prepared by the staff of the Scientific and Technical Information Center, SIRA.

10539275 99
20170314745 99
9109162 75
9646093 75
20050089749 75
20150130346 75
20150269204 75
4591689 68
5186249 68
7392163 68
20010046558 68
20050106574 68
20110003808 68
5774249 62
5999117 62
6749768 62
6749894 62
8098282 62
8868379 62
10444489 62
20020137625 62
20040001966 62
20040069969 62
20050106712 62
20060212502 62
20080069735 62
20090015670 62
20100069660 62
20110151019 62
20120147382 62
20130261976 62
20140322337 62
20160215211 62
20170003497 62
3554073 56
3554085 56
3554088 56
3554091 56
3554093 56
3554095 56
3554098 56
3554099 56
3554120 56
3554135 56
3554137 56
3554147 56
3554153 56
3554155 56
3554165 56
3554167 56

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
17/657,972	04/05/2022	Avi SHUA	15635.0005-00000	2547

22852 7590 05/26/2022

FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

EXAMINER
CENTRAL, DOCKET

ART UNIT	PAPER NUMBER
OPAP	

NOTIFICATION DATE	DELIVERY MODE
05/26/2022	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

regional-desk@finnegan.com

Decision Granting Request for Prioritized Examination (Track I)	**Application No.** 17/657,972	**Applicant(s)** SHUA, Avi	
	Examiner james W tunstall	**Art Unit** OMBL	**AIA (FITF) Status** Yes

1. THE REQUEST FILED 05 April 2022 IS **GRANTED** .

The above-identified application has met the requirements for prioritized examination

A. ☑ for an original nonprovisional application (Track I).
B. ☐ for an application undergoing continued examination (RCE).

2. **The above-identified application will undergo prioritized examination**. The application will be accorded special status throughout its entire course of prosecution until one of the following occurs:

A. filing a **petition for extension of time** to extend the time period for filing a reply;

B. filing an **amendment to amend the application to contain more than four independent claims, more than thirty total claims** , or a multiple dependent claim;

C. filing a **request for continued examination** ;

D. filing a notice of appeal;

E. filing a request for suspension of action;

F. mailing of a notice of allowance;

G. mailing of a final Office action;

H. completion of examination as defined in 37 CFR 41.102; or

I. abandonment of the application.

Telephone inquiries with regard to this decision should be directed to james tunstall at (703)756-1733. In his/her absence, calls may be directed to Petition Help Desk at (571) 272-3282.

/VANITHA M ELGART/ Petitions Examiner, OPET	

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING or 371(c) DATE	GRP ART UNIT	FIL FEE REC'D	ATTY.DOCKET.NO	TOT CLAIMS	IND CLAIMS
17/657,972	04/05/2022		2240	15635.0005-00000	20	3

CONFIRMATION NO. 2547

FILING RECEIPT

22852
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413



Date Mailed: 04/11/2022

Receipt is acknowledged of this non-provisional utility patent application. The application will be taken up for examination in due course. Applicant will be notified as to the results of the examination. Any correspondence concerning the application must include the following identification information: the U.S. APPLICATION NUMBER, FILING DATE, NAME OF FIRST INVENTOR, and TITLE OF INVENTION. Fees transmitted by check or draft are subject to collection.

Please verify the accuracy of the data presented on this receipt. If an error is noted on this Filing Receipt, please submit a written request for a corrected Filing Receipt, including a properly marked-up ADS showing the changes with strike-through for deletions and underlining for additions. If you received a "Notice to File Missing Parts" or other Notice requiring a response for this application, please submit any request for correction to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections provided that the request is grantable.

Inventor(s)

Avi SHUA, Tel Aviv-Yafo, ISRAEL;

Applicant(s)

Orca Security, Tel Aviv-Yafo, ISRAEL;

Power of Attorney: None

Domestic Priority data as claimed by applicant

This application is a CON of PCT/IB2022/052670 03/23/2022
which claims benefit of 63/180,048 04/26/2021

Foreign Applications for which priority is claimed (You may be eligible to benefit from the **Patent Prosecution Highway** program at the USPTO. Please see http://www.uspto.gov for more information.) - None.
Foreign application information must be provided in an Application Data Sheet in order to constitute a claim to foreign priority. See 37 CFR 1.55 and 1.76.

Permission to Access Application via Priority Document Exchange: Yes

Permission to Access Search Results: Yes

Applicant may provide or rescind an authorization for access using Form PTO/SB/39 or Form PTO/SB/69 as appropriate.

If Required, Foreign Filing License Granted: 04/08/2022

The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is **US 17/657,972**

Projected Publication Date: 10/27/2022

Non-Publication Request: No

Early Publication Request: No
Title

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Preliminary Class

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications: No

PROTECTING YOUR INVENTION OUTSIDE THE UNITED STATES

Since the rights granted by a U.S. patent extend only throughout the territory of the United States and have no effect in a foreign country, an inventor who wishes patent protection in another country must apply for a patent in a specific country or in regional patent offices. Applicants may wish to consider the filing of an international application under the Patent Cooperation Treaty (PCT). An international (PCT) application generally has the same effect as a regular national patent application in each PCT-member country. The PCT process **simplifies** the filing of patent applications on the same invention in member countries, but **does not result** in a grant of "an international patent" and does not eliminate the need of applicants to file additional documents and fees in countries where patent protection is desired.

Almost every country has its own patent law, and a person desiring a patent in a particular country must make an application for patent in that country in accordance with its particular laws. Since the laws of many countries differ in various respects from the patent law of the United States, applicants are advised to seek guidance from specific foreign countries to ensure that patent rights are not lost prematurely.

Applicants also are advised that in the case of inventions made in the United States, the Director of the USPTO must issue a license before applicants can apply for a patent in a foreign country. The filing of a U.S. patent application serves as a request for a foreign filing license. The application's filing receipt contains further information and guidance as to the status of applicant's license for foreign filing.

Applicants may wish to consult the USPTO booklet, "General Information Concerning Patents" (specifically, the section entitled "Treaties and Foreign Patents") for more information on timeframes and deadlines for filing foreign patent applications. The guide is available either by contacting the USPTO Contact Center at 800-786-9199, or it can be viewed on the USPTO website at http://www.uspto.gov/web/offices/pac/doc/general/index.html.

For information on preventing theft of your intellectual property (patents, trademarks and copyrights), you may wish to consult the U.S. Government website, http://www.stopfakes.gov. Part of a Department of Commerce initiative, this website includes self-help "toolkits" giving innovators guidance on how to protect intellectual property in specific countries such as China, Korea and Mexico. For questions regarding patent enforcement issues, applicants may call the U.S. Government hotline at 1-866-999-HALT (1-866-999-4258).

LICENSE FOR FOREIGN FILING UNDER

Title 35, United States Code, Section 184

Title 37, Code of Federal Regulations, 5.11 & 5.15

GRANTED

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Bureau of Industry and Security, Department of Commerce (15 CFR parts 730-774); the Office of Foreign AssetsControl, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

NOT GRANTED

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).

SelectUSA

The United States represents the largest, most dynamic marketplace in the world and is an unparalleled location for business investment, innovation, and commercialization of new technologies. The U.S. offers tremendous resources and advantages for those who invest and manufacture goods here. Through SelectUSA, our nation works to promote and facilitate business investment. SelectUSA provides information assistance to the international investor community; serves as an ombudsman for existing and potential investors; advocates on behalf of U.S. cities, states, and regions competing for global investment; and counsels U.S. economic development organizations on investment attraction best practices. To learn more about why the United States is the best country in the world to develop technology, manufacture products, deliver services, and grow your business, visit http://www.SelectUSA.gov or call +1-202-482-6800.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371(C) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
17/657,972	04/05/2022	Avi SHUA	15635.0005-00000

CONFIRMATION NO. 2547

IMPROPER CPOA LETTER

22852
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

OC00000133065419

Date Mailed: 04/11/2022

NOTICE REGARDING POWER OF ATTORNEY

This is in response to the power of attorney filed 04/05/2022. The power of attorney in this application is not accepted for the reason(s) listed below:

- The power of attorney has not been accepted because the party who is giving power has not been identified. Power of attorney may only be signed by the applicant for patent (37 CFR 1.42) or the patent owner. A party who is not the applicant must become the applicant in accordance with 37 CFR 1.46(c) and appoint any power of attorney in compliance with 37 CFR 3.71 and 3.73. For a reissue application, reexamination proceeding, or supplemental examination proceeding, a patent owner who was not the applicant under 37 CFR 1.46 must appoint any power of attorney in compliance with 37 CFR 3.71 and 3.73. See 37 CFR 1.32(b)(4).

/mordonez/

Office of Data Management, Application Assistance Unit (571) 272-4000, or (571) 272-4200, or 1-888-786-0101

PATENT APPLICATION FEE DETERMINATION RECORD

Substitute for Form PTO-875

Application or Docket Number: 17/657,972

APPLICATION AS FILED - PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA	SMALL ENTITY RATE($)	SMALL ENTITY FEE($)	OR	OTHER THAN SMALL ENTITY RATE($)	OTHER THAN SMALL ENTITY FEE($)
BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A			N/A	320
SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A			N/A	700
EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A			N/A	800
TOTAL CLAIMS (37 CFR 1.16(i))	20 minus 20 =	*			OR	x 100 =	0.00
INDEPENDENT CLAIMS (37 CFR 1.16(h))	3 minus 3 =	*				x 480 =	0.00
APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $310 ($155 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).						420
MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))							0.00
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL			TOTAL	2240

APPLICATION AS AMENDED - PART II

		(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE($)	SMALL ENTITY ADDITIONAL FEE($)	OR	OTHER THAN SMALL ENTITY RATE($)	OTHER THAN SMALL ENTITY ADDITIONAL FEE($)
AMENDMENT A	Total (37 CFR 1.16(i))	*	Minus	**	=	x =		OR	x =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	x =		OR	x =	
	Application Size Fee (37 CFR 1.16(s))									
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

		(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE($)	SMALL ENTITY ADDITIONAL FEE($)	OR	OTHER THAN SMALL ENTITY RATE($)	OTHER THAN SMALL ENTITY ADDITIONAL FEE($)
AMENDMENT B	Total (37 CFR 1.16(i))	*	Minus	**	=	x =		OR	x =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	x =		OR	x =	
	Application Size Fee (37 CFR 1.16(s))									
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.

** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".

*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".

The "Highest Number Previously Paid For" (Total or Independent) is the highest found in the appropriate box in column 1.

Doc Code: PA..
Document Description: Power of Attorney

PTO/AIA/82B (07-13)
Approved for use through 03/31/2021. OMB 0651-0035
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number

POWER OF ATTORNEY BY APPLICANT

I hereby revoke all previous powers of attorney given in the application identified in either the attached transmittal letter or the boxes below.

Application Number	Filing Date

(Note: The boxes above may be left blank if information is provided on form PTO/AIA/82A.)

☑ I hereby appoint the Patent Practitioner(s) associated with the following Customer Number as my/our attorney(s) or agent(s), and to transact all business in the United States Patent and Trademark Office connected therewith for the application referenced in the attached transmittal letter (form PTO/AIA/82A) or identified above: 22852

OR

☐ I hereby appoint Practitioner(s) named in the attached list (form PTO/AIA/82C) as my/our attorney(s) or agent(s), and to transact all business in the United States Patent and Trademark Office connected therewith for the patent application referenced in the attached transmittal letter (form PTO/AIA/82A) or identified above. (Note: Complete form PTO/AIA/82C.)

Please recognize or change the correspondence address for the application identified in the attached transmittal letter or the boxes above to:

☑ The address associated with the above-mentioned Customer Number

OR

☐ The address associated with Customer Number:

OR

☐ Firm or Individual Name					
Address					
City		State		Zip	
Country					
Telephone		Email			

I am the Applicant (if the Applicant is a juristic entity, list the Applicant name in the box):

Orca Security LTD.

☐ Inventor or Joint Inventor (title not required below)

☐ Legal Representative of a Deceased or Legally incapacitated Inventor (title not required below)

☑ Assignee or Person to Whom the Inventor is Under an Obligation to Assign (provide signer's title if applicant is a juristic entity)

☐ Person Who Otherwise Shows Sufficient Proprietary Interest (e.g., a petition under 37 CFR 1.46(b)(2) was granted in the application or is concurrently being filed with this document) (provide signer's title if applicant is a juristic entity)

SIGNATURE of Applicant for Patent

The undersigned (whose title is supplied below) is authorized to act on behalf of the applicant (e.g., where the applicant is a juristic entity).

Signature		Date (Optional)	
Name	Avi Shua		
Title	CEO		

NOTE: Signature - This form must be signed by the applicant in accordance with 37 CFR 1.33. See 37 CFR 1.4 for signature requirements and certifications. If more than one applicant, use multiple forms.

☑ Total of 1 forms are submitted.

This collection of information is required by 37 CFR 1.131, 1.32, and 1.33. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 3 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

The application data sheet is part of the provisional or nonprovisional application for which it is being submitted. The following form contains the bibliographic data arranged in a format specified by the United States Patent and Trademark Office as outlined in 37 CFR 1.76.
This document may be completed electronically and submitted to the Office in electronic format using the Electronic Filing System (EFS) or the document may be printed and included in a paper filed application.

Secrecy Order 37 CFR 5.2:

☐ Portions or all of the application associated with this Application Data Sheet may fall under a Secrecy Order pursuant to 37 CFR 5.2 (Paper filers only. Applications that fall under Secrecy Order may not be filed electronically.)

Inventor Information:

Inventor 1 — Remove

Legal Name

Prefix	Given Name	Middle Name	Family Name	Suffix
	Avi		SHUA	

Residence Information (Select One) US Residency ● Non US Residency Active US Military Service

City	Tel Aviv-Yafo	Country of Residence i	IL

Mailing Address of Inventor:

Address 1	c/o Orca Security
Address 2	HaMasger St 66
City	Tel Aviv-Yafo
State/Province	
Postal Code	6721714
Country i	IL

All Inventors Must Be Listed - Additional Inventor Information blocks may be generated within this form by selecting the **Add** button. Add

Correspondence Information:

Enter either Customer Number or complete the Correspondence Information section below.
For further information see 37 CFR 1.33(a).

☐ **An Address is being provided for the correspondence Information of this application.**

Customer Number	22852		
Email Address		Add Email	Remove Email

Application Information:

Title of the Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES		
Attorney Docket Number	15635.0005-00000	**Small Entity Status Claimed**	☐
Application Type	Nonprovisional		
Subject Matter	Utility		
Total Number of Drawing Sheets (if any)	16	**Suggested Figure for Publication (if any)**	4

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

Filing By Reference:

Only complete this section when filing an application by reference under 35 U.S.C. 111(c) and 37 CFR 1.57(a). Do not complete this section if application papers including a specification and any drawings are being filed. Any domestic benefit or foreign priority information must be provided in the appropriate section(s) below (i.e., "Domestic Benefit/National Stage Information" and "Foreign Priority Information").

For the purposes of a filing date under 37 CFR 1.53(b), the description and any drawings of the present application are replaced by this reference to the previously filed application, subject to conditions and requirements of 37 CFR 1.57(a).

Application number of the previously filed application	Filing date (YYYY-MM-DD)	Intellectual Property Authority or Country

Publication Information:

☐ Request Early Publication (Fee required at time of Request 37 CFR 1.219)

☐ **Request Not to Publish.** I hereby request that the attached application not be published under 35 U.S.C. 122(b) and certify that the invention disclosed in the attached application **has not and will not** be the subject of an application filed in another country, or under a multilateral international agreement, that requires publication at eighteen months after filing.

Representative Information:

Representative information should be provided for all practitioners having a power of attorney in the application. Providing this information in the Application Data Sheet does not constitute a power of attorney in the application (see 37 CFR 1.32).
Either enter Customer Number or complete the Representative Name section below. If both sections are completed the customer Number will be used for the Representative Information during processing.

Please Select One:	● Customer Number	US Patent Practitioner	◯ Limited Recognition (37 CFR 11.9)
Customer Number	22852		

Domestic Benefit/National Stage Information:

This section allows for the applicant to either claim benefit under 35 U.S.C. 119(e), 120, 121, 365(c), or 386(c) or indicate National Stage entry from a PCT application. Providing benefit claim information in the Application Data Sheet constitutes the specific reference required by 35 U.S.C. 119(e) or 120, and 37 CFR 1.78.
When referring to the current application, please leave the "Application Number" field blank.

Prior Application Status	Pending		Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)
	Continuation of	PCT/IB2022/052670	2022-03-23

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

Prior Application Status	Pending		Remove
Application Number	Continuity Type	Prior Application Number	Filing or 371(c) Date (YYYY-MM-DD)
PCT/IB2022/052670	Claims benefit of provisional	63180048	2021-04-26
Additional Domestic Benefit/National Stage Data may be generated within this form by selecting the **Add** button.			Add

Foreign Priority Information:

This section allows for the applicant to claim priority to a foreign application. Providing this information in the application data sheet constitutes the claim for priority as required by 35 U.S.C. 119(b) and 37 CFR 1.55. When priority is claimed to a foreign application that is eligible for retrieval under the priority document exchange program (PDX)[i] the information will be used by the Office to automatically attempt retrieval pursuant to 37 CFR 1.55(i)(1) and (2). Under the PDX program, applicant bears the ultimate responsibility for ensuring that a copy of the foreign application is received by the Office from the participating foreign intellectual property office, or a certified copy of the foreign priority application is filed, within the time period specified in 37 CFR 1.55(g)(1).

			Remove
Application Number	Country[i]	Filing Date (YYYY-MM-DD)	Access Code[i] (if applicable)
Additional Foreign Priority Data may be generated within this form by selecting the **Add** button.			Add

Statement under 37 CFR 1.55 or 1.78 for AIA (First Inventor to File) Transition Applications

☐ This application (1) claims priority to or the benefit of an application filed before March 16, 2013 and (2) also contains, or contained at any time, a claim to a claimed invention that has an effective filing date on or after March 16, 2013.
NOTE: By providing this statement under 37 CFR 1.55 or 1.78, this application, with a filing date on or after March 16, 2013, will be examined under the first inventor to file provisions of the AIA.

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

Authorization or Opt-Out of Authorization to Permit Access:

When this Application Data Sheet is properly signed and filed with the application, applicant has provided written authority to permit a participating foreign intellectual property (IP) office access to the instant application-as-filed (see paragraph A in subsection 1 below) and the European Patent Office (EPO) access to any search results from the instant application (see paragraph B in subsection 1 below).

Should applicant choose not to provide an authorization identified in subsection 1 below, applicant **must opt-out** of the authorization by checking the corresponding box A or B or both in subsection 2 below.

NOTE: This section of the Application Data Sheet is **ONLY** reviewed and processed with the **INITIAL** filing of an application. After the initial filing of an application, an Application Data Sheet cannot be used to provide or rescind authorization for access by a foreign IP office(s). Instead, Form PTO/SB/39 or PTO/SB/69 must be used as appropriate.

1. Authorization to Permit Access by a Foreign Intellectual Property Office(s)

A. Priority Document Exchange (PDX) - Unless box A in subsection 2 (opt-out of authorization) is checked, the undersigned hereby **grants the USPTO authority** to provide the European Patent Office (EPO), the Japan Patent Office (JPO), the Korean Intellectual Property Office (KIPO), the State Intellectual Property Office of the People's Republic of China (SIPO), the World Intellectual Property Organization (WIPO), and any other foreign intellectual property office participating with the USPTO in a bilateral or multilateral priority document exchange agreement in which a foreign application claiming priority to the instant patent application is filed, access to: (1) the instant patent application-as-filed and its related bibliographic data, (2) any foreign or domestic application to which priority or benefit is claimed by the instant application and its related bibliographic data, and (3) the date of filing of this Authorization. See 37 CFR 1.14(h)(1).

B. Search Results from U.S. Application to EPO - Unless box B in subsection 2 (opt-out of authorization) is checked, the undersigned hereby **grants the USPTO authority** to provide the EPO access to the bibliographic data and search results from the instant patent application when a European patent application claiming priority to the instant patent application is filed. See 37 CFR 1.14(h)(2).

The applicant is reminded that the EPO's Rule 141(1) EPC (European Patent Convention) requires applicants to submit a copy of search results from the instant application without delay in a European patent application that claims priority to the instant application.

2. Opt-Out of Authorizations to Permit Access by a Foreign Intellectual Property Office(s)

☐ A. Applicant **DOES NOT** authorize the USPTO to permit a participating foreign IP office access to the instant application-as-filed. If this box is checked, the USPTO will not be providing a participating foreign IP office with any documents and information identified in subsection 1A above.

☐ B. Applicant **DOES NOT** authorize the USPTO to transmit to the EPO any search results from the instant patent application. If this box is checked, the USPTO will not be providing the EPO with search results from the instant application.

NOTE: Once the application has published or is otherwise publicly available, the USPTO may provide access to the application in accordance with 37 CFR 1.14.

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

Applicant Information:

Providing assignment information in this section does not substitute for compliance with any requirement of part 3 of Title 37 of CFR to have an assignment recorded by the Office.

Applicant 1 — Remove

If the applicant is the inventor (or the remaining joint inventor or inventors under 37 CFR 1.45), this section should not be completed. The information to be provided in this section is the name and address of the legal representative who is the applicant under 37 CFR 1.43; or the name and address of the assignee, person to whom the inventor is under an obligation to assign the invention, or person who otherwise shows sufficient proprietary interest in the matter who is the applicant under 37 CFR 1.46. If the applicant is an applicant under 37 CFR 1.46 (assignee, person to whom the inventor is obligated to assign, or person who otherwise shows sufficient proprietary interest) together with one or more joint inventors, then the joint inventor or inventors who are also the applicant should be identified in this section.

Clear

● Assignee	Legal Representative under 35 U.S.C. 117	Joint Inventor
Person to whom the inventor is obligated to assign.	Person who shows sufficient proprietary interest	

If applicant is the legal representative, indicate the authority to file the patent application, the inventor is:

Name of the Deceased or Legally Incapacitated Inventor:

If the Applicant is an Organization check here. ☒

Organization Name	Orca Security		
Mailing Address Information For Applicant:			
Address 1	HaMasger St 66		
Address 2			
City	Tel Aviv-Yafo	**State/Province**	
Country	IL	Postal Code	6721714
Phone Number		Fax Number	
Email Address			

Additional Applicant Data may be generated within this form by selecting the Add button. Add

Assignee Information including Non-Applicant Assignee Information:

Providing assignment information in this section does not substitute for compliance with any requirement of part 3 of Title 37 of CFR to have an assignment recorded by the Office.

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

Assignee 1

Complete this section if assignee information, including non-applicant assignee information, is desired to be included on the patent application publication. An assignee-applicant identified in the "Applicant Information" section will appear on the patent application publication as an applicant. For an assignee-applicant, complete this section only if identification as an assignee is also desired on the patent application publication.

Remove

If the Assignee or Non-Applicant Assignee is an Organization check here. ☐

Prefix	**Given Name**	Middle Name	**Family Name**	Suffix

Mailing Address Information For Assignee including Non-Applicant Assignee:

Address 1			
Address 2			
City		**State/Province**	
Country i		Postal Code	
Phone Number		Fax Number	
Email Address			

Additional Assignee or Non-Applicant Assignee Data may be generated within this form by selecting the Add button. Add

Signature:

Remove

NOTE: This Application Data Sheet must be signed in accordance with 37 CFR 1.33(b). **However, if this Application Data Sheet is submitted with the INITIAL filing of the application and either box A or B is not checked in subsection 2 of the "Authorization or Opt-Out of Authorization to Permit Access" section, then this form must also be signed in accordance with 37 CFR 1.14(c).**

This Application Data Sheet **must** be signed by a patent practitioner if one or more of the applicants is a **juristic entity** (e.g., corporation or association). If the applicant is two or more joint inventors, this form must be signed by a patent practitioner, **all** joint inventors who are the applicant, or one or more joint inventor-applicants who have been given power of attorney (e.g., see USPTO Form PTO/AIA/81) on behalf of **all** joint inventor-applicants.

See 37 CFR 1.4(d) for the manner of making signatures and certifications.

Signature	/Christopher C. Johns/			Date (YYYY-MM-DD)	2022-04-05
First Name	Christopher	Last Name	Johns	Registration Number	68664

Additional Signature may be generated within this form by selecting the Add button. Add

PTO/AIA/14 (01-22)
Approved for use through 05/31/2024. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	15635.0005-00000
	Application Number	
Title of Invention	SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	

This collection of information is required by 37 CFR 1.76. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 23 minutes to complete, including gathering, preparing, and submitting the completed application data sheet form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1 The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these records.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3 A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent CooperationTreaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC PAYMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	04/05/2022 11:26:09 AM ET	15635.0005-00000

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Wanda Whittington
PATENT CENTER #	60604981	AUTHORIZED BY	Christopher Johns
CUSTOMER #	22852	FILING DATE	-
CORRESPONDENCE ADDRESS	-	FIRST NAMED INVENTOR	Avi SHUA

Payment Information

PAYMENT METHOD	PAYMENT TRANSACTION ID	PAYMENT AUTHORIZED BY
CARD / 1007	E202245B28310402	Wanda Whittington

FEE CODE	DESCRIPTION	ITEM PRICE($)	QUANTITY	ITEM TOTAL($)
1830	PROCESSING FEE, EXCEPT IN PROVISIONAL APPLICATIONS	140.00	1	140.00
1817	REQUEST FOR PRIORITIZED EXAMINATION	4200.00	1	4200.00
1111	UTILITY PATENT APPL. SEARCH FEE	700.00	1	700.00
1011	BASIC FILING FEE - UTILITY (PAPER FILING ALSO REQUIRES NON-ELECTRONIC FILING FEE UNDER 1.16(T))	320.00	1	320.00
1311	PATENT APPL. EXAMINATION FEE	800.00	1	800.00

1081	UTILITY APPL. SIZE FEE > THAN 100 PAGES (FEE X PAGES >= 101 IN 50 PAGE INCREMENTS)	420.00	1	420.00

TOTAL AMOUNT: **$6,580.00**

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Doc Code: PA..
Document Description: Power of Attorney

PTO/AIA/82A (07-13)
Approved for use through 11/30/2014. OMB 0651-0051
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FOR POWER OF ATTORNEY TO ONE OR MORE REGISTERED PRACTITIONERS

NOTE: This form is to be submitted with the Power of Attorney by Applicant form (PTO/AIA/82B) to identify the application to which the Power of Attorney is directed, in accordance with 37 CFR 1.5, unless the application number and filing date are identified in the Power of Attorney by Applicant form. If neither form PTO/AIA/82A nor form PTO/AIA82B identifies the application to which the Power of Attorney is directed, the Power of Attorney will not be recognized in the application.

Application Number	To be assigned
Filing Date	Herewith
First Named Inventor	Avi SHUA
Title	Systems and Methods for Managing Cyber Vulnerabilities
Art Unit	To be assigned
Examiner Name	To be assigned
Attorney Docket Number	15635-0005-00000

SIGNATURE of Applicant or Patent Practitioner

Signature	/Christopher C. Johns/	Date (Optional)	5 April 2022
Name	Christopher C. Johns	Registration Number	68,664
Title (if Applicant is a juristic entity)			
Applicant Name (if Applicant is a juristic entity)			

NOTE: This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4(d) for signature requirements and certifications. If more than one applicant, use multiple forms.

☑ *Total of 1 forms are submitted.

This collection of information is required by 37 CFR 1.131, 1.32, and 1.33. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 3 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Doc Code: TRACK1.REQ
Document Description: TrackOne Request

PTO/AIA/424 (04-14)

CERTIFICATION AND REQUEST FOR PRIORITIZED EXAMINATION UNDER 37 CFR 1.102(e) (Page 1 of 1)

First Named Inventor:	Avi SHUA	Nonprovisional Application Number (if known):	
Title of Invention:	Systems and Methods for Managing Cyber Vulnerabilities		

APPLICANT HEREBY CERTIFIES THE FOLLOWING AND REQUESTS PRIORITIZED EXAMINATION FOR THE ABOVE-IDENTIFIED APPLICATION.

1. The processing fee set forth in 37 CFR 1.17(i)(1) and the prioritized examination fee set forth in 37 CFR 1.17(c) have been filed with the request. The publication fee requirement is met because that fee, set forth in 37 CFR 1.18(d), is currently $0. The basic filing fee, search fee, and examination fee are filed with the request or have been already been paid. I understand that any required excess claims fees or application size fee must be paid for the application.

2. I understand that the application may not contain, or be amended to contain, more than four independent claims, more than thirty total claims, or any multiple dependent claims, and that any request for an extension of time will cause an outstanding Track I request to be dismissed.

3. The applicable box is checked below:

I. ☑ Original Application (Track One) - Prioritized Examination under § 1.102(e)(1)

i. (a) The application is an original nonprovisional utility application filed under 35 U.S.C. 111(a). This certification and request is being filed with the utility application via EFS-Web.

---OR---

(b) The application is an original nonprovisional plant application filed under 35 U.S.C. 111(a). This certification and request is being filed with the plant application in paper.

ii. An executed inventor's oath or declaration under 37 CFR 1.63 or 37 CFR 1.64 for each inventor, **or** the application data sheet meeting the conditions specified in 37 CFR 1.53(f)(3)(i) is filed with the application.

II. ☐ Request for Continued Examination - Prioritized Examination under § 1.102(e)(2)

i. A request for continued examination has been filed with, or prior to, this form.
ii. If the application is a utility application, this certification and request is being filed via EFS-Web.
iii. The application is an original nonprovisional utility application filed under 35 U.S.C. 111(a), or is a national stage entry under 35 U.S.C. 371.
iv. This certification and request is being filed prior to the mailing of a first Office action responsive to the request for continued examination.
v. No prior request for continued examination has been granted prioritized examination status under 37 CFR 1.102(e)(2).

Signature /Christopher C. Johns/	Date 5 April 2022
Name (Print/Typed) Christopher C. Johns	Practitioner Registration Number 68,664

Note: *This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4(d) for signature requirements and certifications. Submit multiple forms if more than one signature is required.**

☐ *Total of 1 forms are submitted.

Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (*i.e.*, GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**04/05/2022 11:26:09 AM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Wanda Whittington
PATENT CENTER #	60604981	FILING DATE	-
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 11

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
15635-0005-00000ADS-as-filed.pdf	8	Application Data Sheet	2174 KB
15635-0005-00000PreliminaryAmendment-as-filed.pdf	9	Preliminary Amendment	36 KB
15635-0005-POAtransmittal-as-filed.pdf	1	Power of Attorney	212 KB
15635GeneralPOA-as-filed.pdf	1	Power of Attorney	131 KB
15635PCTDeclaration-as-	1	Oath or Declaration filed	315 KB

filed.pdf

Warning: The PDF file has been signed with a digital signature and the legal effect of the document will be based on the contents of the file not the digital signature.

159635-0005Track1Form-as-filed.pdf		2	Track One Request	127 KB
15635-00005-00000Specification-as-filed.pdf		123	-	6938 KB
15635-00005-00000Specification-as-filed-SPEC.pdf	(1-98)	98	Specification	6005 KB
15635-00005-00000Specification-as-filed-CLM.pdf	(99-122)	24	Claims	910 KB
15635-00005-00000Specification-as-filed-ABST.pdf	(123-123)	1	Abstract	21 KB
15635-0005-00000Drawings-as-filed.pdf		16	Drawings-only black and white line drawings	448 KB
15635-00005-00000-NewApp-TransmittalLetter-as-filed.pdf		3	Transmittal of New Application	79 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
15635-0005-00000ADS-as-filed.pdf	7CD4E8D245F7FC3DD349BF800F4F0D9A94A4F34E451CF0D4F7A39ACF23AEA6356029018CD2EDB376E6B0047A71D9204B765F9AD0ABAD35266B8F115F475CC0A9

15635-0005-00000PreliminaryAmendment-as-filed.pdf	DA69EFFA5A9C047C754C84C22187D9C5F107C98647E48B5F261E0BF9934675D1D22E132C354EEDDA633BE476751D16A46275A7831453AB6AFFCB39CF98D495EB
15635-0005-POAtransmittal-as-filed.pdf	267FDDDD55C4D5B3C6751C8253E4363FA34851873B25757DE4D3AE399093F05617A1726C1D4D0B60C908053999D85E8772725AA991ADFCC76CD79E8EC2578CD4
15635GeneralPOA-as-filed.pdf	275B1054FB615D4935BCC79B92958FF7D7198FB5F5EB5CF36192D6C775ED64AC462B75DF860B484D949CA3E6CC216407E0584DB4CD3EA5D9F1509BA38E19BEB9
15635PCTDeclaration-as-filed.pdf	34FEBA79F8F10C14AEEDAEBB3836E33182E541EEB8A4180B860DEE98254E7E553EA960300612C06BAFA802DA1265DD3A0697C8575C76E40F85DA9D1AB25133C9
159635-0005Track1Form-as-filed.pdf	FAEBBA68996FD8DC65186D4F687D3F5400DBAD144EBE96CB0A7FE9FF2F8BFF4B8C038EC80A774A7EBB0D3BA290A2B3D72209D5F28F1591AC45D23F44853CF899
15635-00005-00000Specification-as-filed.pdf	DB7986AA65A557956C274F9E0B9913BD11301B2AD30C1289F7933D5373A8AF76AB92D3E5046CA65089B9DCCFC8A6F4BDC33A7C3ED809A61B56333DA16BE592DE
15635-00005-00000Specification-as-filed-SPEC.pdf	DE10E47D0998A57BB254FEBB32F9B32F8163C349CA9D10889914D4E4056D26FA9463B39C68EAB15DE337C3EBA1BCBB312C6D57FAAB1E1273F116D5FDAD5BA863
15635-00005-00000Specification-as-filed-CLM.pdf	7FECF3CD9FC72961B992A46894D435DE1D20D387BB0244FFFEFF620E55FB208BA280EBA007C78F00BAA333D286832C523EAF6F69551550F0B6AECBDB1D7FE668
15635-00005-00000Specification-as-filed-ABST.pdf	F4B0520B834AC8C6F1E9010603DFC4989F0F5863606126F324FCCCF5D5A2C332EABB950112B7A008C86887BE3B4B1D2FFB3A0E7DC7AC8F8131B045A0A8CB671F
15635-0005-00000Drawings-as-filed.pdf	ED1CA5A2F3D7EDF2BE0BE870F328599B75FEA330D29BE080BD777742DFF73119854F01CD327E65442BC888854642E1A35DBABBF9D113A9D6505FD8E4F1ACCBFD

15635-00005-00000-NewApp-TransmittalLetter-as-filed.pdf	A213D727C4F81EDEA982DA3BEBF0E360573BF9198247E4B19C169CF934C640E043B60B792F929FFE0F31466E811CDAADCDB56EE94B2ACC179419071ECB85DD6F

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PATENT
Attorney Docket No. 15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

New U.S. Patent Application
Title: SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES (AS AMENDED)
Being a Continuation of PCT International Application No. PCT/IB2022/052670, filed March 23, 2022
Inventor: Avi SHUA

Commissioner:

We enclose the following papers for filing in the United States Patent and Trademark Office under 35 U.S.C. 111 (A) as a Continuation application of PCT International Application No. PCT/IB2022/052670, filed March 23, 2022.

1. Application having one hundred twenty-two (122) pages, 2 independent claims, and 20 claims total is enclosed.

2. Drawings including twenty-one (21) sheets of drawings (Figures 1, 2A-2D, and 3 -13) are enclosed.

3. An Application Data Sheet (ADS) is enclosed. The ADS includes a claim for priority.

4. A Certification and Request for Prioritized Examination Under 37 CFR 1.102(e) (Track One) (PTO/AIA424) is enclosed. Prioritized examination of the application is hereby requested.

5. A Preliminary Amendment is enclosed including a reference to the PCT application, PCT/IB2022/052670 and listing of claims cancelling claims 1-20 and 41-159.

6. A declaration signed by the inventor is enclosed.

7. The application will be assigned to **ORCA SECURITY.** The assignment document will be submitted separately for recordation in this application.

8. A Power of Attorney by Applicant (PTO/AIA/82B) and Transmittal for Power of Attorney Form (PTO/AIA/82A) are attached, granting the Power to the

attorneys/agents associated with FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER, L.L.P., **Customer No. 22,852**.

9. Please address all correspondence to FINNEGAN, HENDERSON, FARABOW, GARRETT and DUNNER, L.L.P., **Customer Number 22,852.**

10. The filing fee Calculated as follows on the basis of the claims existing in the Prior Application as amended by the Preliminary Amendment filed herewith:

Basic Utility Application Filing Fee (Electronic Filing for Small Entity)					$320	$320.00
Search Fee					$700	$700.00
Examination Fee					$800	$800.00
	Number of Claims		Basic	Extra		
Total Claims	20	-	20		x $50 =	$0.00
Independent Claims	2	-	3		x $240 =	$0.00
☐ Presentation of Multiple Dep. Claim(s)					+ $430 =	$00.00
Additional Fee for Paper Filing New Application = $400						$00.00
Size Fee: EFS-Web Filing Total Application Pages (specification, drawings, preliminary amendment filed with original application – DO NOT INCLUDE sequence listings)	**150** x .75 - 100 ÷ 50 = 1* x $400 = 400 *Rounded up to next whole number					$400.00
Request for Prioritized Examination (Track One)						$4200.00
Request for Prioritized Examination (Track One) Processing Fee =$140						$140.00
TOTAL FEES DUE						$6560.00

11. The fee of $6160.00 is submitted herewith.

12. The Commissioner is hereby authorized to charge any fees which may be required including fees due under 37 C.F.R. § 1.16 and any other fees due under 37 C.F.R. § 1.17 or credit any overpayment during the pendency of this application to Deposit Account No. 06-0916.

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: April 5, 2022

By: /Christopher C. Johns
Christopher C. Johns
Reg. No. 68,664
(202) 408-4000

100
Cloud Infrastructure
106
102
User Device
105
Network
107A
107B
107C
107D
VM
113A
113B
113C
113D
109A
109B
109C
109D
Database
115
Load
Balancer
111A
111B
111C
111D
Storage
101
Scanning
System
103A
103B
103C
103D
Database

FIG. 1

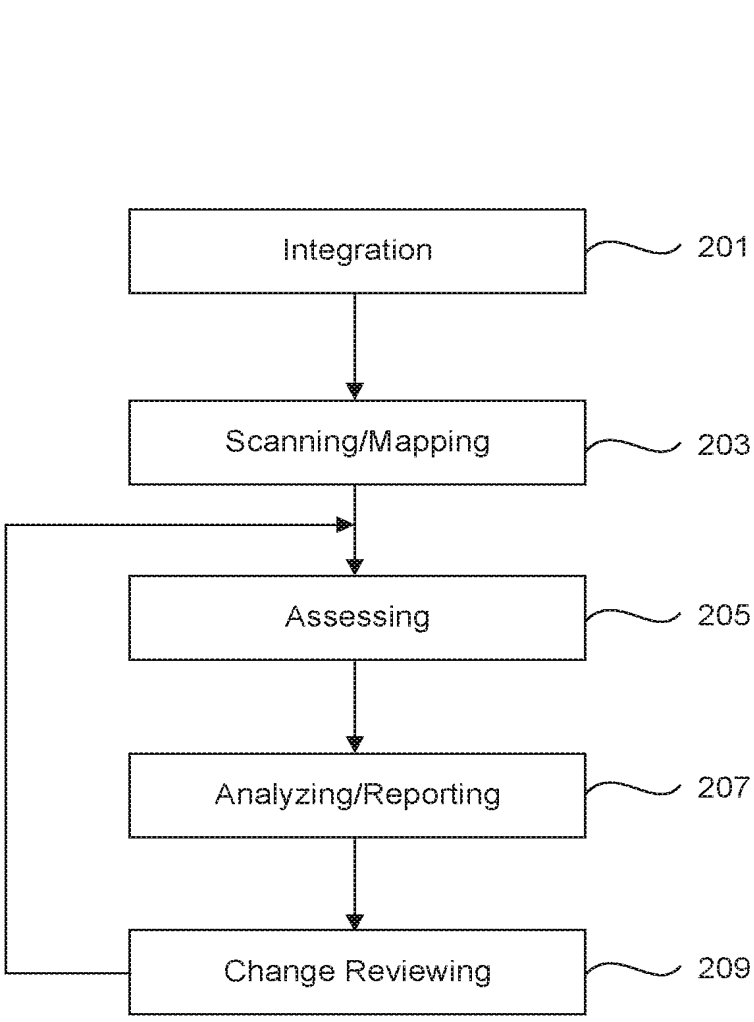

200
Integration
201
Scanning/Mapping
203
Assessing
205
Analyzing/Reporting
207
Change Reviewing
209

FIG. 2A

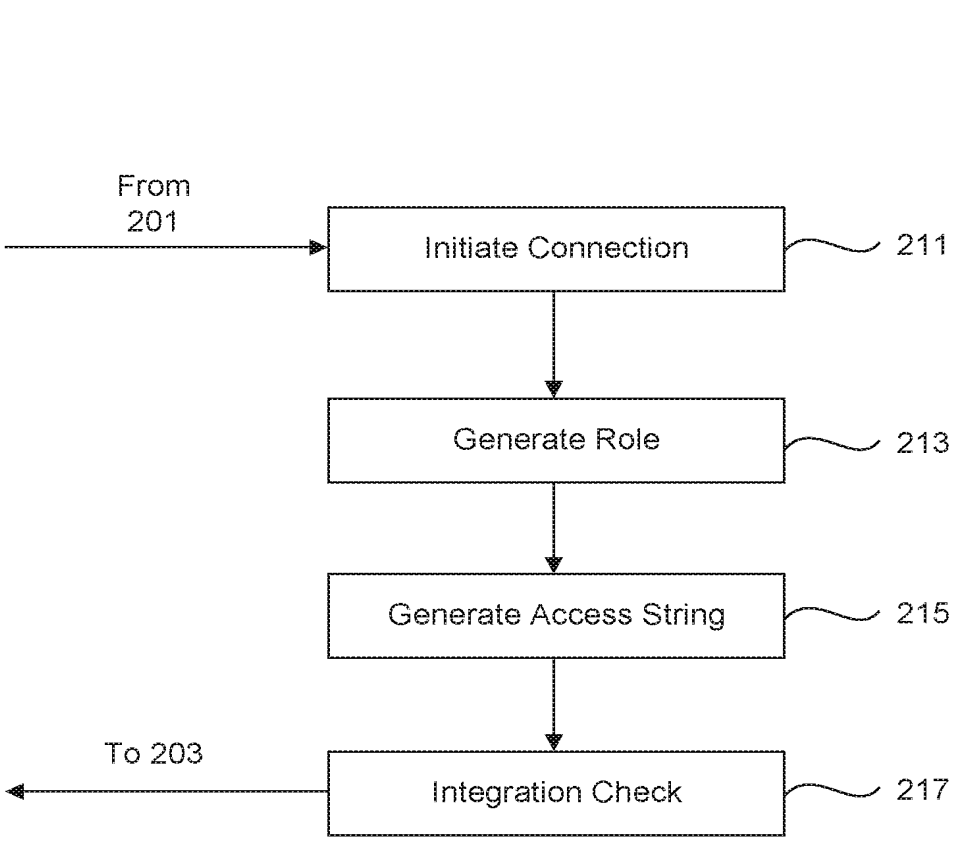

210
From
201
Initiate Connection
211
Generate Role
213
Generate Access String
215
To 203
Integration Check
217

FIG. 2B

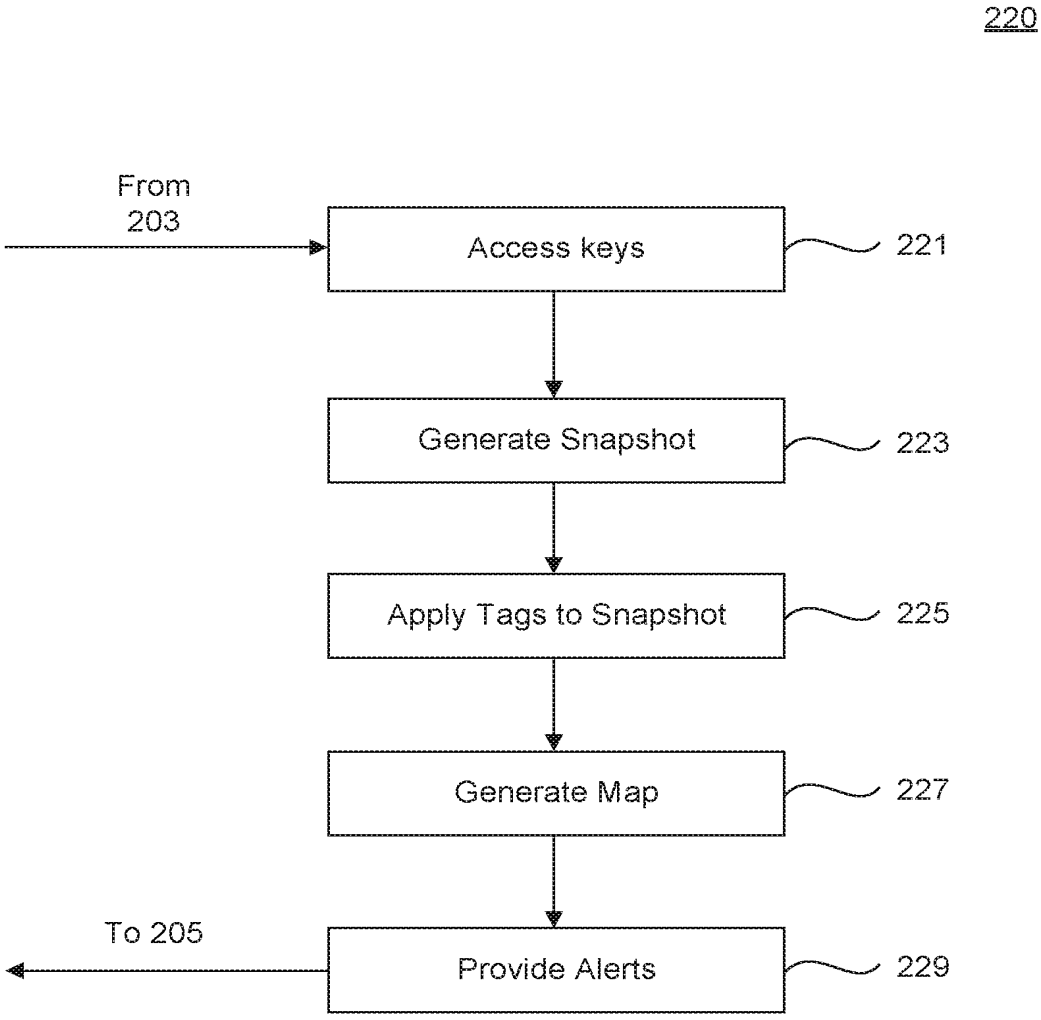

220
From
203
Access keys
221
Generate Snapshot
223
Apply Tags to Snapshot
225
Generate Map
227
To 205
Provide Alerts
229

FIG. 2C

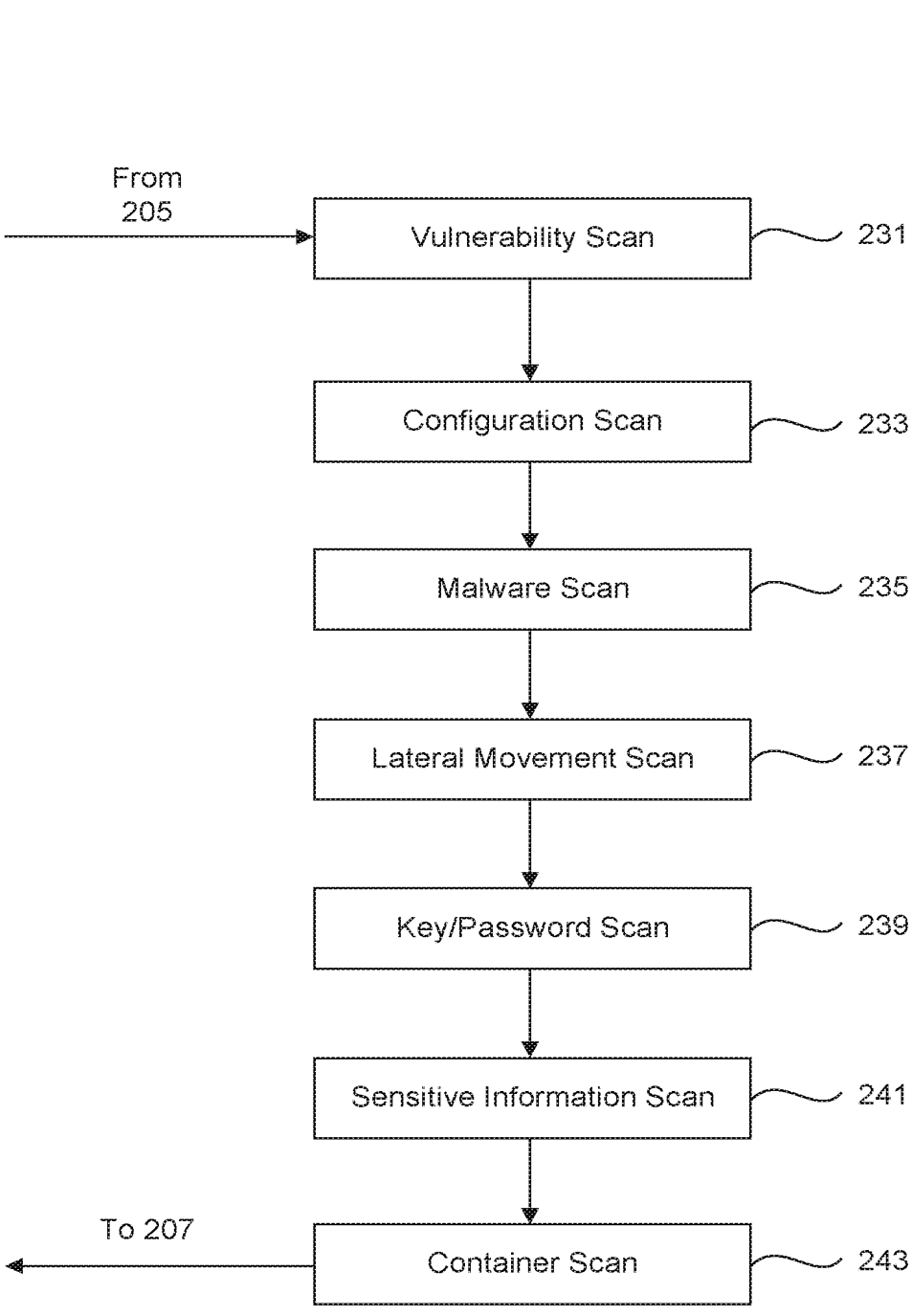

230
From 205
Vulnerability Scan
231
Configuration Scan
233
Malware Scan
235
Lateral Movement Scan
237
Key/Password Scan
239
Sensitive Information Scan
241
To 207
Container Scan
243

FIG. 2D

300
Establish a trusted relationship between a source account in a cloud environment and a scanner account
302
Using the established trust relationship, utilize at least one cloud provider API to identify workloads in the source account
304
Use the at least one cloud provider API to query a geographical location of at least one of the identified workloads
306
Receive an identification of the geographic location
308
Use the cloud provider APIs to access block storage volumes of the at least one workload
310
Determine a file-system of the at least one workload
312
Mount the block storage volumes on a scanner based on the determined file-system
314
Activate a scanner at the geographic location
316
Reconstruct from the block storage volumes a state of the workload
318
Assess the reconstructed state of the workload to extract insights
320

FIG. 3

List Known Vulnerabilities
409A
List Known Vulnerabilities
409B
List Known Vulnerabilities
409C
407
List Known Vulnerabilities
405A
405B
405C
411A
411B
411C
Lookup Known Vulnerabilities of Software Version
Lookup Known Vulnerabilities of Software Version
Identify Software Version
Identify Software Version
Identify Software Version
Lookup Known Vulnerabilities of Software Version
403A
403B
403A
Installed Software Identified
Installed Software Identified
Installed Software Identified
Block Storage Volume
401A
Block Storage Volume
401B
Block Storage Volume
401C
400
Processor

FIG. 4

Cloud Provider API
501
503
505
507
509
Block Storage Volume
Software Type Installed
Software Version Installed
List of Known Vulnerabilities
515
Port Accessibility
511
Network Accessibility
513
Associated Vulnerability
Processing Unit
500

FIG. 5

600
Access at least one cloud provider API to determine a plurality of entities capable of routing traffic in a virtual cloud environment associated with a target account containing the workload
602
Query the at least one cloud provider API to determine at least one networking configuration of the entities
604
Build a graph connecting the plurality of entities based on the networking configuration
606
Access a data structure identifying services publicly accessible via the Internet and capable of serving as an internet proxy
608
Integrate the identified services into the graph
610
Traverse the graph to identify at least one source originating via the Internet and reaching the workload
612
Output a risk notification associated with the workload
614

FIG. 6

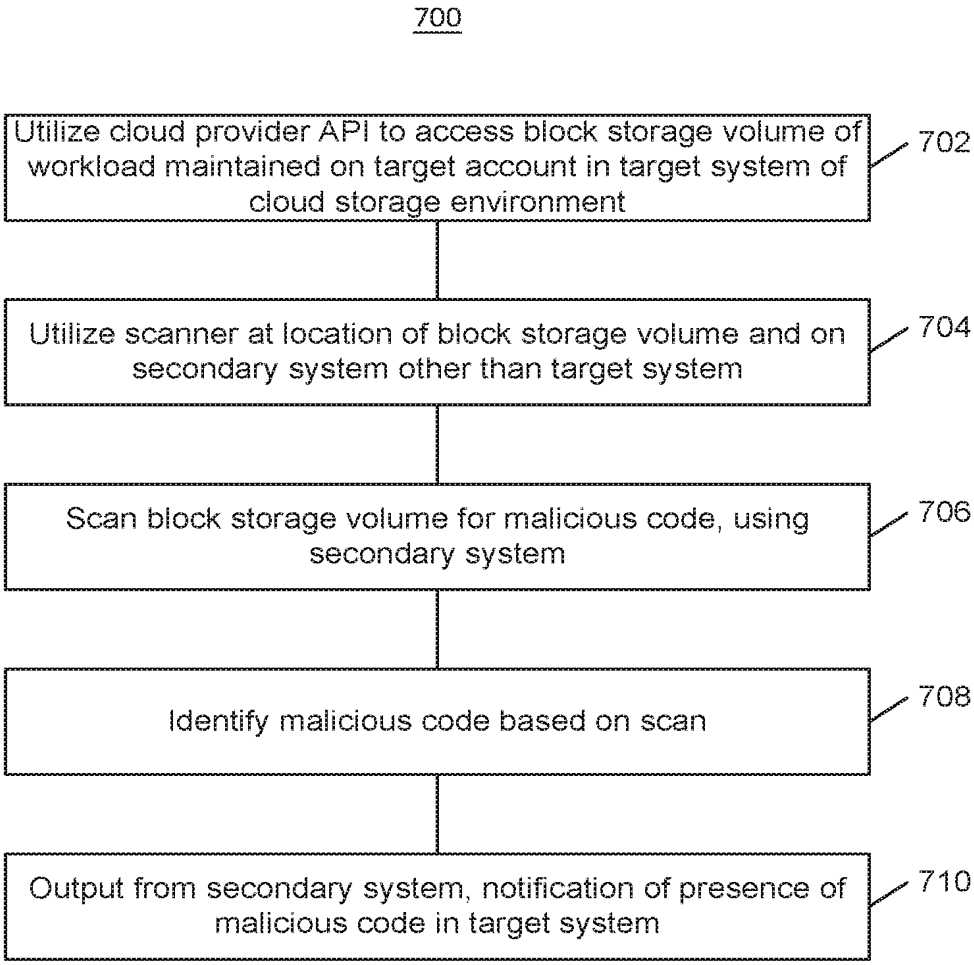

700
Utilize cloud provider API to access block storage volume of workload maintained on target account in target system of cloud storage environment
702
Utilize scanner at location of block storage volume and on secondary system other than target system
704
Scan block storage volume for malicious code, using secondary system
706
Identify malicious code based on scan
708
Output from secondary system, notification of presence of malicious code in target system
710

FIG. 7

807A
Trust
811A
801
803
805
807B
809
Forward Analysis
813
Identify Assets
Identify Risks Associated with Assets
Identify Relationships
Network Connectivity
Receive Identification of Asset
Output Display Signal
Backward Analysis
Mechanism of Network Proxying
811B
807C

FIG. 8

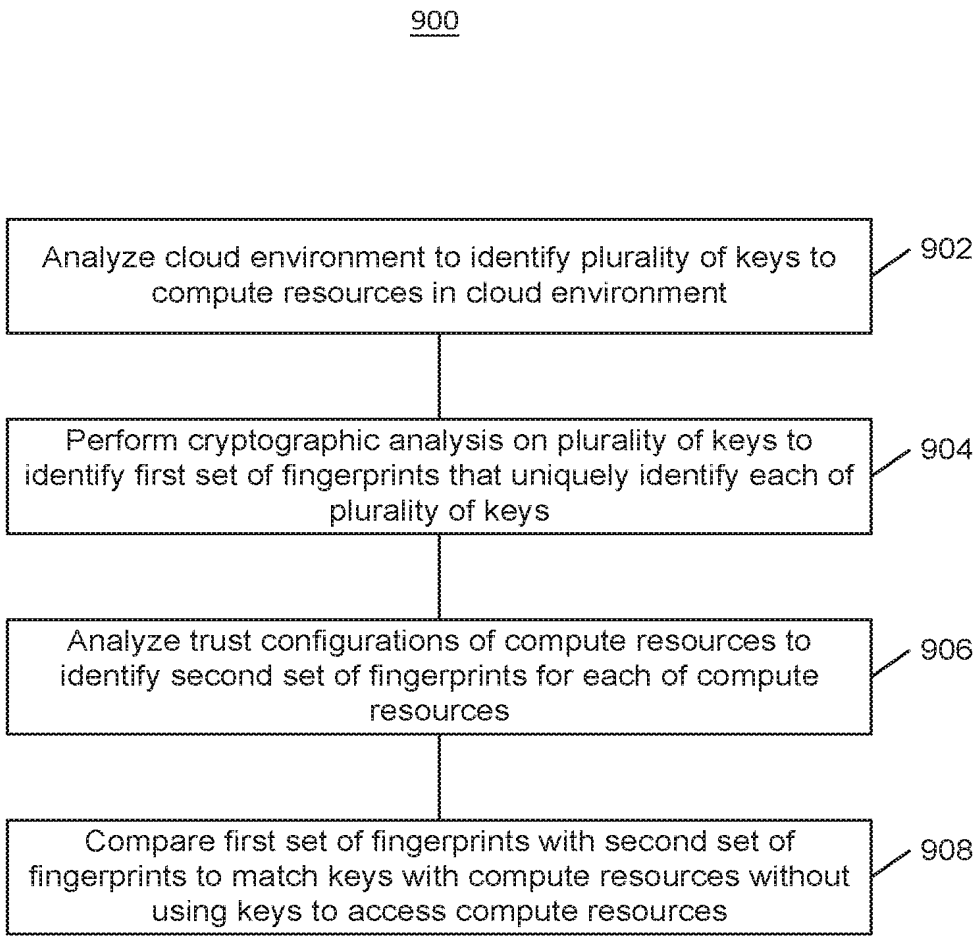

900
Analyze cloud environment to identify plurality of keys to compute resources in cloud environment
902
Perform cryptographic analysis on plurality of keys to identify first set of fingerprints that uniquely identify each of plurality of keys
904
Analyze trust configurations of compute resources to identify second set of fingerprints for each of compute resources
906
Compare first set of fingerprints with second set of fingerprints to match keys with compute resources without using keys to access compute resources
908

FIG. 9

1000
Access a primary account maintained in a cloud environment
1002
Receive information defining a structure of the primary account, wherein the structure includes a plurality of assets
1004
Deploy, inside the primary account, at least one ephemeral scanner configured to periodically scan at least one block storage volume and to output metadata defining the at least one block storage volume
1006
Receive a transmission of the metadata from the at least one ephemeral scanner
1008
Analyze the received metadata to identify a plurality of cybersecurity vulnerabilities
1010
Correlate each of the identified plurality of cybersecurity vulnerabilities with one of the plurality of assets
1012
Generate a report correlating the plurality of cybersecurity vulnerabilities with the plurality of assets
1014

FIG. 10

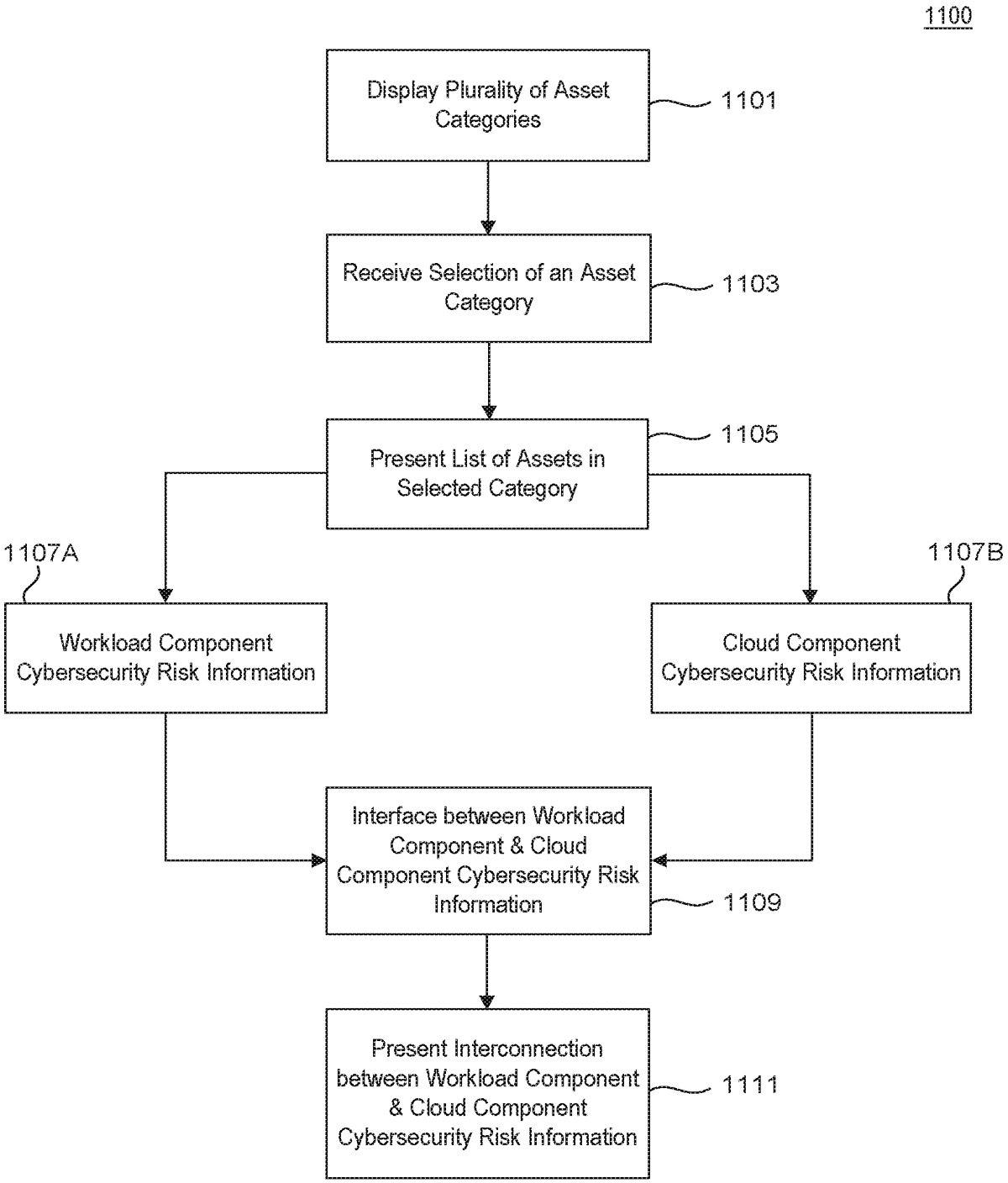

1100
Display Plurality of Asset Categories
1101
Receive Selection of an Asset Category
1103
Present List of Assets in Selected Category
1105
1107A
1107B
Workload Component Cybersecurity Risk Information
Cloud Component Cybersecurity Risk Information
Interface between Workload Component & Cloud Component Cybersecurity Risk Information
1109
Present Interconnection between Workload Component & Cloud Component Cybersecurity Risk Information
1111

FIG. 11

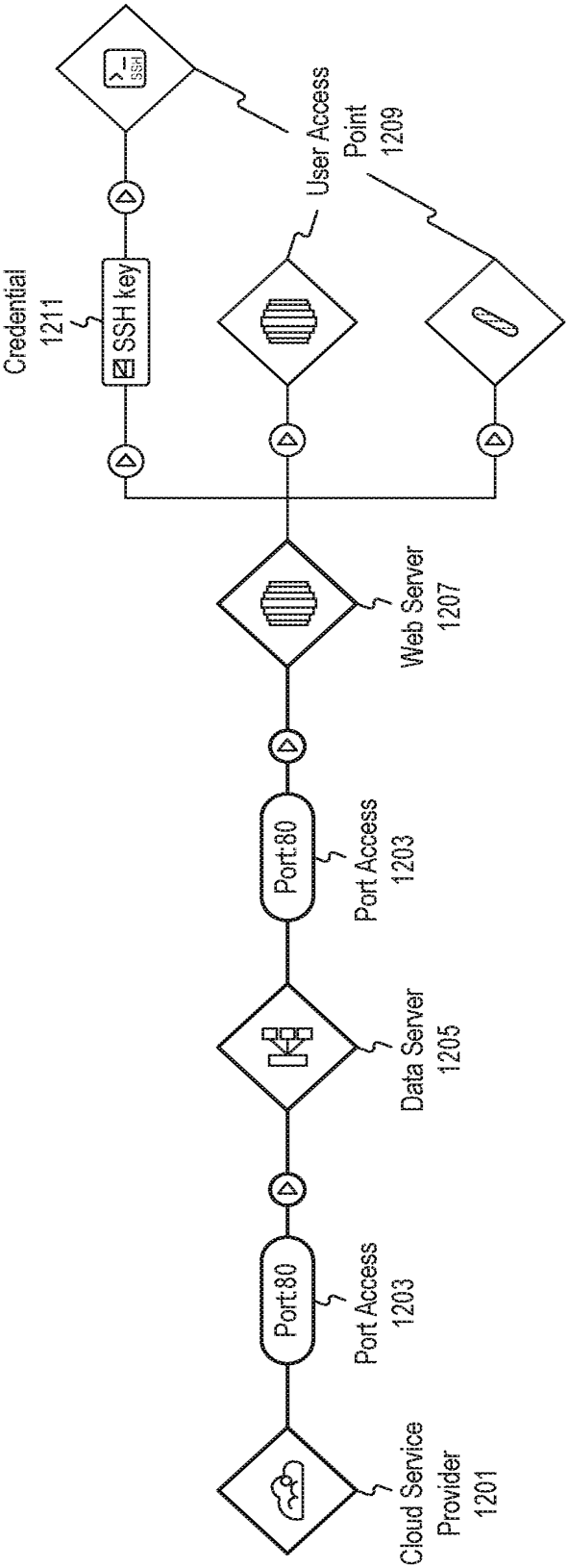

Credential
1211
SSH key
User Access
Point
1209
Web Server
1207
Port:80
Port Access
1203
Data Server
1205
Port:80
Port Access
1203
Cloud Service
Provider
1201

FIG. 12

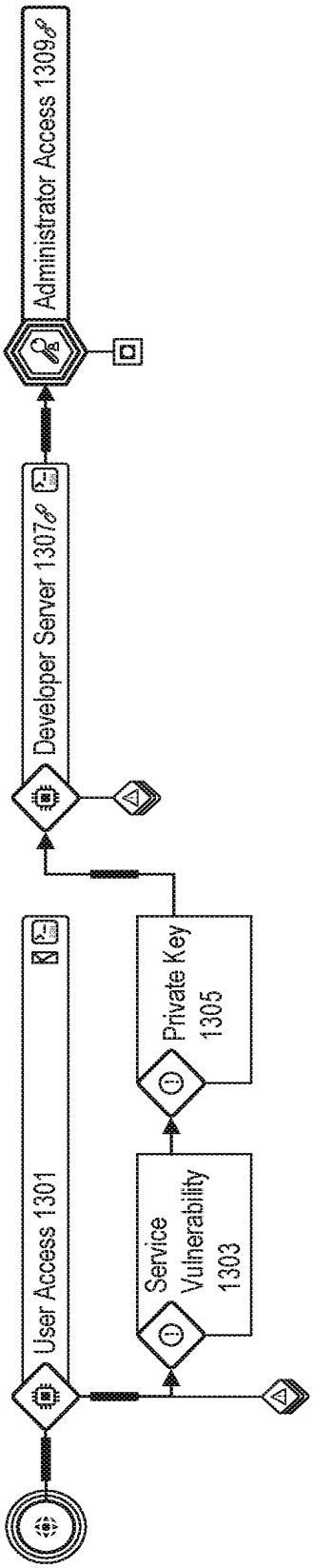

Administrator Access 1309
Developer Server 1307
User Access 1301
Service Vulnerability 1303
Private Key 1305

FIG. 13

SYSTEMS AND METHODS FOR SIDE SCANNING

PRIORITY

[001] This application claims the benefit of U.S. Provisional Application No. 63/180,048 filed on April 26, 2021, the contents of which are hereby incorporated by reference.

TECHNICAL FIELD

[002] Disclosed embodiments include new systems and methods for securing cloud infrastructure to help meet compliance mandates, without disrupting business operations in live environments.

BACKGROUND

[003] Modern organizations typically depend on cloud infrastructure for data storage and processing. Data storage and processing nodes, among other things, power today's electronic infrastructure. In contrast to early network-centric days, an organization may have little in the way of "on-prem" (on-premises) systems and may run its entire business through systems having shared resources. Such an infrastructure may be the entire inventory of cloud assets for an organization, including running and stopped workloads of all types: virtual machines, containers, storage objects, load balancers, IAM (Identity and Access Management) configurations, and more. Organizations are searching for effective ways to scan their entire cloud estate to look for risks stemming from vulnerabilities, misconfigurations, malware, lateral movement risk, weak and leaked passwords, and improperly secured PII.

[004] The infrastructure of these organizations are plainly different than '90s-style physical servers running on bare metal. Existing agents and scanners from the on-prem days have been simply reused in cloud environments without reconfiguration. Before the cloud, it was required to secure physical hosts. That meant spending time installing multiple security agents—one for each server. IP addresses were assigned to physical assets and they seldom changed. Even then, however, agent integration was tedious and coverage rarely reached 100% of assets. In a cloud environment, scaling up and down is a frequent occurrence—possibly thousands of times per hour across multiple clouds—and all within a CI/CD pipeline that builds an organization's infrastructure. There are containers and virtual machines to contend with, and agents carry high operational costs.

[005] This causes problems of inefficiency and overuse of resources, including excess processor cycles on regular scans. In turn, such problems and inefficiencies can lead to a higher monetary cost for hosted resources as some cloud providers charge per processing cycle.

[006] Relying on agents for security visibility in the cloud also has drawbacks. Visibility is limited to only those assets that are already known and accessible to an agent, requiring that such assets are capable of having the agent installed and maintained in the first place. The assets also must have ongoing network connectivity to the backend, something that makes the development cycle inefficient as developers tend not to be inclined to deploy agents on virtual

machines, in containers, and in serverless configurations—let alone dealing with their never-ending maintenance—thus leading to insecure implementations.

[007] An authenticated scan allows for direct host access using remote protocols such as SSH or RDP. The scanner uses a privileged account to log in and determine how secure each host is from an inside vantage point. While authenticated scans can successfully discover potential vulnerabilities, they are limited as they require a privileged account on each scanned host. Furthermore, scans use significant system resources during the test procedures and require opening ports that by themselves pose a security risk.

[008] An unauthenticated scan can only examine publicly visible information and is not able to provide detailed information about assets. It essentially acts as a friendly attacker. An unauthenticated scan can easily miss identifying some assets and vulnerabilities, making it much less effective. For example, a website at https://mydomain.com/email_campaign not linked from a main website will not be scanned unless the scanner is manually configured. This can leave organizations exposed to vulnerabilities in areas where the scanner cannot reach.

[009] While unauthenticated scanners act like an attacker, they often get stuck where a real attacker would not. For example, a CAPTCHA can easily prevent automatic mechanisms (such as scanners) from registering. But such techniques would not have any effect on a human attacker.

[0010] Cloud security posture managers (CSPMs) are scanning tools developed specifically for the cloud. Rather than going inside the machine, a CSPM analyzes the cloud configuration itself for weaknesses. CSPMs are used to discover, assess, and solve cloud misconfigurations but provide shallow coverage where cloud security is concerned because CSPMs have difficulty detecting critical risks such as vulnerabilities, malware, and misconfigurations within the workloads themselves. Organizations choosing to combine agent-based solutions with a CSPM end up getting flooded with separate alerts that lack context which results in alert fatigue on behalf of security analysts.

[0011] Therefore, there is a need for improved security scanning systems that solve the problem with existing systems and are more adapted for use in cloud infrastructures.

[0012] The foregoing background is for illustrative purposes and is not intended as a discussion of the scope of the prior art or an admission of prior art.

SUMMARY

[0013] This summary is provided for the convenience of the reader to present a basic understanding of a few exemplary embodiments of this disclosure, and does not wholly define the breadth of the disclosure. This summary is not an extensive overview of all contemplated embodiments and is intended to neither identify key or critical elements of all embodiments nor to delineate the scope of any or all aspects. Its sole purpose is to present some features of one or more embodiments in a simplified form as a prelude to the more detailed description presented later. For convenience, the term “embodiments” may be used herein to refer to one

or more features or combinations of features described herein. A description associated with an embodiment does not necessarily apply to all embodiments.

[0014] Embodiments of the present disclosure may deliver instant-on, workload level visibility across cloud infrastructure assets without running opcodes on the target scanned workloads, helping organizations to: (1) Detect risks such as vulnerabilities, malware, misconfigurations, lateral movement risk, and unsecured sensitive data; (2) See cloud inventory at every layer: infrastructure, operating system, applications, and data; (3) Discover and see previously missed assets.

[0015] Embodiments of the present disclosure may further use an out-of-band process to reach cloud workloads through the runtime storage layer, combining this with information gathered from cloud provider APIs, including but not limited to K8S APIs.

DRAWINGS

[0016] The drawings and figures contained herein are a part of this specification, illustrate several embodiments consistent with the disclosure and, together with the description, serve to explain principles of the disclosure.

[0017] **Fig. 1** is a schematic block diagram illustrating an exemplary embodiment of a network including computerized systems, consistent with disclosed embodiments.

[0018] **Fig. 2A** is a schematic block diagram illustrating an exemplary embodiment of a process for integration, scanning, assessment, analyzing/reporting, and change reviewing, consistent with disclosed embodiments.

[0019] **Fig. 2B** is a schematic block diagram illustrating an exemplary embodiment of a process for integration, consistent with disclosed embodiments.

[0020] **Fig. 2C** is a schematic block diagram illustrating an exemplary embodiment of a process for scanning/mapping, consistent with disclosed embodiments.

[0021] **Fig. 2D** is a schematic block diagram illustrating an exemplary embodiment of a process for assessment, consistent with disclosed embodiments.

[0022] **Fig. 3** is a block diagram of method for insight extraction, consistent with disclosed embodiments.

[0023] **Fig. 4** depicts a cybersecurity system performing a side scanning function to protect against potential vulnerabilities, consistent with disclosed embodiments.

[0024] **Fig. 5** of the disclosed embodiments describes a method of operating a cybersecurity system performing a side scanning function to protect against potential vulnerabilities, consistent with disclosed embodiments.

[0025] **Fig. 6** is a block diagram of method for risk prioritization, consistent with disclosed embodiments.

[0026] **Fig. 7** is a block diagram illustrating an exemplary process of cyber security scanning for a cloud environment, consistent with the disclosed embodiments.

[0027] **Fig. 8** is a schematic block diagram illustrating an exemplary embodiment of a system for performing visualization of forward and backward facing threats, consistent with the disclosed embodiments.

[0028] **Fig. 9** is a block diagram illustrating an exemplary process of matching keys with compute resources in a cloud environment, consistent with the disclosed embodiments.

[0029] **Fig. 10** is a block diagram of a method for deployment of ephemeral scanners, consistent with disclosed embodiments.

[0030] **Fig. 11** represents a schematic block diagram illustrating an exemplary embodiment of a method for providing a dashboard aggregating risk information, consistent with disclosed embodiments.

[0031] **Fig. 12** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of displaying aggregated cybersecurity risk information, consistent with disclosed embodiments.

[0032] **Fig. 13** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of a flow path for aggregated risk information, consistent with disclosed embodiments.

DETAILED DESCRIPTION

[0033] Exemplary embodiments are described with reference to the accompanying drawings. The figures are not necessarily drawn to scale. While examples and features of disclosed principles are described herein, modifications, adaptations, and other implementations are possible without departing from the spirit and scope of the disclosed embodiments. For example, with this detailed description provides a few examples, these implementations are provided as examples only and are not restrictive of the claim concepts that follow or any of the descriptions herein. Also, the words "comprising," "having," "containing," and "including," and other similar forms are intended to be equivalent in meaning and be open ended in that an item or items following any one of these words is not meant to be an exhaustive listing of such item or items or meant to be limited to only the listed item or items. It should also be noted that as used herein and in the appended claims, the singular forms "a," "an," and "the" include plural references unless the context clearly dictates otherwise.

[0034] In the following description, various working examples are provided for illustrative purposes. However, it is to be understood that the present disclosure may be practiced without one or more of these details.

[0035] It is intended that one or more aspects of any mechanism may be combined with one or more aspect of any other mechanisms, and such combinations are within the scope of this disclosure.

[0036] Various embodiments are described herein with reference to a system, method, device, or computer readable medium. It is intended that the disclosure of one is a disclosure of all. For example, it is to be understood that disclosure of a computer readable medium described

herein also constitutes a disclosure of methods implemented by the computer readable medium, and systems and devices for implementing those methods, via for example, at least one processor. It is to be understood that this form of disclosure is for ease of discussion only, and one or more aspects of one embodiment herein may be combined with one or more aspects of other embodiments herein, within the intended scope of this disclosure.

[0037] Embodiments described herein may refer to a non-transitory computer readable medium containing instructions that when executed by at least one processor, cause the at least one processor to perform a method. Non-transitory computer readable medium may include any medium capable of storing data in any memory in a way that may be read by any computing device with a processor to carry out methods or any other instructions stored in the memory. The non-transitory computer readable medium may be implemented as hardware, firmware, software, or any combination thereof. Moreover, the software may preferably be implemented as an application program tangibly embodied on a program storage unit or computer readable medium consisting of parts, or of certain devices and/or a combination of devices. The application program may be uploaded to, and executed by, a machine having any suitable architecture. Preferably, the machine may be implemented on a computer platform having hardware such as one or more central processing units ("CPUs"), a memory, and input/output interfaces. The computer platform may also include an operating system and microinstruction code. The various processes and functions described in this disclosure may be either part of the microinstruction code or part of the application program, or any combination thereof, which may be executed by a CPU, whether or not such a computer or processor is explicitly shown. In addition, various other peripheral units may be connected to the computer platform such as an additional data storage unit and a printing unit. Furthermore, a non-transitory computer readable medium may be any computer readable medium except for a transitory propagating signal.

[0038] Memory employed herein may include a Random Access Memory (RAM), a Read-Only Memory (ROM), a hard disk, an optical disk, a magnetic medium, a flash memory, other permanent, fixed, volatile or non-volatile memory, or any other mechanism capable of storing instructions. The memory may include one or more separate storage devices collocated or disbursed, capable of storing data structures, instructions, or any other data. The memory may further include a memory portion containing instructions for the processor to execute. The memory may also be used as a working scratch pad for the processors or as a temporary storage.

[0039] Some embodiments may involve at least one processor. A processor may be any physical device or group of devices having electric circuitry that performs a logic operation on input or inputs. For example, the at least one processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU),

digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations. The instructions executed by at least one processor may, for example, be pre-loaded into a memory integrated with or embedded into the controller or may be stored in a separate memory.

[0040] In some embodiments, the at least one processor may include more than one processor. Each processor may have a similar construction, or the processors may be of differing constructions that are electrically connected or disconnected from each other. For example, the processors may be separate circuits or integrated in a single circuit. When more than one processor is used, the processors may be configured to operate independently or collaboratively. The processors may be coupled electrically, magnetically, optically, acoustically, mechanically or by other means that permit them to interact.

[0041] Consistent with the present disclosure, disclosed embodiments may involve a network. A network may constitute any type of physical or wireless computer networking arrangement used to exchange data. For example, a network may be the Internet, a private data network, a virtual private network using a public network, a Wi-Fi network, a LAN or WAN network, and/or other suitable connections that may enable information exchange among various components of the system. In some embodiments, a network may include one or more physical links used to exchange data, such as Ethernet, coaxial cables, twisted pair cables, fiber optics, or any other suitable physical medium for exchanging data. A network may also include a public switched telephone network ("PSTN") and/or a wireless cellular network. A network may be a secured network or unsecured network. In other embodiments, one or more components of the system may communicate directly through a dedicated communication network. Direct communications may use any suitable technologies, including, for example, BLUETOOTH™, BLUETOOTH LE™ (BLE), Wi-Fi, near field communications (NFC), or other suitable communication methods that provide a medium for exchanging data and/or information between separate entities.

[0042] Certain embodiments disclosed herein may also include a computing device for cloud cybersecurity, the computing device may include processing circuitry communicatively connected to a network interface and to a memory, wherein the memory contains instructions to be executed. The computing devices may be devices such as mobile devices, desktops, laptops, tablets, or any other devices capable of processing data. Such computing devices may include a display such as an LED display, augmented reality (AR), virtual reality (VR) display.

[0043] "Software" as used herein refers broadly to any type of instructions, whether referred to as software, firmware, middleware, microcode, hardware description language, or otherwise. Instructions may include code (e.g., in source code format, binary code format, executable code format, or any other suitable format of code). The instructions, when executed by the one or more processors, may cause the processing system to perform the various functions described in further detail herein.

[0044] The one or more processors may be implemented with any combination of general-purpose microprocessors, microcontrollers, digital signal processors (DSPs), field programmable gate array (FPGAs), programmable logic devices (PLDs), controllers, state machines, gated logic, discrete hardware components, dedicated hardware finite state machines, or any other suitable entities that can perform calculations or other manipulations of information.

[0045] Aspects of this disclosure may provide technical solutions to challenges associated with cloud cybersecurity. Disclosed embodiments include methods, systems, devices, and computer-readable media. For ease of discussion, a system is described below with the understanding that the disclosed details may equally apply to methods, devices, and computer-readable media.

[0046] Embodiments of the present disclosure include technology referred to as "SideScanning." In contrast to some existing systems and techniques, these embodiments may provide a distinct advantage because the technology does not necessarily require entering into each workload to inspect data. Rather, some embodiments use an out-of-band process to reach cloud workloads through a runtime storage layer, combining this with metadata gathered from API provided through a cloud service provider's system, thus providing visibility of cloud environments both at a low level and with context, without the requirement for an agent or network scanner.

[0047] **Fig. 1** is a schematic block diagram 100 illustrating an exemplary embodiment of a network including computerized systems, consistent with the disclosed embodiments. Diagram 100 includes user device 102, network 105, and cloud infrastructure 106.

[0048] Cloud infrastructure 106 includes scanning system 101, databases 103A-103D, virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115. While particular numbers and arrangements of devices, systems, and connections, are depicted in exemplary Fig. 1, in some embodiments, each of the devices, systems, or connections may be omitted, duplicated, or modified. For example, in some embodiments, databases 109A-109D may exist as only a single database; in other embodiments, cloud infrastructure 106 may exist as one or more distinct or combined infrastructures (e.g., operated by the same or different cloud services). In some embodiments, scanning system 101 and/or databases 103A-103D may be part of cloud infrastructure 106 (and may be connected to the various other systems and devices in cloud infrastructure 106); in other embodiments, scanning system 101 and/or databases 103A-103D may be separate from cloud infrastructure 106 (e.g., connected to the systems and devices in cloud infrastructure 106 through network 105).

[0049] Scanning system 101, in some embodiments, may include one or more computer systems. Each of the one or more computer systems may include memory storing instructions and at least one CPU configured to execute those instructions to perform operations as

discussed herein. In some embodiments, the instructions cause the CPU to perform scanning operations. In some embodiments, scanning system 101 may perform a scanning operation on one or more workloads (e.g., systems, devices, resources, etc.) in cloud infrastructure 106.

[0050] User device 102, in some embodiments, may include a mechanism operated by a user to control scanning system 101. For example, in some embodiments, user device 102 may be any of a personal computer, a server, a thin client, a tablet, a personal digital assistant, a smartphone, a kiosk, or any other mechanism enabling data input. User device 102 may be operated to instantiate functionality, access data, or otherwise interact with scanning system 101 via network 105, as described herein.

[0051] Databases 103A-103D include data stores for use by scanning system 101. In some embodiments, one or more of databases 103A-103D may be implemented as a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. In some embodiments, one or more of databases 103A-103D may be implemented as flat file stores, data stores, or other non-database storage systems. In some embodiments, databases 103A-103D may be implemented using one or more of ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[0052] Network 105 may be implemented as one or more interconnected data networks. For example, network 105 may include one or more of any type of network (including infrastructure) that provides communications, exchanges information, and/or facilitates the exchange of information, such as the Internet, a Local Area Network, a near field communication (NFC) network, or other suitable connection(s) that enables the sending and receiving of information between the components of system 100. Network 105 may be implemented using wireless connections, wired connections, or both. In some embodiments, one or more components of system 100 can communicate through network 105. In some embodiments, one or more components of system 100 may communicate directly through one or more dedicated communication links. While particular devices and systems are shown as connected to network 105, in some embodiments, more or fewer devices and systems may be connected to network 105.

[0053] Cloud infrastructure 106 may be implemented as a set of devices and systems offered by a single cloud service provider. For example, cloud infrastructure 106 may comprise devices and systems that are part of Amazon Web Services, Microsoft Azure, Google Cloud Platform, IBM Cloud, Alibaba Cloud, or any other cloud platform provider. In some embodiments, one or more of the devices and systems in cloud infrastructure may require authentication or other identity validation for access. For example, to access virtual machine 107A, a user may be required to enter a password or provide a key. Systems (e.g., scanning system 101 or user

device 102) may administer or interact with cloud infrastructure 106 using a cloud service provider's system (not pictured).

[0054] Virtual machines 107A-107D may include one or more devices and systems that implement a virtualized/emulated version of a computer. A virtual machine may be implemented as an emulated version of a computer - including an operating system, memory, storage, graphics processing - such that it can be indistinguishable from a standard (non-virtual) machine to a running program. A computer system, referred to as a "host," may operate virtual machines 107A-107D, referred to as "guests," by dividing the resources of the host between the virtual machines such that each virtual machine is isolated from one another. This means that in some embodiments, one virtual machine, and the operating system(s) and application(s) running thereon, is only able to access the resources that are allocated to that virtual machine and cannot access resources allocated to other virtual machines. For example, if a host has 32 gigabytes of random access memory (RAM), and is hosting three virtual machines, the host may segment 8 gigabytes of RAM to each virtual machine such that each virtual machine may only access data in that 8 gigabytes of RAM and not any of the other 24 gigabytes. Examples of commercial virtual machine software and services include VMWare Workstation, VMWare Server, VMWare ESXi, VirtualBox, Parallels Desktop, Parallels RAS, Amazon Machine Image, Amazon ECS, Kubernetes, Microsoft Hyper-V, and Xen.

[0055] Databases 109A-109D may include data stores for use by devices and systems in cloud infrastructure 106. In some embodiments, one or more of databases 109A-109D may be implemented as a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. In some embodiments, one or more of databases 109A-109D may be implemented as flat file stores, data stores, or other non-database storage systems. In some embodiments, databases 109A-109D may be implemented using one or more of ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances. Databases 109A-109D may store data usable by devices or systems in cloud infrastructure 106. The data, in some embodiments, may include e-commerce data (e.g., shipments, orders, inventory), media data (e.g., pictures, movies, streaming data), financial data (e.g., banking data, investment data), or other data.

[0056] Storage 111A-111D may include storage systems for use by devices and systems in cloud infrastructure 106. In some embodiments, one or more of storage 111A-111D may be implemented as a hard drive, a RAID array, flash memory, optical storage, or any other kind of storage. Each of 111A-111D may include one or more filesystems (e.g., Amazon Elastic File System, GlusterFS, Google File System, Hadoop Distributed File System, OpenZFS, S3, Elastic Block Storage).

[0057] In some embodiments, systems and devices of cloud infrastructure 106 may use databases 109A-109D to store data that is accessed frequently (where, for example, access is

required within a few milliseconds), and may use storage 111A-111D to store data that is accessed less frequently (where, for example, access is required within a few minutes or hours).

[0058] Keystores 113A-113D may include systems storing keys for accessing data and functionality. For example, to access certain data or systems, a system may require the use of passwords or keys in keystores 113A-113D for authentication. The data and functionality that the keys grant access to may be part of cloud infrastructure 106 or may be separate from cloud infrastructure 106. For example, keystores 113A-113D may include systems that store public and private keys (e.g., for use via SSH), may store passwords (e.g., login information for websites or programs), may store tokens (e.g., one-time passcodes), or the like. In some embodiments, keystores 113A-113D may be implemented as one or more of Amazon Web Services KMS, Azure Key Vault, or Google KMS.

[0059] Load balancer 115 may include one or more systems that balance incoming requests between the different systems and devices of cloud infrastructure 106. For example, load balancer 115 may be configured to determine usage (e.g., processor load, used storage capacity) of systems or devices in cloud infrastructure 106 to assist in determining where to route an incoming request from network 105 to store data, perform processing, or retrieve data.

[0060] Load balancer 115 may be configured to receive an incoming request from user device 102. Upon receipt of the request, load balancer 115 may consult a data store (part of or separate from load balancer 115; not pictured) to determine usage or forecasted usage of various systems or devices in cloud infrastructure 106, and may forward the request to the systems or devices having the lowest usage or forecasted usage.

[0061] **Fig. 2A** is a schematic block diagram illustrating an exemplary embodiment of a process 200 for integration, scanning, assessment, and review, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2A are executed using software and hardware of scanning system 101. In some embodiments, the steps in Fig. 2A may be performed in an order other than those depicted in Fig. 2A. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step of Fig. 2A may be used to provide context or other information for use in another step of Fig. 2A. In other embodiments, the steps in Fig. 2A may be executed by other devices.

[0062] Process 200 begins with step 201. In step 201, scanning system 101 may execute a process of integration. The integration process may be performed by scanning system 101 with cloud infrastructure 106. In some embodiments, the integration process includes creating a connection between an account on scanning system 101 and an account on cloud infrastructure 106. The process of integration in step 201 may, in some embodiments, be implemented as described below with respect to Fig. 2B.

[0063] In step 203, scanning system 101 may execute a process of scanning/mapping. The scanning/mapping process may be performed by scanning system 101 with cloud infrastructure 106. In some embodiments, the scanning and mapping process may include analyzing data

relating to cloud infrastructure 106 using scanning system 101, by reading cloud infrastructure 106 through the connection made in step 201 and generate a "map" (e.g., a data structure or data collection) representing systems and devices in cloud infrastructure 106. The process of scanning and mapping in step 203 may, in some embodiments, be implemented as described below with respect to Fig. 2C.

[0064] In step 205, scanning system 101 may execute a process of assessing. The assessing process may be performed by scanning system 101. In some embodiments, the assessing process may include reviewing vulnerabilities, infrastructure, interconnections, data, and other information, using scanning system 101. The process of assessing in step 205 may, in some embodiments, be implemented as described below with respect to Fig. 2D.

[0065] In step 207, scanning system 101 may execute a process of analyzing/reporting. The analyzing/reporting may be performed by scanning system 101. In some embodiments, the change reviewing process comprises scanning system 101 scanning cloud infrastructure 106 again to determine the differences between an earlier observed snapshot of cloud infrastructure 106 and the current state thereof. In some embodiments, scanning system 101 may analyze information (e.g., steps 203 and/or 205) to generate reports. The reports may list what each vulnerability is, where it is located, and its priority. In this way, security engineers and DevOps teams may be able to easily assess how to best allocate their time and attention.

[0066] In some embodiments, analyzing/reporting in step 207 may include scanning system 101 combining conclusions from different environmental perspectives (e.g., metadata) into a single model. For example, scanning system 101 may map the running services on cloud infrastructure 106 and consider collected vulnerability data. Scanning system 101, in some embodiments, may generate a visualization of the map for review, listing where the vulnerabilities are on a two-dimensional graphical representation of cloud infrastructure 106.

[0067] In some embodiments, scanning system 101 may indicate in the map whether an asset (e.g., a system or device) is Internet-facing and easily accessible to attackers, exposed only to internal assets or if it is private altogether (and may be less critical). For example, based on the contextual map, scanning system 101 may perform a "forward" analysis of the specific asset under identification to identify at least one possible Internet-originating attack vector to the asset. Alternatively or additionally, scanning system 101 may perform a "backward" analysis of the specific asset to identify exposure risk to assets downstream of the specific asset, wherein the downstream exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset. Both backward and forward analysis can be recursively, for one or more hops, Scanning system 101 may present these analyses to a user via graphical user interface or expose them via an API.

[0068] As one example, scanning system 101 may determine a vulnerability in a web service. In some embodiments, scanning system 101 may score the vulnerability as: high risk if the web service is connected to the Internet (e.g., has at least one public port forwarded through a

firewall to the web service, is able to be accessed through a load balancer, is able to be accessed via reverse proxy); medium risk if the web service is only accessible internally (e.g., because of firewall configuration); and low risk if access to the web server is blocked by a configuration of cloud infrastructure 106. As another example, if a machine is stopped (e.g., turned off, not running, not connected), a vulnerability, but one that is less likely to be exploited because the machine isn't running. This affects its risk score and other mitigating factors.

[0069] In step 209, scanning system 101 may execute a process of change reviewing. The change reviewing may be performed by scanning system 101. In some embodiments, the change reviewing process may include scanning system 101 scanning cloud infrastructure 106 again to determine the differences between an earlier observed snapshot of cloud infrastructure 106 and the current state thereof.

[0070] In some embodiments, scanning system 101 may monitor cloud change, or "delta," logs (e.g., Cloud event logs such as AWS CloudTrail, monitoring a network-related change in state, a trust-related change in state, or a disk configuration-related change in state in at least one of the primary asset group or a secondary asset group) and contextualize associated risks discovered within a customer cloud account. In some embodiments, scanning system 101 may generate a new map of assets based on a review of all systems or devices in cloud infrastructure 106 (e.g., as described above with respect to step 203), and compare the new map to the map generated earlier in step 203 to determine which assets have changed. In some embodiments, this comparison may require comparing data, metadata, network connections, software configurations, firewall configurations, or any other aspects related to cloud infrastructure 106, including, for example, determining new assets that are created or assets for which their configuration has changed.

[0071] After step 209, as depicted in Fig. 2A, process 200 may return to step 205. Scanning system 101 may then perform step 205 again. In some embodiments, steps 205, and 207 may be performed only on the devices or systems that have experienced a change that was detected by step 209. In some embodiments, process 200 may return to step 203 after step 209.

[0072] **Fig. 2B** is a schematic block diagram illustrating an exemplary embodiment of a process 210 for integration, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2B are executed using software and hardware of scanning system 101.Integration, in some embodiments, is a process for creating a trust relationship between scanning system 101 and cloud infrastructure 106.

[0073] In some embodiments, the steps in Fig. 2B may be performed in an order other than those depicted in Fig. 2B. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step of Fig. 2B may be used to provide context or other information for use in another step of Fig. 2B. In other embodiments, the steps in Fig. 2B may be executed by other devices.

[0074] Process 210 may begin with step 211 to initiate a connection to cloud infrastructure 106. In step 211, scanning system 101 may send a message to user device 102 instructing a user to authenticate, or log in, to a cloud service provider's system operating cloud infrastructure 106. For example, the user may use a username, password, one-time password, two-factor authentication, or any other authentication mechanism to gain access to a cloud service provider's system.

[0075] Concurrently with or after the first message, scanning system 101 may send a second message to user device 102, instructing the user to generate a role. The second message may include instructions for the user to follow to generate the role. In step 213, a user may provide (e.g., via a keyboard at user device 102) a role definition to the cloud service provider's system. In some embodiments, the role definition includes read-only permissions and permissions to read a block storage layer (containing block storage volumes). In some embodiments, scanning system 101 provides a role formation template (e.g., an Amazon Web Services CloudFormation Template) for use with cloud infrastructure 106 to create the necessary role. In step 213, the user may utilize user device 102, for example, by copying and pasting a URL of the template, downloading and uploading the template to the cloud service provider's system, or selecting the template from a list of templates.

[0076] In step 215, the cloud service provider's system may generate a string (e.g., a "key" or "resource name") for use by scanning system 101. In some embodiments, this string may be used to enable access by scanning system 101 to the workload of cloud infrastructure as permitted by the generated role. A user using user device 102 may copy the string and paste it into a user interface presented by scanning system 101 on user device 102. Other aspects of transmitting this string to scanning system 101 are possible as well.

[0077] In step 217, scanning system 101 determines that it is able to access cloud infrastructure 106. For example, scanning system 101 may attempt to perform a command such as authenticating to cloud infrastructure 106 using the string received in step 215 or attempting to perform a command using cloud infrastructure (e.g., requesting a listing of files stored on storage 111A). Once scanning system 101 determines that it is able to access cloud infrastructure 106, the process may return to Fig. 2A and step 203.

[0078] **Fig. 2C** is a schematic block diagram illustrating an exemplary embodiment of a process 220 for scanning/mapping, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2C are executed using software and hardware of scanning system 101. In some embodiments, the steps in Fig. 2C may be performed in an order other than those depicted in Fig. 2C. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step of Fig. 2C may be used to provide context or other information for use in another step of Fig. 2C. In other embodiments, the steps in Fig. 2C may be executed by other devices.

[0079] Process 220 may begin with step 221 where scanning system 101 initiates a process to access keys, such as those keys stored in keystores 113A-113D. Scanning system 101 may utilize the string (e.g., a key or resource name) received in process 210 to authenticate in order to retrieve keys stored in keystores 113A-113D. Once authenticated, scanning system 101 may access one, some, or all keys stored in keystores 113A-113D. In some embodiments, step 221 may comprise utilizing pre-established trust relationships, such as AWS trust policies, instead of creating a new relationship.

[0080] In step 223, scanning system 101 may generate a "snapshot" of devices and systems in cloud infrastructure 106. In some embodiments, generating a snapshot may include reading devices and systems in cloud infrastructure, such as storage 111A-111D, databases 109A-109D, and virtual machines 107A-107D, and copying the information read from those devices and systems to storage at scanning system 101. Generating a snapshot, in some embodiments, may include recording a reference count to data blocks in one or more of storage 111A-111D, databases 109A-109D, and virtual machines 107A-107D, and copying each block them like a copy and write operation. In some embodiments, scanning system 101 may generate an Elastic Block Storage snapshot in step 223.

[0081] In some embodiments, scanning system 101 uses one or more keys retrieved from keystores 113A-113D to encrypt "snapshots" of cloud infrastructure.

[0082] In step 225, scanning system 101 may apply "tags" to the snapshot. This process may include, in some embodiments, adding information to the snapshot to identify the snapshot as being associated with scanning system 101. Scanning system 101, in some embodiments, may be configured to delete only snapshots with the associated tags.

[0083] In step 227, scanning system 101 may generate a map of cloud infrastructure 106. In some embodiments, this map may be in a form of graph: a plurality of interconnected vectors connecting the plurality of systems and devices, based on the networking configuration. Scanning system 101, or other devices, may traverse the map to identify vectors originating in the Internet and reaching the devices and systems in cloud infrastructure 106. In some embodiments, generating the map may include enumerating Internet-accessible services that are capable of serving as an Internet proxy. In some embodiments, a user may control scanning system 101 (e.g., via user device 102) to display the map on a graphical user interface.

[0084] In some embodiments, generating the map may include enumerating properties of all assets, including: virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, load balancer 115, log files or databases, API gateway resources, API gateway REST APIs, Autoscaling groups, CloudTrail logs, CloudFront services, volumes, snapshots, VPCs, subnets, route tables, network ACLs, VPC endpoints, NAT gateways, ELB and ALB, ECR repositories, ECS clusters, services, and tasks, EKS, S3 bucket and Glacier storage, SNS topics, IAM roles, policies, groups, users, KMS keys, and Lambda functions. In some embodiments, generating the map in step 227 may further include analyzing devices or

systems for a subset of risk related situations, including determinations of compromise situations (e.g., where an attacker has already gained access), imminent compromise situations (e.g., a known attack vector exists and can be used, such as a data store that is exposed to the public Internet without authentication), hazardous situations (e.g., a serious security implication, but no full attack vector exists), or informational situations (e.g., when storage 111A-111D has a limited amount of free space, or an unexploitable vulnerability exists).

[0085] Generating the map in step 227 may include recording information such a region identifier, site identifier, datacenter identifier, physical address, network address, workload name, or any other identifier which may be acquired via an API provided through a cloud service provider's system.

[0086] In step 229, scanning system 101 may provide alerts, e.g., to user device 102, indicating any situations found during the process of generating the map. For example, scanning system 101 may send one or more of an email, a popup alert, a text message, or other notification to user device 102.

[0087] After performing step 229, the process may return to Fig. 2A and step 205.

[0088] **Fig. 2D** is a schematic block diagram illustrating an exemplary embodiment of a process 230 for assessing, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2D are executed using software and hardware of scanning system 101. In other embodiments, the steps in Fig. 2D may be executed by other devices. In some embodiments, scanning system 101 may execute the steps in Fig. 2D against a snapshot created in step 223 of Fig. 2C and stored at scanning system 101. In some embodiments, the steps in Fig. 2D may be performed in an order other than those depicted in Fig. 2D. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step of Fig. 2D may be used to provide context or other information for use in another step of Fig. 2D.

[0089] Process 230 begins with step 231. In step 231, scanning system 101 may perform a step of vulnerability scanning. In some embodiments, step 231 comprises extracting everything in the snapshot, including operating system packages, installed software applications, libraries, and program language libraries such as Java archives, Python packages, Go modules, Ruby gems, PHP packages, and Node.js modules, or other software applications.

[0090] In some embodiments, step 231 may determine library versions, software versions, and other identifying characteristics of software and operating systems in the snapshot. Scanning system 101 may then try to match them to known vulnerabilities stored in a vulnerability database (e.g., one of databases 103A-103D). The vulnerability database, in some embodiments, may include vulnerability data from: NVD, WPVulnDB, US-CERT, Node.js Security Working Group, OVAL – Red Hat, Oracle Linux, Debian, Ubuntu, SUSE, Ruby Advisory Database, JVN, Safety DB(Python), Alpine secdb, PHP Security Advisories Database, Amazon ALAS, RustSec Advisory Database, Red Hat Security Advisories, Microsoft MSRC, KB,

Debian Security Bug Tracker, Kubernetes security announcements, Exploit Database, Drupal security advisories, JPCERT. The vulnerability database may also include other vulnerability data (including, e.g., manually-added vulnerability data or vulnerability sources not listed above).

[0091] In step 233, scanning system 101 may perform a step of configuration scanning. In step 233, scanning system 101 may gather configuration information—such as a list of users of each system or device (e.g., virtual machines 107A-D), each system's or device's services, password hashes, and application-specific configurations for software/services such as Apache, Nginx, SSH, and other services. In some embodiments, scanning system 101 may perform a first analysis on all information collected in step 233 to remove sensitive information (e.g., social security numbers, passwords, birthdates) before proceeding to review the configuration information. In some embodiments, step 233 may comprise verifying adherence of the systems or devices in cloud infrastructure 106 to standards or benchmarks established by an external entity, such as the Center for Internet Security.

[0092] In some embodiments, scanning system 101 may perform a benchmarking process to detect misconfigurations of any services based on the information gathered in step 233. For example, scanning system 101 may determine the software version of each service and examine it against known vulnerabilities (e.g., stored in database 103A).

[0093] In some embodiments, scanning system 101 may determine bugs or other configuration risks that might only be exploitable from internal machines, because such bugs and risks can facilitate an attacker's lateral movement.

[0094] In some embodiments, scanning system 101 may evaluate network misconfigurations and their implications. Scanning system 101 may query devices and systems capable of routing traffic (e.g., load balancer 115, routers, switches, firewalls, and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations.

[0095] In step 235, scanning system 101 may perform a step of malware scanning. In some embodiments, scanning system 101 may perform malware scanning across all filesystems in the snapshot (e.g., gathered from virtual machines 107A-107D or storage 111A-111D). Scanning system 101 may use multiple malware scanning software solutions to perform a malware scan against the filesystems, including one sourced from another vendor, such as bucketAV, Trend Micro Cloud One, Sophos Cloud Optix, Crowdstrike Falcon CWP, or others. In some embodiments, malware scanning in step 235 comprises utilizing signatures, heuristics, or sandboxing capabilities to deduce whether there is an infection on the machine.

[0096] In step 237, scanning system 101 may perform a step of lateral movement scanning. An attacker who establishes a network foothold usually attempts to move laterally from one resource to another in search of rich targets such as valuable data. Stolen passwords and keys unlock access to servers, files, and privileged accounts. In some embodiments, scanning

system 101 may gather keys from each scanned system or device (e.g., virtual machines 107A-107D or storage 111A-111D). In some embodiments, scanning system 101 searches for passwords, scripts, shell history, repositories, or other data that may contain passwords, cloud access keys, SSH keys, or other key/password/access information that provide unchecked access to important resources. In some embodiments, scanning system 101 searches for such keys/passwords/access information and calculates a "hash" (a mathematical fingerprint) of each string. Scanning system 101 then attempts to match the hashed strings to hashes of strings that that are stored on different systems or devices. This will be used to detect the potential lateral movement between assets.

[0097] In step 239, scanning system 101 may perform a step of key/password scanning. As one example situation, suppose there is a weak or unprotected password stored (in plain text) in storage 111A. For example, if a personal email account has been compromised, the passwords may be known about in advance. Scanning system 101 may search the snapshot for similar usernames or login names, and, either using known dictionaries or the account owner's previously leaked passwords (stored in, e.g., database 103A), may attempt to login to one or more systems or devices in cloud infrastructure 106, and may record the result thereof.

[0098] In some embodiments, scanning system 101 may perform a "fuzzy search" on any usernames found in a password database (e.g., database 103A) to determine existing targets for password testing. In some embodiments, the fuzzy search uses the Damerau-Levenshtein edit distance algorithm.

[0099] As one example, scanning system 101 may determine that a leaked username includes the email address johndoe@acme.com. Scanning system 101 may try to match passwords from john.doe@acme.net, johndoe@gmail.com, and other variations. Scanning system 101 may perform a similar process against leaked passwords (e.g., if a leaked password is "Victory@19," scanning system may attempt to log in using "Victory@20."

[00100] In step 241, scanning system 101 may perform a step of sensitive information scanning. In some embodiments, scanning system 101 may search the snapshot for sensitive information, such as personally identifiable information (PII), Social Security numbers, healthcare information, or credit card numbers. In some embodiments, scanning system 101 may search data repository history as well. This is because it is not uncommon for an entire production environment repository to be cloned, with no one remembering the copy contains sensitive information. In some situations, detecting sensitive data not secured is critical in adherence to data privacy regulations.

[00101] To be certain that such alerts do not constitute false positives, in some embodiments, scanning system 101 may perform statistical scans against the data. For example, it is possible for a random number to resemble a Social Security number, yet it is extremely unlikely for the majority of a file with thousands of numbers to be valid Social Security Number by pure chance.

[00102] In step 243, scanning system 101 may perform a step of container scanning. In some embodiments, scanning system 101 may apply one or more of the preceding steps of Fig. 2D against containerized environments. In some embodiments, in order to do so, scanning system 101 reconstructs a container runtime layered file system (LFS) before recursively running one or more of steps 231-241 on the reconstructed file system.

[00103] Scanning system 101 may read network information from the snapshot in order to determine which services within which containers are exposed externally and within which ports they are accessible. For example, scanning system 101 may identify a port on which a vulnerable application is accessible based on known software vulnerabilities for a versions of software application. Scanning system 101 may query network accessibility information via an API provided through a cloud service provider's system and may use it to identify specific vulnerabilities susceptible to attack.

[00104] After performing step 243 on one or more containers, the process may return to Fig. 2A and step 207 (as discussed above).

[00105] **Foundational Techniques**

[00106] Aspects of this disclosure may include establishing a trusted relationship between a source account in a cloud environment and a scanner account. A trusted relationship, as used herein, may refer to a secure communication channel between at least two accounts (e.g., domains). The secure communications channel may include, for example, an administration link, communication link, a connection or communication network configured with security protocols, or any other secure relationship between two networked entities.. A trusted relationship between two accounts may enable user accounts and global groups to be used in a domain other than the domain where the accounts are defined. A source account, as used herein, may refer to a domain, a location on a network server, or anything used to access system resources. A cloud environment, as used herein, may refer to a platform implemented on, hosted on, and/or accessing servers that may be reached via the Internet or other shared network. A scanner account, as used herein, may refer to any type of account associated with a scanner. A scanner, as used herein, may refer to a device for examining, reading, or monitoring electronically accessible information. Fig. 1 illustrates one example of a scanner, which is depicted in an exemplary manner as scanning system 101.

[00107] By way of example, establishing a trusted relationship between a source account in a cloud environment and a scanner account may include an integration process (creating a connection between an account on scanning system 101 and an account on cloud infrastructure 106) discussed with reference to Fig. 2A. By way of one example, scanning system 101 of Fig. 1 may create a trusted relationship between a source account in a cloud environment and a scanner account.

[00108] Aspects of this disclosure may include using the established trust relationship, utilizing at least one cloud provider API to identify workloads in the source account. A cloud

provider API, as used herein, may refer to any application program interface that allows an end user to interact with a cloud provider's service. A cloud provider, may include any entity through whom information is accessible over the Internet or other shared network. Workloads, as used herein, may refer to systems, devices, or resources in a network (or available via a network) such as a cloud infrastructure. By way of example, Fig. 1 illustrates examples of a workload: virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115. In some embodiments, scanning system 101 of Fig. 1 may perform a scanning operation on workload (e.g., systems, devices, resources, etc.) in cloud infrastructure 106.

[00109] In some embodiments, scanning system 101 of Fig. 1 may use an API to detect workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115 in the source account.

[00110] Aspects of this disclosure may involve using the at least one cloud provider API to query a geographical location of at least one of the identified workloads. Querying, as used herein, may refer to a request for data or information from a resource. A geographical location, as used herein, may refer to the data center where this workload is served, a position on Earth, a location in a network, an address of a server or other hardware, or any other information capable of identifying a locus, region, or position. Internet geolocation involves software capable of deducing a geographic position of a device connected to the Internet. For example, the device's IP address may be used to determine the country, city, or ZIP code, determining its geographical location. By way of example, scanning system 101 of Fig. 1 may query any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 for their geographical location using a cloud provider API.

[00111] In some embodiments, the geographic location may include an identifier of a physical site. An identifier, as used herein, may refer to anything that allows for recognition of an object, location, or resource. A physical site, as used herein, may refer to an actual location. By way of example, scanning system 101 of Fig. 1 may query any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 for an identifier of their physical site (e.g., address, postal code, or any other physical descriptor) using a cloud provider API.

[00112] In some embodiments, scanning system 101 of Fig. 1 may use an out-of-band process to reach cloud workloads through a runtime storage layer, in combination with information gathered from cloud provider APIs.

[00113] Aspects of this disclosure may include receive an identification of the geographic location. An identification of the geographic location, as used herein, may refer to a recognition, association, or classification of the geographic location. By way of example, scanning system 101 of Fig. 1 may receive a geographical location, after querying for it, by interrogating one or

more workloads, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, or keystores 113A-113D of cloud infrastructure 106. For example, the scanning system 101 of Fig. 1 may receive address, postal code, or any other physical descriptor of any workloads in cloud infrastructure 106. In some embodiments, scanning system 101 of Fig. 1 may receive a listing locating the workloads of cloud infrastructure 106 on a two-dimensional graphical representation.

[00114] In some embodiments, the identification of the geographic location may include an identification of a data center, at least one of a data center name, an Internet Protocol (IP) address, a name of the cloud provider, or a unique identity.. A data center, as used herein, may refer to a facility that centralizes an organization's shared IT operations and equipment for the purposes of storing, processing, and/or disseminating data and applications. A data center, for example, may include a dedicated space within a building, or a group of buildings used to house computer systems and associated components, such as telecommunications and storage systems. An identification of a data center, as used herein, may refer to a recognition, association, or classification of a data center. A data center name, as used herein, may refer to a series of letters or numbers, a word, or set of words or numbers, an address, or a code identifying the data center. An Internet Protocol (IP) address, as used herein, may refer to a unique string of characters that identifies a computer using the Internet Protocol to communicate over a network. A cloud provider, as used herein, may refer to a company that offers any components of cloud computing (e.g., infrastructure as a service (IaaS), software as a service (SaaS) or platform as a service (PaaS)). A unique identity, as used herein, may refer to a distinctive numeric or alphanumeric string that is associated with a single entity within a given system.

[00115] By way of example, scanning system 101 of Fig. 1 may query for and receive a geographical location such as a data center name/city and state, data center address, IP address / specific cloud identifier of the workload, name of cloud provider (e.g., Amazon Web Services), or a specific identifier of information of any of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00116] In some embodiments, scanning system 101 of Fig. 1 may receive a listing indicating locations of the workloads of cloud infrastructure 106 on a two-dimensional graphical representation.

[00117] In some other embodiments, scanning system 101 may receive additional information (in addition to geographic location information) such as whether the workload is Internet-facing and easily accessible to attackers, or if it is private (and may be less critical).

[00118] Aspects of this disclosure may include using the cloud provider APIs to access block storage volumes of the at least one workload. Block storage volumes, as used herein, may refer to data files maintained on Storage Area Networks (SANs) or in cloud-based storage

environments. Each storage volume may act as an individual hard drive configured by a storage administrator. Accessing block storage volumes, as used herein, may refer to obtaining, examining, or retrieving data.

[00119] As described with relation to process 210 of Fig. 2B (beginning with step 211 to initiate a connection to cloud infrastructure 106), scanning system 101 may send a second message to user device 102, instructing the user to generate a role. A role may refer to a type associated with a node that may be assigned or defined. The second message may include instructions for the user to follow to generate the role. In step 213, a user may provide (e.g., via a keyboard at user device 102) a role definition to the cloud service provider's system. In some embodiments, the role definition includes read-only permissions and permissions to read a block storage layer (containing block storage volumes). In some embodiments, scanning system 101 provides a role formation template (e.g., an Amazon Web Services CloudFormation Template) for use with cloud infrastructure 106 to create the necessary role. In step 213, the user may utilize user device 102, for example, by copying and pasting a URL of the template, downloading and uploading the template to the cloud service provider's system, or selecting the template from a list of templates.

[00120] By way of example, scanning system 101 of Fig. 1 may access the block storage volume of any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00121] Aspects of this disclosure may include determining a file-system of the at least one workload. A file-system, as used herein, may refer to a method and/or a data structure that the operating system uses to control how data is stored and retrieved.

[00122] By way of example, scanning system 101 of Fig. 1 may identify the type of file-system of any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106. Some file systems may include NTFS, FAT32, ext3, exFAT, HFS, HFS+, HPFC, UFS, ZFS.

[00123] Aspects of this disclosure may include mounting the block storage volumes on a scanner based on the determined file-system. Mounting the block storage volumes, as used herein, may refer to a process by which the operating system makes files and directories (such as block storage volumes) on a storage device available for users to access via the computer's file system. A scanner, as used herein, may refer to a device for examining, reading, or monitoring electronically accessible information. By way of example, Fig. 1 illustrates one example of a scanner, scanning system 101 of Fig. 1. In some embodiments, mounting may include selecting a driver corresponding to the determined file system. In some embodiments, to mount the block storage volumes on the scanner, may include creating a snapshot of the block storage volumes; and mounting the snapshot of the block storage volumes on the scanner. A snapshot of the block storage volumes, as used herein, may refer to a current state or earlier state of the block storage volumes.

[00124] In some embodiments, the at least one processor may be further configured to encrypt the snapshot of the block storage volumes and mount the encrypted snapshot of the block storage volumes on the scanner. Encrypting a snapshot, as used herein, may refer to a process of encoding information (such as a snapshot). This process may convert the original representation of the information, known as plaintext, into an alternative form known as ciphertext.

[00125] In some embodiments, the scanner may use a privileged account to log in and determine how secure each host is from an inside vantage point. While authenticated scans can successfully discover potential vulnerabilities, they may be limited to the extent they require a privileged account on each scanned host. Furthermore, scans use significant system resources during the test procedures and require opening ports that by themselves pose a security risk.

[00126] Aspects of this disclosure may include activating a scanner at the geographic location. Activating a scanner, as used herein, may refer to starting up or initiating a device for examining, reading, or monitoring data. By way of example, scanning system 101 of Fig. 1 may start up a scanner (a different scanner than scanning system 101) at the geographic location identified by scanning system 101.

[00127] Aspects of this disclosure may include reconstructing from the block storage volumes a state of the workload. Reconstructing, as used herein, may refer to rebuilding or reforming a state of the workload. A state of the workload, as used herein, may refer to a condition of a workload at a certain time. By way of example, scanning system 101 of Fig. 1 may rebuild the block storage volumes as an earlier version of any of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00128] In some embodiments, the reconstructed state of the workload may include at least two of an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content. An indication of an installed application, as used herein, may refer to an identifier of a software. A version of an installed application, as used herein, may refer to a particular form, variant, edition, or revision of a piece or pieces of software. An operating system configuration, as used herein, may refer to a manner in which components are arranged to make up an operating system or a computer system. An application configuration, as used herein, may refer to a manner in which components are arranged to make up an application or a computer system. A profile configuration, as used herein, may refer to a manner in which components are arranged to make up a profile or a computer system. A log, as used herein, may refer to a detailed list of an application information, system performance, or user activities. Database content, as used herein, may refer to any collection of data, or information, that is specially organized for rapid search and retrieval by a computer. In some embodiments, scanning system 101 of Fig. 1 may reconstruct any workload, such as virtual machines 107A-

107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 from the block storage volumes to an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content.

[00129] Aspects of this disclosure may include assessing the reconstructed state of the workload to extract insights. Assessing the reconstructed state of the workload, as used herein, may refer viewing, utilizing, or evaluating the reconstructed state of the workload. Insights, as used herein, may refer to information or knowledge attained based on an analysis of some data. Extracting insights, as used herein, may refer to pulling information or data about, for example, a vulnerability associated with the workload or a composition of installed applications associated with the workload. By way of example, scanning system 101 of Fig. 1 may evaluate the reconstructed state of any of workloads, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 to pull valuable data and information.

[00130] In some embodiments, insights may include at least one of a vulnerability associated with the workload or a composition of installed applications associated with the workload. A vulnerability, as used herein, may refer to any weakness within an organization's information systems, internal controls, or system processes that can be exploited. A composition of installed applications associated with the workload, as used herein, may refer to a structure or group of accessible computer programs. By way of example, scanning system 101 of Fig. 1 may evaluate the reconstructed state of any workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 to pull valuable data and information regarding a vulnerability with the workload.

[00131] In some embodiments, a scanner may be deployed at the geographical location. Deploying a scanner, as used herein, may refer to installing, activating, utilizing, or bringing a scanner into effective action . By way of example, Fig. 1 illustrates one example of a scanner, scanning system 101 of Fig. 1 that is deployed.

[00132] In some embodiments, scanning system 101 of Fig. 1 may update the reconstructed state of the workload based on at least one change to the block storage volumes . Updating the reconstructed state of the workload, as used herein, may refer to overhauling or refurbishing the reconstructed state of the workload. A change to the block storage volumes, as used herein, may refer to any shift, alteration, or modification to the block storage volumes.

[00133] **Fig. 3** illustrates a block diagram of method 300 for insight extraction, consistent with disclosed embodiments. In some embodiments, the method may include ten (or more or less) steps:

[00134] Block 302: Establish a trusted relationship between a source account in a cloud environment and a scanner account. In some embodiments, establishing a trusted relationship

between a source account in a cloud environment and a scanner account may include an integration process (creating a connection between an account on scanning system 101 and an account on cloud infrastructure 106) discussed with reference to Fig. 2A. By way of one example, scanning system 101 of Fig. 1 may create a trusted relationship between a source account in a cloud environment and a scanner account.

[00135] Block 304: Using the established trust relationship, utilize at least one cloud provider API to identify workloads in the source account. In some embodiments, scanning system 101 of Fig. 1 may use an API to detect workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115 in the source account.

[00136] Block 306: Use the at least one cloud provider API to query a geographical location of at least one of the identified workloads. In some embodiments, scanning system 101 of Fig. 1 may query any of workloads virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 for their geographical location using a cloud provider API.

[00137] Block 308: Receive an identification of the geographic location. In some embodiments, scanning system 101 may indicate in the map additional information regarding the workload (e.g., a system or device). For example, scanning system 101 may provide: an identification of a data center (e.g., a name or number associated with the data center where the workload is located), at least one of a data center name (e.g., a name or address associated with the data center where the workload is located), Internet Protocol (IP) address, name of the cloud provider, or a unique identity (e.g., any other information related to the workload that may be used).

[00138] Block 310: Use the cloud provider APIs to access block storage volumes of the at least one workload. In some embodiments, scanning system 101 of Fig. 1 may access the block storage volume of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106. In some embodiments, accessing the block storage may include taking the mean of generating snapshots.

[00139] Block 312: Determine a file-system of the at least one workload. In some embodiments, scanning system 101 of Fig. 1 may identify the type of file-system any of workloads (virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106).

[00140] Block 314: Mount the block storage volumes on a scanner based on the determined file-system. In some embodiments, to mount the block storage volumes on the scanner, may include instructing the operating system to map a directory structure logically to a physical storage device. A storage volume may be mounted after it is attached and formatted

for use by a server's operating system. Mounting may include creating a snapshot of the block storage..

[00141] Block 316: Activate a scanner at the geographic location. In some embodiments, scanning system 101 of Fig. 1 may start up a scanner (a different scanner than scanning system 101) at the geographic location identified by scanning system 101.

[00142] Block 318: Reconstruct from the block storage volumes a state of the workload. In some embodiments, scanning system 101 of Fig. 1 may rebuild the block storage volumes to an earlier version of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106.

[00143] Block 320: Assess the reconstructed state of the workload to extract insights. In some embodiments, scanning system 101 of Fig. 1 may evaluate the reconstructed state of workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D of cloud infrastructure 106 to derive valuable data and information.

[00144] **Vulnerability Management Techniques**

[00145] **Fig. 4** depicts a cybersecurity system performing a side scanning function to protect against potential vulnerabilities. Referring to Fig. 4, a processor 400 configured to identify in a cloud environment one or more block storage volumes 401A-C communicatively connected to a cloud provider API where the block storage volumes 401A-C may be contained. When a block storage volume, e.g., 401A is identified, the processor 400 may be configured to perform the function of identifying the software installed 403A in or relating to block storage volume 401. Upon identifying installed software 403A, the processor may be further configured to identify the version of software 405A operating in the installed software 403A. Based on the installed software 403A and the version of the software 405A, the processor may be configured to identify a listing of known vulnerabilities 407 that may relate to the installed software 403A and the version installed 405A. When the listing of known vulnerabilities 407 is obtained, the system may list the known vulnerabilities 409A for that software version 411A, which can be communicated to the processor 400 of the end user. This information may be processed and displayed to an end user using processor 400.

[00146] **Fig. 5** of the disclosed embodiments describes a method of operating a cybersecurity system performing a side scanning function to protect against potential vulnerabilities. Referring to Fig. 5, a processor unit 500 may be configured to perform the method . When initiated either manually or automatically, processor unit 500 may be communicatively connected with a cloud provider API 501 to initiate a cybersecurity function as disclosed above. Contained digitally on a memory device and accessed by a cloud provider API 501 is a series of block storage volumes 503. Processor unit 500 may be capable of performing a vertical and horizontal scan for network access or security information of a block storage volume 503 to detect potential vulnerabilities to the block storage volume 503 or any processing unit 500 designed to access said block storage volume 503.

[00147] Based on the accessed block storage volume 500, processor unit 500 with the disclosed embodiments of a cybersecurity system may perform an identification of the type of software installed 505 connected with block storage volume 503. Following identification of the type of software installed 505, the processing unit may then identify an installed version 507 of said installed software 505 for a version comprised of a unique identifier based on a combination of letters, numbers, or similar unique identifiers.

[00148] Upon recognition of an installed software version 507 of the installed software 505, processing unit 500 with the disclosed embodiments of the disclosed scanning system 101 may provide a list of known vulnerabilities 509 for review by said scanning system 101 and/or its end user and maintainer of said block storage volumes 503. Based on the list of known vulnerabilities 509, the disclosed scanning system 101 or its end user may identify one or more ports of accessibility to said block storage volume 503 that may be accessed by a known, associated vulnerability 513 from listed vulnerabilities 509 to determine an avenue for potential vulnerability 513 to access and infiltrate block storage volume 503.

[00149] In some embodiments, a cyber security scanning system for a cloud environment 101 may include a processor 500 to operate said system. This processor 500 may include central processing units and other similar computer-enabling equipment for processing and executing commands based on the information inputted to said system. The processor 500 may be communicatively connected to a computer network or series of networks to accomplish said cyber security function.

[00150] As an example embodiment, a processor unit 500 may be configured to use a cloud-provider API that may communicate with one or more specified computer-readable media across a digital network. This can be accomplished through internet protocols, internet control message protocols, transmission control protocol, or user datagram protocol. Cloud provider API 501 may be one of several forms of middleware, interface, middle layer, or other systems of interfacing applications. A processor 300 may be one or more computer processing units, central processing unit, server, microcomputer, mainframe, and any other manifestation of digital computing.

[00151] Further to one of several possible embodiments, a cloud-provider API 501 may be configured to access a block-storage volume 503 of a workload maintained in a cloud-storage environment. This may be accomplished through a system of computer-readable media communicatively connected. Said block-storage volume 303 may be contained on a Storage Area Network (SAN) or similar cloud-based memory storage environment. The block storage volume 503 may be contained in smaller storage volumes with an associated identifier unique to that portion of said block storage volume 503. In some embodiments, the block-storage volume 503 of a workload may have multiple paths for the storage volume to be reaggregated and retrieved quickly.

[00152] Among several embodiments, a scanning system 101 may comprise a system for identifying an installed software application in the accessed block-storage volume 503. This identification of installed software may be accomplished by accessing installed software 505 files through signature verification, root license, or authorized user lists. The installed software 505 may be located and identified within applications such as file storage, database storage, and virtual machine file system volumes. The identification of said installed software application 507 may be processed, analyzed, and communicated to the scanning system 101 for processing, cataloging, and protection through encryption and various methods of layered cyber defense.

[00153] Further, the scanning system 101 described herein may include functionality to analyze installed software applications to determine the associated software version. The software application version 507 may identify the software version based on unique version name, unique version number, and may be based on unique states of the currently installed computer software 505.

[00154] One of many embodiments disclosed above may include the scanning system 101 having the ability to access a data structure of known software vulnerabilities 509 for a plurality of versions of software applications. The known software vulnerabilities 509 may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software. The scanning system 101 may be able to access and identify software vulnerabilities for mitigation, rectification, correction, and fortification.

[00155] In one embodiment, a cybersecurity system may also perform scanning according to scanning system 101 by performing a lookup of the identified installed software version 507 in the data structure to identify known vulnerabilities 509. This function can be performed by the scanning system 101 according to Fig. 1 performing a query of the installed software 505 for unique version number or designator and comparing to, amongst many things, a set of likely or potential vulnerabilities to that software version for potential deficiencies or cybersecurity threats known or suspected to similar software types and versions. This query may be performed according to a predetermined set of values, to include previously identified unique version numbers or designators that may contain the known list of previously identified vulnerabilities.

[00156] Among many embodiments, embodiments of the disclosed scanning system 101 may query the cloud provider API 501 to determine network accessibility information 511 related to the workload. In order to accomplish this query of the cloud provider API 301, the scanning system 101 may involve index of search results and display of said search results, followed by

processing and grouping search results. Network accessibility information 511 may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, or Hadoop nodes, among others.

[00157] If the system detects a vulnerable application 513, one embodiment may identify one or more port on which said vulnerable application is accessible. In one of several embodiments, the scanning system 101 may detect a vulnerable application in one or more computation process. In another embodiment, the cybersecurity system may perform a network accessibility query in a separate process. Further, a disclosed embodiment may perform these separate functions in subsequent and sequential steps of the same process. A person having skill would understand that an authorized user or an authorized scanning system 101 can perform these functions concurrently and subsequently by an authorized user or an authorized cybersecurity system while performing the same function as the disclosed embodiment.

[00158] Upon gathering network accessibility information and the identified port to identify one or more vulnerabilities susceptible to attack from outside the workload, a disclosed embodiment would have the functionality to perform processes to gather, display, and mitigate a discovered vulnerability in order to minimize the likelihood and effectiveness of a cyber threat outside of and attempting to access a workload through a known or previously encountered type of cyber threat. This functionality may include collecting and organizing the vulnerabilities according to type or category of vulnerability, displaying the gathered data for an end user or maintainer, and implementing security features automatically or manually by a user or maintainer such as security patches, password or passcode changes and suggestions for users to do the same, and malicious code eradication.

[00159] As one of several possible embodiments, the scanning system 101 may also, upon identification of one or more vulnerabilities, implement remedial actions via one or more processors. Remedial actions may include, among other things, notification to an end user of an identified threat, compensation through a revised security code to mitigate the potential threat, publication of the identified threat and vulnerability in a log or record of detected vulnerabilities, and communication of the sensed vulnerability and threat to a server operator or maintainer to fortify the protections of workloads existing on similar environments.

[00160] In a similar embodiment to above, the remedial measure may include transmission of an alert to a device associated with an administrator. The alert may be, amongst others, written, auditory, and visual for processing and use by an administrator of said scanning system 101. Said administrator may take action based on the received alert, to include eliminating the cyber threat through mitigation measures, change in cybersecurity posture, or removing the workload from the cybersecurity threat environment.

[00161] In another embodiment, a query of the cloud provider API 501 to determine network accessibility information related to the workload may be performed by at least one processor 500 configured to examine data sources associated with the workload. The data examined may include user data, system processing data, accessibility data, clock cycles, storage input/output, or similar data processors. A query of the cloud provider API 501 may be automated or manually-initiated. Based on the said query, the network accessibility information 511 related to the workload may change based on the data sources associated with said workload.

[00162] In the above processor and similar embodiments, further configuration may include a process to determine network accessibility information 511 based on the examined data sources. The examined data sources may include various cloud-based workloads, internet protocols, transmission communication protocols, or other methods and systems of memory and data storage.

[00163] As an exemplary embodiment, network accessibility information 511 includes at least one of: data from an external data source, cloud provider information, or at least one network capture log. These embodiments of an external data source may include data from the operating environment of the cloud-based environment, an external operating system for a computer processing unit 500, or other similar computer readable media. Cloud provider information may further include information that may identify the network accessibility information 511 vertically or horizontally to fully describe the associated workload. Network capture logs may be automated or manually updated to include possible vulnerabilities 513 and threats to the cloud-based storage medium.

[00164] Further, the disclosed embodiments may include an installed software application 505, with the at least one processor 500 configured to extract data from at least one of operating system packages, libraries, or program language libraries. This data may be extracted through a system query, random access algorithm, or similar automated process. Operating system packages may include systems operable on Microsoft, Apple, Linux, and similar operating systems. Libraries may consist of a series of files, folders, and databases of information stored on one of any indexed data repositories. A program language library may contain several of an exemplary program languages including but not limited to Javascript, Swift, Scala, Go, Python, Elm, Ruby, C#, C++ and other similar sources of software code.

[00165] As another exemplary embodiment, a scanning system 101 may also include a processor 500 configured to identify installed software application 505 based on the extracted data. The processor 500 may perform only this function or this function among many to accomplish the layered cybersecurity defense described herein this disclosed cloud-based security environment. The identification of installed software by said processor may include identifying the software by unique version number or designator, recognized source code,

metadata associated with the installed software application files, or similar software-identifying information.

[00166] One embodiment may include an additional function wherein the installed software application 505 that has been identified includes one or more scripts. These scripts may be processed through various computer readable languages to include Javascript, C#, C++ and other forms of computer code.

[00167] One embodiment contemplated by the disclosed scanning system 101 may also include data structure includes aggregated vulnerability data 309. This aggregated vulnerability data 509 may be compiled by an end user or maintainer from within the cloud-based environment of the current data structure as well as compilation from similar data sets and aggregation of common threats to data structures likely to experience similar vulnerabilities. This aggregation of vulnerability data 509 may be contained within the data structure and it may be collectively aggregated to provide for a more robust and layered cybersecurity defense posture.

[00168] In an embodiment of the disclosed scanning system 101, the aggregated vulnerability data 509 may include data from one or more third-party vendors. These vendors may include operators of the cloud-based server environment, providers of networking and internet communication, methods of layered authentication, and other similar providers of services directly related and in communication with the cloud-based cybersecurity system.

[00169] As an additional exemplary embodiment, the aggregated vulnerability data 309 may include data collected by a scanner. This scanner may involve use of continuous or periodic monitoring of the workload. The scanner may perform security screenings of the various workloads vertically or horizontally to identify network identification information, port accessibility, and associated vulnerabilities. Any scan performed may be communicated to the scanning system 101 that may be responsible for performing and logging the results of the scan and may be able to initiate follow-on processes and protocols to protect the data contained in the workload that is the subject of scanning.

[00170] An embodiment of the disclosed scanning system 101 may also include aggregated vulnerability data 509 that may include at least one of an advisory, an exploit, a security announcement, or a known bug. An advisory may include notification to a system maintainer or user of the potential vulnerability, may log notice of the advisory, and may recommend possible user or maintainer actions to potentially address said advisory. An exploit may further consist of an automated system response designed to take advantage of the sensed vulnerability data. The exploit can be further reflected in the aggregated vulnerability data and protocols can be written into the cybersecurity infrastructure to prevent said exploit from gaining access and permissions to unauthorized areas of the workload storage environment.

[00171] A person having ordinary skill in the art would appreciate the above described embodiments are among many potential embodiments, to include a method of operating a scanning system 101 similar to the one described above. A disclosed embodiment contemplates this method to be accomplished through manual user operation, automated computer processes, or similar manners of operation. These manners of operation and those contemplated similar to them would allow the scanning system 101 described and disclosed to execute its operations as the system above describes.

[00172] One of several embodiments of the disclosed method of operating a scanning system 101 may include a cyber security scanning system for a cloud environment 101 may include a processor 500 to operate said system. This processor may include central processing units and other similar computer-enabling equipment for processing and executing commands based on the information inputted to said system. The processor may be communicatively connected to a computer network or series of networks to accomplish said cyber security function.

[00173] As an example embodiment, a processor unit 500 may be configured to use a cloud-provider API 501 that may communicate with one or more specified computer-readable media across a digital network. This can be accomplished through internet protocols, internet control message protocols, transmission control protocol, or user datagram protocol. Cloud provider API 501 may be one of several forms of middleware, interface, middle layer, or other systems of interfacing applications. A processor may be one or more computer processing units, central processing unit, server, microcomputer, mainframe, and any other manifestation of digital computing.

[00174] Further to one of several possible embodiments, a cloud-provider API 501 may be configured to access a block-storage volume 503 of a workload maintained in a cloud-storage environment. This may be accomplished through a system of computer-readable media communicatively connected. Said block-storage volume 503 may be contained on a Storage Area Network (SAN) or similar cloud-based memory storage environment. The block storage volume 503 may be contained in smaller storage volumes with an associated identifier unique to that portion of said block storage volume 503. In some embodiments, the block-storage volume 503 of a workload may have multiple paths for the storage volume to be reaggregated and retrieved quickly.

[00175] Among several embodiments, a method of operating a scanning system 101 may comprise a system for identifying an installed software application 505 in the accessed block-storage volume 503. This identification of installed software 505 may be accomplished by accessing installed software files through signature verification, root license, or authorized user lists. The installed software 505 may be located and identified within applications such as file storage, database storage, and virtual machine file system volumes. The identification of said installed software application 505 may be processed, analyzed, and communicated to the

method of operating a scanning system 101 for processing, cataloging, and protection through encryption and various methods of layered cyber defense.

[00176] Further, a method of operating a scanning system 101 described herein may include functionality to analyze installed software applications 505 to determine the associated software version 507. The software application version 507 may identify the software version based on unique version name, unique version number, and may be based on unique states of the currently installed computer software.

[00177] One of many embodiments disclosed above may include the method of operating a scanning system 101 having the ability to access a data structure of known software vulnerabilities 509 for a plurality of versions of software applications 507. The known software vulnerabilities 509 may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software. The cyber security system may be able to access and identify software vulnerabilities 309 for mitigation, rectification, correction, and fortification.

[00178] In one embodiment, a method of operating a scanning system 101 may also perform scanning according to scanning system 101 by performing a lookup of the identified installed software version 507 in the data structure to identify known vulnerabilities 309. This function can be performed by the scanning system 101 according to Fig. 1 performing a query of the installed software 505 for unique version number 507 or designator and comparing to, amongst many things, a set of likely or potential vulnerabilities to that software version for potential deficiencies or cybersecurity threats known or suspected to similar software types and versions. This query may be performed according to a predetermined set of values, to include previously identified unique version numbers or designators that may contain the known list of previously identified vulnerabilities.

[00179] Among many embodiments, embodiments of the disclosed method of operating a scanning system 101 may query the cloud provider API 501 to determine network accessibility information 511 related to the workload. In order to accomplish this query of the cloud provider API 501, the scanning system 101 may involve index of search results and display of said search results, followed by processing and grouping search results. Network accessibility information 511 may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, or Hadoop nodes, among others.

[00180] If the system detects a vulnerable application, one embodiment may identify one or more port on which said vulnerable application is accessible. In one of several embodiments,

the scanning system 101 may detect a vulnerable application in one or more computation process. In another embodiment, the scanning system 101 may perform a network accessibility query in a separate process. Further, a disclosed embodiment may perform these separate functions in subsequent and sequential steps of the same process. A person having skill would understand that an authorized user or an authorized cybersecurity system may perform these functions concurrently and subsequently while performing the same function as the disclosed embodiment.

[00181] Upon gathering network accessibility information 511 and the identified port 513 to identify one or more vulnerabilities susceptible to attack from outside the workload, a disclosed embodiment would have the functionality to perform processes to gather, display, and mitigate a discovered vulnerability in order to minimize the likelihood and effectiveness of a cyber threat outside of and attempting to access a workload through a known or previously encountered type of cyber threat. This functionality may include collecting and organizing the vulnerabilities according to type or category of vulnerability, displaying the gathered data for an end user or maintainer, and implementing security features automatically or manually by a user or maintainer such as security patches, password or passcode changes and suggestions for users to do the same, and malicious code eradication.

[00182] A person having ordinary skill in the art would similarly understand that the above disclosed system can be disclosed using a suitable non-transitory computer-readable medium performing each of the disclosed functions. The disclosed embodiment may include a non-transitory computer readable medium with a scanning system 101 similar to the one described above on a non-transitory computer-readable medium. A disclosed embodiment contemplates this system to be accomplished through a medium that may contain a central processing unit, virtual machine, or a similar non-transitory medium. These manners of operation and those contemplated similar to them would allow the scanning system 101 described and disclosed to execute its operations as the system above describes.

[00183] One of several embodiments of the disclosed non-transitory computer readable medium with a scanning system 101 may include a cyber security scanning system for a cloud environment 101 may include a processor 500 to operate said system. This processor 500 may include central processing units and other similar computer-enabling equipment for processing and executing commands based on the information inputted to said system. The processor 500 may be communicatively connected to a computer network or series of networks to accomplish said cyber security function.

[00184] As an example embodiment, a processor unit 300 may be configured to use a cloud-provider API 501 that may communicate with one or more specified computer-readable media across a digital network. This can be accomplished through internet protocols, internet control message protocols, transmission control protocol, or user datagram protocol. Cloud provider API 501 may be one of several forms of middleware, interface, middle layer, or other

systems of interfacing applications. A processor 500 may be one or more computer processing units, central processing unit, server, microcomputer, mainframe, and any other manifestation of digital computing.

[00185] Further to one of several possible embodiments, a cloud-provider API 501 may be configured to access a block-storage volume 503 of a workload maintained in a cloud-storage environment. This may be accomplished through a system of computer-readable media communicatively connected. Said block-storage volume 503 may be contained on a Storage Area Network (SAN) or similar cloud-based memory storage environment. The block storage volume 503 may be contained in smaller storage volumes with an associated identifier unique to that portion of said block storage volume 503. In some embodiments, the block-storage volume 503 of a workload may have multiple paths for the storage volume to be reaggregated and retrieved quickly.

[00186] Among several embodiments, a non-transitory computer readable medium with a scanning system 101 may comprise a system for identifying an installed software application 505 in the accessed block-storage volume 503. This identification of installed software 505 may be accomplished by accessing installed software files through signature verification, root license, or authorized user lists. The installed software 505 may be located and identified within applications such as file storage, database storage, and virtual machine file system volumes. The identification of said installed software application 505 may be processed, analyzed, and communicated to the method of operating a scanning system 101 for processing, cataloging, and protection through encryption and various methods of layered cyber defense.

[00187] Further, a non-transitory computer readable medium with a scanning system 101 described herein may include functionality to analyze installed software applications 505 to determine the associated software version 507. The software application version 507 may identify the software version based on unique version name, unique version number, and may be based on unique states of the currently installed computer software.

[00188] One of many embodiments disclosed above may include the non-transitory computer readable medium with a scanning system 101 having the ability to access a data structure of known software vulnerabilities 509 for a plurality of versions of software applications 507. The known software vulnerabilities 509 may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software. The scanning system 101 may be able to access and identify software vulnerabilities for mitigation, rectification, correction, and fortification.

[00189] In one embodiment, a non-transitory computer readable medium with a scanning system 101 may also perform scanning according to scanning system 101 by performing a

lookup of the identified installed software version in the data structure to identify known vulnerabilities 509. This function can be performed by the scanning system 101 according to Fig. 1 performing a query of the installed software for unique version number 307or designator and comparing to, amongst many things, a set of likely or potential vulnerabilities 309 to that software version 307 for potential deficiencies or cybersecurity threats known or suspected to similar software types and versions. This query may be performed according to a predetermined set of values, to include previously identified unique version numbers or designators that may contain the known list of previously identified vulnerabilities 509.

[00190] Among many embodiments, embodiments of the disclosed non-transitory computer readable medium with a scanning system 101 may query the cloud provider API 501 to determine network accessibility information 511 related to the workload. In order to accomplish this query of the cloud provider API 501, the scanning system 101 may involve index of search results and display of said search results, followed by processing and grouping search results. Network accessibility information 511 may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, or Hadoop nodes, among others.

[00191] If the system detects a vulnerable application, one embodiment may identify one or more ports 515 on which said vulnerable application is accessible. In one of several embodiments, the scanning system 101 may detect a vulnerable application in one or more computation processes. In another embodiment, the scanning system 101 may perform a network accessibility query in a separate process. Further, a disclosed embodiment may perform these separate functions in subsequent and sequential steps of the same process. A person having skill would understand that an authorized user or an authorized cybersecurity system can perform these functions concurrently and subsequently while performing the same function as the disclosed embodiment.

[00192] Upon gathering network accessibility information 511 and the identified port 515 to identify one or more vulnerabilities susceptible to attack from outside the workload, a disclosed embodiment would have the functionality to perform processes to gather, display, and mitigate a discovered vulnerability in order to minimize the likelihood and effectiveness of a cyber threat outside of and attempting to access a workload through a known or previously encountered type of cyber threat. This functionality may include collecting and organizing the vulnerabilities according to type or category of vulnerability, displaying the gathered data for an end user or maintainer, and implementing security features automatically or manually by a user or maintainer such as security patches, password or passcode changes and suggestions for users to do the same, and malicious code eradication.

[00193] Another disclosed embodiment may include a method of using a cloud provider API 501, accessing block storage volume 503 of a workload maintained in a cloud storage environment. The method may include manual user operation, automated system operation, systematic and random operating parameters for the system to provide its cybersecurity and similar security functions. The accessed block storage volume may be contained as a collection of block units organized together, may be a series of individual blocks that can be reorganized to form a new storage volume, and may have the ability to be disaggregated and reaggregated as necessary to accomplish its storage and cybersecurity functions.

[00194] The disclosed method may further comprise a system that may analyze the identified installed software application 505 to determine an associated software version 507. In analyzing an installed software application 305, the disclosed embodiment may include a method of querying the installed software application to access a software version 507 that may include a series of letters, numbers, and other identifying characters to differentiate the version of software currently operative and its associated identifying characteristics and capabilities.

[00195] A method of the disclosed may also include accessing a data structure of known software vulnerabilities 509 for a plurality of versions of software applications 507. The disclosed method may determine a data structure of known software vulnerabilities 509 of one among many operative versions of software installed in the monitored storage system as compared to historical data of similar storage systems.

[00196] The disclosed method as described herein may further include performing a lookup of the identified installed software version 507 in the data structure to identify known vulnerabilities 509. The lookup of identified installed software 507 may be previously indexed based on unique version identifier, and the indexed information may include known vulnerability information 509, network accessibility information 511, network protocols, and administrator identifier information. Based on the said indexed known vulnerabilities 509, the scanning system 101 disclosed may take actions to mitigate a cybersecurity threat and reinforce cybersecurity defenses at the type of cybersecurity threat, as previously discussed.

[00197] Further to the above, a disclosed method may also include querying the cloud provider API 501 to determine network accessibility information 511 related to the workload. A query of the cloud provider API 501 to determine network accessibility information 511 related to the workload may be automatically or manually initiated. Network accessibility information 511 similar to the above embodiment may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, and Hadoop nodes, among others.

[00198] A disclosed method of one embodiment also may provide a method of identifying at least one port 515 on which the vulnerable application is accessible. The identification of one port 515 on which the vulnerable application is accessible may be for the purpose of identifying

a cyber weakness, and it may include the ability to change the port currently accessible to mitigate the said cyber weakness.

[00199] Yet another disclosed embodiment of a disclosed method may include using the network accessibility information 511 and the identified at least one port 515 to identify one or more vulnerabilities 513 susceptible to attack from outside the workload. The vulnerability 513 identified as susceptible to attack from outside the workload may include, among others, missing data encryption, OS command injection, SQL injection, buffer overflow, missing authentication, missing authorization, unrestricted upload of dangerous file types, reliance on untrusted inputs in a security decision, cross-site scripting and forgery, download of codes without integrity checks, broken algorithms, URL redirection, path traversal, software bugs, weak passwords, and previously infected software.

[00200] A method of the disclosed embodiment may further include implementing a remedial action in response to the identified one or more vulnerabilities 513. Said remedial action may include, among other things, notification to an end user of an identified threat, compensation through a cybersecurity patch, publish of the identified threat and vulnerability in a log or record of detected vulnerabilities, and communication of the sensed vulnerability and threat to a server operator and maintainer to fortify the protections of workloads existing on similar environments.

[00201] The disclosed method may also include wherein the remedial measure includes transmitting an alert to a device associated with an administrator. Said alert may be, amongst others, written, auditory, and visual for processing and use by an administrator of said cybersecurity system. Said alert may further be logged and catalogued for future identification of known threats and vulnerabilities to similar software application versions.

[00202] A disclosed method of the present embodiment may also provide querying the cloud provider API 501 to determine the network accessibility information 311 related to the workload and examining data sources associated with the workload. A query of the cloud provider API 501 to determine network accessibility information 511 related to the workload may be automatically or manually initiated. Network accessibility information 511 similar to the above embodiment may include connection quality, alternative paths between nodes in a network, and the ability to avoid blockage in said networks. The workloads associated with this query may include applications, services, capabilities, and specific processes such as virtual machines, databases, containers, and Hadoop nodes, among others.

[00203] The method of the above disclosed embodiment may further determine network accessibility information 511 by determining the network accessibility information 511 based on the examined data sources. The data examined may include user data, system processing data, accessibility data, clock cycles, storage input/output, and similar processes. A query of the cloud provider API 501 may be automated or manually-initiated. Based on the said query, the

network accessibility information 511 related to the workload may change based on the data sources associated with said workload.

[00204] In the method of one disclosed embodiment, network accessibility information 511 may include at least one of data from an external data source, cloud provider information, or at least one network capture log. These embodiments may include data from the operating environment of the cloud-based environment, an external operating system for a computer processing unit, or other similar computer readable media. Network capture logs may be automated or manually updated to include possible vulnerabilities and threats to the cloud-based storage medium.

[00205] A method of the disclosed embodiment may also include identifying the installed software application 507 by extracting data from at least one of OS packages, libraries, or program language libraries. This data may be extracted through a system query, random access algorithm, or similar automated process. Operating system packages may include systems operable on Microsoft, Apple, Linux, and similar operating systems. Libraries may consist of a series of files, folders, and databases of information stored on one of any indexed data repositories. A program language library may contain several of an exemplary program languages including but not limited to Javascript, Swift, Scala, Go, Python, Elm, Ruby, C#, C++ and other similar sources of software code.

[00206] Another method of the disclosed embodiment may identify the installed software application 507 based on the extracted data. The method of the processor 500 may perform only this function or this function among many to accomplish the layered cybersecurity defense described herein this disclosed cloud-based security environment.

[00207] A further disclosed method of the present embodiment may include at least one processor 500 further configured to identify a version 507 of the installed software application 505. The method of the processor may perform only this function or this function among many to accomplish the layered cybersecurity defense described herein this disclosed cloud-based security environment.

[00208] **Prioritization Techniques**

[00209] Fig. 6 is a block diagram of method 600 for risk prioritization, consistent with disclosed embodiments.

[00210] Aspects of this disclosure may include a cloud-based cybersecurity system for assessing internet exposure of a cloud-based workload with embodiments configured to access at least one cloud provider API to determine a plurality of entities capable of routing and/or filtering traffic in a virtual cloud environment associated with a target account containing the workload. Routing traffic, as used herein, may refer to the process of selecting a path for traffic (e.g., cells, blocks, frames, packets, calls, messages, or other units of data) in a network or between or across multiple networks. The cloud-based cybersecurity system, in some

embodiments, may comprise at least one processor configured to assess internet exposure as described herein.

[00211] A cloud provider API, as used herein, may refer to any application program interface that allows the end user to interact with a cloud provider's service. Workloads, as used herein, may refer to devices in a network. By way of example, workloads may include systems, devices, resources in a cloud infrastructure. By way of example, Fig. 1 illustrates examples of a workload, such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, or load balancer 115. In some embodiments, scanning system 101 of Fig. 1 may perform a scanning operation on any workload (e.g., systems, devices, resources, etc.) in cloud infrastructure 106.

[00212] In some embodiments, scanning system 101 of Fig. 1 may use an API to detect workloads such as virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115 in the source account. In some embodiments, workloads may exist as virtual machines, while in other embodiments, workloads may exist as discrete, physical devices. For example, in some embodiments, a layer 3 routing system could be implemented as a physical device (e.g., a router or switch) or a virtual device (e.g., a routing system instantiated as a virtual device on a computer).

[00213] Scanning system 101 may query devices and systems capable of routing and filtering traffic (e.g., load balancer 115, routers, switches, firewalls, security groups, API Gateways and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations. These devices can be cloud native and / or unmanaged (e.g. NGINX proxy)

[00214] A cloud environment, as used herein, may refer to a platform implemented on, hosted on, and/or accessing servers that are accessed over the Internet. A scanner account, as used herein, may refer to any type of account associated with a scanner. A scanner, as used herein, may refer to a device for examining, reading, or monitoring something. By way of example, Fig. 1 illustrates one example of a scanner, scanning system 101 of Fig. 1. A target account containing the workload, as used herein, may refer to a location on a network server selected as an aim of connection, for example. In some embodiments, scanning system 101 of Fig. 1 may determine entities capable of routing traffic.

[00215] In some embodiments, the plurality of entities includes a virtual network appliance. A virtual network appliance, as used herein, may refer to a machine focused on virtualizing network functionality. In some embodiments, a typical network virtual appliance may include functionality directed to various layer four to seven. That is, such appliances may implement functionality associated with OSI model layers 4-7 (Transport, Session, Presentation, and Application). Such functionality may include, for example, a firewall, WAN optimizer, application delivery controllers, routers, load balancers, IDS/IPS, proxies, or SD-WAN edge

device. In some embodiments, scanning system 101 of Fig. 1 may determine a virtual network appliance capable of routing traffic.

[00216] In some embodiments, the virtual network appliance is at least one of a load balancer, a firewall, a proxy, or a router. A load balancer, as used herein, may refer to any device that acts as a reverse proxy and distributes network or application traffic across a number of servers. In some embodiments, load balancer 115 may comprise one or more systems that balance incoming requests between the different systems and devices of cloud infrastructure 106. For example, load balancer 115 may be configured to determine usage (e.g., processor load, used storage capacity) of systems or devices in cloud infrastructure 106 to assist in determining where to route an incoming request from network 105 to store data, perform processing, or retrieve data.

[00217] Load balancer 115 may be configured to receive an incoming request from user device 102. Upon receipt of the request, load balancer 115 may consult a data store (part of or separate from load balancer 115; not pictured) to determine usage or forecasted usage of various systems or devices in cloud infrastructure 106, and may forward the request to the systems or devices having the lowest usage or forecasted usage. A firewall, as used herein, may refer to a device or system that monitors incoming and outgoing network traffic and permits or blocks data packets based on a set of security rules. A proxy, as used herein, may refer to a device or system that translates traffic between networks or protocols. In some embodiments, a proxy may be implemented as an intermediary a device or system separating end-user devices from destinations devices (e.g., web servers) by routing traffic through them. Proxies (or proxy servers) may be configured to provide varying levels of functionality, security, and privacy depending on use case, needs, or company policy. A router, as used herein, may refer to a device or system that connects a local network to the internet.

[00218] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to query the at least one cloud provider API to determine at least one networking configuration of the entities. Querying, as used herein, may refer to a request for data or information from a resource. A cloud provider API, as used herein, may refer to any application program interface that allows an end user to interact with a cloud provider's service. A cloud provider may include any entity through whom information is accessible over the Internet or other shared network. At least one networking configuration of the entities, as used herein, may refer to a manner in which components are arranged to make up an operating system, networking system, or a computer system. In some embodiments, scanning system 101 of Fig. 1 may send a request to determine at least one networking configuration of the entities.

[00219] In some embodiments, the networking configuration is at least one of a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration.

[00220] A routing configuration, as used herein, may refer to a manner in which components are arranged to make up a routing system. For example, a routing configuration may provide for dynamic or static "routes" between devices on a network or on networks.

[00221] A proxy configuration, as used herein, may refer to a manner in which components are arranged to make up a proxy system. For example, a proxy configuration may provide for a configuration of how a client can route traffic through a central point (e.g., a proxy server), in order to relay multiple clients' requests from a single point on a network.

[00222] A load balancing configuration, as used herein, may refer to a manner in which components are arranged to make up a load balancing system. For example, a loan balancing configuration may define how resources such as uplinks or distributed applications may be utilized by devices on the same (or another) network, in order to balance usage of multiple redundant uplinks or distributed application nodes.

[00223] A firewall configuration, as used herein, may refer to a manner in which components are arranged to make up a firewall system. For example, a firewall configuration may comprise rules that determine what traffic may be forwarded, which devices are permitted to communicate with one another, or what protocols may be used for communication between devices.

[00224] A VPN (virtual private network) configuration, as used herein, may refer to a manner in which components are arranged to make up a VPN system. A VPN may refer to a type of network that extends a private network across a public network and enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network. For example, a VPN configuration may provide rules or other information by which data is sent or received over a single, encrypted route, such that intermediate points cannot determine the content, source, or destination of the data.

[00225] In some embodiments, scanning system 101 of Fig. 1 may send a request to determine at least a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration of the entities.

[00226] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to build a graph connecting the plurality of entities based on the networking configuration. A graph connecting the plurality of entities, as used herein, may refer to a diagram showing the relation between variable quantities (e.g., entities). In some embodiments, the graph may include any data structures that includes the connections from one entity to another. In some embodiments, scanning system 101 of Fig. 1 may construct a graph or diagram linking the plurality of entities based on the networking configuration. In some embodiments, graph includes a data structure sequentially connecting entities. A data structure sequentially connecting entities, as used herein, may refer to an organization, management, and/or storage format that enables efficient access and modification of data such as entities.

[00227] In some embodiments, a graph may include directional vectors indicating directions of dataflow. Directional vectors, as used herein, may refer to a quantity having direction as well as magnitude (indicating, for example, what kinds of traffic are allowed to flow, what ports are open, what protocols are allowed), especially as determining the position of one point in space relative to another. Directions of dataflow, as used herein, may refer to the movement of data through a system comprised of software, hardware or a combination of both along a particular course. In some embodiments, scanning system 101 of Fig. 1 may construct a graph including a data structure connecting entities sequentially or a graph including directional vectors showing the direction of data flow.

[00228] In some embodiments, to build the graph, the at least one processor may be configured to identify individual entities as nodes and connect the nodes. Nodes, as used herein, may refer to a device in a network (e.g., devices in cloud infrastructure 100). Connecting nodes, as used herein, may refer to linking or bonding devices or data points in a larger network. In some embodiments, the graph includes port numbers associated with the workload. Port numbers, as used herein, may refer to a way to identify a specific process to which an internet or other network message is to be forwarded when it arrives at a server and/or which services are running on a server. In some embodiments, the graph includes a path from the at least one source to the workload. A path from the at least one source to the workload, as used herein, may refer to a string of characters used to uniquely identify a location in a directory structure.

[00229] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to access a data structure identifying services publicly accessible via the Internet and capable of serving as an internet proxy. Services capable of serving as an internet proxy, as used herein, may refer to a system or router that provides a gateway between users and the internet (e.g., network 105). In some embodiments, scanning system 101 of Fig. 1 may access a configuration detecting services over the internet and capable of serving as an internet proxy.

[00230] Aspects of this disclosure may include embodiments configured to integrate the identified services into the graph. Integrating identified services into a graph, as used herein, may refer to the act of bringing together smaller components (the identified services) into a single system that functions as one (the graph). For example, scanning system 101 may integrate information on services available in each node into the data structure, associated with each node in the data structure.

[00231] Aspects of this disclosure may include embodiments in which a cloud-based cybersecurity system is configured to traverse the graph to identify at least one source originating via the Internet and reaching the workload. Traversing the graph, as used herein, may refer to a device, such as scanning system 101, accessing the systems represented by the nodes in the graph, to determine which systems are accessible and how. For example,

scanning system 101 may analyze the graph to detect sources coming from the internet and reaching the workload, by accessing a first system, and traversing between systems in a depth-first, breadth-first, or other manner, until a path to an external network (e.g., the Internet) is reached.

[00232] In some embodiments, the at least one source may be a potential malicious source. In some embodiments, the source may include a code in any part of a software system or script that is intended to cause undesired effects, security breaches or damage to a system. (In some embodiments, the Internet at large may be considered to be a potentially malicious source.)

[00233] Aspects of this disclosure may include embodiments configured to output a risk notification associated with the workload. Outputting a risk notification, as used herein, may refer to sending out data such as a message indicating a possibility of danger. In some embodiments, scanning system 101 of Fig. 1 may produce a notification indicating a risk that is related with the workload. Examples of such risk notifications may include, for example, an electronic mail message, a visual alert, an audio alert, an audio-visual alert,

[00234] In some embodiments, the risk notification includes one or more resolution recommendations. Resolution recommendations, as used herein, may refer to a suggestion or proposal as to the best course of action in view of the identified risk.

[00235] **Fig. 6** illustrates a block diagram of method 600 performed by a processor (e.g., a processor associated with scanning system 101), consistent with disclosed embodiments. In some embodiments, the processor may operate on instructions stored in a non-transitory computer readable medium. In some embodiments, the method may include seven (or more or less) steps:

[00236] Block 602: Access at least one cloud provider API to determine a plurality of entities capable of routing traffic in a virtual cloud environment associated with a target account containing the workload. In some embodiments, scanning system 101 of Fig. 1 may determine entities capable of routing traffic, consistent with the above-disclosed embodiments.

[00237] Block 604: Query the at least one cloud provider API to determine at least one networking configuration of the entities. In some embodiments, scanning system 101 of Fig. 1 may send a request to determine at least one networking configuration of the entities, consistent with the above-disclosed embodiments.

[00238] Block 606: Build a graph connecting the plurality of entities based on the networking configuration. In some embodiments, scanning system 101 of Fig. 1 may construct a graph or diagram linking the plurality of entities based on the networking configuration, consistent with the above-disclosed embodiments.

[00239] Block 608: Access a data structure identifying services publicly accessible via the Internet and capable of serving as an internet proxy. In some embodiments, scanning

system 101 of Fig. 1 may access a configuration detecting services over the internet and capable of serving as an internet proxy, consistent with the above-disclosed embodiments.

[00240] Block 610: Integrate the identified services into the graph. In some embodiments, scanning system 101 of Fig. 1 may incorporate the services into the graph, consistent with the above-disclosed embodiments.

[00241] Block 612: Traverse the graph to identify at least one source originating via the Internet and reaching the workload. In some embodiments, scanning system 101 of Fig. 1 may analyze the graph to detect sources coming from the internet and reaching the workload, consistent with the above-disclosed embodiments.

[00242] Block 614: Output a risk notification associated with the workload. In some embodiments, scanning system 101 of Fig. 1 may produce a notification indicating a risk that is related with the workload, consistent with the above-disclosed embodiments.

[00243] **Techniques for Malware Detection Using Secondary System**

[00244] Aspects of this disclosure may provide a technical solution to the challenging technical problem of malware detection on a primary system using a secondary system other than the primary system. In existing technologies, malware detection system may be run on a primary system that itself may be susceptible to malware. The malware may detect existence of the malware detection system and deceive it to avoid being detected. To resolve such a technical problem, a secondary system may be provided to host the malware detection system, which may isolate the malware detection system from potential exposure or influence from the malware that may reside on the primary system. The disclosed technical solutions may increase success rate of malware detection and thus enhance security level of the primary system.

[00245] A cyber security system, as used herein, may refer to a system including any combination of software and hardware for enhancing security of a device, a platform, or another system in a network environment. By way of example, the cyber security may be implemented as a system including scanning system 101 in Fig. 1. As another example, scanning system 101 may include the disclosed cyber security system as a subset. A cloud environment, as used herein, may refer to a computing environment running on a cloud. By way of example, the cloud environment may include or be part of cloud infrastructure 106 in Fig. 1.

[00246] Consistent with disclosed embodiments, at least one processor may be configured to utilize a cloud provider API to access a block storage volume of a workload maintained on a target account in a target system of a cloud storage environment. Utilizing, as used herein, may refer to an operation of using, deploying, enacting, enabling, activating, allocating, invoking, calling, or any operation of putting a thing into use in a computer environment. An API refers to an application programming interface herein. A cloud provider in this disclosure may refer to a cloud service provider (e.g., a service provider of cloud computing, cloud storage, or any services provided to users on demand over a network). A cloud provider API, as used herein, may refer to an API provided, prepared, enabled, activated, programmed,

or written by a cloud provider. A block storage volume in this disclosure may refer to a storage volume in a block storage scheme. In some embodiments, a block storage volume may be connected, disconnected, or reconnected to a system (e.g., a physical computer, a virtual machine, a network interface, a cloud, or any combination of hardware or software modules) without interfering (e.g., shutting down, halting, rebooting, or any manner of interrupting) the operation status of the system or other running tasks on the system. For example, a block storage volume may be implemented as a virtual disk. In some embodiments, a block storage volume can be associated with an account of a user of the system, the system that it is connected to, or both.

[00247] Block storage, as used herein, may refer to a data storage scheme or model in which data is saved to storage media in fixed-sized raw data chunks (referred to as "blocks"). The raw data blocks may contain no metadata. Each block may be associated with a unique address as metadata assigned to the block. Storage blocks may be controlled by an operating system (OS) and may be accessed by a protocol (e.g., iSCSI, Fibre Channel, or Fibre Channel over Ethernet).

[00248] A storage volume, as used herein, may refer to an identifiable unit (e.g., a physical unit or a logical, virtual unit) of data storage. The storage volume may be mounted to a device via an operation system and be configured with a specific file system (e.g., New Technology File System) assigned a system-unique name or number that identifies the storage volume. The storage volume may represent a named, logical area of storage that enables users and applications to access data on the underlying device. By way of example, a storage volume may be a logical disk that represents a named, logical area of a physical storage device (e.g., a hard disk drive, solid-state drive, compact disc read-only memory, digital video disk, floppy disk, or any other type of storage device). In some embodiments, a logical storage volume may span multiple physical storage devices (e.g., hard disks) and appear as a single, contiguous storage area that works like a physical volume. Storage volumes may be flexibly configured, such as being expanded, contracted, mirrored, stripped, or adapted to support multiple disks (e.g., redundant array of independent disks).

[00249] A workload, as used herein, may refer to a specific application, service, capability, or a specific amount of work that can be run on a cloud resource, system, or infrastructure. By way of example, a workload may be a virtual machine, a database, a container, a Hadoop node, an application, a storage object, a load balancer, or an IAM (Identity and Access Management) configuration. A cloud storage environment in this disclosure may refer to a computing environment of a cloud computing model (referred to as "cloud storage") that stores data on the Internet through a cloud computing provider that manages and operates data storage as a service. A cloud storage environment may be managed and controlled by an operating system. The operation system may be susceptible to malware and security vulnerabilities, and may become a primary system (or referred to as a "target system" herein)

that needs malware detection. By way of example, the cloud storage environment may include or be part of cloud infrastructure 106 in Fig. 1.

[00250] A target system of a cloud storage environment in this disclosure may refer to an operating system or a computing system with an installed operating system, which manages and controls the cloud storage environment or a subsystem of the target system. The target system may maintain one or more accounts for users of the cloud storage environment. A target account of the target system in this disclosure may refer to an account of a user of a cloud infrastructure (e.g., a cloud environment or a cloud storage environment).

[00251] By way of example, to utilize the cloud provider API to access the block storage volume (e.g., existing in storage 111A-111D in Fig. 1) of the workload, the at least one processor (e.g., a processor in scanning system 101 of Fig. 1) may communicate with the cloud storage environment (e.g., cloud infrastructure 106 in Fig. 1) via sending and receiving data (e.g., data packets) over a network (e.g., network 105 in Fig. 1), and enable one or more APIs provided by the cloud provider for accessing the block storage volume. For example, the communication and API enabling may be performed through the process of integration as described in step 201 of Fig. 2. In some embodiments, the at least one processor may create a connection (e.g., by logging in) between a first account on scanning system 101 and a second account on cloud infrastructure 106, then cause the second account to authorize access privileges to the first account via the cloud provider API for accessing storage 111A-111D in cloud infrastructure 106.

[00252] Consistent with disclosed embodiments, the at least one processor may be configured to utilize a scanner at a location of the block storage volume and on a secondary system other than the target system. A scanner, as used herein, may refer to an application, a program, a service, a process, a thread, a function, or any executable codes for performing a scanning process on a computer system to obtain information (e.g., related to structures, vulnerabilities, security issues, or any information of the computer system).

[00253] A location of the block storage volume in this disclosure may refer to a system (e.g., a physical computer, a virtual machine, a network interface, a cloud, a data center, or any combination of hardware or software modules) or an identifier associated with the system that the block storage volume is connected to. In some embodiments, the location of the block storage volume may include at least one of the target account, a secondary system account, a cloud provider account, or a third party account. By way of example, the secondary system account may be an account connected with the target account (e.g., connected by an integration process similar to step 201 in Fig. 2A). The cloud provider account may be an account connected to at least one of the target account or the secondary system account. The third party account may be an account hold by a third party and connected to at least one of the target account, the secondary system account, or the cloud provider account.

[00254] In some embodiments, the at least one processor may determine the location of the block storage volume based on at least one of the target account, a secondary system account, a cloud provider account, or a third party account. By way of example, to determine the location of the block storage volume, the at least one processor may consult a lookup table, a list, or a database that stores a relationship record between an account (e.g., a target account, a secondary system account, a cloud provider account, or a third party account) and a system (e.g., a physical computer, a virtual machine, a network interface, a cloud, or any combination of hardware or software modules) that the block storage volume is connected to. Based on the relationship record, the at least one processor may identify the system as the location using the account as a key.

[00255] A secondary system, as used herein, may refer to a computer system running at an environment or conditions not affected by another system (referred to as a "primary system" or "target system" herein). In some embodiments, the secondary system may have an operating system different from an operating system of the target account. By way of example, the target account may have a WINDOWS® operating system, and the secondary system may have a LINUX® operating system.

[00256] In some embodiments, the secondary system includes at least one of a virtual machine (e.g., any of virtual machines 107A-107D in Fig. 1), a container, or a serverless function. A container, as used herein, may refer to an OS-level virtualization instance assigned with resources by a non-virtual computer operating system (OS). A non-virtual computer operating system (OS) may create multiple isolated user space instances that may function like real computers from the point of view of programs running in them. A computer program running on a non-virtual operating system can access all available resources (e.g., connected devices, files, folders, network shares, CPU power, or any other software or hardware capabilities) of the computer. In contrast, a computer program running inside of a container can only access the resources assigned to the container.

[00257] A serverless function, as used herein, may refer to a computer function hosted in a cloud environment that may allocate resources on demand to perform a function (e.g., scanning) for a user of the cloud environment. The user of the serverless function may be free from concerns of capacity planning, configuration, management, maintenance, fault tolerance, or scaling of containers, virtual machines, or physical servers. When a serverless function is not running, no computing resources may be allocated to it (e.g., using no provisioned server, thus named "serverless"). By way of example, a serverless function may be implemented as any combination of Lambda functions, event sources, and other computing resources. A serverless function may be invoked by the user of a cloud infrastructure (e.g., a cloud environment or a cloud environment).

[00258] By way of example, to utilize the scanner at the location of the block storage volume (e.g., existing in storage 111A-111D in Fig. 1), the at least one processor (e.g., a

processor in scanning system 101 of Fig. 1) may communicate with the secondary system (e.g., one or more of virtual machines 107A-107D in Fig. 1) other than the target system (e.g., the main system operating and managing storage 111A-111D) and activate the secondary system for scanning. For example, to activate the secondary system, the at least one processor may transmit a scanner (e.g., in the form of a computer program or executable codes) to the secondary system, and the secondary system may allocate computing resources (e.g., CPU powers, memory space, network ports, or any other software or hardware resources) in preparation of running the scanner. The scanner may be programmed to be stored at the location of the block storage volume or to start the scanning at the location of the block storage volume. For example, the block storage volume may be connected to a particular system in cloud infrastructure 106 in Fig. 1, and the scanner may be stored at that system (e.g., on storage connected to the system). When the scanner begins scanning, the scanner may start the scanning at that system before scanning other systems.

[00259] In some embodiments, to utilize the scanner, the at least one processor may suspend an operation of the scanner after the scan of the block storage volume. Suspending, as used herein, may refer to an operation of pausing, stopping, ceasing, freezing, halting, hanging, terminating, killing, or any manner or temporarily or permanently setting aside an ongoing process or computer program. By way of example, after the scanner completes scanning the block storage volume, the at least one processor may detect the completion by an active manner (e.g., by periodically polling the secondary system for a status of the scanning) or a passive manner (e.g., by receiving a message from the secondary system indicating the completion of the scanning). The at least one process may then send data to the secondary system, the data including a signal or computer-executable codes for suspending the operation of the scanner. After receiving the data, the secondary system may suspend the scanner and release the computing resources previously allocated to the scanner.

[00260] In some embodiments, to utilize the scanner, the at least one processor may modify a pre-utilized scanner at the location of the block storage volume based on information related to the target account to obtain a modified scanner, and then utilize the modified scanner. A pre-utilized scanner in this disclosure may refer to a scanner that has been previously utilized (e.g., on the same secondary system) and not destroyed. In some embodiments, the information related to the target account may include data or parameters indicating a status or a change of the target account, such as, for example, newly created files, deleted files, changed files, or any data indicating a state of the target account. For example, the pre-utilized scanner at the block storage volume may use a configuration file. The configuration file may include data or parameters indicating a previous status (e.g., list of existing files, public network ports, Internet-accessible data channels, or any operation state parameters) of the target account. The information related to the target account may include data indicating a current status of the target system. The current status of the target system may be different from the previous status

of the target system (e.g., including changed files, changed public network ports, changed Internet-accessible data channels, or any changed operation state parameters). The at least one processor may modify the pre-utilized scanner by modifying the configuration file to reflect the current status of the target system. Then, the at least one processor may utilize the modified scanner.

[00261] Consistent with disclosed embodiments, the at least one processor may be configured to scan the block storage volume for malicious code, using the secondary system. Scanning, as used herein, may refer to an operation of traversing, checking, inspecting, reading, examining, searching, or any manner of looking up. Malicious code, as used herein, may refer to an application, a computer program, a service, a process, a thread, a function, a script, or any type of executable codes that may be executed on a computer without authorization and with a malicious or harmful intent (e.g., manipulating the computer into dangerous behaviors, writing changes or add-ons to existing computer programs or files, stealing computer files, copying sensitive information without authorization, damaging the computer itself, disabling one or more functions of the computer, or attacking another computer using the computer as a disguise). For example, malicious code may include attack scripts, computer viruses, computer worms, Trojan horses, backdoors, or any other malicious executable content. In some embodiments, the malicious code may include a rootkit. A rootkit in this disclosure may refer to any type of computer software designed to enable access to a computer or an area of its software without authorization.

[00262] By way of example, to scan the block storage volume (e.g., existing in storage 111A-111D in Fig. 1) for malicious code, the secondary system (e.g., one or more of virtual machines 107A-107D in Fig. 1) may run the utilized scanner with allocated computing resources to look up databases (e.g., a blacklist) of known malicious files, look up databases storing patterns of malicious files and identify the patterns, look up databases of cryptographic hashes that identify malicious files, run potential malicious code in a sandbox (e.g., a confined and isolated environment for running programs without causing changes outside the confined and isolated environment), determine whether there are scanned files that are not in a database (e.g., a whitelist) of known good files, detect changes over time in the scanned environment, or any combination thereof.

[00263] In some embodiments, scanning the block storage volume may include scanning disk-backed memory. A disk-backed memory in this disclosure may refer to a storage space implemented on a disk (e.g., a hard disk) and functioning like a volatile memory from the perspective of a running program. By way of example, scanning the block storage volume may include scanning files stored in the disk-backed memory. In some embodiments, the disk-backed memory may include at least one of a page file or a cache file. A page file, as used herein, may refer to a file containing one or more pages (also referred to as “memory pages” or “virtual pages”), each of the one or more pages being a fixed-length contiguous block of virtual

memory and described by a single entry in a page table. A page file may be stored on a disk (e.g., a hard disk) for memory management of a virtual memory used by an operating system. A cache file, as used herein, may refer to a file storing data so that future requests for the stored data can be served faster. The data stored in a cache file may be the result of a computation or a copy of the data stored elsewhere.

[00264] Consistent with disclosed embodiments, the at least one processor may be configured to identify malicious code based on the scan. By way of example, if the secondary system scans the block storage volume by consulting databases of known malicious files and finds a scanned file matches a record in the databases, the at least one processor may identify the matched scanned file as the malicious code. In another example, if the secondary system scans the block storage volume by consulting databases storing patterns of malicious files and identifies a pattern of a scanned file matches one or more of the stored patterns, the at least one processor may identify the scanned file as the malicious code. In yet another example, if the secondary system scan the block storage volume by consulting databases of cryptographic hashes that identify malicious files and finds that a cryptographic hash of a scanned file matches one or more of the cryptographic hashes that identify malicious files, the at least one processor may identify the scanned file as the malicious code.

[00265] In yet another example, if the secondary system scans the block storage volume by running potential malicious code in a sandbox and finds that running a scanned file causes malicious or harmful changes or behavior (e.g., an attempt to modify a system setting, an attempt to access files outside of a set of directories, an attempt to perform a network mapping process), the at least one processor may identify the scanned file as the malicious code. In yet another example, if the secondary system scans the block storage volume to determine whether there are scanned files that are not in a database of known good files and finds that a scanned file is not in the database of known good files, the at least one processor may identify the scanned file as the malicious code. As yet another example, if the secondary system scans the block storage volume to detect changes over time in the scanned environment and finding that a scanned file causes a malicious or harmful change in the scanned environment over time, the at least one processor may identify the scanned file as the malicious code. It should be noted that the identification of the malicious code based on the scan may be implemented in various manners and is not limited to the examples described herein.

[00266] Consistent with disclosed embodiments, the at least one processor may be configured to output from the secondary system, a notification of a presence of malicious code in the target system. In some embodiments, the notification may be a pop-up window, a pop-pup dialog, a flashing symbol, an email, a text message, a mobile application notice, a phone call, a warning sound, a tactile feedback (e.g., a vibration), or any type of communications. By way of example, the secondary system may trigger a communication channel between the secondary system and the target system and feed one or more parameters to the

communication channel, in which the one or more parameters represent the presence of the malicious code. The secondary system may then send data representing the notification via the triggered communication channel including the one or more parameters to the target system. After receiving the data, the target system may parse the data and identify the one or more parameters. The target system may then display the notification (e.g., as a pop-up window on a screen) that present the presence of the malicious code.

[00267] Consistent with disclosed embodiments, the at least one processor may be further configured to identify a source of the malicious code. The notification of the presence of malicious code outputted from the secondary system in the target system may include information related to the identified source of the malicious code. In some embodiments, the source of the malicious code may include identity of a developer or distributor of the malicious code, the landing location (e.g., a network port or a vulnerable access point) where the malicious code is imported to the target system, source codes of the malicious code, or any information or data indicating the origination of the malicious code. The information related to the identified source of the malicious code may include a text, a picture, a symbol, a video, a link, a voice, or any type of data for presenting and describing the identified source of the malicious code.

[00268] By way of example, **Fig. 7** is a block diagram illustrating an exemplary process 300 of cyber security scanning for a cloud environment, consistent with the disclosed embodiments. While the block diagram may be described below in connection with certain implementation embodiments presented in other figures, those implementations are provided for illustrative purposes only, and are not intended to serve as a limitation on the block diagram. For example, the steps in Fig. 7 may be executed in any order, steps may be duplicated, or steps may be omitted. In some embodiments, the process 300 may be performed by at least one processor (e.g., a CPU) of a computing device or system (e.g., scanning system 101 in Fig. 1) to perform operations or functions described herein, and may be described hereinafter with reference to Figs. 1-2D by way of example. In some embodiments, some aspects of the process 300 may be implemented as software (e.g., program codes or instructions) that are stored in a memory or a non-transitory computer-readable medium. In some embodiments, some aspects of the process 300 may be implemented as hardware (e.g., a specific-purpose circuit). In some embodiments, the process 700 may be implemented as a combination of software and hardware.

[00269] Fig. 7 includes process blocks 702-710. At block 702, at least one processor may utilize a cloud provider API to access a block storage volume of a workload maintained on a target account in a target system of a cloud storage environment.

[00270] At block 704, the at least one processor may utilize a scanner at a location of the block storage volume and on a secondary system other than the target system. In some embodiments, the secondary system may include at least one of a virtual machine, a container,

or a serverless function. In some embodiments, the secondary system may have an operating system different from an operating system of the target account.

[00271] In some embodiments, to utilize the scanner, the at least one processor may suspend an operation of the scanner after the scan of the block storage volume. In some embodiments, to utilize the scanner, the at least one processor may modify a pre-utilized scanner at the location of the block storage volume based on information related to the target account to obtain a modified scanner, and the utilize the modified scanner.

[00272] At block 706, the at least one processor may scan the block storage volume for malicious code, using the secondary system. In some embodiments, to scan the block storage volume, the at least one processor may scan disk-backed memory. For example, the disk-backed memory includes at least one of a page file or a cache file. In some embodiments, the malicious code includes a rootkit.

[00273] At block 708, the at least one processor may identify malicious code based on the scan. At block 710, the at least one processor may output from the secondary system, a notification of a presence of malicious code in the target system.

[00274] Consistent with disclosed embodiments, besides blocks 702-710, the at least one processor may further determine the location of the block storage volume based on at least one of the target account, a secondary system account, a cloud provider account, or a third party account.

[00275] Consistent with disclosed embodiments, besides blocks 302-310, the at least one processor may further identify a source of the malicious code. The notification outputted in block 710 may include information related to the identified source of the malicious code.

[00276] **Forward and Rearward Facing Attack Vector Visualization Techniques**

[00277] **Fig. 8** is a schematic block diagram illustrating an exemplary embodiment of a system for performing visualization of forward and backward facing threats.

[00278] In one of many embodiments, the system may include a graphical user interface system for providing forward and backward facing attack vector visualizations. A graphical user interface system may include systems similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows 10 Mobile, PalmOS-Web OS, or Firefox OS. Forward facing attack vectors may be those external entities that may access an analyzed asset. Backward facing attack vectors may include the possible impacts to an analyzed asset and those impacts' possible effect. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of attack vectors.

[00279] In an embodiment, the system may comprise at least one processor configured to identify assets 801 in a cloud environment. The at least one processor may include a processor such as user device 102 of Fig. 1. The at least one processor may include a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a

cloud-based storage volume. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service.

[00280] Another embodiment contemplated may consist of at least one processor configured to identify risks 803 associated with each of the identified asset. The identified risks 803 associated with each of the identified assets may be listed sequentially based on probability of risk, severity of risk, or similar method of ordering the risks. The identified risks 303 associated with an identified asset may be scored based on the level of risk to the asset and may be categorized as a high risk, medium risk, or low risk. Once the level of risk can be determined by scanning system 101, the level of risk can be conveyed and displayed to a user device 102 with a representative graphical display.

[00281] An exemplar embodiment may additionally comprise at least one processor configured to identify relationships 805 between at least some of the identified assets, the relationships including at least one of a trust 807A, a network connectivity 807B, or a mechanism of network proxying 807C. A trust 807A may establish a relationship and may involve cryptography, digital signatures, electronic certificates, or similar method of establishing a trusting relationship. A network connectivity 807B may include wireless or wired connections and may be established once a relationship of trust 807A is established. A mechanism of network proxying 807C can be accommodated once trust 807A and network connectivity 807B are established through any number of connected servers to accomplish proxying.

[00282] A further embodiment may also comprise a processor configured to receive an identification 809 of a specific asset under investigation. The specific asset under investigation may be a physical or virtual machine that has software operating a specific version that is vulnerable to any number of associated risks. The identification 809 of a specific asset under investigation may involve an IP address, universally unique identifier, or similar system to uniquely or semi-uniquely identify the specific asset under investigation.

[00283] A system among several embodiments may include a processor configured to perform a forward analysis 811A of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment. A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. The analysis performed by scanning system 101 may determine the origin of the forward threat, the nature of the threat to the analyzed system, and the seriousness of the threat originating from the Internet. For example, scanning system 101 may identify at least one possible attack vector to the workload currently being analyzed. The network outside

of the cloud environment may identify new or existing threats and attack vectors currently threatening the associated system and workload.

[00284] An exemplary embodiment may further comprise a processor configured to perform a backward analysis 811B of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset. A backward analysis 811B of the associated machine may look downstream to related workloads and systems connected to the analyzed systems. A scan downstream may involve, among several embodiments, the downstream exposure risk including an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset. The at least one exposure risk may include one or more entry point to access the exposed asset and any related machines potentially vulnerable to attack from the analyzed and exposed asset.

[00285] Another embodiment may comprise a processor configured to output a signal 813 to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points. The display of a forward or backward path to a specific asset may be one of several visual displays such as linear, graphical, or through computer-generated images. The visualization of a plurality of entry points and lateral movement risks with the plurality of entry points may include visualization of the plurality of paths may sequence the paths based on likelihood of access via that path or the severity of potential threat in a path of access to an asset.

[00286] An example embodiment may comprise a system wherein the network outside the cloud environment includes the Internet. The network outside the cloud environment may include access to the Internet in order for the system to access a plurality of analyzed and potentially vulnerable assets.

[00287] Another embodiment may comprise a system wherein the assets in the cloud environment include at least one of a virtual machine, a network appliance, a storage appliance, a compute instance, or an engine instance. A virtual machine may include an emulated version of a computer - including an operating system, memory, storage, graphics processing - such that it can be indistinguishable from a standard (non-virtual) machine to a running program. A network appliance may include a plurality of servers connected through the Internet in order to establish a system and network of stored assets and workloads. A storage appliance may be comprised by a server designed to store workloads and assets to be analyzed. A compute instance may involve input and processing of data entries that cause the analyzed asset to be accessed and run. An engine instance may include a search algorithm run to analyze and

search the analyzed workloads or virtual machines to determine the software and data stored on the said virtual machine or stored instance.

[00288] One among many embodiments may comprise a system wherein identifying the assets in a cloud environment includes identifying the assets based on at least one of an identity and access management policy, an organization policy, or an access policy. An identity and access management policy may comprise a software tool or similar method of establishing identity of users attempting to access a system to verify identities of users and control access to an associated workload or system functionality. An organization policy may include a policy applied to all users by the controlling organization of a workload. An access policy may determine the authorized users of an associated system.

[00289] One possible embodiment may comprise a system wherein the presentation of the forward and backward paths indicates alternative paths connecting between the specific asset and an upstream asset or a downstream asset. The indication will be via a visualization showing specific alternative paths connecting specific assets or it may visualize a plurality of alternative paths connecting specific assets. Alternative paths connecting between the specific asset and an upstream asset or a downstream asset may involve a direct connection, such as via the Internet, or an indirect connection via one or more series of servers acting as proxy servers between the analyzed asset and the upstream or downstream asset.

[00290] An exemplary embodiment may further comprise a system wherein the visualization includes a presentation of the alternative paths. The alternative paths may be presented visually through a graphical representation, sequential list of alternative paths, or a computer-generated image that represents the plurality of alternative paths.

[00291] Another embodiment may comprise a system wherein the presentation of the forward and backward paths indicates port numbers for each pathway. The port numbers may be indicated by a series of alphanumeric characters (e.g., “HTTP” or “80”) or other identifier.

[00292] One among several exemplary embodiments may further comprise a system wherein the visualization of the entry points indicates at least one entry point of risk. The visualization may indicate one or more points of entry to a given analyzed system. An entry point of risk may involve a port of access to the workload, and may involve a one or more entry points visualized for use by user device 102.

[00293] Another exemplary embodiment may comprise a system wherein the at least one processor is further configured to monitor network activities of the assets in a cloud environment. The processor may monitor continuously, intermittently, or as directed by an authorized user to direct monitoring of the analyzed network activities of the assets. The cloud environment may be, among several embodiments, the cloud infrastructure 106.

[00294] An embodiment contemplated may comprise a system further configured to detect a potential risk associated with the specific asset based on a network activity of the specific asset. The potential risk may be associated to a specific asset based on historical

threats to the specific asset or a list of possible threats to the specific asset based on its current network activities.

[00295] Another embodiment may comprise a system further configured to detect a potential risk associated with the specific asset based on a network activity of an upstream asset of the specific asset. The potential risk associated with the specific asset may be identified by scanning system 101 and determine whether a potential risk can be classified as high risk, medium risk, or low risk. The potential risk may be identified based on the specific asset, the network activity performed by the specific asset, or the specific asset type or related software of the upstream asset.

[00296] One of many embodiments may comprise a system further configured to detect a potential risk associated with the specific asset based on a network activity of a downstream asset of the specific asset. The potential risk may be identified based on the specific asset, the network activity performed by the specific asset, or the specific asset type or related software of the upstream asset. For example, scanning system 101 may perform a scan on an asset being analyzed, and based on the one or more entry points of risk, may determine if the downstream asset of the specific asset poses a potential risk to the analyzed system.

[00297] Another exemplary embodiment may comprise a method for generating a graphical user interface for providing forward and backward facing attack vector visualizations. The method may include a method of generating a graphical user interface similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows 10 Mobile, PalmOS-Web OS, or Firefox OS. A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. Backward analysis 811B may include the possible impacts to an analyzed asset and those impacts' possible effect. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of attack vectors.

[00298] One exemplar embodiment may further comprise a method of identifying assets 801 in a cloud environment. A method may include at least one processor that may include a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service.

[00299] One of several embodiments may comprise a method of identifying risks 803 associated with each of the identified assets 801. The method of identifying risks 803 associated with each of the identified assets 801 may be listed sequentially based on probability of risk, severity of risk, or similar method of ordering the risks. The identified risks 803

associated with an identified asset may be scored based on the level of risk to the asset and may be categorized as a high risk, medium risk, or low risk. Once the level of risk can be determined by scanning system 101, the level of risk can be conveyed and displayed to a user device 102 with a representative graphical display.

[00300] Another embodiment contemplated may comprise a method for identifying relationships 305 between at least some of the identified assets 801, the relationships including at least one of a trust 807A, a network connectivity 807B, or a mechanism of network proxying 807C. A trust 807A may establish a relationship and may involve cryptography, digital signatures, electronic certificates, or similar method of establishing a trusting relationship. A network connectivity 807B may include wireless or wired connections and may be established once a relationship of trust 807A is established. A mechanism of network proxying 807C can be accommodated once trust 807A and network connectivity 807B are established through any number of connected servers to accomplish proxying.

[00301] An embodiment may further comprise a method of receiving an identification of a specific asset 809 under investigation. The specific asset under investigation may be a physical or virtual machine that has software operating a specific version that is vulnerable to any number of associated risks 803. The identification 809 of a specific asset under investigation may involve an IP address, universally unique identifier, or similar system to uniquely or semi-uniquely identify the specific asset under investigation.

[00302] An embodiment may also comprise a method of performing a forward analysis 811A of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment. A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. A forward analysis may involve analysis of the specific asset under identification to identify at least one possible Internet-originating attack vector to the asset. The analysis performed by scanning system 101 may determine the origin of the forward threat, the nature of the threat to the analyzed system, and the seriousness of the threat originating from the Internet. For example, scanning system 101 may identify at least one possible attack vector to the workload currently being analyzed. The network outside of the cloud environment may identify new or existing threats and attack vectors currently threatening the associated system and workload.

[00303] One embodiment may comprise a method of performing a backward analysis 811B of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset. Backward analysis 811B may include the possible impacts to an analyzed asset and those impacts' possible effect. A scan downstream may involve, among several embodiments, the downstream exposure risk including an identification

of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset. The at least one exposure risk may include one or more entry point to access the exposed asset and any related machines potentially vulnerable to attack from the analyzed and exposed asset.

[00304] An embodiment may comprise a method of outputting a signal 813 to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points. The display of a forward or backward path to a specific asset may be one of several visual displays such as linear, graphical, or through computer-generated images. The visualization of a plurality of entry points and lateral movement risks with the plurality of entry points may include visualization of the plurality of paths may sequence the paths based on likelihood of access via that path or the severity of potential threat in a path of access to an asset.

[00305] Another embodiment may comprise a method wherein the network outside the cloud environment includes the Internet. The network outside the cloud environment may include access to the Internet in order for the system to access a plurality of analyzed and potentially vulnerable assets.

[00306] One of several embodiments may comprise a method wherein the assets in the cloud environment include at least one of a virtual machine, a network appliance, a storage appliance, a compute instance, or an engine instance. A virtual machine may include an emulated version of a computer - including an operating system, memory, storage, graphics processing - such that it can be indistinguishable from a standard (non-virtual) machine to a running program. A network appliance may include a plurality of servers connected through the Internet in order to establish a system and network of stored assets and workloads. A storage appliance may be comprised by a server designed to store workloads and assets to be analyzed. A compute instance may involve input and processing of data entries that cause the analyzed asset to be accessed and run. An engine instance may include a search algorithm run to analyze and search the analyzed workloads or virtual machines to determine the software and data stored on the said virtual machine or stored instance.

[00307] An embodiment may further comprise a method wherein identifying assets in a cloud environment includes identifying the assets based on at least one of an identity and access management policy, an organization policy, or an access policy. An identity and access management policy may comprise a software tool to verify identities of users and control access to an associated workload or system functionality. An organization policy may include a policy applied to all users by the controlling organization of a workload. An access policy may determine the authorized users of an associated system.

[00308] An embodiment may also comprise a method wherein the presentation of the forward and backward paths indicates alternative paths connecting between the specific asset

and an upstream or a downstream asset. The indication may be via a visualization showing specific alternative paths connecting specific assets or it may visualize a plurality of alternative paths connecting specific assets.

[00309] Another exemplar embodiment may comprise a method wherein the visualization includes a presentation of the alternative paths. The alternative paths may be presented visually through a graphical representation, sequential list of alternative paths, or a computer-generated image that represents the plurality of alternative paths.

[00310] Another embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations.

[00311] One embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor 800, are configured to perform operations identifying assets 801 in a cloud environment. the instructions stored on a non-transitory computer-readable medium may include at least one processor 800 that may include a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service.

[00312] An embodiment may further comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor 300 to perform operations comprising identifying risks 803 associated with each of the identified asset. The instructions performed of identifying risks 803 associated with each of the identified assets 301 may be listed sequentially based on probability of risk, severity of risk, or similar method of ordering the risks.

[00313] Another embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising identifying relationships 805 between at least some of the identified assets 301, the relationships including at least one of a trust 807A, network connectivity 807B, or a mechanism of network proxying 807C.

[00314] An exemplar embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising receiving an identification of a specific asset 809 under investigation. The instructions may include receiving information identifying the specific asset 809. The specific asset 809 under investigation may be a physical or virtual machine that has software operating a specific version that is vulnerable to any number of associated risks 803.

[00315] One among many embodiments may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are

configured to cause the at least one processor to perform operations comprising performing a forward analysis 811A of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment. . A forward analysis 811A of the specific asset may involve those external entities that may access an analyzed asset. And the scanning system 101 may identify at least one possible attack vector to the workload currently being analyzed. The network outside of the cloud environment may identify new or existing threats and attack vectors currently threatening the associated system and workload.

[00316] An embodiment may further comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising performing a backward analysis 811B of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset. Backward analysis 811B may include the possible impacts to an analyzed asset and those impacts' possible effect. The at least one exposure risk may include one or more entry point to access the exposed asset and any related machines potentially vulnerable to attack from the analyzed and exposed asset.

[00317] An embodiment may comprise a non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising outputting a signal 813 to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points. The display of a forward or backward path to a specific asset may be one of several visual displays such as linear, graphical, or through computer-generated images. The visualization of a plurality of entry points and lateral movement risks with the plurality of entry points may include visualization of the plurality of paths may sequence the paths based on likelihood of access via that path or the severity of potential threat in a path of access to an asset.

[00318] **Passive Key Identification Techniques**

[00319] Aspects of this disclosure may provide a technical solution to the challenging technical problem of identifying access keys (e.g., passwords, Secure Shell (SSH) keys, or cloud keys) to compute resources (e.g., machines, containers, storages, or any hardware or software component) in a cloud environment without using the access keys to access the compute resources. In existing technologies, scanning systems may identify access keys to compute resources and may need to verify the identified keys by actually accessing the compute resources using the access keys. In such cases, the actual accesses may generate logs or records (e.g., for successful or failed accesses) that can constitute activity patterns

similar to an attacker to the system. Such activity patterns may distract a system administrator (e.g., an individual or a computer program) to misidentify the scanning system as an attacker. Further, actual accesses may create loads on both the scanning system and the compute resources and may reduce the computing resources and power for other tasks. To resolve such a technical problem, a cryptographic analysis may be performed to identify a first set of fingerprints of the identified keys, and trust configurations of the compute resources may be analyzed to identify a second set of fingerprints of the compute resources. The first and second set of fingerprints may be compared to match keys with the compute resources without using the keys to access the compute resources. By doing so, the scanning system may perform its function without raising false alarms to the system administrator, while reducing computing costs to the cloud environment.

[00320] A cyber security system, as used herein, may refer to a system including any combination of software and hardware for enhancing security of a device, a platform, or another system in a network environment. By way of example, the cyber security may be implemented as a system including scanning system 101 in Fig. 1. As another example, scanning system 101 may include the disclosed cyber security system as a subset. A compute resource, as used herein, may refer to a database, a virtual machine, a storage, a keystore, a scanning system, a load balancer, a server, a computer, a container, or any physical or virtual component integrated in or communicatively connected to a cloud infrastructure. By way of example, the compute resource may be scanning system 101, any of databases 103A-103D, any of virtual machines 107A-107D, any of databases 109A-109D, any of storage 111A-111D, any of keystores 113A-113D, or load balancer 115 in cloud infrastructure 106 in Fig. 1.

[00321] A key to access a compute resource, as used herein, may refer to computerized data that includes a credential for granting permission to an accessor to access the compute resource, or a service or function provided by the compute resource. By way of example, the key may include at least one of a password, a remote access key, a script containing a password, a cloud key (e.g., a credential file for accessing a cloud service, such as an AWS credential file), an Secure Shell (SSH) key, an AWS key, a private component of a private-public key pair, or any type of credential that provide checked or unchecked access to the compute resource. Matching a key to a compute resource, as used herein, may refer to confirming, verifying, validating, or any operation or process of determining that the key is valid for an accessor (e.g., another compute resource) to access the compute resource. For example, to match a key to a compute resource, an accessor may test by using the key to access the compute resource and determines that the key is matched to the compute resource if such access is successful. As another example, to match a key to a compute resource, an accessor may perform methods or algorithms to verify that the key is matched to the compute resource without using the key to access the compute resource.

[00322] Consistent with disclosed embodiments, at least one processor may be configured to analyze a cloud environment to identify a plurality of keys to the compute resources in the cloud environment. In some embodiments, the at least one processor may analyze the cloud environment to identify the plurality of keys to the compute resources in the cloud environment in any of steps 231, 237, or 239 in FIG. 2D. In some embodiments, the at least one processor may analyze the cloud environment to identify the plurality of keys to the compute resources in the cloud environment in any step described in association with FIGS. 2A-2D. In some embodiments, at least one of the plurality of keys may include at least one of a password (e.g., a combination of alphanumeric characters and symbols), a script (e.g., a PYTHON® script or a shell script) containing a password, a cloud key, or an Secure Shell (SSH) key. For example, a cloud key may be a key for accessing a cloud resource.

[00323] A cloud environment, as used herein, may refer to a computing environment running on a cloud. By way of example, the cloud environment may include or be part of cloud infrastructure 106 in Fig. 1. Analyzing a cloud environment, as used herein, may refer to an operation or a process of separating or dividing the cloud environment into parts or components (e.g., physical components or logic components), then determining nature and relationship of the parts or components based on data associated with the parts or components.

[00324] By way of example, if the cloud environment is cloud infrastructure 106 in Fig. 1, to analyze cloud infrastructure 106, the at least one processor may access cloud infrastructure 106 (e.g., via network 105 or via internal connections in cloud infrastructure 106) and obtain a list of compute resources in cloud infrastructure 106 (e.g., scanning system 101, any of databases 103A-103D, any of virtual machines 107A-107D, any of databases 109A-109D, any of storage 111A-111D, any of keystores 113A-113D, or load balancer 115). For example, the at least one processor may obtain the list of compute resources by reading a configuration file from a database (e.g., one of databases 109A-109D), or visit available compute resources one after another through communicative connections between the compute resources to generate the list of compute resources. After obtaining the list of compute resources in cloud infrastructure 106, the at least one processor may determine the nature of the compute resources, such as their types, numbers, functions, or any other characteristics or features. The at least one processor may further determine the relationship between the compute resources, such as communicative connections, access privileges, data input/output directions, or controls between the compute resources.

[00325] Identifying a key to a compute resource, as used herein, may refer to an operation or a process of locating, recognizing, or any operation or process or analyzing computerized data or information to determine that the computerized data or information is a key to the compute resources. By way of example, the at least one processor may analyze cloud infrastructure 106 in Fig. 1 to recognize a list of compute resources in cloud infrastructure 106 (e.g., scanning system 101, any of databases 103A-103D, any of virtual machines 107A-

107D, any of databases 109A-109D, any of storage 111A-111D, any of keystores 113A-113D, or load balancer 115). Then, the at least one processor may read computerized data stored in a compute resource and identify that some of the computerized data are one or more keys to other compute resources in cloud infrastructure 106. For example, to recognize the keys, the at least one processor may read and compare the computerized data with records stored in keystores 113A-113D, and determine that the computerized data is a key if it matches a record in any of keystores 113A-113D. As another example, the at least one processor may read the computerized data and check its syntaxes, text string patterns, file formats, file properties, encryption manners, library versions, software versions, or any other characteristics or features of the computerized data, and determine that the computerized data is a key if its checked characteristics or features fit a predetermined pattern of a key or fit an entry in a dictionary of keys. For example, the dictionary of keys may be stored in a keystore (e.g., any of keystores 113A-113D).

[00326] In some embodiments, the plurality of keys may be stored in at least one workload. A workload, for example, may refer to a specific application, service, capability, or a specific amount of work that can be run on a cloud resource, system, or infrastructure. By way of example, a workload may be a virtual machine, a database server, a container, a Hadoop node, an application, a storage server, a load balancer, or an IAM (Identity and Access Management) configuration. By way of example, with reference to FIG. 1, cloud infrastructure 106 may include workloads such as scanning system 101, databases 103A-103D, virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115.

[00327] Consistent with disclosed embodiments, at least one processor may be configured to perform a cryptographic analysis on the plurality of keys to identify a first set of fingerprints that uniquely identify each of the plurality of keys. The first set of fingerprints may be non-functional. In some embodiments, the at least one processor may perform the cryptographic analysis in step 237 of FIG. 2D. In some embodiments, the at least one processor may perform the cryptographic analysis in any step described in association with FIGS. 2A-2D.

[00328] A cryptographic analysis on data, as used herein, may refer to an operation or process of identifying ciphertext, ciphers and cryptosystems contained in the data to discover hidden aspects of the data that may improve or weaken security of the data or compute resources associated with the data. In some embodiments, to perform the cryptographic analysis on the plurality of keys, the at least one processor may perform an algorithm on the plurality of keys to convert the plurality of keys into another form of data. For example, the at least one processor may perform a hashing process on the plurality of keys (e.g., by inputting values of the keys to a hash function) to obtain corresponding hash values of respective keys.

[00329] A fingerprint of a key, as used herein, may refer to non-functional data (e.g., a numeric value, an alphanumeric string, or any combination of letters, numbers, or symbols)

generated based on the key and may uniquely identify the key. By way of example, a fingerprint may be generated by performing a hashing process on a key or a part of the key. The hashing process may include, for example, an MD5 algorithm, an SHA-1 algorithm, an SHA-2 algorithm, an SHA-3 algorithm, a RIPEMD-160 algorithm, a BLAKE2 algorithm, a BLAKE3 algorithm, or any type of cryptographic hash algorithms. In such cases, the fingerprint may be a hash value of the key, a part (e.g., a truncated part) of the hash value of the key, a hash value of a part of the key, or a part (e.g., a truncated part) of the hash value of the part of the key. The hash value may be unique (e.g., non-overlap with any hash value of any other key) and non-functional (e.g., unable to be used as a key to access a compute resource). To identify the first set of fingerprints, the at least one processor may perform the hashing process on the plurality of keys to obtain a plurality of hash values, and search the cloud environment to check whether there exists any compute resource that stores the plurality of hash values.

[00330] In some embodiments, at least one of the first set of fingerprints may be non-identical to any key of the plurality of keys. For example, if the first set of fingerprints are hash values, they may be non-identical to the plurality of keys.

[00331] Consistent with disclosed embodiments, at least one processor may be configured to analyze trust configurations of the compute resources to identify a second set of fingerprints for each of the compute resources. In some embodiments, the at least one processor may analyze the trust configurations in step 233 of FIG. 2D. In some embodiments, the at least one processor may analyze the trust configurations in any step described in association with FIGS. 2A-2D.

[00332] A trust configuration, as used herein, may refer to data (e.g., a stored file, a database entry, a value stored in a data structure, or any computerized information) that stores a pre-established trust relationship between at least two compute resources. For example, a trust configuration may be a file storing a trust policy (e.g., a public key infrastructure (PKI), a digital signature system, or an AWS trust policy). Analyzing a trust configuration, as used herein, may refer to an operation or a process of separating or dividing compute resources associated with the trust configuration into units or groups, then determining nature and relationship of the units or groups based on relationship data associated with the units or groups. For example, the relationship data may include flags, indicators, or any type of value that indicates a first compute resource has an access privilege to a second compute resource.

[00333] A fingerprint for a compute resource, as used herein, may refer to non-functional data (e.g., a numeric value, an alphanumeric string, or any combination of letters, numbers, or symbols) generated based on a key for accessing the compute resource and may uniquely identify the key. By way of example, a fingerprint for a compute resource may be generated by applying a cryptographic hash function (e.g., an SHA-1 or SHA-2 function) to a key for accessing the compute resource to obtain a sequence of bytes (e.g., a hash value or a modified

hash value), and the sequence of bytes may be the fingerprint for the compute resource. The key may be uniquely associated with the compute resource.

[00334] For example, to identify the second set of fingerprints, the at least one processor may analyze a trust configuration file (e.g., a file storing fingerprints and their associated data) by reading the byte stream of the trust configuration file to determine whether any sequence of bytes that fits to a predetermined pattern (e.g., an alphanumeric string pattern). The sequences of bytes that fit to the predetermined pattern may be determined as the second set of fingerprints.

[00335] Consistent with disclosed embodiments, at least one processor may be configured to compare the first set of fingerprints with the second set of fingerprints to match keys with the compute resources without using the keys to access the compute resources. In some embodiments, the at least one processor may compare the first set of fingerprints with the second set of fingerprints in any step described in association with FIGS. 2A-2D. Comparing a first fingerprint and a second fingerprint, as used herein, may refer to performing a textual, numerical, or semantic comparison of the first fingerprint and the second fingerprint, or performing an algorithmic conversion of at least one of the first fingerprint or the second fingerprint to further perform a textual, numerical, or semantic comparison of the conversion results. By way of example, a first compute resource may store a key (e.g., a public key) for accessing a second compute resource. The at least one processor may analyze the cloud environment to identify the key from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. A third compute resource (e.g., may be the same as or different from the second compute resource) may store in its trust configurations a second fingerprint of the key. The second fingerprint may be generated by a different processor at a different time. The at least one processor may then analyze the trust configurations of the third compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other. If the first fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource. As can be seen, after comparison, if the first set of fingerprints are matched with the second set of fingerprints, keys may be matched with the compute resources, and the at least one processor needs not use the keys to access the compute resources to confirm that the keys are matched to the compute resources.

[00336] Consistent with disclosed embodiments, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the first set of fingerprints. A multi-machine interaction, as used herein, may refer to data input/output, operation control, status polling, scanning, or any type of an interaction between a plurality of machines in a cloud environment. Analyzing a multi-machine interaction, as used herein, may refer to an operation

or a process of separating or dividing machines in the cloud environment into units or groups, then determining nature and relationship of the units or groups based on relationship data associated with the units or groups. For example, the relationship data may be the first set of fingerprints.

[00337] By way of example, to analyze the multi-machine interaction, the at least one processor may compare the first set of fingerprints with the second set of fingerprints. If a first fingerprint of the first set of fingerprints matches with a second fingerprint of the second set of fingerprints, the at least one processor may determine that a first compute resource associated with the first fingerprint may have an inter-machine interaction (e.g., having an ability to access the other) with a second compute resource associated with the second fingerprint. By way of example, a first compute resource may store a key (e.g., a private key) for accessing a second compute resource. The at least one processor may analyze the cloud environment to identify the key from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. A third compute resource (e.g., may be the same as or different from the second compute resource) may store in its trust configurations a second fingerprint of the key. The at least one processor may then analyze the trust configurations of the third compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other (e.g., being the same). If the first fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource, and may further determine an inter-machine interaction in which the first compute resource has ability to access the second compute resource.

[00338] Consistent with disclosed embodiments, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the plurality of keys. For example, the at least one processor may use the plurality of keys as the relationship data for determining nature and relationship of compute resources in the cloud environment. By way of example, to analyze the multi-machine interaction, the at least one processor may compare a first key for accessing a first one of the compute resources and a second key for accessing a second one of the compute resources. If the first key matches with (e.g., is identical to) the second key, the at least one processor may determine that the first one of the compute resources may have an interaction (e.g., having an ability to access the other) with the second one of the compute resources.

[00339] By way of example, **Fig. 9** is a block diagram illustrating an exemplary process 900 of matching keys with compute resources in a cloud environment, consistent with the disclosed embodiments. While the block diagram may be described below in connection with certain implementation embodiments presented in other figures, those implementations are provided for illustrative purposes only, and are not intended to serve as a limitation on the block

diagram. For example, the steps in Fig. 9 may be executed in any order, steps may be duplicated, or steps may be omitted. In some embodiments, the process 900 may be performed by at least one processor (e.g., a CPU) of a computing device or system (e.g., scanning system 101 in Fig. 1) to perform operations or functions described herein, and may be described hereinafter with reference to Figs. 1-2D by way of example. In some embodiments, some aspects of the process 900 may be implemented as software (e.g., program codes or instructions) that are stored in a memory or a non-transitory computer-readable medium. In some embodiments, some aspects of the process 300 may be implemented as hardware (e.g., a specific-purpose circuit). In some embodiments, the process 300 may be implemented as a combination of software and hardware.

[00340] Fig. 9 includes process blocks 902-908. At block 902, at least one processor may analyze a cloud environment to identify a plurality of keys to the compute resources in the cloud environment. In some embodiments, the plurality of keys may be stored in at least one workload. In some embodiments, at least one of the plurality of keys may include at least one of a password (e.g., a combination of alphanumeric characters and symbols), a script (e.g., a PYTHON® script or a shell script) containing a password, a cloud key, or an Secure Shell (SSH) key.

[00341] At block 904, the at least one processor may perform a cryptographic analysis on the plurality of keys to identify a first set of fingerprints that uniquely identify each of the plurality of keys. The first set of fingerprints may, in some embodiments, be non-functional. In some embodiments, at least one of the first set of fingerprints may be non-identical to the key.

[00342] At block 906, the at least one processor may analyze trust configurations of the compute resources to identify a second set of fingerprints for each of the compute resources. As discussed above, to identify the second set of fingerprints, in some embodiments, the at least one processor may analyze a trust configuration file (e.g., a file storing fingerprints and their associated compute resources) by reading the byte stream of the trust configuration file to determine whether any sequence of bytes that fits to a predetermined pattern (e.g., an alphanumeric string pattern). The sequences of bytes that fit to the predetermined pattern may be determined as the second set of fingerprints.

[00343] At block 908, the at least one processor may compare the first set of fingerprints with the second set of fingerprints to match keys with the compute resources without using the keys to access the compute resources. As discussed above, in some embodiments, a first compute resource may store a key (e.g., a private component of a private-public key pair) to a second compute resource, and the second compute resource may store in its trust configurations a second fingerprint of the key. The at least one processor may analyze the cloud environment to identify the key from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. The at least one processor may then analyze the trust configurations of the second compute resource to identify

the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other (e.g., being identical). If the first fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource.

[00344] Consistent with disclosed embodiments, besides blocks 902-908, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the first set of fingerprints. By way of example, to analyze the multi-machine interaction, the at least one processor may compare the first set of fingerprints with the second set of fingerprints. As discussed above, in some embodiments, a first compute resource may store a key (e.g., a private key) for accessing a second compute resource. The at least one processor may analyze the cloud environment to identify the private component from the first compute resource, then may perform the cryptographic analysis described herein to identify a first fingerprint of the key. A third compute resource (e.g., may be the same as or different from the second compute resource) may store in its trust configurations a second fingerprint of the key. The at least one processor may then analyze the trust configurations of the third compute resource to identify the second fingerprint. The at least one processor may then compare the first fingerprint and the second fingerprint to determine whether they match each other (e.g., being the same). If the first fingerprint matches with the second fingerprint, the at least one processor may determine that the key identified from the first compute resource is a key for accessing the second compute resource, and may further determine an inter-machine interaction in which the first compute resource has ability to access the second compute resource.

[00345] Consistent with disclosed embodiments, besides blocks 902-908, the at least one processor may further analyze a multi-machine interaction in the cloud environment using the plurality of keys. By way of example, to analyze the multi-machine interaction, the at least one processor may compare a first key for accessing a first one of the compute resources and a second key for accessing a second one of the compute resources.

[00346] **Hybrid Ephemeral Scanner Techniques**

[00347] Aspects of this disclosure may include accessing a primary account maintained in a cloud environment. A primary account as used herein, may refer to a principal or main identity created for a person in a computer or computing system. A primary account may also be created for machine entities, such as service accounts for running programs, system accounts for storing system files and processes, and root and administrator accounts for system administration. A cloud environment, as used herein, may refer to a platform implemented on, hosted on, and/or accessing servers that are accessed over the Internet. An example of a cloud environment is cloud infrastructure 106 in Fig. 1. In some embodiments, scanning system 101 of Fig. 1 may gain access to an account within cloud infrastructure 106.

[00348] Aspects of this disclosure may include receiving information defining a structure of the primary account, wherein the structure includes a plurality of assets. Information as used herein, may refer to data received. Structure of the primary account as used herein, may refer to a set up or of the primary account. A plurality of assets as used herein, may refer to one or more data, devices, or components within an organization's system. For example, the plurality of assets may include any of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, load balancer 115, log files or databases, API gateway resources, API gateway REST APIs, Autoscaling groups, CloudTrail logs, CloudFront services, volumes, snapshots, VPCs, subnets, route tables, network ACLs, VPC endpoints, NAT gateways, ELB and ALB, ECR repositories, ECS clusters, services, and tasks, EKS, S3 bucket and Glacier storage, SNS topics, IAM roles, policies, groups, users, KMS keys, and Lambda functions.

[00349] In some embodiments, scanning system 101 of Fig. 1 may receive information relating to the structure of an account. In such an embodiment, the structure of the account may include virtual machines 107A-D and databases 109A-109D, for example. In some embodiments, the information may be acquired from another device, received via a one way communication, received in response to a request, retrieved from a storage device, or generated.

[00350] In some embodiments, the information may exclude raw data of the primary account. Raw data as used herein, may refer to a collection of data as gathered before it has been processed, cleaned, or analyzed. For example, raw data may include usage data, passwords, and cache.

[00351] Aspects of this disclosure may include deploying, inside the primary account, at least one ephemeral scanner configured to scan at least one block storage volume and to output metadata defining the at least one block storage volume. In some embodiments, the output may exclude raw data of the primary account. In some embodiments, the ephemeral scanner may be deployed inside a secondary account for which trust has been established with a primary account. For example, a secondary account on another system may have an existing trust relationship (such as a cloud trust policy) with a first account. The ephemeral scanner may be operated inside of that secondary account to scan block storage associated with the first account. Each account may be hosted on the same system or on different systems.

[00352] Further, in some embodiments, the at least one ephemeral scanner may be configured to periodically scan at least one block storage volume. For example, the scanner may be configured to scan at least one block storage volume once per hour, once per day, once per week, or more or less often.

[00353] Further, in some embodiments, the at least one ephemeral scanner may be periodically deployed to scan at least one block storage volume. For example, the scanner may

be deployed for scanning at least one block storage volume once per hour, once per day, once per week, or more or less often.

[00354] Outputting metadata as used herein, may refer to producing a set of data that describes and gives information about other data. In some embodiments, scanning system 101 of Fig. 1 may scan a block storage volume and output data related to it (e.g., vulnerability information, configuration information, malware information, risk analysis information, and sensitive information). Periodically scanning as used herein, may refer to conducting a scan occurring or recurring at regular intervals. In some embodiments, the ephemeral scanner is configured to perform vulnerability scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for security vulnerabilities on the device. In some embodiments, the ephemeral scanner is configured to perform configuration scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for configuration information related to the device. In some embodiments, the ephemeral scanner may scan the block storage volume in order to look for security issues. In some embodiments, the ephemeral scanner is configured to perform malware scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for malware information related to the device. As discussed above with respect to Fig. 2D, in step 235, scanning system 101 may perform a step of malware scanning. In some embodiments, scanning system 101 may perform malware scanning across all filesystems in the snapshot (e.g., gathered from virtual machines 107A-107D or storage 111A-111D). Scanning system 101 may use multiple malware scanning software solutions to perform a malware scan against the filesystems, including one sourced from another vendor, such as bucketAV, Trend Micro Cloud One, Sophos Cloud Optix, Crowdstrike Falcon CWP, or others. In some embodiments, malware scanning in step 235 comprises utilizing signatures, heuristics, or sandboxing capabilities to deduce whether there is an infection on the machine.

[00355] In some embodiments, the ephemeral scanner is configured to perform lateral-movement risk analysis of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for lateral-movement risk information related to the device. For example, in some embodiments, scanning system 101 may perform a “backward” analysis of the specific asset to identify exposure risk to assets downstream of the specific asset, wherein the downstream exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset.

[00356] Further, as discussed above with respect to Fig. 2D, in step 237, scanning system 101 may perform a step of lateral movement scanning. An attacker who establishes a network foothold usually attempts to move laterally from one resource to another in search of rich targets such as valuable data. Stolen passwords and keys unlock access to servers, files,

and privileged accounts. In some embodiments, scanning system 101 may gather keys from each scanned system or device (e.g., virtual machines 107A-107D or storage 111A-111D). In some embodiments, scanning system 101 searches for passwords, scripts, shell history, repositories, or other data that may contain passwords, cloud access keys, SSH keys, or other key/password/access information that provide unchecked access to important resources. In some embodiments, scanning system 101 searches for such keys/passwords/access information and calculates a "hash" (a mathematical fingerprint) of each string. Scanning system 101 then attempts to match the hashed strings to hashes of strings that that are stored on different systems or devices. This will be used to detect the potential lateral movement between assets.

[00357] In some embodiments, the ephemeral scanner is configured to perform sensitive information scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for sensitive information related to the device. As discussed above with respect to Fig. 2D, in step 241, scanning system 101 may perform a step of sensitive information scanning. In some embodiments, scanning system 101 may search the snapshot for sensitive information, such as personally identifiable information (PII), Social Security numbers, healthcare information, or credit card numbers. In some embodiments, scanning system 101 may search data repository history as well. This is because it is not uncommon for an entire production environment repository to be cloned, with no one remembering the copy contains sensitive information. In some situations, detecting sensitive data not secured is critical in adherence to data privacy regulations.

[00358] In some embodiments, the ephemeral scanner is configured to perform container scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for container information related to the device. As discussed with respect to Fig. 2D, in step 243, scanning system 101 may perform a step of container scanning. In some embodiments, scanning system 101 may apply one or more of the preceding steps of Fig. 2D against containerized environments. In some embodiments, in order to do so, scanning system 101 reconstructs a container runtime layered file system (LFS) before recursively running one or more of steps 231-241 on the reconstructed file system.

[00359] In some embodiments, the ephemeral scanner is configured to perform keys and password scanning of the at least one block storage volume. In such an embodiment the ephemeral scanner may scan the block storage volume to check for keys and password information related to the device. As discussed above with respect to Fig. 2D, in step 239, scanning system 101 may perform a step of key/password scanning. As one example situation, suppose there is a weak or unprotected password stored (in plain text) in storage 111A. For example, if a personal email account has been compromised, the passwords may be known about in advance. Scanning system 101 may search the snapshot for similar usernames or login names, and, either using known dictionaries or the account owner's previously leaked

passwords (stored in, e.g., database 103A), may attempt to login to one or more systems or devices in cloud infrastructure 106, and may record the result thereof.

[00360] In some embodiments, defining the at least one block storage volume includes presenting risk data without sharing consumer data or data that was used to identify the risk data. Risk data as used herein, may refer to information relating to causing harm. Consumer data as used herein, may refer to a user's personal information or any information trail user's leave behind as a result of their computer or Internet use. Data used to identify the risk as used herein, may refer to information for determining possible issues. In some embodiments, scanning system 101 of Fig. 1 may present information without sharing a user's personal information or any information trail user's leave behind as a result of their computer or Internet use.

[00361] In some embodiments, the metadata defining the at least one block storage volume includes at least one of: an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, or a profile configuration. An indication of an installed application as used herein, may refer to a sign or signal of a stored program. A version of an installed application as used herein, may refer to an older or newer form of a stored program. An operating system configuration as used herein, may refer to one or more computer system settings that have been set by default automatically or manually by a given program or the user. An application configuration as used herein, may refer to one or more computer program settings that have been set by default automatically or manually by a given program or the user. A profile configuration as used herein, may refer to one or more computer program file settings that have been set by default automatically or manually by a given program or the user.

[00362] Aspects of this disclosure may include receiving a transmission of the metadata from the at least one ephemeral scanner. In some embodiments, the transmission may exclude raw data of the primary account. A transmission of metadata as used herein, may refer to movement of a set of data that describes and gives information about other data. In some embodiments, scanning system 101 of Fig. 1 may receive data excluding any raw data (e.g., data as gathered before it has been processed, cleaned or analyzed).

[00363] Aspects of this disclosure may include analyzing the received metadata to identify a plurality of cybersecurity vulnerabilities. A plurality of cybersecurity vulnerabilities as used herein, may refer to a weakness in a system. For example, a cybersecurity vulnerability may be exploited by cybercriminals to gain unauthorized access to a computer system. In some embodiments, scanning system 101 of Fig. 1 may analyze the received data in order to identify any weakness in a system (e.g., to avoid cybercriminals gaining unauthorized access to a computer system). In some embodiments, vulnerability information may include information stored in a system.

[00364] Aspects of this disclosure may include correlating each of the identified plurality of cybersecurity vulnerabilities with one of the plurality of assets. In some embodiments, scanning system 101 of Fig. 1 may connect a security risk with an asset in the cloud environment. In such embodiments, the connection or correlation of vulnerability to asset may allow the system to address the vulnerability.

[00365] Aspects of this disclosure may include generating a report correlating the plurality of cybersecurity vulnerabilities with the plurality of assets. A report as used herein, may refer to a document containing information. For example, a report may notify the administrator of a website or application about a problem such as a security issue or vulnerability in the system that should be addressed. In some embodiments, scanning system 101 of Fig. 1 may create a report providing information related to the vulnerability and asset. In such an embodiment, the report may allow for the vulnerability to be addressed and resolved.

[00366] In such embodiments, the system may allow scanners (e.g., ephemeral scanners, scanning system 101) to run inside an account, including an Amazon Web Services, Azure, or GCP account. In this mode, a cloud cybersecurity service may generate ephemeral scanners inside an account that performs the same actions as a SAAS node but are logically hosted inside the account.

[00367] In some embodiments, the at least one processor is further configured to receive a transmission of updated metadata defining the at least one block storage volume in response to at least one change to the at least one block storage volumes. Updated metadata as used herein, may refer to a renewed set of data that describes and gives information about other data.

[00368] **Fig. 10** is a block diagram of a method 1000 for deployment of ephemeral scanners, consistent with disclosed embodiments. In some embodiments, the method may include seven (or more or less) steps:

[00369] Block 1002: Access a primary account maintained in a cloud environment. In some embodiments, scanning system 101 of Fig. 1 may gain access to an account within cloud infrastructure 106 using a password stored in a database, for example.

[00370] Block 1004: Receiving information defining a structure of the primary account, wherein the structure includes a plurality of assets. In some embodiments, the information may exclude raw data of the primary account. In some embodiments, scanning system 101 of Fig. 1 may receive data relating to the structure of an account. In such an embodiment, the structure of the account may include virtual machines 107A-D and databases 109A-109D, for example.

[00371] Block 1006: Deploying, inside the primary account, at least one ephemeral scanner configured to periodically scan at least one block storage volume and to output metadata defining the at least one block storage volume. In some embodiments, the output may exclude raw data of the primary account. In some embodiments, scanning system 101 of Fig. 1 may scan a block storage volume and output data related to it (e.g., vulnerability information,

configuration information, malware information, risk analysis information, and sensitive information).

[00372] Block 1008: Receiving a transmission of the metadata from the at least one ephemeral scanner. In some embodiments, the transmission may exclude raw data of the primary account. In some embodiments, scanning system 101 of Fig. 1 may receive data excluding any raw data (e.g., data as gathered before it has been processed, cleaned or analyzed).

[00373] Block 1010: Analyzing the received metadata to identify a plurality of cybersecurity vulnerabilities. For example, a cybersecurity vulnerability may be exploited by cybercriminals to gain unauthorized access to a computer system. In some embodiments, scanning system 101 of Fig. 1 may analyze the received data in order to identify any weakness in a system (e.g., to avoid cybercriminals gaining unauthorized access to a computer system).

[00374] Block 1012: Correlating each of the identified plurality of cybersecurity vulnerabilities with one of the plurality of assets. In some embodiments, scanning system 101 of Fig. 1 may connect a security risk with an asset in the cloud environment. In such embodiments, the connection or correlation of vulnerability to asset may allow the system to address the vulnerability.

[00375] Block 1014: Generating a report correlating the plurality of cybersecurity vulnerabilities with the plurality of assets. For example, a report may notify the administrator of a website or application about a problem such as a security issue or vulnerability in the system that should be addressed. In some embodiments, scanning system 101 of Fig. 1 may create a report providing information related to the vulnerability and asset. In such an embodiment, the report may allow for the vulnerability to be addressed and resolved.

[00376] **Risk Information Aggregation Techniques**

[00377] **Fig. 11** represents a schematic block diagram 1100 illustrating an exemplary embodiment of a method for providing a dashboard aggregating risk information.

[00378] Assets operating in a cloud environment face myriad cybersecurity risks that vary in type and nature of threat with increasing regularity. In order to combine the varied and multiple cybersecurity risks an asset may encounter in a cloud environment into a single view to allow for faster and more flexible processing, prioritization and mitigation, a system and method of aggregating cybersecurity risks into a single dashboard is needed. Aggregation of multiple cybersecurity risks into a single view for an administrator is needed. In one of many embodiments, an exemplary embodiment may include a graphical user interface system for providing comprehensive cloud environment risk inventory visualization. A graphical user interface system may include systems similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows 10 Mobile, PalmOS-Web OS, or Firefox OS. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage

111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of risk inventories. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D.

[00379] An exemplary embodiment may also include at least one processor. The at least one processor may be part of a personal computer, tablet, smartphone, or virtual machine for processing threats and vulnerabilities to a cloud-based storage volume. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations.

[00380] In an embodiment may include a processor configured to cause a display to present a plurality of asset categories (step 1101). A display may be graphical, sequential, or multi-dimensional. A plurality may include at least one asset category to be displayed. Asset categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category.

[00381] An account category may further include listing of assets based on type of account associated with that user. A type of account may include active user account, passive user account, administrator account, maintainer account, a system account, superuser account, or a guest user account.

[00382] A container category may include one or more containers currently listed in the system. A container may involve specific versions of programming language runtimes, libraries required to run software, or another method of packaging applications abstracted from the cloud environment.

[00383] A database category may include one or more databases managed and maintained in relation to the system. Databases may include a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. Databases may be implemented using ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[00384] An image category may contain one or more images based on image type, size, and content. Images may include images of one or more virtual machines.

[00385] A managed service category may include one or more managed services listed in the above system, to include at least client or customer-owned systems being managed by a third party entity.

[00386] A messaging service category may include systems/programs communicating with other systems/programs using services such as Google Cloud Pub and Sub communications, AWS SQS, or similar queues found in Information Systems.

[00387] A monitoring category may include centralized or decentralized methods of security monitoring systems.

[00388] A network category may include one or more systems of establishing a form of communicative connectivity between systems, e.g., TCP-IP services.

[00389] A storage category may list one or more methods of data storage, to include Direct Attached Storage, Network Attached Storage, SSD Flash Drive Arrays, Hybrid Flash Arrays, Hybrid Cloud Storage, Backup Software, Backup Appliances, Cloud Storage, or similar. Storage may include data structures, instructions, or any other data to be contained in a storage medium.

[00390] A user category may include one or more users of a given system along with identifying information relating to its unique or semi-unique identifier, level of authorized access and permissions, and storage volumes accessible by said user.

[00391] An access category may include one or more levels of access permission and a listing of users or devices granted access at a given level of access permission to a system.

[00392] A virtual machine category may list one or more virtual machines accessible to a system and the levels and types of access granted to a given virtual machine.

[00393] A serverless category may list one or more storage volumes stored on a serverless computing system such as Kubeless, Fission, Google App Engine, AWS Lambda, Google Cloud Functions, IBM Cloud Function, IBM Cloud Code Engine, Microsoft Azure Functions, Cloudflare Workers, and Compute@Edge.

[00394] In an embodiment, an exemplar may include a processor configured to receive, via an input device, a selection of a particular asset category (step 1103). An input device may include a personal computer, tablet, smartphone, or virtual machine. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations. A selection of one or more asset categories may be received and updated as new visualizations are developed based on the current threats to an asset in that category. Asset categories may be selected or deselected by an end user. Once a particular asset

category is selected, it may be added to a list of selected asset categories with a list for deselected categories as well. The list of selected asset categories may organized per category or may be compiled to include all current threats in a comprehensive selected asset threat list.

[00395] One exemplary embodiment may further include a processor configured to cause the display to present a list of assets in the selected category that have cyber security risks (step 1105). When an asset in a selected category has a cyber security risk associated with it, the asset may appear in a list of all listed categories having a cyber security risk associated with it, and may be organized by individual category, type of cyber security risk, or some combination or sequential ordering methods based on the information available to the processor and its end user.

[00396] In one of many exemplary embodiments a processor may be configured to retrieve workload component cybersecurity risk information (step 1107A) for each listed asset. When a category of assets is selected, a processor may be configured to order those assets according to available cybersecurity risk information available to said processor, and may organize by type of risk, type of asset, or similar type of sequentialization. Cybersecurity risk information may be determined from historical threats to the workload component, the current active cybersecurity threats to workload components of certain types, or some combination of the same.

[00397] In an embodiment, a processor may be configured to retrieve cloud component cybersecurity risk information (step 1107B) for each listed asset. Cloud component cybersecurity risk information may include identification of the risk based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset. Cybersecurity risk information to a cloud component may be determined from historical threats within the cloud environment 106, current active cybersecurity threats to similar workload components in the cloud environment 106 or the level of risk to assets within the cloud environment 106, as outlined below.

[00398] An exemplary embodiment may include a system configured to cause the display to present a common interface (step 1109) providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information. The common interface may be configured to access component cybersecurity risk information based on the nature of the risk, the likelihood of the risk, or the severity of the risk to the component, including recursive risk – risk that can happen from an attacker getting access to a system and using it as a mean to reach other system. On the same interface, a different interface, or a similar and related interface, a display may present cloud component cybersecurity risk information based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset.

[00399] As an exemplary embodiment, the system may be configured to cause the display to present, in the common interface, an interconnection between the workload

component cybersecurity risk information and the cloud component cybersecurity risk information (step 1111). With this display presented, an exemplary embodiment may allow for a user to prioritize data from additional sources. The interconnection between workload component cybersecurity risk information and cloud component cybersecurity risk information may generate an interface between them and may be presented as an interconnection between them. The common interface presented may include direct interconnections between the component cybersecurity risk information and the cloud component cybersecurity risk information or links from one set of cybersecurity risk information. For instance, the component cybersecurity risk information may include a link to access the cloud component cybersecurity risk information. Conversely, the cloud component cybersecurity risk information may include a link to access the component cybersecurity risk information.

[00400] In one embodiment, a system may be configured to provide the plurality of asset categories include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. A category may list the currently identified entities of that category type as described below. An account category may include one or more listed accounts of users of the system. An account category may further include listing of assets based on type of account associated with that user. A type of account may include active user account, passive user account, administrator account, maintainer account, a system account, superuser account, or a guest user account.

[00401] A container category may include one or more containers currently listed in the system. A container may involve specific versions of programming language runtimes, libraries required to run software, or another method of packaging applications abstracted from the cloud environment.

[00402] A database category may include one or more databases managed and maintained in relation to the system. Databases may include a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. Databases may be implemented using ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[00403] An image category may contain one or more images based on image type, size, and content. Images may include images of one or more virtual machines displayed. A managed service category may include one or more managed services listed in the above system, to include at least client or customer-owned systems being managed by a third party entity.

[00404] A messaging service category may include services of users communicating with at least one other user such as Google Cloud Pub and Sub communications, AWS SQS, or similar queues found in Information Systems.

[00405] A monitoring category may include centralized or decentralized methods of security monitoring systems.

[00406] A network category may include one or more systems of establishing a form of communicative connectivity between systems, e.g., TCP-IP services.

[00407] A storage category may list one or more methods of data storage, to include Direct Attached Storage, Network Attached Storage, SSD Flash Drive Arrays, Hybrid Flash Arrays, Hybrid Cloud Storage, Backup Software, Backup Appliances, Cloud Storage, or similar. Storage may include data structures, instructions, or any other data to be contained in a storage medium.

[00408] A user category may include one or more users of a given system along with identifying information relating to its unique or semi-unique identifier, level of authorized access and permissions, and storage volumes accessible by said user.

[00409] An access category may include one or more levels of access permission and a listing of users or devices granted access at a given level of access permission to a system.

[00410] A virtual machine category may list one or more virtual machines accessible to a system and the levels and types of access granted to a given virtual machine.

[00411] A serverless category may list one or more storage volumes stored on a serverless computing system such as Kubeless, Fission, Google App Engine, AWS Lambda, Google Cloud Functions, IBM Cloud Function, IBM Cloud Code Engine, Microsoft Azure Functions, Cloudflare Workers, and Compute@Edge.

[00412] In one of many exemplary embodiments, a system may further include the common interface configured to display information relating to at least one of an asset type, a risk, a region, or an account. A common interface may include the ability to configure and reconfigure the asset type, risk, region, or account to appear in any order and at multiple formats for displaying the information relating to those categories.

[00413] In another embodiment, a system may provide the common interface configured to display description for each listed asset. The display of a description for each listed asset may be listed vertically or horizontally. The display may be displayed as written text, representative graphical figures, or similar method of displaying the aforementioned types of information for an asset.

[00414] In another embodiment, a system may further provide the common interface is configured to display at least one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, or an authentication. A vulnerability may include vulnerability data from a vulnerability database such as NVD, WPVulnDB, US-CERT, Node.js Security Working Group, OVAL – Red Hat, Oracle

Linux, Debian, Ubuntu, SUSE, Ruby Advisory Database, JVN, Safety DB(Python), Alpine secdb, PHP Security Advisories Database, Amazon ALAS, RustSec Advisory Database, Red Hat Security Advisories, Microsoft MSRC, KB, Debian Security Bug Tracker, Kubernetes security announcements, Exploit Database, Drupal security advisories, JPCERT. An insecure configuration may include software flaws or misconfigurations, non-encrypted files, improper file or directory permissions, unpatched security flaws in server software, enabled or accessible administrative and debugging functions, administrative account vulnerabilities, SSL certificates and encryption settings not properly configured, or a similar misconfiguration. These misconfigurations may be discovered by performing a scan, e.g., may query devices and systems capable of routing traffic (e.g., load balancer 115, routers, switches, firewalls, and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations. Malware indicated may include Adware, Botnets, Cryptojacking, Malvertising, Ransomware, Remote Administration Tools (RATs), Rootkits, Spyware, Trojans, Virus Malware, Worm Malware, or similar attack vehicles. A neglected asset may include an asset that has been improperly maintained, patched, or similar cybersecurity security measure. A data at risk may include any packet of data that may be exposed to a cybersecurity threat. A lateral movement may include any pathway between assets where a cybersecurity risk can travel from one affected asset to another affected asset. An authentication may include a username, password, one-time password, two-factor authentication, or any other authentication mechanism to gain access to a cloud service provider's system. A common interface may display more than one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, and an authentication separately or concurrently for an asset. A display of said information may be ordered and listed in any order depending on user preference and input.

[00415] An exemplary embodiment may further include a system wherein the common interface is configured to display one or more possible attack vectors reaching the each listed asset. Generating this common interface display may include generating a map of an asset to include recording information such as a region identifier, site identifier, datacenter identifier, physical address, network address, workload name, or any other identifier which may be acquired via an API provided through a cloud service provider's system and demonstrating via a display the myriad attack vectors currently available to an analyzed asset. Display of the one or more possible attack vectors reaching the each listed asset may be displayed as written text, representative graphical figures, or similar method of displaying the one or more possible attack vectors.

[00416] Another exemplary embodiment may further include a system wherein the common interface is configured to display a recommended mitigation tactic for the each listed asset. A recommended mitigation tactic may be generated based on historical data of the

cybersecurity risk, the vulnerability attributed to the listed asset, or the data contained on the each listed asset. Mitigation tactics may be selected based on user input or may be implemented automatically based on the mitigation tactics likelihood of a successful mitigation of the cybersecurity risk. A mitigation tactic may include increased frequency of scanning, heightened access control measures, firewalls and antivirus software patches, increased patch management scheduling, continuous workload monitoring, or similar tactics.

[00417] In another exemplary embodiment is a system wherein the common interface is configured to display one or more workload metrics associated with the each listed asset. Workload metrics may be a statistical representation of the current performance of the workload, the capacity for additional processing of the workload, and the historical performance of the workload based on statistical trend.

[00418] Another exemplary embodiment may include a system wherein the at least one processor is further configured to provide a cybersecurity report for the each listed asset. A cybersecurity report may be generated by scanning system 101 following a complete scan of an analyzed asset, and may include a snapshot of one or more of the number of total cybersecurity threats to a listed asset, number of cybersecurity threats mitigated by recent performance, methods of cybersecurity risk mitigation in the listed asset, rate of cybersecurity threats detected in the listed asset, and possibility of failed cybersecurity mitigation tactics based on the frequency and nature of the historical cybersecurity threats. Data collected to generate a cybersecurity report may include, among many things, operating system packages, installed software applications, libraries, and program language libraries such as Java archives, Python packages, Go modules, Ruby gems, PHP packages, and Node.js modules, other software applications, library versions, software versions, and other identifying characteristics of software and operating systems, lists of users of each system or device (e.g., virtual machines 107A-D), each system's or device's services, password hashes, and application-specific configurations for software/services such as Apache, Nginx, SSH, and other services, bugs or other configuration risks, malware scan results, passwords, scripts, shell history, repositories, or other data that may contain passwords, cloud access keys, SSH keys, or other key/password/access information that provide unchecked access to important resources.

[00419] In another embodiment, a system may include the at least one processor is configured to identify a risk level distribution among the listed assets. The at least one processor may be configured to list risk level distribution by highest level of risk, lowest level of risk, or largest number of cybersecurity risks attributed among the listed assets. A risk level may be high if the web service is connected to the Internet (e.g., has at least one public port forwarded through a firewall to the web service, is able to be accessed through a load balancer, is able to be accessed via reverse proxy). A risk level may be medium risk if the web service is only accessible internally (e.g., because of firewall configuration). A risk level may be low risk if access to the web server is blocked by a configuration of cloud infrastructure 106.

[00420] Another exemplary embodiment may include a system wherein the at least one processor is configured to receive a search query for a specific risk. A search query may be input by type of risk, name of a risk, or another semi-unique manner of identifying a specific cybersecurity risk. A search may be initiated by end user operating user device 102, by a maintainer or administrator of an analyzed asset, or by a virtual machine associated with operating scanning system 101, A search may be initiated by inputting a number of inputs, to include keywords, phrases, file type, or similar identifiers unique or semi-unique to a given cybersecurity risk. Upon initiating a search, the results generated by the search may be stored in the analyzed asset, transmitted to the operator of a processor running side scanning system 101, or generated for review by an end user operating user device 102. The search and display of a specific risk may be based on historical trends of cybersecurity risks in the listed asset, severity of cybersecurity risk to a listed asset, or a historical search for past cybersecurity threats to a listed asset.

[00421] Another embodiment may include a system wherein the at least one processor is configured to identify one or more assets vulnerable to the specific risk. For instance, a processor operating scanning system 101 may be configured to perform a scan of a set of one or more assets with potential exposure to a given risk, and upon identifying vulnerable pathways for an attack vector to reach a set of identified assets, may generate a list of assets that are potentially or currently exposed to one or more specific risk. Assets vulnerable to a specific risk may be determined to be vulnerable based on the type of the specific risk, the existence of specific risks in adjacent or similar workloads, and related risks that may also cause an asset to be vulnerable based on the specific risk determined.

[00422] In another embodiment is a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing a display to present a plurality of asset categories (step 1101). The method may include the use of a graphical user interface system and may include systems similar in appearance and functionality to Microsoft Windows, Mac OS, Ubuntu Unity, Gnome Shell, Android, Apple iOS, Blackberry OS, Windows 10 Mobile, PalmOS-Web OS, or Firefox OS. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D. The cloud environment may include service provided by a third party such as Amazon Web Services, Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP, or similar service. Visualizations provided by the above system may include graphical, sequential, or multi-dimensional displays of risk inventories. For example, cloud infrastructure 106 may be the cloud environment consisting of virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115, and the cloud-based storage volume may be contained in storage 111A-111D.

[00423] One of several embodiments may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising receiving, via an input device, a selection of a particular asset category. A method may include a personal computer, tablet, smartphone, or virtual machine. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations. A selection of one or more asset categories may be received and updated as new visualizations are developed based on the current threats to an asset in that category. Asset categories may be selected or deselected by an end user. Once a particular asset category is selected, it may be added to a list of selected asset categories with a list for deselected categories as well. The list of selected asset categories may organized per category or may be compiled to include all current threats in a comprehensive selected asset threat list.

[00424] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing the display to present a list of assets in the selected category that have cybersecurity risks (step 1105). A display may be graphical, sequential, or multi-dimensional. A list of selected categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. When an asset in a selected category has a cyber security risk associated with it, the asset may appear in a list of all listed categories having a cyber security risk associated with it, and may be organized by individual category, type of cyber security risk, or some combination or sequential ordering methods based on the information available to the processor and its end user.

[00425] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising retrieving workload component cybersecurity risk information 307A for each listed asset. When a category of assets is selected, a processor may be configured to order those assets according to available cybersecurity risk information available to said processor, and may organize by type of risk, type of asset, or similar type of sequentialization. Cybersecurity risk information may be determined from historical threats to the workload component, the current active cybersecurity threats to workload components of certain types, or some combination of the same.

[00426] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising retrieving cloud component cybersecurity risk information 307B. Cloud component cybersecurity risk information may include identification of the risk based on a semi-unique identifier, the nature of the risk, the

likelihood of the risk, or the severity of the risk to each listed asset. Cybersecurity risk information to a cloud component may be determined from historical threats within the cloud environment 106, current active cybersecurity threats to similar workload components in the cloud environment 106 or the level of risk to assets within the cloud environment 106, as outlined above.

[00427] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing the display to present a common interface 309 providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information. The common interface may be configured to access component cybersecurity risk information based on the nature of the risk, the likelihood of the risk, or the severity of the risk to the component. On the same or a similar and related interface, a display may present cloud component cybersecurity risk information based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset.

[00428] Another embodiment may include a method for providing comprehensive cloud environment risk inventory visualization in a graphical user interface comprising causing the display to present, in the common interface, an interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information. The interconnection between workload component cybersecurity risk information 307A and cloud component cybersecurity risk information 307B may generate an interface between them 309 and may be presented as an interconnection between them 311. The common interface presented may include direct interconnections between the component cybersecurity risk information and the cloud component cybersecurity risk information or links from one set of cybersecurity risk information. For instance, the component cybersecurity risk information may include a link to access the cloud component cybersecurity risk information. Conversely, the cloud component cybersecurity risk information may include a link to access the component cybersecurity risk information.

[00429] In another embodiment, a method may include wherein the plurality of asset categories include at least one of an account category, an authentication category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. A category may list the currently identified entities of that category type as described below. An account category may include one or more listed accounts of users of the system. An account category may further include listing of assets based on type of account associated with that user. A type of account may include active user account, passive user account, administrator account, maintainer account, a system account, superuser account, or a guest user account.

[00430] A container category may include one or more containers currently listed in the system. A container may involve specific versions of programming language runtimes, libraries required to run software, or another method of packaging applications abstracted from the cloud environment.

[00431] A database category may include one or more databases managed and maintained in relation to the system. Databases may include a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. Databases may be implemented using ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[00432] An image category may contain one or more images based on image type, size, and content. Images may include images of one or more virtual machines displayed.

[00433] A managed service category may include one or more managed services listed in the above system, to include at least client or customer-owned systems being managed by a third party entity.

[00434] A messaging service category may include services of users communicating with at least one other user such as Google Cloud Pub and Sub communications, AWS SQS, or similar queues found in Information Systems.

[00435] A monitoring category may include centralized or decentralized methods of security monitoring systems.

[00436] A network category may include one or more systems of establishing a form of communicative connectivity between systems, e.g., TCP-IP services.

[00437] A storage category may list one or more methods of data storage, to include Direct Attached Storage, Network Attached Storage, SSD Flash Drive Arrays, Hybrid Flash Arrays, Hybrid Cloud Storage, Backup Software, Backup Appliances, Cloud Storage, or similar. Storage may include data structures, instructions, or any other data to be contained in a storage medium.

[00438] A user category may include one or more users of a given system along with identifying information relating to its unique or semi-unique identifier, level of authorized access and permissions, and storage volumes accessible by said user.

[00439] An access category may include one or more levels of access permission and a listing of users or devices granted access at a given level of access permission to a system.

[00440] A virtual machine category may list one or more virtual machines accessible to a system and the levels and types of access granted to a given virtual machine.

[00441] Another embodiment may include a method wherein the common interface is configured to display information relating to at least one of an asset type, a risk, a region, or an account. A common interface may include the ability to configure and reconfigure the asset

type, risk, region, or account to appear in any order and at multiple formats for displaying the information relating to those categories.

[00442] Another embodiment may include a method wherein the common interface is configured to display description for each listed asset. The display of a description for each listed asset may be listed vertically or horizontally. The display may be displayed as written text, representative graphical figures, or similar method of displaying the aforementioned types of information for an asset.

[00443] Another embodiment may include a method wherein the common interface is configured to display at least one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, or an authentication. A vulnerability may include vulnerability data from a vulnerability database such as NVD, WPVulnDB, US-CERT, Node.js Security Working Group, OVAL – Red Hat, Oracle Linux, Debian, Ubuntu, SUSE, Ruby Advisory Database, JVN, Safety DB(Python), Alpine secdb, PHP Security Advisories Database, Amazon ALAS, RustSec Advisory Database, Red Hat Security Advisories, Microsoft MSRC, KB, Debian Security Bug Tracker, Kubernetes security announcements, Exploit Database, Drupal security advisories, JPCERT. An insecure configuration may include software flaws or misconfigurations, non-encrypted files, improper file or directory permissions, unpatched security flaws in server software, enabled or accessible administrative and debugging functions, administrative account vulnerabilities, SSL certificates and encryption settings not properly configured, or a similar misconfiguration. These misconfigurations may be discovered by performing a scan, e.g., may query devices and systems capable of routing traffic (e.g., load balancer 115, routers, switches, firewalls, and proxies) using an API provided through a cloud service provider's system to determine network configurations, and may evaluate them against known problematic configurations or other configurations. Malware indicated may include Adware, Botnets, Cryptojacking, Malvertising, Ransomware, Remote Administration Tools (RATs), Rootkits, Spyware, Trojans, Virus Malware, Worm Malware, or similar attack vehicles. A neglected asset may include an asset that has been improperly maintained, patched, or similar cybersecurity security measure. A data at risk may include any packet of data that may be exposed to a cybersecurity threat. A lateral movement may include any pathway between assets where a cybersecurity risk can travel from one affected asset to another affected asset. An authentication may include a username, password, one-time password, two-factor authentication, or any other authentication mechanism to gain access to a cloud service provider's system. A common interface may display more than one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, and an authentication separately or concurrently for an asset. A display of said information may be ordered and listed in any order depending on user preference and input.

[00444] Another embodiment may include a method wherein the common interface is configured to display one or more possible attack vectors reaching the each listed asset. Generating this common interface display may include generating a map of an asset to include recording information such as a region identifier, site identifier, datacenter identifier, physical address, network address, workload name, or any other identifier which may be acquired via an API provided through a cloud service provider's system and demonstrating via a display the myriad attack vectors currently available to an analyzed asset. Display of the one or more possible attack vectors reaching the each listed asset may be displayed as written text, representative graphical figures, or similar metho of displaying the one or more possible attack vectors.

[00445] Another embodiment may include a method wherein the common interface is configured to display a recommended mitigation tactic for the each listed asset. A recommended mitigation tactic may be generated based on historical data of the cybersecurity risk, the vulnerability attributed to the listed asset, or the data contained on the each listed asset. Mitigation tactics may be selected based on user input or may be implemented automatically based on the mitigation tactics likelihood of a successful mitigation of the cybersecurity risk. A mitigation tactic may include increased frequency of scanning, heightened access control measures, firewalls and antivirus software patches, increased patch management scheduling, continuous workload monitoring, or similar tactics.

[00446] Another embodiment may include a method wherein the common interface is configured to display one or more workload metrics associated with the each listed asset. Workload metrics may be a statistical representation of the current performance of the workload, the capacity for additional processing of the workload, and the historical performance of the workload based on statistical trend.

[00447] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing a display to present a plurality of asset categories 301. A plurality may include at least one asset category to be displayed. Asset categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category.

[00448] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising receiving, via an input device, a selection of a particular asset category (step 1103). An input device may include a personal computer, tablet, smartphone, or virtual

machine. A processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations. A selection of one or more asset categories may be received and updated as new visualizations are developed based on the current threats to an asset in that category. Asset categories may be selected or deselected by an end user. Once a particular asset category is selected, it may be added to a list of selected asset categories with a list for deselected categories as well. The list of selected asset categories may organized per category or may be compiled to include all current threats in a comprehensive selected asset threat list.

[00449] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing the display to present a list of assets in the selected category that have cybersecurity risks (step 1105). A display may be graphical, sequential, or multi-dimensional. A list of selected categories may include at least one of an account category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category. When an asset in a selected category has a cyber security risk associated with it, the asset may appear in a list of all listed categories having a cyber security risk associated with it, and may be organized by individual category, type of cyber security risk, or some combination or sequential ordering methods based on the information available to the processor and its end user.

[00450] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising retrieving workload component cybersecurity risk information for each listed asset (step 1107A). When a category of assets is selected, a processor may be configured to order those assets according to available cybersecurity risk information available to said processor, and may organize by type of risk, type of asset, or similar type of sequentialization. Cybersecurity risk information may be determined from historical threats to the workload component, the current active cybersecurity threats to workload components of certain types, or some combination of the same.

[00451] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the

operations comprising retrieving cloud component cybersecurity risk information for each listed asset (step 1107B). Cloud component cybersecurity risk information may include identification of the risk based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset. Cybersecurity risk information to a cloud component may be determined from historical threats within the cloud environment 106, current active cybersecurity threats to similar workload components in the cloud environment 106 or the level of risk to assets within the cloud environment 106, as outlined above.

[00452] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing the display to present a common interface providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information for each listed asset (step 1109). The common interface may be configured to access component cybersecurity risk information based on the nature of the risk, the likelihood of the risk, or the severity of the risk to the component. On the same or a similar and related interface, a display may present cloud component cybersecurity risk information based on a semi-unique identifier, the nature of the risk, the likelihood of the risk, or the severity of the risk to each listed asset.

[00453] Another embodiment may include a non-transitory computer readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations for matching keys with compute resources, the operations comprising causing the display to present, in the common interface, an interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information (step 1111). The interconnection between workload component cybersecurity risk information (step 1107A) and cloud component cybersecurity risk information (step 1107B) may generate an interface between them (step 1109) and may be presented as an interconnection between them (step 1111). The common interface presented may include direct interconnections between the component cybersecurity risk information and the cloud component cybersecurity risk information or links from one set of cybersecurity risk information. For instance, the component cybersecurity risk information may include a link to access the cloud component cybersecurity risk information. Conversely, the cloud component cybersecurity risk information may include a link to access the component cybersecurity risk information.

[00454] In some embodiments, a processor may cause a display of one or more risks aggregated in the same visualization. **Fig. 12** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of displaying aggregated cybersecurity risk information, consistent with disclosed embodiments. A visualization may be generated from one or more sources based on scanning system 101. For example, a visual display may include a

cloud provider visualization 1201, displaying where the multiple sources of data may be derived from. A cloud service provider may include services such as Cloudflare, Amazon Web Services, Google Cloud, IBM Cloud, Oracle Cloud, Microsoft Azure, or similar. This is further displayed via a communicative connection through an access port 1203 to a data server 1205.

[00455] Data server 1205 may include servers such as 109A-D. Data server 1205 may then display risk information that may be communicatively passed through web server 1207, demonstrating one or more possible sources of the one or more cybersecurity risk information. Data server 1205 may be communicatively connected to a web server 1207 (through access port 1203, which may be the same or a different port from port 1203 between cloud service provider 1201 and data server 1205), which can then be communicatively connected to one or more users via user access points 1209. This may include, among other things, user access that may be verified or validated via a credential 1211. Each of these connections may demonstrate one or more risk paths that may be exploited in a single view to allow for prioritization of threats and analysis of possible users who may exploit those risks and vulnerabilities, as discussed above. This may allow an administrator to prioritize risks to a given storage volume as well as assess lateral storage volumes that may be similarly risked based on the assessed threats in a single image.

[00456] **Fig. 13** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of a flow path for aggregated risk information, consistent with disclosed embodiments. In another embodiment, a processor may be configured to display a flow path of aggregated risk information. In one view, a displayed visualization may include user access 1301, which displays risk information such as service vulnerability 1303 and an insecure private key 1303. These vulnerabilities or risks may be displayed to an administrator via developer server 1307 to an administrator through administrator access 1309. As previously discussed, this will allow an administrator to visualize myriad risks to a storage volume in a single view that will allow an administrator to prioritize risks and analyze possible exploiters of this risk information on lateral systems and storage volumes.

[00457] In another embodiment, risk information may be shown in one or more views. This may involve a risk path similar to the exemplary embodiment of Fig. 13. In an exemplary embodiment, in a single visualization data from multiple sources may be displayed. This may involve identifying risk information such as the origin of the risk and potential exploiters of a risk. In some embodiments, this visualization may be done for a specific risk. In another embodiment, this visualization may be done for a combination of risks.

[00458] Disclosed embodiments may include any one of the following bullet-pointed features alone or in combination with one or more other bullet-pointed features, whether implemented as a method, by at least one processor, and/or stored as executable instructions on non transitory computer readable media:

- establishing a trusted relationship between a source account in a cloud environment and a scanner account;
- using the established trust relationship, utilize at least one cloud provider API to identify workloads in the source account;
- using the at least one cloud provider API to query a geographical location of at least one of the identified workloads;
- receiving an identification of the geographic location;
- using the cloud provider APIs to access block storage volumes of the at least one workload;
- determining a file-system of the at least one workload;
- mounting the block storage volumes on a scanner based on the determined file-system;
- activating a scanner at the geographic location;
- reconstructing from the block storage volumes a state of the workload; and
- assessing the reconstructed state of the workload to extract insights.
- wherein the geographic location includes an identifier of a physical site.
- wherein mounting includes selecting a driver corresponding to the determined file system.
- wherein the at least one processor is further configured to deploy a scanner at the geographical location.
- wherein the identification of the geographic location comprises an identification of a data center, at least one of a data center name, Internet Protocol (IP) address, name of the cloud provider, or a unique identity.
- wherein the reconstructed state of the workload includes at least two of an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content;
- wherein the at least one processor is further configured to update the reconstructed state of the workload based on at least one change to the block storage volumes.
- wherein the insights comprise at least one of a vulnerability associated with the workload or a composition of installed applications associated with the workload.
- wherein to mount the block storage volumes on the scanner, the at least one processor is configured to create a snapshot of the block storage volumes;
- wherein to mount the block storage volumes on the scanner, the at least one processor is configured to mount the snapshot of the block storage volumes on the scanner
- wherein the at least one processor is further configured to encrypt the snapshot of the block storage volumes;

- wherein the at least one processor is further configured to mount the encrypted snapshot of the block storage volumes on the scanner;
- using a cloud provider API, access a block storage volume of a workload maintained in a cloud storage environment;
- identifying an installed software application in the accessed block storage volume;
- analyzing the identified installed software application to determine an associated software version;
- accessing a data structure of known software vulnerabilities for a plurality of versions of software applications;
- performing a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and
- performing at least one of query the cloud provider API to determine network accessibility information related to the workload, identify at least one port on which the vulnerable application is accessible, use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.
- implementing a remedial action in response to the identified one or more vulnerabilities.
- wherein the remedial measure includes transmitting an alert to a device associated with an administrator.
- wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises examining data sources associated with the workload
- wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises determining the network accessibility information based on the examined data sources.
- wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.
- identifying the installed software application comprises extracting data from at least one of operating system packages, libraries, or program language libraries;
- identifying the installed software application comprises identifying the installed software application based on the extracted data;
- identifying a version of the installed software application;
- wherein the identified installed software application includes one or more scripts;
- wherein the data structure includes aggregated vulnerability data;
- wherein the aggregated vulnerability data includes data from one or more third-party vendors;

- wherein the aggregated vulnerability data includes data collected by a scanner;
- wherein the aggregated vulnerability data includes at least one of an advisory, an exploit, a security announcement, or a known bug.
- querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.
- accessing at least one cloud provider API to determine a plurality of entities capable of routing traffic in a virtual cloud environment associated with a target account containing the workload;
- querying the at least one cloud provider API to determine at least one networking configuration of the entities;
- building a graph connecting the plurality of entities based on the networking configuration;
- accessing a data structure identifying services publicly accessible via the internet and capable of serving as an internet proxy;
- integrating the identified services into the graph;
- traversing the graph to identify at least one source originating via the internet and reaching the workload; and
- outputting a risk notification associated with the workload.
- wherein the plurality of entities includes a virtual network appliance.
- wherein the virtual network appliance is at least one of a load balancer, a firewall, a proxy, or a router.
- wherein the networking configuration is at least one of a routing configuration, a proxy configuration, a load balancing configuration, a firewall configuration, or a VPN configuration.
- wherein the graph includes a data structure sequentially connecting entities.
- wherein the graph includes directional vectors indicating directions of dataflow.
- wherein building the graph comprises identifying individual entities as nodes
- wherein building the graph comprises connecting the nodes.
- wherein the graph includes port numbers associated with the workload.
- wherein the graph includes a path from the at least one source to the workload.
- utilizing a cloud provider API to access a block storage volume of a workload maintained on a target account in a target system of a cloud storage environment;
- utilizing a scanner at a location of the block storage volume and on a secondary system other than the target system;
- scanning the block storage volume for malicious code, using the secondary system;

- identifying malicious code based on the scan; and
- outputting from the secondary system, a notification of a presence of malicious code in the target system.
- wherein the location of the block storage volume includes at least one of: the target account, a secondary system account, a cloud provider account, or a third party account.
- wherein scanning the block storage volume includes scanning disk-backed memory.
- wherein the disk-backed memory includes at least one of a page file or a cache file.
- wherein the secondary system includes at least one of a virtual machine, a container, or a serverless function.
- wherein the secondary system has an operating system different from an operating system of the target account.
- wherein the malicious code includes a rootkit.
- wherein utilizing a scanner includes suspending an operation of the scanner after the scan of the block storage volume.
- wherein utilizing a scanner includes modifying a pre-utilized scanner at the location of the block storage volume based on information related to the target account to obtain a modified scanner;
- wherein utilizing a scanner includes utilizing the modified scanner.
- identifying assets in a cloud environment;
- identifying risks associated with each of the identified asset;
- identifying relationships between at least some of the identified assets, the relationships including at least one of a trust, a network connectivity, or a mechanism of network proxying;
- receiving an identification of a specific asset under investigation;
- performing a forward analysis of the specific asset under investigation to identify at least one possible attack vector reaching the specific asset via a network outside the cloud environment;
- performing a backward analysis of the specific asset to identify at least one exposure risk to one or more assets that is in a downstream of the specific asset, wherein the at least one exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and a lateral movement risk associated with the exposed asset; and
- outputting a signal to cause on a display to present a presentation of forward and backward paths associated with the specific asset, thereby enabling visualization of a plurality of entry points and lateral movement risks associated with the plurality of entry points.
- wherein the network outside the cloud environment includes the Internet.

- wherein the assets in the cloud environment include at least one of: a virtual machine, a network appliance, a storage appliance, a compute instances, or an engine instance.
- wherein identifying the assets in a cloud environment includes identifying the assets based on at least one of: an identity and access management policy, an organization policy, or an access policy.
- wherein the presentation of the forward and backward paths indicates alternative paths connecting between the specific asset and an upstream asset or a downstream asset.
- wherein the visualization includes a presentation of the alternative paths.
- wherein the presentation of the forward and backward paths indicates port numbers for each pathway.
- wherein the visualization of the entry points indicates at least one entry point at risk.
- monitoring network activities of the assets in a cloud environment.
- detecting detect a potential risk associated with the specific asset based on the monitored network activities.
- detecting a potential risk associated with the specific asset based on a network activity of the specific asset.
- detecting a potential risk associated with the specific asset based on a network activity of an upstream asset of the specific asset.
- detecting a potential risk associated with the specific asset based on a network activity of a downstream asset of the specific asset.
- analyzing a cloud environment to identify a plurality of keys to the compute resources in the cloud environment;
- performing a cryptographic analysis on the plurality of keys to identify a first set of fingerprints that uniquely identify each of the plurality of keys, the first set of fingerprints being non-functional;
- analyzing trust configurations of the compute resources to identify a second set of fingerprints for each of the compute resources; and
- comparing the first set of fingerprints with the second set of fingerprints to match keys with the compute resources without using the keys to access the compute resources.
- wherein the plurality of keys are stored in at least one workload.
- wherein at least one of the first set of fingerprints is not identical to any key of the plurality of keys.
- wherein at least one of the plurality of keys includes at least one of a password, a script containing a password, a private component of a private-public key pair, a cloud key, or an Secure Shell (SSH) key.
- testing validity of at least one of the plurality of keys.
- analyzing a multi-machine interaction in the cloud environment using the first set of

fingerprints.

- analyzing the multi-machine interaction includes comparing the first set of fingerprints with the second set of fingerprints.
- analyzing a multi-machine interaction in the cloud environment using the plurality of keys.
- accessing a primary account maintained in a cloud environment;
- receiving information defining a structure of the primary account, wherein the structure includes a plurality of assets, and wherein the information excludes raw data of the primary account;
- deploying, inside the primary account or inside a secondary account for which trust has been established with the primary account, at least one ephemeral scanner configured to scan at least one block storage volume and to output metadata defining the at least one block storage volume, the output excluding raw data of the primary account;
- receiving a transmission of the metadata from the at least one ephemeral scanner, wherein the transmission excludes raw data of the primary account;
- analyzing the received metadata to identify a plurality of cybersecurity vulnerabilities;
- correlating each of the identified plurality of cybersecurity vulnerabilities with one of the plurality of assets; and
- generating a report correlating the plurality of cybersecurity vulnerabilities with the plurality of assets.
- wherein defining the at least one block storage volume includes presenting risk data without sharing consumer data or data that was used to identify the risk data.
- wherein the metadata defining the at least one block storage volume includes at least one of: an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, or a profile configuration.
- receiving a transmission of updated metadata defining the at least one block storage volume in response to at least one change to the at least one block storage volumes.
- wherein the ephemeral scanner is configured to perform vulnerability scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform configuration scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform malware scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform lateral-movement risk analysis of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform sensitive information scanning of the at least one block storage volume.

- wherein the ephemeral scanner is configured to perform container scanning of the at least one block storage volume.
- wherein the ephemeral scanner is configured to perform keys and password scanning of the at least one block storage volume.
- causing a display to present a plurality of asset categories;
- receiving, via an input device, a selection of a particular asset category;
- causing the display to present a list of assets in the selected category that have cyber security risks;
- for each listed asset, retrieving workload component cybersecurity risk information, and
- for each listed asset. retrieving cloud component cybersecurity risk information;
- for each listed asset, causing the display to present a common interface providing access to the workload component cybersecurity risk information and cloud component cybersecurity risk information; and
- for each listed asset, causing the display to present, in the common interface, an interconnection between the workload component cybersecurity risk information and the cloud component cybersecurity risk information.
- wherein the plurality of asset categories include at least one of an account category, an authentication category, a container category, a database category, an image category, a container category, a managed service category, a messaging service category, a monitoring category, a network category, a storage category, a user category, a access category, a virtual machine category, or a serverless category.
- wherein the common interface is configured to display information relating to at least one of an asset type, a risk, a region, or an account.
- wherein the common interface is configured to display description for each listed asset.
- wherein the common interface is configured to display at least one of a vulnerability, an insecure configuration, an indication of a presence of malware, a neglected asset, a data at risk, a lateral movement, or an authentication.
- wherein the common interface is configured to display one or more possible attack vectors reaching the each listed asset.
- wherein the common interface is configured to display a recommended mitigation tactic for the each listed asset.
- wherein the common interface is configured to display one or more workload metrics associated with the each listed asset.
- wherein the at least one processor is further configured to provide a cybersecurity report for the each listed asset.
- wherein the at least one processor is configured to identify a risk level distribution among the listed assets.

- receiving a search query for a specific risk; or
- identifying one or more assets vulnerable to the specific risk.

[00459] Other embodiments will be apparent to those skilled in the art from consideration of the specification and practice of the disclosed embodiments disclosed herein. It is intended that the specification and examples be considered as exemplary only, with a true scope and spirit of the disclosed embodiments being indicated by the following claims.

[00460] Computer programs based on the written description and methods of this specification are within the skill of a software developer. The various programs or program modules can be created using a variety of programming techniques. One or more of such software sections or modules can be integrated into a computer system, non-transitory computer readable media, or existing software.

[00461] Moreover, while illustrative embodiments have been described herein, the scope includes any and all embodiments having equivalent elements, modifications, omissions, combinations (e.g., of aspects across various embodiments), adaptations or alterations based on the present disclosure. The elements in the claims are to be interpreted broadly based on the language employed in the claims and not limited to examples described in the present specification or during the prosecution of the application. These examples are to be construed as non-exclusive. Further, the steps of the disclosed methods can be modified in any manner, including by reordering steps or inserting or deleting steps. It is intended, therefore, that the specification and examples be considered as exemplary only, with a true scope and spirit being indicated by the following claims and their full scope of equivalents.

CLAIMS:

1. A system for inspecting data, the system comprising:

at least one processor configured to:

establish a trusted relationship between a source account in a cloud environment and a scanner account;

using the established trust relationship, utilize at least one cloud provider API to identify workloads in the source account;

use the at least one cloud provider API to query a geographical location of at least one of the identified workloads;

receive an identification of the geographic location;

use the cloud provider APIs to access block storage volumes of the at least one workload;

determine a file-system of the at least one workload;

mount the block storage volumes on a scanner based on the determined file-system;

activate a scanner at the geographic location;

reconstruct from the block storage volumes a state of the workload; and

assess the reconstructed state of the workload to extract insights.

2. The system of claim 1, wherein the geographic location includes an identifier of a physical site.

3. The system of claim 1, wherein mounting includes selecting a driver corresponding to the determined file system.

4. The system of claim 1, wherein the at least one processor is further configured to deploy a scanner at the geographical location.

5. The system of claim 1, wherein the identification of the geographic location comprises an identification of a data center, at least one of a data center name, Internet Protocol (IP) address, name of the cloud provider, or a unique identity.

6. The system of claim 1, wherein the reconstructed state of the workload includes at least two of an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content.

7. The system of claim 1, wherein the at least one processor is further configured to update the reconstructed state of the workload based on at least one change to the block storage volumes .

8. The system of claim 1, wherein the insights comprise at least one of a vulnerability associated with the workload or a composition of installed applications associated with the workload.

9. The system of claim 1, wherein to mount the block storage volumes on the scanner, the at least one processor is configured to:

create a snapshot of the block storage volumes; and

mount the snapshot of the block storage volumes on the scanner.

10. The system of claim 9, wherein the at least one processor is further configured to:

encrypt the snapshot of the block storage volumes; and

mount the encrypted snapshot of the block storage volumes on the scanner.

11. A method for inspecting data, comprising:

establishing a trusted relationship between a source account in a cloud environment and a scanner account;

using the established trust relationship, utilizing at least one cloud provider API to identify workloads in the source account;

using the at least one cloud provider API to query a geographical location of at least one of the identified workloads;

receiving an identification of the geographic location;

using the cloud provider APIs to access block storage volumes of the at least one workload;

determining a file-system of the at least one workload;

mounting the block storage volumes on a scanner based on the determined file-system;

activating a scanner at the geographic location;

reconstructing from the block storage volumes a state of the workload; and

assessing the reconstructed state of the workload to extract insights.

12. The method of claim 11, wherein the geographic location includes an identifier of a physical site.

13. The method of claim 11, wherein mounting includes selecting a driver corresponding to the determined file system.

14. The method of claim 11, wherein the at least one processor is further configured to deploy a scanner at the geographical location.

15. The method of claim 11, wherein the identification of the geographic location comprises at least one of an identification of a data center, at least one of a data center name, Internet Protocol (IP) address, name of the cloud provider, or a unique identity.

16. The method of claim 11, wherein the reconstructed state of the workload includes at least two of an indication of an installed application, a version of an installed application, an operating system configuration, an application configuration, a profile configuration, a log, or a database content.

17. The method of claim 11, further comprising updating the reconstructed state of the workload based on at least one change to the block storage volumes.

18. The method of claim 11, wherein the insights comprise at least one of a vulnerability associated with the workload or a composition of installed applications associated with the workload.

19. The method of claim 11, wherein mounting the block storage volumes on the scanner comprising:

creating a snapshot of the block storage volumes; and

mounting the snapshot of the block storage volumes on the scanner.

20. A non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising:

establishing a trusted relationship between a source account in a cloud environment and a scanner account;

using the established trust relationship, utilizing at least one cloud provider API to identify workloads in the source account;

using the at least one cloud provider API to query a geographical location of at least one of the identified workloads;

receiving an identification of the geographic location;

using the cloud provider APIs to access block storage volumes of the at least one workload;

determining a file-system of the at least one workload;

mounting the block storage volumes on a scanner based on the determined file-system;

activating a scanner at the geographic location;

reconstructing from the block storage volumes a state of the workload; and

assessing the reconstructed state of the workload to extract insights.

21. A cyber security system for a cloud environment, the system comprising:

at least one processor configured to:

using a cloud provider API, access a block storage volume of a workload maintained in a cloud storage environment;

identify an installed software application in the accessed block storage volume;

analyze the identified installed software application to determine an associated software version;

access a data structure of known software vulnerabilities for a plurality of versions of software applications;

perform a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and

perform at least one of:

query the cloud provider API to determine network accessibility information related to the workload;

identify at least one port on which the vulnerable application is accessible; and

use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.

22. The system of claim 21, wherein the at least one processor is further configured to implement a remedial action in response to the identified one or more vulnerabilities.

23. The system of claim 22, wherein the remedial measure includes transmitting an alert to a device associated with an administrator.

24. The system of claim 21, wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises

examining data sources associated with the workload; and

determining the network accessibility information based on the examined data sources.

25. The system of claim 21, wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.

26. The system of claim 21, wherein to identify the installed software application, the at least one processor is configured to:

extract data from at least one of operating system packages, libraries, or program language libraries; and

identify the installed software application based on the extracted data.

27. The system of claim 21, wherein the at least one processor is further configured to identify a version of the installed software application.

28. The system of claim 21, wherein the identified installed software application includes one or more scripts.