

System and method for active detection of malicious network resources

FIELD

[001] The described technique relates to the field of computing, namely to systems and methods for active detection of malicious network resources.

BACKGROUND

[002] Network resources, within the scope of this specification, refer to computing devices, i.e., software and hardware complexes configured to exchange data with a computer network, for example, the Internet, and located within the network infrastructure. In particular, a network resource in terms of this application could be a workstation connected to a local area network. Also, a network resource could be a laptop, netbook, tablet, smartphone, Internet of Things (IoT) device, server, router, firewall, or, without limitation, any other software and hardware solution that meets the above criteria.

[003] The term "network infrastructure" is understood within the scope of this specification in the popular sense; it is a set of software and hardware creating the basis for effective data communication, which could be targeted by malefactors. This could be, for example, a corporate computer network containing servers, workstations, routers, etc.

[004] This implies that the network infrastructure has a so-called perimeter, i.e, a conditional border formed by the network architecture itself, by the settings of the hardware and software used, and enabling to separate the network resources located "inside" the infrastructure, such as a workstation of one of the company's computer network users, and resources located "outside", for example, servers used for google.com site hosting. Within the scope of this specification the first devices are called internal network devices, and the second ones - external network devices.

[005] A malicious network resource, within the scope of this specification, is an external network resource, that is, the one located "outside" the infrastructure and belonging to a malefactor. At the same time, it is assumed that there is at least one compromised network resource "inside" the infrastructure, that is, the one that was originally intended to perform legitimate tasks, but then has been compromised by the malefactor or malware.

[006] By way of non-limiting example of a compromised network resource there could be a workstation compromised during phishing attack. The malwares, installed in it because of a compromise, exchange instructions with an external network resource, the malefactor's server.

[007] Detection of such an exchange of instructions between one of the internal devices of the protected infrastructure and an external server belonging to the malefactor is the infrastructure compromise indicator.

[008] The traditional approach to detection of such situations is based on the analysis of traffic, i.e., the information which the computing devices constituting this infrastructure exchange with each other, and also with the external network devices.

[009] The traditional approach usually implies placing a computing device inside the protected infrastructure, for example, a server that performs the traffic receiving and analyzing functions. As a result of the analysis, traffic anomalies are detected, that is, the data indicating that at least one device, being the part of the protected infrastructure, performs a typical, “anomalous” operations (for example, data exchange with an external device belonging to a malefactor), which is considered an indicator of compromise.

[010] The traffic itself may look, for example, as follows:

```
{ "ts":1607594574.388979,"uid":"CpuE6k2FNg9n3hbQel","id.orig_h":"111.222.333.444","id.orig_p":12238,"id.resp_h":"167.172.158.20","id.resp_p":443,"proto":"tcp","service":"ssl","duration":103.96006202697754,"orig_bytes":14136,"resp_bytes":202748,"conn_state":"S1","missed_bytes":0,"history":"ShADad","orig_pkts":76,"orig_ip_bytes":17188,"resp_pkts":208,"resp_ip_bytes":122280,"service_count":0,"ts_last_usage":1607594574.388979}
```

```
{ "ts":1607594574.388979,"uid":"CpuE6k2FNg9n3hbQel","id.orig_h":"111.222.333.444","id.orig_p":12238,"id.resp_h":"167.172.158.20","id.resp_p":443,"proto":"tcp","service":"ssl","duration":119.43929314613342,"orig_bytes":14934,"resp_bytes":204104,"conn_state":"S1","missed_bytes":0,"history":"ShADad","orig_pkts":82,"orig_ip_bytes":18226,"resp_pkts":216,"resp_ip_bytes":212756,"service_count":0,"ts_last_usage":1607594574.388979}
```

[011] At the same time, it is worth noting that at present, almost all traffic, both legitimate and malicious, is encrypted. This means that direct "reading" of data is very difficult if not impossible. Therefore, the common approach to traffic analysis involves the use of machine learning; models are trained in one way or another on the basis of knowingly legitimate traffic, and they help to detect situations when traffic ceases to resemble the legitimate one.

[012] However, all currently known methods of traffic analysis by machine learning have a common serious problem: with this approach, false responses often occur when legitimate traffic is taken for a malicious one. Internal computing devices legitimately exchange information with external legitimate computing devices (for example, web servers of contracting companies, online stores, etc.), but some features of this traffic trigger the traffic analysis systems that generate a warning about potential compromise of the infrastructure. This creates a significant unjustified burden on cybersecurity systems and seriously complicates practical application of such solutions.

[013] Many solutions based on network traffic analysis are known in the art.

[014] For example, the prior art discloses the solution US 8533819 B2 (AT&T Corp. AT&T Intellectual Property II LP), "METHOD AND APPARATUS FOR DETECTING COMPROMISED HOST COMPUTERS", published on July 10, 2013. It discloses an approach to detecting compromised hosts (computers) based on traffic analysis. The described approach is based on the selection of periodic

flow patterns (traffic fragments) between the client and the server, and further, on the classification of suspicious computing devices according to their traffic profiles. However, the said solution does not disclose a method for detecting the very fact of compromising any computing device, for example, the computing device that was first attacked by malefactors; it is designed to detect botnets, i.e., set of devices that are already known to perform the malefactors' tasks. At the same time, the solution described below, firstly, solves the problem of detecting the compromise of devices that have not been previously known, and secondly, it simplifies solving this problem by comparing the unique imprints of services running on an external suspicious computing device.

[015] The prior art also knows the patent US 7512980 B2 (Cisco Technology Inc.) "PACKET SAMPLING FLOW-BASED DETECTION OF NETWORK INTRUSIONS", published on March 31, 2009, which discloses the detection of traffic anomalies based on statistical analysis, that is, on data accumulation and learning some decision rules and classifiers from the accumulated data. This method will have the aforementioned disadvantage of frequent false responses; besides, such an approach will require accumulation of sufficient amount of data, that means large consumption of time and resources.

[016] The prior art also knows the solution described in US patent 9043920 B2 (Tenable Inc) "SYSTEM AND METHOD FOR IDENTIFYING EXPLOITABLE WEAK POINTS IN A NETWORK", published on January 02, 2014. The publication discloses a method for identifying network vulnerabilities by external scanning of computing devices. However, this method is based on the analysis of the so-called "trust relationship", which include open ports, TCP connections, connections of the protected infrastructure devices to the external systems, their reception of external traffic, etc., and such "trust relationship" is constantly analyzed for all the protected infrastructure devices, that means large consumption of time and resources. Moreover, it is well known that external scanning, i.e., the one that is not performed from a compromised infrastructure, is by no means always effective, since a malefactor, who has compromised the infrastructure, usually has the ability to detect and block such scanning, because IP addresses of those computing devices from which scanning is performed, will differ from IP addresses of the devices included in the compromised infrastructure, which are known to the malefactor.

[017] In addition, the prior art knows the solution US 10630674 B2 (Citrix Systems Inc), "SYSTEMS AND METHODS FOR PERFORMING TARGETED SCANNING OF A TARGET RANGE OF IP ADDRESSES TO VERIFY SECURITY CERTIFICATES", published on January 31, 2019. This solution discloses a method for detecting fake security certificates and suspicious activity by scanning a range of IP addresses and identifying all the certificates (for example, SSL certificates) served on all the ports open on the computing devices belonging to this IP address range. However, this method, firstly, does not disclose a method for determining the malicious activity of a computing device, for which there is no coincidence of the identified security certificates with certificates known from the Internet

resource registry, and secondly, it is characterized by the reduced performance due to the need for performing additional requests to external systems.

[018] The prior art also knows the patent RU 2 722 693 C1 (Group-IB TDS LLC), “METHOD AND SYSTEM FOR DETECTING THE INFRASTRUCTURE OF A HARMFUL PROGRAM OR CYBERSHIP”, published on June 03, 2020. This patent proposes a method for detecting the infrastructure of a malicious program or a malefactor, wherein a request is received comprising the infrastructure element and a tag on this element belonging to the malicious program or a malefactor; on the basis of further analysis of this data, there is determining the connections of the obtained element with other elements of the infrastructure belonging to certain malicious programs or malefactors, thus detecting this infrastructure. Despite the fact that this well-known solution, like the one described below, uses the information about IP addresses, domains associated with them, ports open on them and running services, it is aimed at achieving a fundamentally different result: detecting a priori unknown infrastructure built by malefactors, that is, an attack tool. Whereas the solution described below is aimed at identifying the fact of the protected infrastructure compromise, that is, the very fact of the attack.

[019] Finally, the prior art discloses the solution US 10284595 B2 (Citrix Systems Inc), "COMBINING INTERNET ROUTING INFORMATION WITH ACCESS LOGS TO ASSESS RISK OF USER EXPOSURE", published on November 10, 2016. This solution is aimed at identification of a network attack, assessment of a network attack risk or mitigation of a network attack by assessing the routing and access logs information. At the same time, in the described method, domain name registrars are additionally monitored to detect changes, and also certificates are monitored. However, the approach based on monitoring domain name registrars does not meet the requirements for fast and accurate detection of compromised computing devices. Moreover, the approaches to detecting anomalies in the network traffic, used in accordance with the proposed method, at the first stage help to obtain more accurate information about a potentially malicious computing device than the periodic processing of routing data described in this known solution.

[020] Based on the results of the prior art study, it may be concluded that there is a need for a technical solution eliminating the disadvantages of the above approaches. The solution described below is designed to solve at least some of the problems revealed in the prior art analysis. Moreover, the proposed method improves the accuracy and expands the coverage of the detected network attacks.

SUMMARY

[021] The object of the proposed technique consists in development of a method and system for active detection of malicious network resources.

[022] The technical result of the claimed technique is improving the accuracy of detecting the protected infrastructure compromise by malefactors or malicious programs.

[023] The technical result is achieved owing to the fact that the system for active detection of malicious network resources comprises an external computing device, the external computing device installed outside an infrastructure and configured to communicate with the infrastructure devices, the external computing device comprising a communication module, a scanning module, a notification module, an analysis module, a data storage module, the data storage module storing an instructions, the instructions, if being executed by a processor, implement the described method, at least one database, the database comprising data on known malicious services, historical data on the activity of various IP addresses, a graph model of the Internet, and an internal computing device, the internal computing device installed within the infrastructure and configured to receive the infrastructure outbound traffic, and comprising at least a communication module, a scanning module, an analysis module, a data storage module, the data storage module storing an instructions, the instructions, if being executed by a processor, implement the described method.

[024] The technical result is also achieved owing to the fact that the method for active detection of malicious network resources, the method being executable by a distributed computer system, the method comprising receiving, by a communication module of an internal computing device, an outbound infrastructure traffic, detecting, by an analysis module of the internal computing device, at least one suspicious external IP address in the outbound traffic, scanning, by a scanning module of the internal computing device, at least one suspicious device located on detected at least one suspicious IP address, and obtaining a list of services running on the suspicious device, sending, by the communication module of the internal computing device, detected at least one suspicious IP address and the obtained list of services to an external computing device, comparing, using an analysis module of the external computing device, the obtained list of services with a data on known malicious services, and in response to a coincidence of at least one service from the obtained service list with the data on known malicious services, considering, by the analysis module of the external computing device, the suspicious device, located on at least one suspicious IP address, to be malicious one, generating, by a notification module of the external computing device, at least one report on detection of malicious activity in the infrastructure, in response to absence of coincidences of services from the obtained service list with the data on known malicious services, computing, by the analysis module of the external computing device, an interval of ownership of at least one suspicious IP address, in response to the fact that the computed duration of the ownership interval is less than a threshold value, considering, by the analysis module of the external computing device, the suspicious device, located on at least one suspicious IP address, to be malicious one, generating, by a notification module of the external computing device, at least one report on detection of malicious activity in the infrastructure, and in response to the fact that the computed duration of the ownership interval exceeds the threshold value, considering, by the analysis module of the external computing device, the suspicious device to be legitimate one.

[025] In other embodiment of the described method in response to inaccessibility of at least one suspicious device, located on detected at least one suspicious IP address, for the scanning module of the internal computing device transmitting, by the communication module of the internal computing device, the information about at least one suspicious IP address to the external computing device, scanning, by a scanning module of the external computing device, the suspicious device located on at least one suspicious IP address, obtaining, by the scanning module of the external computing device, a list of services running on the suspicious device.

[026] In another embodiment of the described method for detecting, by the analysis module of the internal computing device, at least one suspicious IP address in the outbound traffic, detecting at least one anomalous traffic fragment, extracting, from the detected at least one anomalous traffic fragment, at least one IP address of the computing device, which this at least one traffic fragment is addressed to, and considering the extracted at least one IP address to be suspicious.

[027] In the other embodiment of the described method for detecting at least one anomalous traffic fragment in the outbound traffic at least one of the following methods is used: detecting by a deep neural network preliminarily trained on a tagged traffic, detecting by a OneClassSVM classifier preliminarily trained on legitimate traffic samples, detecting by machine learning models, such as RandomForestClassifier, XGBoost, KNN, AdaBoost, etc., preliminarily trained on tagged samples of traffic, searching in the outbound traffic by preformed feature signatures, using sliding window methods and collecting statistical features of traffic logs.

[028] In the other embodiment of the described method for comparing the obtained list of services with a data on known malicious services, a graph model of the Internet is used.

[029] In the other embodiment of the described method for computing the interval of ownership of at least one suspicious IP address historical data on the activity of IP addresses is used.

[030] In the other embodiment of the described method, at least one generated report on the detection of malicious activity is transmitted via at least one of the following communication methods: e-mail, SMS, MMS, push notifications, instant messenger messages, API events.

BRIEF DESCRIPTION OF THE DRAWINGS

[031] The accompanying drawings, which are included for additional understanding of the described method and form a part of this specification, illustrate the technique embodiments and together with the specification serve to explain the principles of the technique.

[032] The claimed technique is illustrated by the following drawings, wherein:

[033] Fig. 1 illustrates an example of computing devices implementing the method described herein.

[034] Fig. 2A illustrates a simplified flow chart of the method for active detection of malicious network resources.

[035] Fig. 2B illustrates a simplified flow chart of one of the steps of the method for active detection of malicious network resources, namely the step of comparing the obtained list of web services with the data on known malicious web services.

[036] Fig. 2C illustrates a simplified alternative flow chart of the method for active detection of malicious network resources.

[037] Fig. 3A illustrates a non-limiting example of a fragment of an Internet graph model used at one of the steps of the method for active detection of malicious network resources.

[038] Fig. 3B illustrates a non-limiting example of a filtered fragment of the Internet graph model obtained at one of the steps of the method for active detection of malicious network resources.

[039] Fig. 4 illustrates one of the possible approaches to computing the ownership interval of at least one suspicious IP address.

[040] Fig. 5 illustrates a non-limiting example of the computing device schematic diagram.

DETAILED DESCRIPTION

[041] Description of the exemplary embodiments of the claimed technique is given below.

[042] The subject matters and features of the present technique, methods for achieving these subject matters and features become apparent by reference to exemplary embodiments. However, the present technique is not limited to the exemplary embodiments disclosed below, and it could be implemented in various forms. The essence given in the specification is nothing, but specific details provided to assist a person skilled in the art in a comprehensive understanding of the technique, and this technique is defined only within the scope of the appended claims.

[043] It is worth noting that in the context of the present specification, unless clearly specified otherwise, the words the "first" and the "second" are used solely to distinguish the nouns to which they refer to from each other, and not for the purposes of describing any specific relationship between these nouns.

[044] In order to implement the described method for active detection of malicious network resources, an external computing device (100) is pre-installed, as shown in Fig. 1, outside the perimeter of the network infrastructure (150) to be protected.

[045] Besides, an internal computing device (120) is installed within the perimeter of the network infrastructure (150) to be protected.

[046] Design and hardware configuration of this computing devices could be any commonly known ones enabling to ensure execution of the functions described below. This could be, by way of example, but not limited to, servers such as Dell™ PowerEdge™ servers. In an alternative embodiment

of the method, the external computing device (100) could be a cloud solution, i.e, a set of servers, and the internal computing device (120) could be, for example, a laptop, workstation or mobile device, such as a smartphone. But in any case, the external (100) and internal (120) computing devices should be configured to exchange traffic among themselves via a computer network, for example, via the Internet (130). Moreover, the internal device (120) should be configured to exchange traffic with the devices of the protected infrastructure (150) over the local network of the protected infrastructure (152).

[047] It is understood that other devices constituting the protected infrastructure (150) are also configured to exchange traffic both through the network, in particular the Internet (130), and over the local network of the protected infrastructure (152).

[048] In order to illustrate a possible situation, the example of which demonstrates functioning of the described system, Fig. 1 also illustrates a malefactor's computing device (190), for example, a server, which a compromised computing device being a part of the protected infrastructure (150) can exchange information with.

[049] The external computing device (100) should comprise the following functional modules: a communication module (102), a data storage module (104), a scanning module (106), an analysis module (108), and also a notification module (110).

[050] The internal computing device (120) should comprise the following functional modules: a communication module (122), a data storage module (124), a scanning module (126), and also an analysis module (128).

[051] The listed modules in terms of their technical implementation could be software modules executed on one computing device, software and hardware modules, for example, at least some of these modules could be implemented as dedicated servers, with each of them performing a corresponding function, or represent an arbitrary combination of the above implementation methods.

[052] In this case, the main function of each of the modules is as follows:

[053] **The communication module** (102) of the external computing device (100) is configured to exchange data with the data storage module (104), the scanning module (106), the analysis module (108), and also the notification module (110). The communication module (102) enables the computing device implementing this method to communicate with a computer network, for example, with the Internet (130). In turn, the communication module provides communication between the computing device (100), implementing this method, with the protected infrastructure devices (150), including the internal computing device (120), using a computer network. Besides, via the computer network, the communication module (102) enables the computing device (100) to obtain additional information, in particular, data on known malicious services and certificates, about various IP addresses and domains, and also their various parameters, such as registration date, compliance (resolve) of a domain name with

an IP address, etc. Implementation of these functions by the communication module (102) has no specific features and could be performed by any well-known method.

[054] **The communication module** (122) of the internal computing device (120) is configured to exchange data with the data storage module (124), the scanning module (126), the analysis module (128), and also with all the other devices of the protected infrastructure (150). The communication module (122) enables the computing device (120) to communicate with the computing device (100) via a computer network, for example, the Internet (130), and with the other devices of the protected infrastructure (150) via a local computer network (152). Implementation of these functions by the communication module (122) has no specific features and could be performed by any well-known method.

[055] **The data storage module** (104) of the external computing device (100) is configured to exchange data with the communication module (102), the scanning module (106) and the analysis module (108). The data storage module (104) provides both permanent and temporary storage of information. For this purpose, the data storage module (104) comprises at least one database (not shown conventionally in Fig. 1), which allows to store information orderly and to carry out search efficiently. Such a database could comprise, by way of example, but not limited to, all historical data available for each IP address: open ports, running services, associated domains, data on known malicious services, and also a graph model of the Internet. The data storage module (104) could also store program instructions, which execution enables the device (100) to implement the described method. Implementation of all these functions by the data storage module (104) has no specific features and could be performed by any well-known method.

[056] **The data storage module** (124) of the internal computing device (120) is configured to exchange data with the communication module (122), the scanning module (126) and the analysis module (128). The data storage module (124) provides both permanent and temporary storage of information. For this purpose, the data storage module (124) comprises at least one database (not shown conventionally in Fig. 1), which allows to store information orderly and to carry out search efficiently. The data storage module (124) could also store program instructions, which execution enables the device (120) to implement the described method. Implementation of all these functions by the data storage module (124) has no specific features and could be performed by any well-known method.

[057] **The scanning module** (106) of the external computing device (100) is configured to exchange data with the communication module (102), the data storage module (104) and the analysis module (108). The scanning module (106) is also configured to scan arbitrary computing devices connected to the Internet, for example, such as a malefactor's server (190), and obtain information about the services running on the ports of each scanned device. Implementation of these functions by the scanning module (106) has no specific features and could be performed by any well-known method.

[058] **The scanning module** (126) of the internal computing device (120) is configured to exchange data with the communication module (122), the data storage module (124) and the analysis module (128). The scanning module (126) is also configured to scan arbitrary computing devices connected to the Internet, for example, such as a malefactor's server (190), and obtain information about the services running on the ports of each scanned device. Implementation of these functions by the scanning module (126) has no specific features and could be performed by any well-known method.

[059] **The analysis module** (108) of the external computing device (100) is configured to exchange data with the communication module (102), the scanning module (106), the data storage module (104) and the notification module (110). The analysis module (108) checks the lists of services received from the communication module (102) or from the scanning module (106) for belonging to known malicious services. Besides, the analysis module (108) computes the IP address ownership interval, that is, the time interval during which some domains referred to this IP address, or some ports were opened on this IP address. Implementation of these functions by the analysis module (108) will be described below with reference to Fig. 2B, Fig. 3A, Fig. 4.

[060] **The analysis module** (128) of the internal computing device (120) is configured to exchange data with the communication module (122), the scanning module (126) and the data storage module (124). The analysis module (128) detects anomalies in the traffic received from the communication module (122), and also identifies the IP addresses of external computing devices which the anomalous traffic was addressed to, for example, such as the malefactor's server (190). Implementation of these functions by the analysis module (128) has no specific features and could be performed by any well-known method.

[061] **The notification module** (110) of the external computing device (100) is configured to exchange data with the analysis module (108) and the communication module (102). The notification module (110) generates and transmits, using the communication module (102), the notification of a potential compromise of the protected network infrastructure by at least one of the following methods: by e-mail, by sending SMS, by sending MMS, by sending a push notification, by a message in an instant messenger, by creating an API event. Implementation of these functions by the notification module (110) has no specific features and could be performed by any well-known method.

[062] Hereafter, with reference to Fig. 2A, the main steps of the method (200) for active detection of malicious network resources are described.

[063] The method (200) begins at step (210) when the outbound traffic of the protected infrastructure (150) is received by the communication module (122) of the internal computing device (120). This could be done by any well-known method; for example, some of the routers included in the protected infrastructure (150) could be configured to mirror all outbound traffic of the protected infrastructure (150) to one of the ports of the communication module (122).

[064] All outbound traffic received by the communication module (122) is transmitted unchanged to the analysis module (128) for analysis. In this case, the method (200) proceeds to step (220). On the step (220) at least one suspicious external (global) IP address is detected in the outbound traffic using the analysis module (128) of the internal computing device (120).

[065] Outbound traffic analysis, which allows to identify at least one suspicious external IP-address in it, could be performed by the analysis module (128) by any well-known method. For example, if at least one anomalous traffic fragment is detected in the outbound traffic, then at least one IP address of a computing device which is an addressee of this at least one traffic fragment is addressed, can be extracted from that anomalous traffic fragment by any well-known method. Said at least one external IP address can be considered to be suspicious one.

[066] In turn, a detection of at least one anomalous traffic fragment in the outbound traffic also can be done in any known way. For example, it can be done:

- using a deep neural network preliminarily trained on the tagged traffic,
- using of OneClassSVM classifier preliminarily trained on legitimate traffic samples,
- using machine learning models, such as RandomForestClassifier, XGBoost, KNN, AdaBoost, etc., preliminarily trained on tagged samples of traffic,
- by searching in the outbound traffic for preformed feature signatures,
- using the sliding window method and collecting statistical features from the traffic logs,
- in some other way without limitation, including hybrid methods of analysis using the above methods.

[067] It is important that the used analysis method allows to make a decision that a certain traffic fragment is anomalous, when predetermined conditions are met.

[068] In modern computer networks, each elementary traffic fragment (packet or datagram) comprises an IP address of the source, that is, the computing device from which this traffic fragment has been sent, address of the port where the service generated this fragment is running, and also an IP address and the port of an external computing device where this traffic fragment is addressed to, such as, for example, a malefactor's server (190). Therefore, at this step, at least one external (global) IP address of the device-recipient becomes known, for example, the IP address 111.222.333.444, which is associated with this anomalous traffic fragment, and therefore, is suspicious, and also the port address, for example, the port 50050.

[069] Upon detecting such at least one suspicious IP address, the method (200) proceeds to step (230). At step (230), at least one suspicious device, such as, for example, a malefactor's server (190), located on at least one suspicious external (global) IP-address, for example, IP address 111.222.333.444, is scanned by the scanning module (126) of the internal computing device (120), and a list of ports open

on this IP address, a list of web services running on these ports and software versions corresponding to these web services, are obtained.

[070] Thus, in contrast to passive methods for detecting malicious activity, according to which they are limited to traffic collection and analysis, and do not have an impact on a suspicious device, the described method also performs an active action aimed at a suspicious device, namely, scanning this device.

[071] As it is known to those skilled in the art, the complete list of ports that could be theoretically open on any predetermined computing device is limited and known in advance.

[072] In one possible embodiment of this method, all ports that could be open on a given suspicious IP address are scanned.

[073] In the other possible embodiment, ports which are scanned are predetermined. For example, a list of ports that are most often used by malefactors, for example, such ports as 22, 80, 443, 2022, 8443, etc., is stored in the database of the data storage module (124), and also the port where at least one abnormal traffic fragment is addressed to, in this example it is the port 50050.

[074] In order to obtain a list of open ports, a so-called port scanner is used, i.e., implementation of an algorithm built, for example, on the known principles of SYN, TCP, UDP, ACK or FIN-scanning, or on any other known principles of port scanning, enabling to distinguish an open port from a closed one.

[075] For each suspicious IP address, using a previously prepared script, a list of ports currently open on a given IP address and also names (identifiers) and versions of the web services running on them are obtained, and the result is stored in the database of the data storage module (124). Actually, obtaining this information is a well-known, standard function of port scanners. Based on the results of the port scanner operation, not only the numbers of open ports but also the names and versions of programs (web services) running on them are determined.

[076] In the other possible embodiment of the described method, the well-known network utilities supplied with some operating systems, such as netstat and nmap utilities, are used to determine the state of ports.

[077] In this case, it is important that scanning of an external suspicious computing device located on an external suspicious IP-address, such as, for example, a malefactor's server (190), is performed by a computing device located in the same local network, inside the same protected infrastructure (150) as the compromised device which a malefactor intends to exchange instructions with. Malefactor's abilities to block such scanning are extremely limited.

[078] First, a malefactor will not know from which internal (local) IP address of the protected infrastructure the scanning is performed, because the internal (local) IP addresses during such scanning are always hidden, since their routing in the global Internet is impossible. Therefore, a malefactor, when trying to determine the scanning source, will only see the external (global) IP address of the same

infrastructure, where the device compromised by the malefactor is located. Secondly, a malefactor will also not be able to block such scanning, because when it blocks traffic from the external (global) IP address of the protected infrastructure (150), the internal computing device compromised by the malefactor will lose the ability of connecting to the server (190), and the malefactor will lose the communication with this compromised device.

[079] As a result of step (230), there is obtaining a list of web services running on open ports of at least one computing device having a suspicious IP address, such as a malefactor's server (190). By way of non-limiting example, such a list may look as follows:

- port 111.222.333.444:22 is open, SSH-2.0-OpenSSH_7.9p1 Ubuntu-10 is running,
- port 111.222.333.444:69 is open, Cisco TFTP Server v.1.1 FTP server is running,
- port 111.222.333.444:443 is open, web service name is not available, (1)
- port 111.222.333.444:4899 is open, Radmin Server v.3.5.2 web service is running,
- port 111.222.333.444:8080 is open, Apache Tomcat v.3.3.2 web server is running,
- port 111.222.333.444:50050 is open, the service name is not available.

[080] This completes step (230), and the method proceeds to step (240), wherein at least one suspicious external (global) IP address and a list of web services running on open ports of at least one computing device having this IP address is sent by the communication module (122) of the internal computing device (120) to an external computing device (100).

[081] Sending the specified data to the external computing device (100) has no specific features and could be performed by the communication module (122) of the internal computing device (120) and the communication module (102) of the external device (100) by any well-known method.

[082] This completes step (240), and the method proceeds to step (250), wherein the obtained list of web services is compared with the data on known malicious web services on the external computing device (100) by the analysis module (108).

[083] It is worth noting that the data on known malicious web services could be preliminarily stored in the database of the data storage module (104) of the external device (100). Moreover, the external device (100) could be configured to receive updates and additions to the information about known malicious web services periodically, for example, once an hour or once a day, via the communication module (102), and store these updates and additions in the database of the data storage module (104).

[084] The data on known malicious web services stored in the database of the data storage module (104) of the external device (100), in particular, include the hash storage of the malicious web services. Hash, in this case, is the result of taking a hash function according to some predetermined algorithm, for example, according to the SHA256 algorithm, from the names of malicious web services or web services, the use of which may be indicative of the protected infrastructure compromise, and also from IP addresses

used by malicious web services, and SSL/TLS certificates used by malicious web services. This storage has no specific features and could be organized by any well-known method enabling to search for a specific hash therein.

[085] In this case, the analysis module (108) includes a pre-built and periodically updated Internet graph model, described in detail below with reference to Fig. 3A. It is worth noting that this is a simplified example given for ease of understanding, therefore, it conventionally does not show a number of graph vertices that could be present in the practical embodiment of such a model. In another possible embodiment of the described method a pre-built and periodically updated Internet graph model can be stored in the database of the data storage module (104) of the external device (100).

[086] The Internet graph model (300) is a graph, at which vertices there are elements of the computer network infrastructure and the parameters of these elements. There could be involved such elements of the computer network infrastructure as, for example, external (global) IP addresses (310), (320), (330), (340), domains (311), (315), (319), (323), (325), ports (312), (316), (332), web services running on these ports (314), (318), (334), SSL and TLS certificates (313), (317), (321), etc. The parameters of the computer network infrastructure elements could be, for example, owners of IP addresses (302), (342), dates of IP address registration (not shown), dates of SSL and TLS certificates creation and expiration (not shown), etc.

[087] The graph vertices are identified at the stage of model building while crawling the Internet. This crawling enables to obtain information about availability, status and parameters of each of the listed elements of the computer network infrastructure.

[088] The graph edges are the links identified during analysis of the said Internet crawling results. In this case, a link is considered to be the revealed match of the parameters of different elements of the computer network infrastructure. Building links between the vertices could be executed by any well-known method, in particular, as described in patent RU2681699C1, where finding at least the first network resource (one graph vertex) and at least the second network resource (one more graph vertex); retrieving information about the found network resources, including at least one parameter of the first network resource and at least one parameter of the second network resource; and in response to the fact that at least one parameter of the first network resource matches at least one parameter of the second network resource, building a link between the first network resource and the second network resource (an edge between two vertices of the graph).

[089] Thus, with reference to Fig. 3, the links between the graph vertices corresponding to the IP addresses (320), (330) and (340) are established, since these three addresses belong to one and the same owner (342). The links between vertices corresponding to domains (315), (319) and (325) are established because one and the same TLS certificate (317) is installed on the websites having these domain names.

[090] Moreover, the links between the vertices corresponding to domains (315) and (319) are established, since both of these domain names refer to one and the same external (global) IP address (310). Similarly, the links between the vertices corresponding to domains (323) and (325) are established, since both of these domain names refer to one and the same IP address (320).

[091] The link between the graph vertices corresponding to the port (316) opened at IP address (310) and port (332) opened at IP address (330) is established, since the same web services (318), (334) are running on these ports, for example, Apache Tomcat web servers of the same v.3.3.2 version.

[092] A certain weight characterizing, for example, the “strength” of the link or some other its parameter could be assigned to each graph edge (each link) in the Internet graph model (300). Computing or assignment of these values could be performed by any well-known method. In one non-limiting example, a link could have more weight the less elements it connects; for example, the link between the IP address (310) and the owner of that address (302) could have more weight than the links between the IP addresses (320), (330), (340) and their owner (342). Alternatively, the first link could have less weight than the second one.

[093] Moreover, the weight assigned to each graph edge could depend on other parameters as well. For example, links built from vertices corresponding to TLS certificates could have more weight than links built from vertices corresponding to the same web services, such as web services (318), (334). Any other known methods of computing or assigning weights to the edges of such a graph are possible without limitations.

[094] The above-described Internet graph model could be configured to be periodically replenished or enriched. The graph model replenishment or enrichment could be performed both by a device implementing the method and by third-party systems, moreover, as by re-crawling the Internet and analyzing the obtained data by the above-described method, as by alternative methods, for example, the methods described below.

[095] In one non-limiting example, the lists of domains published by domain registrars could be analyzed, and thus new domains could be identified. In the other example, it is possible to analyze network traffic on the Internet and extract new domain names from it. In the other example, new domains could be extracted from TLS/SSL certificate data, and also from the code or traffic of malicious files analyzed by third-party systems. Similarly, it is possible to analyze the lists of the issued SSL and TLS certificates, thus obtaining information about new certificates and domains, and enriching the graph model. The said analysis of the lists, network traffic, malicious file code and certificates could be performed by any well-known method.

[096] The Internet graph model, created in advance and constantly replenished in this way, is a part of the analysis module (108), using which, at step (250), the list of services obtained at step (240) is compared with the data on known malicious web services.

[097] The main steps in step (250) are described below with reference to Fig. 2B.

[098] Step (250) begins at step (251), wherein the names of web services from the list of web services running on open ports of at least one computing device having a suspicious external IP address is obtained and a hash function from each of them is taken. It is worth noting that the hash function is taken according to the same algorithm that has been used to build the hash storage of malicious web services, for example, according to the SHA256 algorithm. Also it is worth noting that selection of an algorithm, such as the SHA256 algorithm, is not a limiting factor, this algorithm is named as an example; alternatively, any other algorithm could be used.

[099] For example, resulting from step (251), four predetermined web service names will be obtained from the above list (1), and from them, in turn, the following hashes will be obtained:

- d1452346b1f850c336f4552f810d48ddf065618a600783455444a5cc43cf9f35
- a78eb06000bec36b79a127b77c640e1a9664502ea201f1bb0ceaf5f94b1432aa (2)
- a83a7ef76cbe8a3a6cfac2a2fdb0b9c9acd6ed0fcee7292ce59319474d31d500
- e488e34ba7c1bd7f68be636ab13e4adc12d4030096fc4bd9d8e24a820d93bbb1.

[0100] This completes step (251), and the method proceeds to step (252), where using the analysis module (108), there is searching for each of the hashes obtained at step (251), for example, the hashes from the list (2) in the hash storage of malicious web services, in the database of the data storage module (104). The method is repeated until the end of the list, for example, the list (2).

[0101] In this case, if the list end is reached, and no hash from the list has been found in the hash storage of malicious web services, the method proceeds to step (253).

[0102] Moreover, in response to detection of at least one of the obtained hashes in the hash storage of malicious web services at step (252), the list of detected malicious web services is added. It is worth noting that in this example, searching for a hash from the list (2) coincides with the hash stored in the hash storage of malicious web services, since the third hash from the list (2) corresponds to Radmin Server web service, the use of which by an external computing device may be indicative of compromise of the protected infrastructure.

[0103] In this case, the list of malicious web services detected during step (250) is added using the analysis module (108) by any well-known method. This list is stored in the database of the data storage module (104).

[0104] Alternatively, it is also possible such an embodiment of the method, wherein in case of detection of at least one malicious web service, execution of step (250) is interrupted and the described method proceeds to step (290), where a computing device located on at least one suspicious IP address is considered malicious, generating, by the notification module (110), at least one report on detection of malicious activity in the protected infrastructure (150) and sending it via the communication module

(102). This completes the described method. In the other possible embodiment of the method, after sending at least one report, the method returns to step (210).

[0105] If searching for all hashes in the hash storage of malicious web services has produced no results, then after reaching the end of the list, for example, the list (2), step (252) ends, and the method proceeds to step (253).

[0106] At step (253), by any well-known method, there is finding, by the analysis module (108), at least one suspicious external (global) IP address, such as the suspicious IP address 111.222.333.444 (310 in Fig. 3A) obtained at step (240), in the Internet graph model.

[0107] Then the method proceeds to step (254), where there is identifying a graph model fragment comprising all the elements associated with the IP address (310). In one of the non-limiting embodiments of the method, for this purpose, all the graph model elements, which the IP address (310) has any link with, are taken. In the other possible embodiment, those elements are taken that are connected to the original element of the protected infrastructure with such a number of edges that does not exceed a preset threshold, for example, elements connected to the original element of the protected infrastructure with no more than 500 edges (links). The identified fragment of the graph model may look, for example, as shown in Fig. 3A.

[0108] The method then proceeds to step (255), where at least one clearing algorithm is applied to the identified graph model fragment. This at least one algorithm could be, without limitation, at least one of any algorithms known in the art. In one non-limiting example, this could be an algorithm that cuts off the graph edges (links) which weight is less than a preset threshold.

[0109] As a result of applying at least one clearing algorithm at step (255), a cleared graph fragment is obtained. Example of such a cleared fragment (350) is shown in Fig. 3B. It can be seen from the attached drawing that as a result of execution of steps (253)-(255) three domains (311), (315) and (319), and two TLS certificates (313) and (317) have been identified in this example.

[0110] This completes step (255), and the method proceeds to step (256), where there is taking a hash function for each of the identified elements of the computing infrastructure or parameters thereof, that is, for each identified vertex of the cleared graph fragment, according to the same algorithm that has been used to build the hash storage of malicious web services, for example, according to the SHA256 algorithm.

[0111] For example, for the above-mentioned vertices of the graph model fragment shown in Fig. 3B, i.e., for three domains, (311), (315) and (319) and two TLS certificates (313) and (317), resulting from step (256), the following five hashes are obtained:

- e19e8ff3840a5568afe531044b48e0485cb2c61e3e4a2e122d7ac8ec236da3f3, (3)
- f1a1b4d398dc0bd98d7caac890c52ae3be52c9028f700a62b582e85a3049f242,

- 0f0599c43dd7d37568c6f41c8016bc232c5033d7ddaf5f83fca26d9dd375e520,
- d53e96f24b3b30832a370df4f01cbcc36eeb68e7f07475073b932ed9f5706eda
- 21488535997ef4bbb6ef1ea3b34e4d7443cff161b42a6ddaea2a4b56644beb1c.

[0112] This completes step (256), and the method proceeds to step (257), where using the analysis module (108), there is searching for each of the hashes obtained at step (256), for example, the hashes from the list (3) in the hash storage of malicious web services, in the database of the data storage module (104). This is done by any well-known method, in the same way as described above with respect to step (252).

[0113] Similarly, analogously to step (252), at step (257), in response to detection of at least one of the obtained hashes in the hash storage of malicious web services, there is adding the list of the detected malicious web services. In particular, in the example under consideration, searching for the fourth hash from the list (3) coincides with the hash stored in the hash storage of malicious web services, since this hash is the hash of the serial number of the TLS certificate for the CobaltStrike penetration testing framework, the use of which by an external computing device may be indicative of compromise of the protected infrastructure.

[0114] In this case, the list of malicious web services detected during step (250) is added using the analysis module (108) by any well-known method. This list is stored in the database of the data storage module (104).

[0115] Alternatively, it is also possible such an embodiment of the method, wherein in case of detection at step (257) of at least one malicious web service, execution of step (250) is interrupted and the described method proceeds to step (290), where a computing device located on at least one suspicious IP address is considered malicious, generating via the notification module (110), at least one report on detection of malicious activity in the protected infrastructure (150) and sending it via the communication module (102). This completes the described method. In the other possible embodiment of the method, after sending at least one report, the method returns to step (210).

[0116] When the end of the list, such as the list (3), is reached, step (257), and along with it the entire step (250) ends, and the method proceeds to step (260).

[0117] At step (260), by any well-known method, it is verified, via the analysis module (108), whether at least one service associated with at least one malicious IP address coincides with the data on known malicious web services. This is done by checking the list of malicious web services detected at step (250).

[0118] If this list contains at least one entry, that is, during step (250) at least one malicious web service associated with at least one suspicious IP address obtained at step (240) has been detected, the described method proceeds to step (290), where a computing device located on at least one suspicious IP

address is considered malicious, generating, by the notification module (110), at least one report on detection of malicious activity in the protected infrastructure (150) and sending it via the communication module (102). This completes the described method. In the other possible embodiment of the method, after sending at least one report, the method returns to step (210).

[0119] If the list of malicious web services detected at step (250) is empty, that is, no malicious web services have been detected during step (250), step (260) ends, and the method proceeds to step (270).

[0120] At step (270), there is computing, by the analysis module (108), an interval of ownership of at least one suspicious IP address, obtained at step (240). This is done by analyzing historical data on the activity of various IP addresses, which data are previously obtained and stored in the database of the data storage module (104) of an external computing device (100).

[0121] Historical data on the activity of various IP addresses is information about when (from what date to what date) on each specific IP address certain ports were opened, certain web services were running, when certain domains referred to each specific IP address. This information could be stored in the database in any known format enabling efficient search by a given parameter, for example, by an IP address.

[0122] In one of the possible embodiments of the described method, the historical data could be obtained from the Internet graph model, described above. In the other possible embodiment, the historical data could be obtained independently, during periodical crawling the Internet, resulting from such crawling, ordering and storing information about what services on what ports of what device with a specific IP address are functioning as of a given date. This crawling as such, as well as the subsequent analysis and ordering of data, are not specific and could be performed by any well-known method obvious to those skilled in the art.

[0123] One of the possible approaches 400 to computing the ownership interval of at least one suspicious IP address is shown in Fig. 4.

[0124] Thus, Fig. 4 illustrates that the ownership interval of the IP address A (402) is computed as the sum of those time intervals when any domains, such as domain (410) old.company.com, domain (420) another.old.company.com, domain (440) company.com, referred to this IP address, or any web services, such as web service (430) Apache Tomcat v.3.3.2, were running on the ports of the device with the given IP address, or the device with the given IP address simply had at least one open port (not shown in Fig. 4).

[0125] Duration of time intervals is summed up, provided that every two adjacent intervals are overlapped at least for one day. As, for example, the interval of functioning the domain (420) another.old.company.com and the interval of functioning the web service (430) Apache Tomcat v.3.3.2 intersect.

[0126] In the other possible embodiment of the described method, the duration of time intervals could be summed up also in case, if some two adjacent intervals are not overlapped on condition that the period, during which there have been no open ports on the device with the given IP address and no domain has referred to it, does not exceed a preset value, for example, one week or one month.

[0127] In this example, the ownership interval of the IP address 111.222.333.444, shown in Figure 4 as the ownership interval (402), will be approximately 7800 days, that corresponds to the period from January, 2000 to May, 2021.

[0128] At the same time, the ownership interval (404) of the other IP address, shown in Fig. 4, will be approximately 120 days, that corresponds to the period from January to May, 2021.

[0129] It is worth noting that “approximately” in this case means that taking a decision, which is described below in respect to step (280), does not require to compute the ownership interval within the accuracy of one day. The said computation could be performed either with a predetermined accuracy, say, within the accuracy of 10 days, or imply rounding of the computed value according to well-known arithmetic rules.

[0130] After computing the ownership interval of at least one suspicious IP address by the analysis module (108), step (270) ends, and the method proceeds to step (280).

[0131] At step (280), it is determined, by the analysis module (108), whether the value of the computed ownership interval of at least one suspicious IP address exceeds a preset threshold value.

[0132] This threshold value could be selected experimentally at the stage of preparing the system implementing the method, or it could be selected automatically depending on various parameters, for example, depending on the properties of the protected infrastructure (150), such as the total number of the computing devices or any other parameters. In this example, the threshold value could be selected as equal to, say, 200 days.

[0133] If the value of the computed ownership interval of at least one suspicious IP address exceeds a preset threshold value, this is interpreted as an indication of the fact that at least one computing device having this at least one suspicious IP address functions on the Internet long enough not to be considered malicious.

[0134] Thus, for example, at step (280), the analysis module (108) determines that the computed ownership interval (402) of the IP address 111.222.333.444 (7800 days) exceeds the preset threshold value (200 days).

[0135] In this case, the device with the specified IP address is considered legitimate. This completes the described method. In the other possible embodiment of the method, after that, the method returns to step (210).

[0136] Alternatively, at step (280), it is determined by the analysis module (108) that the computed ownership interval (404) by some other IP address B (120 days) does not exceed the preset threshold value (200 days).

[0137] In response to this, the method proceeds to step (290), where the computing device located on this IP address is considered malicious, generating, by the notification module (110), at least one report on detection of malicious activity in the protected infrastructure (150) and sending it via the communication module (102). This completes the described method. In the other possible embodiment of the method, after sending at least one report, the method returns to step (210).

[0138] As described above in reference to Fig. 1 and the notification module (150), this module enables to generate and send (via the communication module) notifications about potential compromise of the protected network infrastructure by at least one of the following methods: by e-mail, by sending SMS, by sending MMS, by sending a push notification, by a message in an instant messenger by generating an API event.

[0139] It is worth noting that use of such a notification tool as API events enables to implement additional integration of the described system with various third-party tools, such as firewalls, security management platforms, SIEM solutions, and so on. Actually, the generation of all the listed notifications, such as emails, SMS, MMS, push notifications, etc. is not specific and could be performed by any well-known method.

[0140] It is worth noting that the scanning module (126) of the internal computing device (120) could also be configured to determine that at least one suspicious device scanned at the above step (230) is inaccessible for scanning. That is, detecting a situation when, in response to scanning at least one suspicious external (global) IP address, it is not possible to get a list of ports open on this IP address.

[0141] It is also worth noting that the external computing device (100) and the internal computing device (120) could be configured to implement an alternative algorithm of the method for active detection of malicious network resources. Such an alternative algorithm (201) is described below with reference to Fig. 2C.

[0142] The method (201) begins at step (211) when the outbound traffic of the protected infrastructure (150) is received by the communication module (122) of the internal computing device (120). Like step (210), this could be done by any well-known method.

[0143] All outbound traffic received by the communication module (122) is transmitted unchanged to the analysis module (128) for analysis. In this case, the method (201) proceeds to step (221), where there is detecting at least one suspicious external IP address in the outbound traffic using the analysis module (128) of the internal computing device (120). This is also done in exactly the same way as at step (220) above.

[0144] In response to the fact that when trying to scan at least one suspicious external (global) IP address by the scanning module (126) of the internal computing device in accordance with the way described for step (230), there is determining that at least one device on at least one suspicious external (global) IP address, for example, a malefactor's server (190), is inaccessible for scanning, at step (231) transmitting the information about at least one suspicious external (global) IP-address to an external computing device (100) using the communication module (122) of the internal computing device (120).

[0145] This completes step (231), and the method proceeds to step (241).

[0146] At step (241), using the scanning module (106) of the external computing device (100), there is scanning at least one suspicious device located on at least one suspicious IP address, for example, IP address 111.222.333.444, obtaining a list of ports open on this IP address, and also a list of web services running on these ports and software versions corresponding to these web services. Actually, scanning is performed in the same way as described for step (230).

[0147] When the scanning module (106) of the external computing device (100) receives a list of ports open on a given IP address, and also a list of web services running on these ports and software versions corresponding to these web services, the listed information is stored in the database of the data storage module (104) of the first computing device (100).

[0148] Then, step (241) ends, and the method proceeds to step (258), which is executed completely similarly to step (250) described above with reference to Fig. 2B.

[0149] Upon completion of step (258), the method proceeds to steps (261), (271), (281) and (291), which are executed in exactly the same way as described above for steps (260), (270), (280) and (290).

[0150] Fig. 5 illustrates a schematic diagram of the computer device (500) processing the data required for embodiment of the claimed solution. As described above in reference to Fig. 1, the modules implementing the described method, in terms of their technical implementation, could be as software modules on a single computing device, as software and hardware modules. For example, at least some of these modules, as well as a combination of these modules, could be implemented as a computing device (500) shown in Fig. 5.

[0151] In general, the device (500) comprises such components as: one or more processors (501), at least one random access memory, or memory (502), data storage means (503), input/output interfaces (504), input/output means (505), networking or, what is the same, data communication means (506).

[0152] The device processor (501) executes main computing operations, required for functioning the device (500) or functionality of one or more of its components. The processor (501) runs the required machine-readable commands, contained in the random-access memory (502).

[0153] The memory (502), typically, is in the form of RAM and comprises the necessary program logic ensuring the required functionality.

[0154] The data storage means (503) could be in the form of HDD, SSD, RAID, networked storage, flash-memory, optical drives (CD, DVD, MD, Blue-Ray disks), etc.

[0155] Interfaces (504) are standard means for connection and operation with server side, e.g. USB, RS232, RJ45, LPT, COM, HDMI, PS/2, Lightning, FireWire, etc. Selection of interfaces (504) depends on the specific device (500), which could be a personal computer, mainframe, server cluster, thin client, smartphone, laptop, etc.

[0156] As input/output means (505) there could be used: keyboard, joystick, display (touch-screen display), projector, touch pad, mouse, trackball, light pen, loudspeakers, microphone, etc.

[0157] Networking means (506) are selected from a device providing network data receiving and transfer, e.g. Ethernet-card, WLAN/Wi-Fi module, Bluetooth module, BLE module, NFC module, IrDa, RFID module, GSM modem, etc. Making use of the means (506) provides an arrangement of data exchange through wire or wireless data communication channel, e.g. WAN, PAN, LAN, Intranet, Internet, WLAN, WMAN or GSM.

[0158] The components of the device (500) are interconnected by the common data bus (510).

[0159] In conclusion it is worth noting that the data given in the specification are the examples only, which do not limit the scope of this technique specified by the claims. It is obvious to those skilled in the art that there could exist other embodiments of this technique compliant with the essence and scope of this technique.

[0160] The exemplary systems and methods illustrated herein could be described in terms of the functional block components. It will be appreciated that such functional blocks could be implemented by any number of hardware and/or software components configured to perform these functions. For example, the system could use various components of an integrated circuit, such as storage elements, processing elements, logic elements, look-up tables, etc., which can perform a variety of functions under control of one or more microprocessors or other control devices. Similarly, the system software elements could be implemented using any programming or scripting language such as C, C ++, C #, Java, JavaScript, VBScript, Macromedia Cold Fusion, COBOL, Microsoft Active Server Pages, assembler language, Perl, PHP, AWK, Python, Visual Basic, stored SQL and PL/SQL procedures, any UNIX shell scripts and Extensible Markup Language (XML), and different algorithms are implemented with any combination of data structures, objects, processes, procedures or other program elements.

[0161] Moreover, the system for monitoring a cyberattack surface could operate on a single computing device, or on several devices interconnected over a network. Additionally, it is worth noting that the system can use any number of conventional techniques for data transmission, signaling, data processing, network management, etc. Additionally, the system can be used to detect or prevent security issues with client scripting languages such as JavaScript, VBScript, etc.

[0162] In this context, the devices mean any computing devices software- and hardware-based, for example, such as: personal computers, servers, smartphones, laptops, tablets, etc.

[0163] A processor, microprocessor, computer, PLC (programmable logic controller) or an integrated circuit configured to execute certain commands (instructions, programs) for data processing could be a data processing device. A processor could be multi-core for parallel data processing.

[0164] Memory devices can include, but not limited to, hard disk drives (HDD), flash memory, ROM (read only memory), solid state drives (SSD), etc.

[0165] It is worth noting that the specified device may include any other devices known in the art, for example, such as sensors, input/output devices, display devices (displays), etc. The input/output device could be, but not limited to, for example, a mouse, keyboard, touch pad, stylus, joystick, track-pad, etc.

[0166] The application materials have represented the preferred embodiment of the claimed technical solution, which shall not be used as limiting the other particular embodiments, which are not beyond the claimed scope of protection and are obvious to persons skilled in the art.

CLAIMS

1. A computer-implementable method for active detection of malicious network resources, the method being executable by a distributed computer system, the method comprising:

- receiving, by a communication module of an internal computing device, an outbound infrastructure traffic,
- detecting, by an analysis module of the internal computing device, at least one suspicious external IP address in the outbound traffic,
- scanning, by a scanning module of the internal computing device, at least one suspicious device located on detected at least one suspicious IP address, and obtaining a list of services running on the suspicious device,
- sending, by the communication module of the internal computing device, detected at least one suspicious IP address and the obtained list of services to an external computing device,
- comparing, using an analysis module of the external computing device, the obtained list of services with a data on known malicious services, and
 - in response to a coincidence of at least one service from the obtained service list with the data on known malicious services:
 - considering, by the analysis module of the external computing device, the suspicious device, located on at least one suspicious IP address, to be malicious one,
 - generating, by a notification module of the external computing device, at least one report on detection of malicious activity in the infrastructure,
 - in response to absence of coincidences of services from the obtained service list with the data on known malicious services,
 - computing, by the analysis module of the external computing device, an interval of ownership of at least one suspicious IP address,
 - in response to the fact that the computed duration of the ownership interval is less than a threshold value,

- considering, by the analysis module of the external computing device, the suspicious device, located on at least one suspicious IP address, to be malicious one,

- generating, by a notification module of the external computing device, at least one report on detection of malicious activity in the infrastructure,

- in response to the fact that the computed duration of the ownership interval exceeds the threshold value, considering, by the analysis module of the external computing device, the suspicious device to be legitimate one.

2. The method of claim 1, wherein in response to inaccessibility of at least one suspicious device, located on detected at least one suspicious IP address, for the scanning module of the internal computing device:

- transmitting, by the communication module of the internal computing device, the information about at least one suspicious IP address to the external computing device,

- scanning, by a scanning module of the external computing device, the suspicious device located on at least one suspicious IP address,

- obtaining, by the scanning module of the external computing device, a list of services running on the suspicious device.

3. The method of claim 1, wherein for detecting, by the analysis module of the internal computing device, at least one suspicious IP address in the outbound traffic:

- detecting at least one anomalous traffic fragment in the outbound traffic,

- extracting from the detected at least one anomalous traffic fragment at least one IP address of a computing device to which the said at least one traffic fragment is addressed,

- considering the extracted at least one IP address to be suspicious.

4. The method of claim 3, wherein detecting at least one anomalous traffic fragment in the outbound traffic, at least one of the following methods is used:

- detecting by a deep neural network preliminarily trained on a tagged traffic,

- detecting by a OneClassSVM classifier preliminarily trained on legitimate traffic samples,

- detecting by machine learning models, such as RandomForestClassifier, XGBoost, KNN, AdaBoost, etc., preliminarily trained on tagged samples of traffic,

- searching in the outbound traffic by preformed feature signatures,

- using sliding window methods and collecting statistical features of traffic logs.

5. The method of claim 1, wherein for comparing the obtained list of services with a data on known malicious services, a graph model of the Internet is used.

6. The method of claim 1, wherein for computing the interval of ownership of at least one suspicious IP address historical data on the activity of IP addresses is used.

7. The method of claim 1, wherein at least one generated report on detection of malicious activity in the infrastructure further transmitting by at least one of the following communication methods:

- e-mail,
- SMS,
- MMS,
- push notifications,
- instant messenger messages,
- API events

8. A system for active detection of malicious network resources, the system comprising:

- an external computing device, the external computing device installed outside an infrastructure and configured to communicate with the infrastructure devices, the external computing device comprising:

- a communication module,
- a scanning module,
- a notification module,
- an analysis module,
- a data storage module, the data storage module storing:
 - an instructions, the instructions, if being executed by a processor, implement the method according to claims 1-7,
 - at least one database, the database comprising:
 - data on known malicious services,
 - historical data on the activity of various IP addresses,
 - a graph model of the Internet,

- an internal computing device, the internal computing device installed within the infrastructure and configured to receive the infrastructure outbound traffic, and comprising at least:

- a communication module,
- a scanning module,
- an analysis module,
- a data storage module, the data storage module storing:
 - an instructions, the instructions, if being executed by a processor, implement the method according to claims 1-7.