

ADAPTIVE ASSET-SPECIFIC ANOMALY DETECTION FOR OT AND IT NETWORKS

FIELD OF THE PRESENTLY DISCLOSED SUBJECT MATTER

The presently disclosed subject matter relates to cybersecurity risks detection
5 and warnings in computer systems.

BACKGROUND

Anomaly detection in operational technology (OT) and information technology
(IT) networks is a crucial aspect of cybersecurity that enables the identification of
suspicious activities and potential threats. However, traditional methods face
10 significant challenges, which often result in deficient performance.

GENERAL DESCRIPTION

In the current context normal network behavior refers to the typical and
expected patterns of activity within a network, including data transmission, protocol
usage, device interactions, user activities, and application behavior. For example, it
15 includes regular user logins during business hours, standard HTTP and HTTPS traffic
for web browsing, typical communication between workstations and servers, and
expected data transfer volumes. Understanding this baseline is essential for detecting
anomalies such as unusual login attempts from foreign IP addresses, unexpected large
data transfers indicative of data exfiltration, or abnormal traffic spikes that may signal
20 a denial of service (DoS) attack. Identifying deviations from normal network behavior
helps in early detection and response to potential security threats.

Normal network behavior varies by asset type due to differing roles and traffic
patterns. For example, a web server typically handles high volumes of HTTP/HTTPS
requests and database queries, while a workstation sees user logins, web browsing,
25 and file server access. Network devices like routers and switches focus on packet
forwarding. Deviations from these patterns, such as unusual outbound traffic from a
web server or off-hours data access from a workstation, can signal security issues.

One problem faced by common anomaly detection techniques lies in the overwhelming amount of data collected from OT/IT network assets. Traditional methods rely on the collection of extensive amounts of data, including both relevant and irrelevant information.

5 Common anomaly detection techniques which rely on big data often utilize fixed models or predefined thresholds. These approaches lack adaptability and fail to consider the specific risks associated with each asset in the network. A static model is deficient in effectively capturing the evolving nature of network threats, limiting the system's ability to detect sophisticated attacks and anomalies. More general big data
10 models create an overwhelming number of cybersecurity alerts, due to noise in the data, including many false positive alerts. As a result, false positive and false negative detections are common, leading to inefficient resource allocation and potential security failures, and are generally associated with alarm fatigue.

The presently disclosed subject matter includes an innovative approach that
15 addresses various challenges in anomaly detection, including those related to data overload, noise reduction, and risk-awareness specificity.

The disclosed approach includes various innovative principles which include, but are not limited to:

- dynamic selection of assets (referred to herein as "critical assets"), which are
20 likely to be attacked by adversaries. This dynamic selection of assets can fit many types of risks and severities and is updated according to ongoing cybersecurity data updates.

- dynamic identification and collection of asset-specific features from each critical asset.

- dynamic configuration and deployment of asset-specific sensors (or
25 "collectors") to the critical assets in the network for the purpose of asset-specific features collection.

- An automatic asset-specific anomaly detectors generation process that is implemented using a large language model (LLM).

- continuously and dynamically updating the above components (including the selection of the critical assets, the selection of the asset-specific features, the configuration and deployment of asset-specific sensors, and generation of asset-specific anomaly detectors) according to ongoing changes detected in the network
5 and ongoing changes in cybersecurity threats, thereby maintaining an asset-specific anomaly detection scheme continuously adapted according to changes dynamically occurring in the network, and to changes in the relevant cybersecurity threats.

According to a first aspect of the presently disclosed subject matter there is provided a computer-implemented method of adaptive anomaly detection in a
10 computer network, the method comprising:

collecting from assets in the network, respective asset-specific features indicative of normal network behavior in a corresponding asset;

applying a first large language model (LLM) on asset-specific features obtained from the assets, wherein the first LLM is trained to generate, based on
15 asset-specific features received from a given asset, a corresponding asset-specific machine learning (ML)-based anomaly detector configured to receive asset-specific features from the given asset, and detect anomalies in the given asset;

further collecting from each asset respective asset-specific features and applying the respective asset-specific features to the corresponding asset-specific
20 anomaly detector for detecting anomalies in the respective asset.

In addition to the above features, the method according to this aspect of the presently disclosed subject matter can optionally comprise one or more of features (i) to (xxii) below, in any desired and technically possible combination or permutation:

I. Wherein the network is an operational technology network or an
25 information technology network or a network combining operational technology network components and information technology network components.

II. The method further comprising training each corresponding asset-specific anomaly detector to detect anomalies based on asset-specific features obtained from a respective critical asset.

30 III. Wherein a ML-based anomaly detector is an autoencoder machine

learning model.

IV. Wherein the autoencoder is trained to receive a sequence of signals, determine a predicted signal based on the sequence, and identify an anomaly if a difference between the predicted signal and observed signal complies with one or
5 more conditions.

V. Wherein the assets include a subset of critical assets selectively identified based on network parameters (including network discovery output and domain controller output).

VI. The method further comprises:
10 for each critical asset, determining asset-specific features based on combined processing of network discovery output (including asset-specific network parameters) pertaining to the critical asset and cybersecurity data indicative of relevant cybersecurity threats.

VII. Wherein determining asset-specific features is performed by a large
15 language model (LLM) trained to receive network discovery output and cybersecurity data, and identifying assets which are relevant to certain types of cybersecurity threats and attacks.

VIII. Wherein the network discovery output is provided as a network graph, where each node in the network graph represents a respective asset in the network
20 and each vertex connecting between two nodes represents a respective connection between assets in the network.

IX. Wherein the critical assets include a selectively identified subset of assets in the network based on network parameters, wherein the method further comprises selecting the critical assets comprising:

25 applying the LLM on network data (including at least network discovery output and domain control output) trained to execute queries with respect to the assets in the network dedicated for determining network parameters with respect to each asset;

assigning a score to each asset according to responses received to the queries;
30 and

selecting the subset of assets according to the score.

X. Wherein in some examples all assets in the network are considered critical assets.

XI. Wherein collecting from each critical asset respective asset-specific
5 features comprises:

providing asset-specific sensors to critical assets in the network; wherein each asset-specific sensor is configured to monitor data in a respective critical asset and collect the respective asset-specific features.

XII. The method further comprises deploying the asset-specific sensors in
10 the network, and receiving at each asset-specific anomaly detector the asset-specific features collected by a respective asset-specific sensor.

XIII. The method further comprises executing an update procedure, comprising:

for at least one asset, determining updated asset-specific features based on
15 combined processing of updated network data (a term used herein to refer collectively to network discovery output and additional information on the network such as domain controller data) pertaining to assets in the network, and updated cybersecurity data indicative of a relevant cybersecurity threat; and

collecting from the at least one asset in the network, respective updated asset-
20 specific features;

applying the first large language model (LLM) on the asset-specific features obtained from the at least one asset, to thereby obtain, for the at least one asset, a corresponding asset-specific machine learning (ML)-based anomaly detector configured to receive the updated asset-specific features from the given asset, and
25 detect anomalies in the network behavior of the given asset;

XIV. The method comprises further collecting from each critical asset respective updated asset-specific features and applying the corresponding asset-specific anomaly detector for detecting anomalies in the critical asset.

XV. Wherein the update procedure further comprises applying the first LLM
30 for selecting an updated subset of critical assets; and performing the updated

procedure on the update subset of critical assets.

XVI. Wherein the update procedure is initiated in response to receiving data update indicating a change in the network data or an update to the cybersecurity data to maintain anomaly detection in the network which is constantly updated according to current network characteristics and currently active cybersecurity threats.

XVII. Wherein the update procedure is initiated periodically to maintain anomaly detection in the network which is constantly updated according to current network characteristics and currently active cybersecurity threats.

XVIII. Wherein determining updated asset-specific features is performed using a large language model (LLM) trained to receive network discovery output and cybersecurity data and identifying assets which are relevant to certain types of cybersecurity threats and attacks.

XIX. Wherein collecting from each critical asset respective updated asset-specific features comprises:
providing or updating asset-specific sensors in critical assets in the network; wherein each asset-specific sensor is configured to monitor data in a respective critical asset and collect the respective updated asset-specific features.

XX. The method further comprises deploying the respective asset-specific sensor at the respective asset in the network.

XXI. The method further comprises monitoring the network for identifying changes in the network and initiating the update procedure in case a change that complies with one or more predefined conditions is identified.

XXII. The method further comprises, responsive to detecting an anomaly, generating a warning indicating the anomaly.

The presently disclosed subject matter further contemplates a computer system comprising at least one processing circuitry being operatively connectable to a computer network, the processing circuitry comprising one or more computer processors configured to execute a method according to the first aspect disclosed above.

The presently disclosed subject matter further contemplates a computer program product comprising a computer readable storage medium retaining a program of instructions, which, when read by a computer processor, causes the computer processor to perform a method according to the first aspect disclosed
5 above.

The presently disclosed subject matter further contemplates a non-transitory program storage device readable by a computer, tangibly embodying a program of instructions executable by the computer to perform a method according to the first aspect disclosed above.

10 The system, the computer program product, and the non-transitory program storage device, disclosed with reference to the first aspect, can optionally comprise one or more of features (i) to (xxii) listed above, mutatis mutandis, in any desired and technically possible combination or permutation.

According to a second aspect of the presently disclosed subject matter there
15 is provided a computer-implemented method of adaptive anomaly detection in a computer network, the method comprising:

selecting from multiple assets in the network a subset of critical assets;

Determining, for each critical asset, a respective set of asset-specific features that are indicative of normal network behavior (or deviation from normal
20 network behavior) in the respective critical asset;

Collecting, from each critical asset in the subset, the respective asset-specific features;

applying a first large language model (LLM) on the asset-specific features obtained from the critical assets; wherein the first LLM is trained to generate,
25 based on the respective asset-specific features of a given critical asset, a corresponding asset-specific machine learning (ML)-based anomaly detector configured to receive asset-specific features from the given asset, and detect anomalies in the given critical asset;

further collecting, from each respective critical asset in the subset of critical
30 assets, the respective asset-specific features, and applying a corresponding asset-specific ML-based anomaly detector for detecting anomalies in the respective

critical asset.

Wherein, in some examples, selecting a subset of critical assets comprises:

applying a LLM on network data trained to execute queries with respect to the
assets in the network dedicated for determining network parameters with respect to
5 each asset;

assigning a score to each asset according to responses received to the queries;

and

selecting the subset of critical assets according to the score.

Wherein in some examples the method further comprises executing an update
10 procedure, comprising:

re-selecting from the multiple assets in the network an updated subset of
critical assets based on updated network data;

for at least one critical asset in the updated subset, determining updated
asset-specific features based on combined processing of updated network data
15 pertaining to assets in the network, and updated cybersecurity data indicative of
a relevant cybersecurity threat; and

collecting from the at least one critical asset, respective updated asset-
specific features;

applying the first large language model (LLM) on the asset-specific features
20 obtained from the at least one critical asset, to thereby obtain, for the at least one
critical asset, a corresponding asset-specific machine learning (ML)-based
anomaly detector configured to receive the updated asset-specific features from
the at least one critical asset, and detect anomalies therein.

According to a third aspect of the presently disclosed subject matter there is
25 provided a computer-implemented method of adaptive anomaly detection in a
computer network (e.g., IT/OT network), the method comprising:

receiving from sensors operating at critical assets in the computer
network, asset-specific features indicative of normal network behavior in the
critical assets; wherein each sensor is configured for providing asset-specific
30 features from a respective critical asset in the computer network;

applying asset-specific features received from a respective critical asset to

a corresponding asset-specific machine learning (ML)-based anomaly detector configured to process the asset-specific features obtained from the respective critical asset, and detect anomalies in the critical asset;

generating a warning in case an output of the corresponding asset-specific machine learning (ML)-based anomaly detector, indicates an anomaly in the
5 respective critical asset.

Wherein in some examples the method according to the third aspect further comprises:

generating the corresponding asset-specific machine learning (ML)-based
10 anomaly detector, comprising:

applying a first large language model (LLM) on asset-specific features obtained from critical assets in the computer network; wherein the first LLM is trained to generate, based on the asset-specific features received from a given critical asset, a corresponding asset-specific machine learning (ML)-based
15 anomaly detector configured to receive asset-specific features from the given asset, and detect anomalies in the given asset; and

training each asset-specific machine learning (ML)-based anomaly detector.

Wherein in some examples the method according to the third aspect
20 further comprises:

executing an update procedure following determining that a change occurred in network data characterizing the computer network or cybersecurity data characterizing cybersecurity risk in the computer network, comprising:

for at least one critical asset, determining updated asset-specific features
25 based on combined processing of updated network data pertaining to assets in the network, and updated cybersecurity data indicative of a relevant cybersecurity threat; and

collecting from the at least one critical asset, respective updated asset-specific features;

30 applying the first large language model (LLM) on the asset-specific features obtained from the critical assets, to thereby obtain an updated corresponding

asset-specific machine learning (ML)-based anomaly detector configured to receive the updated asset-specific features from the critical asset, and detect anomalies in the critical asset.

5 The presently disclosed subject matter further contemplates a computer system comprising at least one processing circuitry operatively connectable to an IT or OT computer network, the processing circuitry comprising one or more computer processors configured to execute a method according to the second and/or third aspect disclosed above.

10 The presently disclosed subject matter further contemplates a computer program product comprising a computer readable storage medium retaining a program of instructions, which, when read by a computer processor, causes the computer processor to perform a method according to the second and/or third aspect disclosed above.

15 The presently disclosed subject matter further contemplates a non-transitory program storage device readable by a computer, tangibly embodying a program of instructions executable by the computer to perform a method according to the second and/or third aspect disclosed above.

20 The systems, the computer program products, and the non-transitory program storage devices, disclosed with reference to the second and third aspects above, can optionally comprise, where applicable, one or more of features (i) to (xxii) listed above, mutatis mutandis, in any desired and technically possible combination or permutation.

BRIEF DESCRIPTION OF THE DRAWINGS

25 In order to understand the presently disclosed subject matter and to see how it may be carried out in practice, the subject matter will now be described, by way of non-limiting examples only, with reference to the accompanying drawings, in which:

Fig. 1 is a high-level block diagram schematically illustrating computer system
10 configured with adaptive asset-specific anomaly detection technology, in accordance with an example of the presently disclosed subject matter;

30 **Fig. 2** is a high-level flowchart showing operations carried out as part of an

adaptive asset-specific anomaly detection process, in accordance with an example of the presently disclosed subject matter;

Fig. 3 is a diagram schematically illustrating discovery and graph generation computer **100** and network risk-assessment computer **200**, in accordance with an example of the presently disclosed subject matter;

Fig. 4 is a diagram schematically illustrating asset-specific, LLM-based anomaly detector computer **300**, in accordance with an example of the presently disclosed subject matter;

Fig. 5 is a flowchart showing a more detailed view of operations carried out as part of the adaptive asset-specific anomaly detection process, in accordance with an example of the presently disclosed subject matter;

Fig. 6 is a flowchart showing operations carried out as part of the generation and application of asset-specific, machine learning (ML)-based detectors, in accordance with an example of the presently disclosed subject matter; and

Fig. 7 is a flowchart showing operations carried out as part of an update procedure in an adaptive asset-specific anomaly detection process, in accordance with an example of the presently disclosed subject matter.

DETAILED DESCRIPTION

In the drawings and descriptions set forth, identical reference numerals indicate those components that are common to different embodiments or configurations. Elements in the drawings are not necessarily drawn to scale.

Unless specifically stated otherwise, as apparent from the following discussions, it is appreciated that, throughout the specification, discussions utilizing terms such as "collecting", "applying", "processing", "determining", "selecting", "deploying" or the like, include an action and/or processes of a computer that manipulate and/or transform data into other data, said data represented as physical quantities, e.g. such as electronic quantities, and/or said data representing the physical objects.

The terms "computer", "computer system", "computer device", "computerized device" or the like, should be expansively construed to include any kind of hardware-based electronic device with one or more data processing circuitries (e.g., digital signal processor (DSP), a GPU, a TPU, a field programmable gate array (FPGA),
5 an application specific integrated circuit (ASIC), microcontroller, microprocessor etc.). Each processing circuitry can comprise, for example, one or more processors operatively connected to computer memory, loaded with executable instructions for executing operations, as further described below.

The operations in accordance with the teachings herein may be performed by
10 a computer specially constructed for the desired purposes, or by a general-purpose computer specially configured for the desired purpose by a computer program stored in a computer readable storage medium.

As used herein, the phrase "for example," "such as", "for instance" and variants thereof, describe non-limiting embodiments of the presently disclosed subject matter.
15 Reference in the specification to "one case", "some cases", "other cases", or variants thereof, means that a particular feature, structure, or characteristic, described in connection with the embodiment(s), is included in at least one embodiment of the presently disclosed subject matter. Thus, the appearance of the phrase "one case", "some cases", "other cases", or variants thereof, does not necessarily refer to the
20 same embodiment(s).

It is appreciated that certain features of the presently disclosed subject matter, which are, for clarity, described in the context of separate embodiments, may also be provided in combination in a single embodiment. Conversely, various features of the presently disclosed subject matter, which are, for brevity, described in the context of
25 a single embodiment, may also be provided separately, or in any suitable sub-combination.

In embodiments of the presently disclosed subject matter, fewer, more and/or different stages than those shown in **Figs. 2, 5, 6, and 7** may be executed. In embodiments of the presently disclosed subject matter, one or more stages illustrated

in the figures may be executed in a different order, and/or one or more groups of stages may be executed simultaneously.

Figs. **1**, **3**, and **4** illustrate general schematics of the system architecture in accordance with certain examples of the presently disclosed subject matter. Elements in Figs. **1**, **3**, and **4** can be made up of any combination of software and hardware and/or firmware that performs the functions as defined and explained herein. Elements in Figs. **1**, **3**, and **4** may be centralized in one location or dispersed over more than one location. For example, each one of computers **100**, **200**, and **300** can be located at a different geographical location, remote from the other computers, or at the same geographical location directly connected to one another.

Furthermore, in some examples of the presently disclosed subject matter, the system may comprise fewer, more, and/or different elements than those shown in Figs. **1**, **3**, and **4**. For example, Figs. **1**, **3**, and **4** show several separate computers, each dedicated for executing certain functions of the system, however it would be clear to any person skilled in the art that the functionalities can be divided otherwise. For instance, in an alternative system-design, different functions assigned to computer **100** or computer **200** can be otherwise distributed to several computers. For instance, network monitoring module **21** in computer **200** can be otherwise implemented in a separate computer device. Likewise, various elements described as distributed over different computers can be otherwise consolidated into a single computer device. For example, operations assigned to computer **200** and computer **300**, can be implemented in a single computer device.

Bearing the above in mind, attention is drawn to Fig. **1**, which is a high-level diagram schematically illustrating a computer system **10** configured with adaptive asset-specific anomaly detection technology, in accordance certain examples of the presently disclosed subject matter. Fig. **1** is a general example which demonstrates various principles of the presently disclosed subject matter. By way of example, components of system **10** shown in Fig. **1** can be connected over any type of communication network, including, for example, any one of the following: the

Internet, a local area network (LAN), wide area network (WAN), metropolitan area network (MAN), any type of telephone network (including for example PSTN with DSL technology) or mobile network (including for example 4G or 5G mobile communication technologies), or any combination thereof.

5 By way of example only, system **10** is shown to include several computers, each configured to perform a certain general task, including network discovery and graph generation computer **100**, risk-assessment computer **200**, and asset-specific, LLM-based adaptive anomaly detection computer **300**. Fig. **2** is a high-level flowchart showing operations carried out as part of an adaptive asset-specific anomaly detection
10 process, in accordance with certain examples of the presently disclosed subject matter. For ease of understanding and by way of example only, the various operations shown in Fig. **2** are described below in conjunction with the components of Fig. **1**.

 At block **201** network mapping that specifies the various assets in the network, their relative location, and their properties, is obtained. In some examples, network
15 mapping is provided as network discovery output. Network discovery is a process dedicated for the identification and understanding of the various assets that constitute an operational technology (OT) network, and for establishing their associations with other Information Technology (IT) systems. By applying network discovery in an organization, valuable information regarding the organization can be
20 obtained, including the identification, characterization, and mapping of the different assets which constitute the OT network, and the identification of connectivity and communication pathways between these assets.

 Network discovery output can be stored in a graph data-structure (also referred to herein as "network graph") for easy and efficient data processing, retrieval,
25 and display. Assets discovered in the network are represented by respective nodes in the graph, and each connection between assets is represented by a respective vertex in the graph connecting between two nodes. Following generation of the graph, various tools dedicated for querying and retrieval of information from the graph can be used for this purpose. In some examples, data discovery and graph generation are

executed by computer **100** in Fig. **1**.

Fig. **1** also schematically illustrates OT network **400**, which includes various assets which constitute the network, including, router, firewalls, switches, hubs, bridges, and endpoints (e.g., server computers or working stations). Notably, networks
5 can include additional types of assets such as gateways and sensors. An OT network can comprise many thousands of assets (in some cases, 100,000 or more) and the graph can likewise include a similar number of graph elements (e.g., nodes or vertices).

At block **202** cybersecurity data is obtained from various sources, such as
10 cybersecurity organizations and frameworks like MITRE ATT&CK®. The cybersecurity data includes, for example, Common Vulnerabilities and Exposures (CVE) and attack techniques (i.e., a specific cybersecurity attack technique used by adversaries). CVE includes warnings about potential cybersecurity risks and alerts on ongoing or new cyberattacks or campaigns, software/hardware vulnerabilities, misconfigurations that
15 create vulnerabilities, etc.

As further explained below, cybersecurity data is used for identifying assets in the network which are relevant to a specific cybersecurity risk, and provide the asset with appropriate measures dedicated for detecting cybersecurity attacks on the asset which are related to that cybersecurity risk.

20 At block **203** a main network risk-assessment procedure is executed. As further explained below, network discovery output (including the network graph generated following network discovery) and cybersecurity data are used together in the process of network risk-assessment and LLM-based anomaly detection.

In some examples, a preliminary risk-assessment procedure is executed before
25 the main network risk assessment risk-assessment procedure. During the preliminary procedure a subset of assets are identified as important (referred to herein as "critical assets") and selected for further processing in the main risk-assessment procedure. This selective approach enables to focus the cybersecurity protection efforts on these value points, rather than on the entire network, which often provides inefficient and

ineffective protection. By focusing anomaly detection only on a selected subset of assets in the network that exhibit higher risk, resource consumption is reduced, and the rates of false positive and false negative anomaly detections are decreased.

Notably, in some cases (e.g., in case of a small network and/or where sufficient
5 processing resources are available) a preliminary risk-assessment procedure is not performed, and the entire process is applied on the entire network without selection, where all assets in the network are considered as critical assets.

During the network risk-assessment procedure a risk profile of the different assets in the network is determined, based on parameters which are extracted from
10 the network discovery output (and additional network data such as the domain control) and the cybersecurity data. The risk profile includes information on the specific types of cybersecurity risks which are relevant to the different assets.

In addition, and as further explained below, relevant features (data elements) that match the relevant cybersecurity risks are determined for each critical asset
15 (referred to herein as "asset-specific features"). Asset-specific features include data which is indicative of normal behavior in the respective asset and can therefore be used to identify changes in the normal behavior for the purpose of anomaly detection. According to some examples, risk-assessment computer **200** is configured to execute the risk-assessment procedure.

At block **205** asset-specific features are collected from each respective critical
20 asset. At block **207** an asset-specific anomaly detector (also referred to herein as "asset-specific ML-based detector") is generated for each critical asset in the network. As explained below, the collected asset-specific features are used as input to a large language model (LLM) trained to generate, based on the respective asset-specific
25 features retrieved from a critical asset, a corresponding asset-specific anomaly detector, configured in turn to continuously receive asset-specific features from the critical asset, and detect anomalies in that asset.

Once asset-specific anomaly detectors are available, each detector is used for processing input of asset-specific features that are continuously being collected from

a particular critical asset, and for providing output indicating whether any anomaly has been identified in the critical asset (**209**). The specific collection of critical assets, their corresponding asset-specific features, and asset-specific anomaly detectors, is also referred to herein as an "asset-specific anomaly detection scheme".

5 As mentioned above, the preliminary risk-assessment procedure and the main risk-assessment procedure use various parameters characterizing assets in the network, many of which are obtained from the network discovery output and cybersecurity data obtained from external sources, such as Common Vulnerabilities and Exposures (CVEs) and attack techniques. Accordingly, changes that may occur in
10 the network and influence any of the network parameters or updated cybersecurity data (updated CVEs which indicate the possibility of new types of threats to the network or new cybersecurity reports or alerts), may affect the risk profile of any one of the assets in the network and therefore affect the assets which are designated as critical assets. Furthermore, these parameters also affect the asset-specific features
15 which are collected from each critical asset. Accordingly, the network is continuously monitored, and in case a change identified in the network or in updated cybersecurity data, is recognized as a change that could affect the selected critical assets and/or the asset-specific features of critical assets, an update procedure is triggered, in which the network risk-assessment is repeated and one or more asset-specific ML-based
20 detectors are updated according to the updated risk-assessment outputs (**211**). This adaptive approach enables to maintain an asset-specific anomaly detection scheme that constantly evolves together with the dynamics of the network and the threats, and thus provides ongoing, dynamic, accurate, and efficient anomaly detection.

Fig. **3** shows a more detailed view of graph generation computer **100** and
25 network risk-assessment computer **200**, and Fig. **4** shows a more detailed view of LLM-based adaptive anomaly detection computer **300**. Fig. **5** is a flowchart showing a more detailed view of operations carried out by computers **100**, **200**, and **300**.

At block **501** network discovery is executed. Computer **100** in Fig. **3** is configured to execute network discovery and generate a respective graph **20**. Notably,

functionalities of computer **100** can be divided differently, for example it can be implemented on multiple computers (e.g., each module illustrated as part of computer **100** can be implemented on a dedicated computer device). If network ontology (e.g., in the form of a graph) is already available, for example, it has already
5 been executed for a different reason, existing information can be used instead of executing network discovery.

There are various methods known in the art for performing network discovery. One known tool is Wireshark, developed and maintained by the Wireshark Foundation. Another known tool is Nmap by Insecure.Org. Nmap is a network
10 scanning tool used for discovering hosts and services on a computer network. It operates by sending specially crafted packets to target devices and analyzing their responses. Nmap can identify open ports, detect service versions, and infer operating systems. Its functionality can be extended with scripts for tasks like vulnerability assessments and advanced host discovery. Widely used in cybersecurity, Nmap helps
15 in network inventory, security auditing, and penetration testing.

In some examples, network discovery is executed as an automatic network discovery process carried out using a trained large language model (e.g., by LLM training module **120**), where the LLM is trained to comprehensively "understand" configuration files. Configuration files are special types of files that store information
20 defining various assets and their interactions in an OT or IT network. Each configuration file describes a respective network asset. Each configuration file contains information (referred to herein as "configuration data") of a respective asset.

The information in configuration files includes for example:

- information on the assets in the network, including data such as IP
25 addresses, hostname, network settings, access control (user access privileges), protocols, applications, etc.
- Information on the network topology, including data on physical and logical relations and links between different assets, the links bandwidth, network security configurations and policies, cybersecurity parameters, routing protocol,

protocol specific configuration, etc.

The LLM incorporates machine learning techniques, and is trained, using a large dataset of configuration files, to extract from configuration files in a network (OT and/or IT networks) relevant configuration data.

5 According to some examples, the training dataset (prepared for example by data preparation module **110**) includes both actual configuration files, as well as synthetic configuration files generated by an innovative approach. The synthetic files can be generated by virtually producing numerous possible network architectures. This approach ensures a diverse and extensive training dataset, enhancing the
10 accuracy and effectiveness of the LLM in understanding and extracting data from real-world configuration files.

 During execution, the LLM is applied (e.g., by LLM execution module **130**) on configuration files obtained from actual OT/IT networks for automatically extracting the relevant configuration data from the files, and generating the network discovery
15 results (e.g., network ontology).

 As further mentioned above in some examples, a graph data-structure representing the network discovery results is generated (**503**). In some examples, the graph **20** is generated by graph generator **139** and stored in a dedicated computer storage, graph-DB **140**, which can be implemented, for example, as part of system **100**
20 or in a separate computer, dedicated for this purpose. The LLM-based automatic discovery process is described in detail in Israeli Patent Application No. 304326, filed by the applicant on July 9, 2023, which is incorporated herein by reference in its entirety.

 At block **502** cybersecurity data is obtained. In some examples, network risk
25 assessment computer **200** comprises cybersecurity data hub **22**, configured to continuously collect and store cybersecurity data obtained from various sources. The collected cybersecurity data is provided to the LLM-based risk assessment module **25**, as further discussed below.

 As explained above, in some cases a preliminary risk-assessment process is

executed to identify critical assets, which are identified to be associated with high risk. These assets include, for example, assets that perform critical functions, store important information, or assets which are located on critical paths in the network.

An example of a critical asset is a file server that is used for storing important
5 information, where any damage or loss to the stored files may pose significant risk or challenges to the owners of the files, such as data loss, Intellectual Property loss, business interruption, downtime, privacy breach, etc. Another example of a critical asset is an endpoint (e.g., working station), which has no particular importance (does not have a critical function and does not store important data), but is located on a
10 critical path leading to another critical asset (e.g., file server). In such cases, lateral movement implemented by an attacker that has penetrated the network in an attempt to advance towards valuable data, is likely to (and in some cases must) pass through this endpoint. A further example of a critical asset is one which is located at the periphery of an OT network, connecting the network to other external networks
15 (e.g., a router connected to the Internet), and may be used for initial breach into the network. A yet further example of a critical asset is an asset (e.g., endpoint) that is configured with a permissive access control list (ACL) (e.g., configured with an extensive list of users granted with access privileges to the asset), and therefore may be vulnerable to privilege escalation attacks.

20 At **505** a preliminary risk-assessment procedure is executed. During this procedure critical information on the network extracted from the network discovery output (e.g., from the configuration files of the assets), is processed, and various parameters characterizing the assets in the network are analyzed and used for selecting critical assets. These parameters, referred to herein as "network
25 parameters", are described in more detail with respect to block 507 below, and include, for example, function of assets, configuration of assets, relative location in the network, physical characteristics, connectivity, access control list (ACL), other security settings and vulnerabilities (e.g., whether a password is required), filesystem parameters, user activity, system activity, installed applications, etc.

In some examples, critical assets are selected based on their location (determined based on the network discovery output data) and function in the following manner. Assets with critical function are identified, and one or more paths that connect each of these assets with vulnerable network entry points from which an
5 attack is likely to start, are identified. The assets located along these paths are analyzed, and part, or all, of these assets are identified as critical assets.

In some examples, critical assets are selected based on a risk-score calculated for each asset, based on the parameters characterizing the asset. High risk-score values indicate that an attack on the corresponding asset may result in substantial
10 damage to the entire network, to an important component in the network, or to important data stored in the network. For example, the subset of critical assets may include all assets that are assigned with a risk-score which is greater than a certain threshold value. Additionally, or alternatively, the subset of critical assets may be limited to 'n' assets assigned with the highest risk-scores. As would be clear to any
15 person skilled in the art, calculation of the risk-score can be implemented in many different ways, which are not described here in detail. For example, a score can be assigned for each one of a plurality of categories (e.g., ACL, connectivity, vulnerabilities, hardware, etc.) based on the observed status according to a predefined scoring scheme, and a collective risk-score can be calculated based on the
20 compilation of the score of each category.

At block **507** a (main) network risk-assessment procedure is executed. In the following discussion it is assumed that a preliminary risk-assessment procedure has been executed and a subset of critical assets have been identified. However, in other examples this may not be the case, and the following operations may be applied on
25 all assets in the network.

As mentioned above, during the network risk-assessment procedure, parameters extracted from the network discovery output and cybersecurity data obtained from external sources, are processed together to determine, for each critical asset, an asset-specific cybersecurity risk profile. In some examples, the risk profile

can be used for determining asset-specific risk mitigation measures, which are selected to mitigate the specific type of risks found in a certain asset. The risk profiles include information indicating the type and level of cybersecurity risk that is associated with the respective asset.

5 Examples of network parameters extracted from the network discovery output and additional network data (e.g., domain controller output), which are considered when generating the risk profiles, include:

- The physical characteristics of the asset, as different types of devices have different vulnerability which depend on their physical structure.
- 10 - The functionality of the asset, where certain functions (e.g., file servers) are associated with a greater risk than others.
- The relative location of the asset within the network, including its proximity to the periphery of the network and to other critical assets such as assets that store valuable data or have some other valuable function, and their
- 15 proximity to critical paths which lead to other critical assets.
- The connectivity of the asset, i.e., to how many other assets each asset is connected, where greater connectively may imply greater risk.
- Users access privileges. This can be done, for example, by overlaying the permission granted to users in the network over the various accesses, to
- 20 thereby obtain the various permissions granted to different users for each asset in the network (including for example, Role-based access control (RBAC) and group-based access control (GBAC)).

Additional parameters extracted from domain controller output, include for example:

- 25 - Different operating systems (OSs) or other software installed on each asset, where different OSs exhibit different vulnerabilities.
- Different updates made to each station, where such updates may have significant influence on the vulnerability of the asset.

Notably, information can be extracted from the domain controller by executing

30 specifically designed queries. While the above parameters characterize specific assets

in the network, the cybersecurity data includes information on current cybersecurity threats and vulnerabilities.

Cybersecurity data can also be used for classifying different cybersecurity threats into respective types. An example of risk classification that can be used is
5 MITRE ATT&CK® classification, which includes: Specification- Reconnaissance, Resource Development, Initial Access, Execution, Persistence, Privilege Escalation, Defense Evasion, Credential Access, Discovery, etc.

According to some examples, during the risk-assessment procedure, the cybersecurity data is compared with the network parameters of each critical asset to
10 thereby identify the risks which are relevant to the asset. The risk profile of each critical asset is updated with information indicating the cybersecurity risks (and risks type) which are relevant to the asset.

Assuming, for example, cybersecurity data indicates a cybersecurity risk of a cyber ransomware campaign that is currently being executed on machines running
15 Windows 10 connected in a certain way to a certain server. This information can be matched to the network parameters of assets in the network for identifying assets, which use Windows 10 and are connected in the prescribed manner to the prescribed server, and are therefore at risk from the ransomware campaign.

Notably, in some examples the operations described with reference to blocks
20 505 and 507 are unified in a single procedure that performs both tasks. By identifying critical assets and determining their respective risk profiles in one streamlined process, the efficiency of the process is improved.

This integrated process involves extracting critical information on the network from the network discovery output and cybersecurity data from external sources as
25 described above, and using this information to determine critical assets and their respective risk profiles. As mentioned above, critical assets can be selected, for example, based on risk-scores calculated considering the network parameters, together with the security data.

At block 509 asset-specific features are determined for each critical asset. As

mentioned above, asset-specific features, collected from each critical asset, are used for generating asset-specific anomaly detectors, where each detector is specifically tailored for detecting anomalies in a corresponding asset according to the asset's type and characteristics. Asset-specific features collected from each critical asset depend on various parameters, such as the type of cybersecurity risk/attack which is being addressed, function of the asset, type of data related to the cybersecurity attack (e.g., data used or targeted by the attack) the asset receives, transmits and processes, and the way the data is being processed or otherwise manipulated by the asset. Thus, based on the relevant risks and risk types defined in the risk profile of each asset, the relevant asset-specific features can be determined.

A few non-limiting examples of asset-specific features are provided. For monitoring assets related to user login activity asset-specific features can include for example, number of concurrent logins for each user, device used for each login attempt, operating system version of the device used for each login attempt, browser or application used for each login attempt, country or region of the source IP address for each login attempt, amount of data transferred during each login session, The length of time for each login session, etc. For monitoring assets related to file access activity assets specific features can include for example, the size of each file accessed, the location of each file accessed, the type of file accessed, the frequency of access for each file, the time of day when each file was accessed, the date when each file was last accessed, the date when each file was created, the date when each file was modified, the path of the file accessed, etc. For assets related to network traffic activity asset-specific features can include for example, the source and destination IP addresses of each network packet, the size of each packet, the type of protocol used (e.g. HTTP, FTP, SSH), the frequency of network traffic (e.g. how often a particular user/device generates network traffic), the geographic location of the source or destination IP address, the country or region of the source or destination IP address, etc.

According to some examples, generation of risk profiles and determination of asset-specific features, and possibly also the selection of critical assets (blocks 505,

507 and 509), are accomplished with the help of a machine learning model specifically trained for this purpose. In some examples, the machine learning model is a large language model (LLM, also referred to herein as "risk-assessment LLM") trained to identify, for each asset, (or each critical asset), the respective relevant risks and risk types, and, based on the risks and/or risk type, determine, for a particular risk, the appropriate features which can be used for detecting anomalies that indicate a cybersecurity attack associated with that risk.

An example of a training process of the risk-assessment LLM is described herein below. The training process can be divided into three layers, each layer being dedicated for performing a certain part of the training process, including a fundamental layer, instruct layer, and reward and reinforcement layer. In the illustrated example, LLM training module 23 (optionally implemented on a dedicated computer device) comprises three respective modules (231, 233, and 235) each module configured for executing operations related to one of the layers.

The fundamental layer (implemented for example, by module 231) is configured to receive a training set of cybersecurity data as input, and to execute word embedding on the input, where the cybersecurity data is tokenized and converted into a respective sequence of word vectors (referred to herein as "embedded tokens"), which preserve the semantic relation and includes next token prediction. The fundamental layer is further configured for training a transformer model (e.g., autoregressive transformer model) which is a neural network model that learns to predict the next token in a sequence, given the previous tokens in the sequence. The transformer model also includes a self-attention mechanism that enables the model to identify relationships and dependencies between different tokens in the input and improve next token prediction.

The instruct (or "instruction") layer (implemented for example, by module 233) is configured to further train the model to follow specific instructions. The instruct layer can be a separate component, or integrated within the model's input pipeline, depending on the specific implementation. The instruct layer is dedicated for fine-

tuning the fundamental layer to improve user interaction with the model. In a deeper sense, the instruct layer causes the weights of the neural network to change, so that the output of the model is better suited to the instruction received as input.

The instruction layer of the LLM enables users to provide guidance to the LLM, 5 guiding the model to generate the desired response to each different instruction (also known as "prompt"). The instruct layer receives, as input, instruction-result pairs, and learns to predict the next word in a sequence, given the instructions and the previous words, such that the appropriate result is provided to a given instruction. The training dataset therefore includes many examples of a variety of pairs of instructions and the 10 respective desired response. During training, the LLM is fed with the instruction-response pairs to facilitate training of the model to provide, in response to a specific prompt, the appropriate risk score and/or asset-specific features.

According to some examples, multiple types of instructions are generated. A first type of instruction-response pairs is dedicated for extracting information from 15 the cybersecurity data, which, as mentioned above, includes various cybersecurity reports, CVEs, etc. In the example of MITRE cybersecurity data, the instructions are trained to extract information such as the attack techniques, the specific risks, the type of risk, their classification, the related parameters of the assets which are relevant to certain risks, etc.

20 A second type of instruction-response pairs are dedicated for determining critical assets (in some examples, essentially executing the risk assessment procedure) by processing (e.g., in the fundamental layer) network data including network discovery output and domain controller output, and extracting network parameters mentioned above, such as physical characteristics, functionality, location in the 25 network (proximity to critical paths), neighboring assets, connectivity, software (e.g., OS), user access privileges, etc. The queries are configured to investigate assets about the network parameters mentioned above, for example, querying about the location and or function of a certain asset, querying whether the asset is connected to the Internet, querying whether the asset has an open port, querying about the number of

connections an asset has, etc. Based on the responses received for each query, a score is calculated for each asset, and the critical assets are selected according to the score as explained above (e.g., by selecting assets with a score greater than a certain threshold value).

5 A third type of instruction-response pairs is dedicated for comparing between the data extracted from the cybersecurity data (the output of the first instructions type) and the network discovery graph (the output of the second instructions type), representing the network, thus identifying assets which are relevant to a certain type of cybersecurity threats and attacks (essentially executing the main assessment
10 procedure).

 The LLM also determines a set of relevant features and assigns the set of features ("asset-specific features") to the asset. In some examples, a fourth type of instruction-response pair is dedicated for associating between information on a given asset, including network parameters such as the type of asset, its function, its location
15 obtained from the network discovery output data and cybersecurity risk data related to the asset, and the respective asset-specific feature of the given asset. The relevant features are those which contribute to the detection of normal and abnormal network behavior at the given asset.

 Notably, if a critical asset is not found to be related to any particular threat, a
20 default or general set of asset-specific features can be used. In some examples, the LLM is trained to transform the network discovery output to textual data, to enable contextual comparison between the cybersecurity data, which is textual, and the network discovery output.

 In some examples, the LLM is configured to select a query relevant to a certain
25 asset from an extensive list of instructions, execute the query, and compare, semantically, the risks and the attack techniques. Considering the example mentioned above, the third type of instructions can be used for determining which assets in the network are relevant to a cyber ransomware campaign being executed on Windows
10 machines connected in a certain way to a certain server.

This hybrid approach enables to train an LLM to process cybersecurity data and extract information in "an ontology aware" manner, such that the risks and attack techniques are selectively and intelligently associated with the relevant assets in the network, based on the discovery output which characterizes the assets in the network.

5 In some examples, computer **200** may further include a reinforcement layer (implemented by module **235**) configured to learn from feedback, and thus further fine-tune the model and provide more accurate results, which are also, in general, more human-accustomed. This fine-tuning process guides the model towards generating more desirable outputs for a target task, in this case providing more accurate information on assets in the network during network discovery in response to a respective instruction. As with the instruct layer, the reinforcement layer also causes the weights of the neural network to change, so that the output of the model is augmented. Once the training process is complete, the trained model is made available for execution.

15 During execution, the trained risk-assessment LLM model is applied on the cybersecurity data and the network graph to determine the critical assets in the network and their respective asset-specific features. In some examples, selection of critical assets and determination of asset-specific features is executed by an LLM-based risk-assessment module **25**, implemented in network risk-assessment computer **200**. In some examples, information on the critical assets and the respective asset-specific features determined for each one of the critical assets, is transmitted to an asset-specific LLM-based anomaly detection computer **300**.

25 At block **511** sensors (or "collector") are configured (**511**) and distributed in the network (**513**). Once the asset-specific features of each critical asset have been determined, a respective asset-specific sensor (also referred to as "sensor" or "collector" in short, which is asset specific and is dynamically updated based on updates to the network and/or cybersecurity data) is configured to operate at each critical asset, to monitor network traffic and other activities in the asset, and continuously collect the asset-specific features. They are dynamically updated based

on updates to the network and/or cybersecurity data.

The asset-specific features, which are collected by each sensor, are used for detecting anomalies occurring in the respective asset, as further explained below (515). A sensor can be, for example, a software program that is installed at a respective asset and is configured to collect data as explained above. In some examples, each
5 sensor can receive, from the risk-assessment LLM, a list of asset-specific features.

In some examples, network risk-assessment computer 200 further includes sensors generation module 27 configured to generate, configure, or update the asset-specific sensors, and sensors deployment module 29 configured to deploy each asset-specific sensor to the respective asset.
10

At block 517 the asset-specific features are used for generating asset-specific anomaly detectors (or "detectors" in short), where each detector is specifically tailored for detecting an anomaly in the respective asset. As explained above, different assets have different characteristics, exhibit a different behavior, and face different types of threats. For example, a file server should be protected against attacks
15 directed on the files (e.g., ransomware), but not necessarily against lateral movement, whereas an endpoint located on a critical path, without any special functionality, should be protected against lateral movement. Thus, each asset-specific anomaly detector is generated to particularly match the threat associated with the respective critical asset it aims to protect.
20

To this end, another machine learning model is used for automatically generating asset-specific anomaly detectors. In some examples, the machine learning model is implemented as a large language module (referred to herein also as "detectors-generating-LLM"). Each detector can be implemented for example, as a
25 machine learning (ML) model specifically configured for detecting an anomaly occurring at a specific critical asset. Turning to Fig. 6, it shows an example of operations carried out as part of a ML-based asset-specific anomaly detectors generation process by detectors-generating-LLM. Operations described with respect to Fig. 6 can be executed for example by asset-specific, LLM-based anomaly detection

computer **300**.

At block **601** (also block **504** in Fig. **5**) a detectors-generating-LLM is trained to automatically generate multiple program codes, each program code dedicated to operating as a detector in a specific critical asset (executed for example, by LLM training module **31** in computer **300**). In some examples, each asset-specific anomaly detector is implemented as a machine learning (e.g., Deep Learning) code specifically designed to detect anomalies in a respective critical asset, based on the corresponding asset-specific features determined for that asset.

According to one non-limiting example, ML-based asset-specific anomaly detectors are implemented as autoencoders, or, more specifically, as time series autoencoders. An autoencoder is trained (e.g., unsupervised autoencoder) to generate a series of signals, each signal being generated based on one or more observed asset-specific features received from an asset at a certain timestamp (e.g., over a time window of a few seconds), study the sequence of signals, and determines a predicted signal based on the sequence. The ML-based asset-specific anomaly detector is further trained to compare the next observed signal with the predicted signals, and identify an anomaly if a difference between the predicted signal and observed signal complies with one or more conditions.

The autoencoder includes an encoder and a decoder. In some examples, the encoder uses a Long Short-Term Memory (LSTM) layer to process the sequence input and reduce its dimensionality. The decoder also uses LSTM layers, but in reverse to the encoder: it attempts to reconstruct the original sequence from the lower-dimensional representation.

The autoencoder is trained to compress the input data, in the current case sequences of normal behavior data (asset-specific features), into a lower-dimensional representation, while striving to minimize the reconstruction error between the input sequence and the output sequence generated by the decoder, and thus capture the significant features in the data.

Following training, the autoencoder can be used for anomaly detection. Any

new sequence is fed through the model to calculate the reconstruction error. Normal sequences should have a lower reconstruction error, while anomalous sequences (which the model was not trained on) will likely have a higher reconstruction error. Normally, a reconstruction error threshold is set, defining an error which is considered
5 an anomaly. In some examples, the threshold is determined based on the distribution of reconstruction errors on a validation dataset.

Notably, an autoencoder, and particularly LSTM autoencoders described above, is merely a non-limiting example, and the presently disclosed subject matter contemplates other possible algorithms, as would be clearly apparent to any person
10 skilled in the art.

The training data includes many examples of program codes (e.g., Python codes) of machine learning models which can provide this input, where each program code is associated with the respective asset-specific features which were used in its creation, thus training the detectors-generation-LLM to automatically generate a
15 specific anomaly detection program code for a given set of asset-specific features.

In some examples, the detectors-generating-LLM is implemented using the three layers described above, fundamental layer, instruction layer, and reinforcement layer (implemented by modules **311**, **313** and **315**, respectively). The fundamental layer is applied on the program code examples. The instruction layer can include, for
20 example, at least one instruction-result pair. The instruction requests the model to build an asset-specific anomaly detector code and includes a code snippet of an anomaly detector and the set of asset-specific features. The instruction result includes the respective asset-specific anomaly detector code, built using the set of features. A reinforcement layer can be applied for making the code more readable and select the
25 best structured code from various possible codes.

Once trained (and the weights are saved), the LLM can be executed to automatically generate a plurality of program codes, each for a respective asset-specific anomaly detector designated for detecting anomalies in a certain asset in the network (**603**; e.g., by LLM execution module **303** in computer **300**). The LLM receives

as input a respective set of asset-specific features collected from each critical asset by the respective asset-specific sensor, and generates, for each critical asset, a respective asset-specific anomaly detector. Fig. 4 shows schematically a collection of ML-based asset-specific anomaly detectors 1-n (**310**), where each detector is configured to
5 receive asset-specific features from a respective asset-specific sensor operatively connected to a certain asset in network **400**.

Since, according to this example, each detector is implemented as a machine learning code, following their creation, each asset-specific anomaly detector is trained (**605**; e.g., by detectors training module **305**). During training, asset-specific features
10 are continuously collected from the respective critical asset and sent to computer **300** for training the respective ML-based anomaly detector. In some examples, computer **300** includes multiple training modules, each dedicated for training a respective ML-based asset-specific anomaly detector, to enable parallel training of the detectors, and reduce training time.

15 Once the ML-based asset-specific anomaly detectors have been trained, each detector can be used for receiving a flow of asset-specific features collected from the respective critical asset, and determining, based on the received features, whether an anomaly is identified in the critical asset (**607**; **519**). In some examples, at the end of training, a collection of asset-specific anomaly detectors (1-n) are available, each
20 specifically configured to detect anomalies at a certain critical asset, based on asset-specific features received from a respective sensor assigned to that same critical asset.

In case an anomaly is identified by any one of the asset-specific anomaly detectors, a respective alert and/or report can be generated and provided (e.g., to a Chief Information Security Officer, or anyone from the cybersecurity team). An alert
25 can be provided in any way conceivable, including email, SMS, WhatsApp, displayed on a computer screen (e.g., in a cybersecurity dashboard), etc.

As explained above, critical assets can be identified according to a calculated risk-score based on a variety of network parameters associated with the respective asset (obtained for example from the network discovery and domain controller), such

as physical characteristics, functionality, location in the network, neighboring assets, connectivity, software (e.g., OS), user access privileges, etc. Asset-specific features are also determined based on such parameters, as well as the risk profile of the asset (indicating the relevant risks). As mentioned above with respect to block **211** in Fig. **2**,
5 parameters that characterize an OT or IT network and cybersecurity data are dynamic by nature and may constantly change, and these changes may affect the risk-score assigned to each asset and the respective asset-specific features.

Changes in the network include, but are not limited to:

- 10 - Changes to the physical layout of the network, which may result from removal or addition of assets to the network or changes in the location of existing assets within the network.
- Updates made to software installed on assets (e.g., OS version update).
- Changes in the function of assets in the network.
- Changes to the connectivity of different assets.
- 15 - Changes to user privileges (ACLs) of different assets.
- Changes in cybersecurity updated CVEs which indicate the possibility of new types of threats to the network or new cybersecurity reports or alerts.

The presently disclosed subject matter further includes an update procedure, dedicated for continuously and automatically updating the asset-specific anomaly
20 detection scheme according to changes detected in the network data and/or the cybersecurity data. The adaptive character of the process enables to continuously maintain a highly accurate and efficient anomaly detection, notwithstanding the dynamic nature of the network and/or cybersecurity threats. The update procedure can be initiated automatically in response to certain indications or events.

25 Referring to Fig. **7**, it shows operations carried out as part of an update procedure. Notably, some operations described with reference to Fig. **7** are the same as those described above with reference to Fig. **5** and are therefore assigned with the same reference numerals. At the onset of the update procedure, the risk-assessment procedure is executed (**701**; **507**). In some examples, the risk-assessment procedure

(and optionally also the preliminary risk-assessment procedure) is executed periodically. Alternatively, or additionally, the risk-assessment procedure can be triggered in response to updated cybersecurity data received by the system (e.g., updated CVEs and/or attack techniques) and/or a change that is identified in the OT/IT
5 network and recognized as one which may have an effect on the risk-score of the assets and/or the respective asset-specific features. For example, in case assets are removed or added to a network, an update procedure may be triggered. In some examples, the network graph generation process may be repeated periodically and/or in response to a detected event (e.g., addition or removal of an asset to the network),
10 and the update procedure is initiated following an update made to the network graph.

According to some examples, network risk-assessment computer **200** comprises network monitoring module **21**, that monitors the network and identifies changes in the network and/or receives updates from external sources and triggers execution of the risk-assessment procedure in response to the changes and updates.
15 Likewise, in some examples, network risk-assessment computer **200** comprises cybersecurity data hub **22**, that can trigger execution of the risk-assessment procedure in response to cybersecurity data updates.

Depending on the risk-assessment procedure output, in some cases, the subset of critical assets is updated as a result of the risk-assessment procedure (**703**),
20 while in other cases the subset of critical assets may stay the same. Asset-specific features are determined for each critical asset (**509**). Notably, in some cases updated asset-specific features may be determined even if the subset of critical assets has not changed. For example, if an update was made to OS version in a critical asset that was also identified as a critical asset before the update, the asset-specific features may still
25 be updated.

In case the subset of critical assets has been updated, new asset-specific sensors are generated, dedicated to operating at the newly identified critical assets. Otherwise, existing asset-specific sensors may be adapted to collect newly determined asset-specific features (**705**). The newly generated asset-specific sensors are deployed

in the network, each at the respective asset (**707**). If previously identified critical assets are no longer identified as critical assets, the respective asset-specific sensors can be deactivated. In some examples, existing sensors are updated remotely to collect a new set of asset-specific features.

5 The asset-specific sensors are used for retrieving the asset-specific features from the critical assets in the network (**515**). The ML-based asset-specific anomaly detectors generation process is repeated as explained above with respect to Fig. **6** (**517**). Existing ML-based asset-specific anomaly detectors can be retrained using newly selected asset-specific features to enable the model to detect anomalies
10 according to recent updates. Once the training process is complete, the newly generated asset-specific ML-based detectors are used for monitoring the respective critical assets and identifying anomalies as explained above.

 While Fig. **7** explains in detail the different operations which are related to an update procedure, in some examples, once triggered, an update procedure may
15 involve the execution of the risk-assessment LLM followed by the detectors-generating-LLM, where the first identifies the updated asset-specific features of each critical asset and facilitates the generation or configuration of updated sensors which are deployed in the network, and the second generates the corresponding updated ML-based anomaly detectors for detecting anomalies in the monitored assets.

20 It will also be understood that the system according to the presently disclosed subject matter may be a suitably programmed computer. Likewise, the presently disclosed subject matter contemplates a computer program being readable by a computer for executing the method of the presently disclosed subject matter. The presently disclosed subject matter further contemplates a machine-readable non-
25 transitory memory tangibly embodying a program of instructions executable by the machine for executing the method of the presently disclosed subject matter.

 It is to be understood that the presently disclosed subject matter is not limited in its application to the details set forth in the description contained herein or illustrated in the drawings. The presently disclosed subject matter is capable of other

embodiments and of being practiced and carried out in various ways. Hence, it is to be understood that the phraseology and terminology employed herein are for the purpose of description and should not be regarded as limiting. As such, those skilled in the art will appreciate that the conception upon which this disclosure is based may
5 readily be utilized as a basis for designing other structures, methods, and systems for carrying out the several purposes of the present presently disclosed subject matter.

CLAIMS:

1. A computer-implemented method of adaptive anomaly detection in a computer network, the method comprising:
 - collecting from assets in the network, respective asset-specific features
5 indicative of normal network behavior in a corresponding asset;
 - applying a first large language model (LLM) on asset-specific features obtained from the assets, wherein the first LLM is trained to generate, based on asset-specific features received from a given asset, a corresponding asset-specific machine learning (ML)-based anomaly detector configured to receive asset-
10 specific features from the given asset, and detect anomalies in the given asset;
 - further collecting, from each asset, respective asset-specific features, and applying the respective asset-specific features to the corresponding asset-specific anomaly detector for detecting anomalies in the respective asset.
2. The method of claim 1 further comprising: training each
15 corresponding asset-specific ML-based anomaly detector to detect anomalies based on asset-specific features obtained from a respective asset.
3. The method of any one of the preceding claims, wherein each corresponding asset-specific ML-based anomaly detector is an autoencoder machine learning model.
- 20 4. The method of any one of the preceding claims further comprising:
 - for each asset, determining asset-specific features based on combined processing of network discovery output pertaining to the asset and cybersecurity data indicative of a relevant cybersecurity threat.
5. The method of claim 4, wherein determining asset-specific features
25 is performed by a large language model (LLM) trained to receive network discovery output and cybersecurity data and identifying assets which are relevant to certain types of cybersecurity threats and attacks.
6. The method of any one of claims 4 to 5, wherein the network
30 discovery output is provided as a network graph, where each node in the network graph represents a respective asset in the network and each vertex connecting between two nodes represents a respective connection between assets in the

network.

7. The method of any one of the preceding claims, wherein the assets include a subset of critical assets selectively identified based on network parameters.

5 8. The method of any one of claims 1 to 6, wherein the assets include a subset of critical assets selectively identified in the network based on network parameters, wherein the method further comprises selecting the critical assets comprising:

applying the first LLM on network data trained to execute queries with
10 respect to the assets in the network dedicated for determining network parameters with respect to each asset;

assigning a score to each asset according to responses received to the queries; and

selecting the subset of assets according to the score.

15 9. The method of any one of claims 1 to 6, wherein all assets in the network are considered critical assets.

10. The method of any one of the preceding claims, wherein collecting from each asset respective asset-specific feature comprises:

providing asset-specific sensors to assets in the network; wherein each
20 asset-specific sensor is configured to monitor data in a respective asset and collect the respective asset-specific features.

11. The method of claim 10 further comprising deploying the asset-specific sensors in the network; and receiving at each asset-specific anomaly detector the asset-specific features collected by a respective asset-specific sensor.

25 12. The method of any one of the preceding claims comprising executing an update procedure, comprising:

for at least one asset, determining updated asset-specific features based on combined processing of updated network data pertaining to assets in the network, and updated cybersecurity data indicative of a relevant cybersecurity threat; and

30 collecting from the at least one asset, respective updated asset-specific features;

applying the first large language model (LLM) on the asset-specific features obtained from the at least one asset, to thereby obtain, for the at least one asset, a corresponding asset-specific machine learning (ML)-based anomaly detector configured to receive the updated asset-specific features from the given asset, and
5 detect anomalies in the given asset.

13. The method of claim 12, wherein the update procedure further comprises applying the first LLM for selecting an updated subset of critical assets; and

performing the update procedure on the updated subset of critical assets.

10 14. The method of any one of claims 12 and 13, wherein the update procedure is initiated in response to receiving a data update indicating a change in the network data or an update to the cybersecurity data to maintain anomaly detection in the network which is constantly updated according to current network characteristics and currently active cybersecurity threats.

15 15. The method of any one of claims 12 to 13, wherein the update procedure is initiated periodically to maintain anomaly detection in the network which is constantly updated according to current network characteristics and currently active cybersecurity threats.

16. The method of any one of claims 12 to 15, wherein determining
20 updated asset-specific features is performed using a large language model (LLM) trained to receive network discovery output and cybersecurity data, and identifying assets which are relevant to certain types of cybersecurity threats and attacks.

17. The method of any one of claims 12 to 16 further comprising:
25 providing or updating asset-specific sensors in the at least one asset in the network; wherein each asset-specific sensor is configured to monitor data in a respective asset and collect the respective updated asset-specific features.

18. The method of claim 17 further comprising deploying the respective asset-specific sensor at the respective asset in the network.

30 19. The method of any one of claims 12 to 18 further comprising: monitoring the network for identifying changes in the network and initiating the

update procedure in case a change that complies with one or more predefined conditions is identified.

20. The method of any one of claims 12 to 19, wherein the change includes one or more of:

- 5 addition or removal of one or more assets to the network;
- a change in a function of at least one asset in the network;
- a change in a relative location of at least one asset in the network; and
- an update made to a software component running on at least one asset in the network.

10 21. The method of any one of the preceding claims further comprising: responsive to detecting an anomaly, generating a warning indicating the anomaly.

 22. A computer program product comprising a computer readable storage medium retaining a program of instructions, which, when read by a computer processor, causes the computer processor to perform a method
15 according to any one of claims 1 to 21.

 23. A non-transitory program storage device readable by a computer, tangibly embodying a program of instructions executable by the computer to perform a method according to any one of claims 1 to 21.

 24. A computer system comprising at least one processing circuitry
20 operatively connectable to a computer network and configured to execute a method according to any one of claims 1 to 21.

 25. A computer-implemented method of adaptive anomaly detection in a computer network, the method comprising:

 selecting from multiple assets in the network a subset of critical assets;
25 determining, for each critical asset, a respective set of asset-specific features that are indicative of normal network behavior in the respective critical asset;

 collecting from each critical asset in the subset the respective asset-specific features;

30 applying a first large language model (LLM) on the asset-specific features obtained from the critical assets; wherein the first LLM is trained to generate,

based on the respective asset-specific features of a given critical asset, a corresponding asset-specific machine learning (ML)-based anomaly detector configured to receive asset-specific features from the given asset, and detect anomalies in the given critical asset;

5 further collecting from each respective critical asset in the subset of critical assets the respective asset-specific features, and applying a corresponding asset-specific ML-based anomaly detector for detecting anomalies in the respective critical asset.

26. The method of claim 25, wherein, when selecting from multiple
10 assets in the network, a subset of critical assets comprises:

 applying a LLM on network data trained to execute queries with respect to the assets in the network dedicated for determining network parameters with respect to each asset;

 assigning a score to each asset according to responses received to the
15 queries; and

 selecting the subset of critical assets according to the score.

27. The method of any one of claims 25 to 26 further comprising:
executing an update procedure, comprising:

 re-selecting from the multiple assets in the network an updated subset of
20 critical assets based on updated network data;

 for at least one critical asset in the updated subset, determining updated asset-specific features based on combined processing of updated network data pertaining to assets in the network, and updated cybersecurity data indicative of a relevant cybersecurity threat; and

25 collecting from the at least one critical asset, respective updated asset-specific features;

 applying the first large language model (LLM) on the asset-specific features obtained from the at least one critical asset, to thereby obtain, for the at least one critical asset, a corresponding asset-specific machine learning (ML)-based
30 anomaly detector configured to receive the updated asset-specific features from the at least one critical asset, and detect anomalies therein.

28. A non-transitory program storage device readable by a computer, tangibly embodying a program of instructions executable by the computer to perform a method according to any one of claims 25 to 28.

29. A computer system comprising at least one processing circuitry
5 operatively connectable to a computer network configured to execute a method according to any one of claims 25 to 28.

30. A computer-implemented method of adaptive anomaly detection in a computer network, the method comprising:

receiving from sensors operating at critical assets in the computer
10 network, asset-specific features indicative of normal network behavior in the critical assets; wherein each sensor is configured for providing asset-specific features from a respective critical asset in the computer network;

applying asset-specific features received from a respective critical asset to a corresponding asset-specific machine learning (ML)-based anomaly detector
15 configured to process the asset-specific features obtained from the respective critical asset, and detect anomalies in the critical asset;

generating a warning in case an output of the corresponding asset-specific machine learning (ML)-based anomaly detector indicates an anomaly in the respective critical asset.

20 31. The computer-implemented method of claim 30 further comprising:

generating the corresponding asset-specific machine learning (ML)-based anomaly detector, comprising:

applying a first large language model (LLM) on asset-specific features
25 obtained from critical assets in the computer network; wherein the first LLM is trained to generate, based on the asset-specific features received from a given critical asset, a corresponding asset-specific machine learning (ML)-based anomaly detector configured to receive asset-specific features from the given asset, and detect anomalies in the given asset; and

30 training each asset-specific machine learning (ML)-based anomaly detector.

32. The method of claim 31 further comprising executing an update procedure following determining that a change occurred in network data characterizing the computer network or cybersecurity data characterizing cybersecurity risk in the computer network, comprising:

5 for at least one critical asset, determining updated asset-specific features based on combined processing of updated network data pertaining to assets in the network, and updated cybersecurity data indicative of a relevant cybersecurity threat; and

 collecting, from the at least one critical asset, respective updated asset-specific features;

10

 applying the first large language model (LLM) on the asset-specific features obtained from the critical assets, to thereby obtain and update a corresponding asset-specific machine learning (ML)-based anomaly detector configured to receive the updated asset-specific features from the critical asset, and detect

15

anomalies in the critical asset.

33. A non-transitory program storage device readable by a computer, tangibly embodying a program of instructions executable by the computer to perform a method according to any one of claims 30 to 32.

34. A computer system comprising at least one processing circuitry

20

configured to execute a method according to any one of claims 30 to 32.