(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
19 April 2018 (19.04.2018)

WIPO | PCT

(10) International Publication Number
WO 2018/071356 A1

(51) International Patent Classification:
G06F 21/55 (2013.01) *G06F 21/60* (2013.01)

(21) International Application Number:
PCT/US2017/055826

(22) International Filing Date:
10 October 2017 (10.10.2017)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

62/407,573	13 October 2016 (13.10.2016)	US
62/407,576	13 October 2016 (13.10.2016)	US
15/725,974	05 October 2017 (05.10.2017)	US

(71) Applicant: **NEC LABORATORIES AMERICA, INC.** [US/US]; 4 Independence Way, Suite 200, Princeton, New Jersey 08540 (US).

(72) Inventors: **TANG, LuAn**; 10 Manley Road, Pennington, New Jersey 08534 (US). **ZHANG, Hengtong**; c/o NEC Laboratories America, Inc., 4 Independence Way, Suite 200, Princeton, New Jersey 08540 (US). **CHEN, Zhengzhang**; 44 York Road, Princeton Junction, New Jersey 08550 (US). **ZONG, Bo**; 6821 Ravens Crest Drive, Plainsboro, New Jersey 08536 (US). **LI, Zhichun**; 306 Sayre Drive, Princeton, New Jersey 08540 (US). **JIANG, Guofei**; 5 Danby Court, Princeton, New Jersey 08540 (US). **YOSHIHIRA, Kenji**; 26 Sherbrooke Drive, Princeton Junction, New Jersey 08550 (US).

(74) Agent: **KOLODKA, Joseph**; NEC Laboratories America, Inc., 4 Independence Way, Suite 200, Princeton, New Jersey 08540 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV,

(54) Title: GRAPH-BASED ATTACK CHAIN DISCOVERY IN ENTERPRISE SECURITY SYSTEMS

(57) Abstract: Methods and systems for detecting anomalous events include detecting anomalous events (42, 43) in monitored system data. An event correlation graph is generated (302) based on the monitored system data that characterizes the tendency of processes to access system targets. Kill chains are generated (310) that connect malicious events over a span of time from the event correlation graph that characterize events in an attack path over time by sorting events according to a maliciousness value and determining at least one sub-graph within the event correlation graph with an above-threshold maliciousness rank. A security management action is performed (412) based on the kill chains.

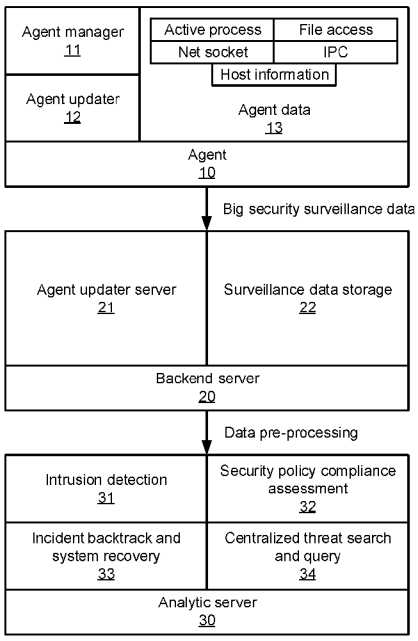

FIG. 1

WO 2018/071356 A1

[Continued on next page]

MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

GRAPH-BASED ATTACK CHAIN DISCOVERY IN ENTERPRISE SECURITY SYSTEMS

RELATED APPLICATION INFORMATION

[0001] This application claims priority to provisional application serial numbers 62/407,573 and 62/407,576, filed on October 13, 2016, both of which are incorporated herein in their entirety.

BACKGROUND

Technical Field

[0002] The present invention relates to computer and network security and, more particularly, to discovery of attack chains from system monitoring logs.

Description of the Related Art

[0003] Enterprise networks are key systems in corporations and they carry the vast majority of mission-critical information. As a result of their importance, these networks are often the targets of attack. Communications on enterprise networks are therefore frequently monitored and analyzed to detect anomalous network communication as a step toward detecting attacks.

[0004] In particular, advanced persistent threat (APT) attacks, which persistently use multiple complex phases to penetrate a targeted network and steal confidential information, have become major threats to enterprise information systems. Existing rule/feature-based approaches for APT detection may only discover isolated phases of an attack. As a result, these approaches may suffer from a high false-positive rate and cannot provide a high-level picture of the whole attack.

SUMMARY

[0005] A method for detecting anomalous events include detecting anomalous events in monitored system data. An event correlation graph is generated based on the monitored system data that characterizes the tendency of processes to access system targets. Kill chains are generated that connect malicious events over a span of time from the event correlation graph that characterize events in an attack path over time by sorting events according to a maliciousness value and determining at least one sub-graph within the event correlation graph with an above-threshold maliciousness rank. A security management action is performed based on the kill chains.

[0006] A system for detecting anomalous events includes an anomaly detection module configured to detect anomalous events in monitored system data. A kill chain module includes a processor configured to generate an event correlation graph based on the monitored system data that characterizes the tendency of processes to access system targets and to generate kill chains that connect malicious events over a span of time from the event correlation graph that characterize events in an attack path over time by sorting events according to a maliciousness value and determining at least one sub-graph within the event correlation graph with an above-threshold maliciousness rank. A security module is configured to perform a security management action based on the kill chains.

[0007] These and other features and advantages will become apparent from the following detailed description of illustrative embodiments thereof, which is to be read in connection with the accompanying drawings.

BRIEF DESCRIPTION OF DRAWINGS

[0008] The disclosure will provide details in the following description of preferred embodiments with reference to the following figures wherein:

[0009] FIG. 1 is a block/flow diagram directed to an automatic security intelligence system architecture in accordance with an embodiment of the present principles.

[0010] FIG. 2 is a block/flow diagram directed to an intrusion detection engine architecture in accordance with an embodiment of the present principles.

[0011] FIG. 3 is a block/flow diagram of a method/system for discovering attack chains in accordance with an embodiment of the present principles.

[0012] FIG. 4 is a block diagram of an intrusion detection system in accordance with an embodiment of the present principles.

[0013] FIG. 5 is a block diagram of a processing system accordance with an embodiment of the present principles.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

[0014] In accordance with the present principles, the present embodiments provide a two-stage framework that takes massive system monitoring logs and record-level alert labels and recovers well-organized attack chains (also referred to herein as "kill chains"). The present embodiments thereby decrease the false positive rate of alert labels. The two-stage framework constructs an event correlation graph from the monitored log data and then generates kill chains from the event correlation graph.

[0015] In an advanced persistent threat (APT) graph, most attack steps are well-camouflaged as normal events whose malicious intent cannot be readily determined. At the same time, rule-based event detectors will generate false positives, marking innocuous events as being potentially malicious. It can be difficult for a system administrator to locate and recover real APT attacks from a high volume of false

positives. The kill chains discovered by the present embodiments can be generated automatically and can be part of a fully automated security system that does not involve the direct intervention by an administrator. No prior knowledge about the attack is needed, nor are any labeled training datasets used.

[0016] This problem is a result of the fact that isolated events do not provide enough contextual information to determine that they are malicious. To this end, the present embodiments jointly consider multiple alerts and normal events to more accurately identify APT attacks. The present embodiments therefore output graphs of linked events that recover the procedures of possible APT attacks, referred to herein as "kill chains."

[0017] Referring now in detail to the figures in which like numerals represent the same or similar elements and initially to FIG. 1, an automatic security intelligence system (ASI) architecture is shown. The ASI system includes three major components: an agent 10 is installed in each machine of an enterprise network to collect operational data; backend servers 200 receive data from the agents 10, pre-process the data, and sends the pre-processed data to an analysis server 30; and an analysis server 30 that runs the security application program to analyze the data.

[0018] Each agent 10 includes an agent manager 11, an agent updater 12, and agent data 13, which in turn may include information regarding active processes, file access, net sockets, number of instructions per cycle, and host information. The backend server 20 includes an agent updater server 21 and surveillance data storage. Analysis server 30 includes intrusion detection 31, security policy compliance assessment 32, incident backtrack and system recovery 33, and centralized threat search and query 34.

[0019] Referring now to FIG. 2, additional detail on intrusion detection 31 is shown. There are five modules in an intrusion detection engine: a data distributor 41 that receives the data from backend server 20 and distributes the corresponding to network level module 42 and host level module 43; network analysis module 42 that processes the network communications (including TCP and UDP) and detects abnormal communication events; host level analysis module 43 that processes host level events, including user-to-process events, process-to-file events, and user-to-registry events; anomaly fusion module 44 that integrates network level anomalies and host level anomalies and refines the results for trustworthy intrusion events; alert ranking module 46 that takes the intrusion events and ranks them according to degree of suspicion; attack chain discovery module 47 that filters out untrustworthy alerts (e.g., by removing alerts that have a rank or degree of suspicion below a threshold value) and organizes trustworthy alerts into kill chains; and visualization module 45 that outputs the detection results to end users.

[0020] Referring now to FIG. 3, additional detail on the attack chain discovery module 46 is shown. As noted above, kill chain discovery includes two stages: event correlation/graph construction block 302 and kill chain generation 310. In the first stage 302, the present embodiments learn the behavioral characteristics of processes as well as the interactive patterns that occur between processes, further discovering the correlations among isolated system events. The monitored system events and alerts are provided as input 304 to the event correlation/graph construction block 302. Event correlation is performed in block 306 and an event triggering graph is created in block 308.

[0021] In kill chain generation 310, a greedy algorithm may be used to generate sub-graphs in block 312, the sub-graphs having a high likelihood of representing the

procedures of APT attacks. Block 314 generates a kill chain from the malicious sub-graph to identify malicious events and organize them into meaningful stories. Block 316 then polishes and refines the kill chains.

[0022] Each system event that is input in block 304 can be represented as a four-tuple $(i_v, t_v, d_v, \delta_v)$, where i_v is an index among all the events, i_v is a host process of the v^{th} event, t_v is the timestamp for the v^{th} event, d_v is the target (e.g., documents, network addresses, etc.) of the v^{th} event, and δ_v is the malicious label generated by the analysis engine (though it should be noted that this label may be incorrect). The set of all system events is denoted as $\mathcal{O} = \{(I_v, t_v, d_v, \delta_v)\}_{\{v=1\}}^{\{|\mathcal{O}|\}}$.

[0023] The present embodiments accomplish the following goals:

[0024] 1. Construct a directed event correlation graph: $G = \{V, E, w\}$, where each vertex $v \in V$ in the graph corresponds to an observed system event and each edge $e \in E$ corresponds to an event triggering correlation, which is inferred from the sequence of system events. The function $w: V \rightarrow \mathbb{R}$ denotes a function that assigns a non-negative value $w(v)$ to each event v based on the anomaly labels generated by the analysis engine (the online anomaly fusion module 44).

[0025] 2. Find subsets of events $S \subset V$ which have high maliciousness scores and are sufficiently compact. The subgraphs that include these subsets of events are considered as kill-chains.

[0026] Thus the first stage 302 finds triggering correlations among isolated events, making use of use of Hawkes processes to model the event data and then describing the proposed model that learns the triggering correlations from observed system sequences.

[0027] Hawkes processes are a family of counting processes that model sequences of "event arrivals" over time. Each event arrival "excites" the process by increasing

the likelihood of event arrivals for some time period after an initial event. The univariate Hawkes process is defined to be a point process $N(t)$ having an intensity function $\lambda(t)$ is defined as:

$$\lambda(t) = \mu + \alpha \sum_{t_\ell < t} \kappa(t - t_\ell)$$

where μ is the non-negative base intensity, α is the non-negative self-exciting coefficient, $\kappa(\cdot)$ is the decay function. The intensity captures the tendency of new events to occur within a time interval after t.

[0028] The multivariate Hawkes process is an extension of the univariate Hawkes process. The multivariate Hawkes process can be used to model the influence among individual dimensions. The intensity function of the i^{th} dimension, based on past events, is defined as:

$$\lambda_i(t) = \mu_i + \sum_{(t_\ell < t)} \alpha_{i i_\ell} \kappa(t - t_\ell)$$

where μ_i is the base intensity of the i^{th} dimension, i_ℓ is the dimension identity of the ℓ^{th} event, $\alpha_{i i_\ell}$ is a coefficient that captures the influence between the i^{th} dimension and the $i_\ell{}^{\text{th}}$ dimension. A larger $\alpha_{i i_\ell}$ corresponds to a greater tendency of the $i_\ell{}^{\text{th}}$ dimension to trigger the i^{th}.

[0029] It is specifically contemplated that the decay function may be defined as $\kappa(x) = e^{-cx}$, where c is an exponential coefficient. It should be understood, however, that this particular definition is only exemplary—other choices for the decay function may be used instead.

[0030] The occurrence of a system event may be caused not only by mechanisms internal to a process of interest itself, but by the influence of events created by other processes. The degree of influence between different kinds of process may vary. For example, version control software tends to interact more with code editors than email

clients. Therefore, events between version control software and code editors are more likely to be correlated. However, the innate character of each host process and the underlying triggering correlations among events are not known in advance. The present embodiments therefore learn the character of each host process as well as interactive tendencies between host processes from the observed individual events, discovering the triggering correlations between events.

[0031] At any given time t_v, for each pair of host process and target, the estimated tendency function $\lambda_{id}(t_v)$ indicates the tendency that a process *i* accesses a target *d* at time t_v. The estimated tendency function may be written as:

$$\lambda_{id}(t_v) = \delta_{id} + \sum_{p \in \mathcal{O}_i}^{t_p < t_v} \alpha_i \kappa(t_v - t_p) + \sum_{j \in E \setminus i} \sum_{q \in \mathcal{O}_{jd}}^{t_q < t_v} b_{ij} \kappa(t_v - t_q)$$

where δ_{id} denotes the base tendency rate for the process *i* to access target *d*, a_i is a process-specific parameter modeling the impact of historical events on *i*'s future behaviors, and b_{ij} is a non-negative parameter representing how likely process j is to trigger processes *i*. Low-rank constraints are added to δ_{id}. These constraints are based on the observation that interactions between processes and targets can be categorized into a limited number of types. $\mathcal{O}_{jd}$ is the sequence of events executed by host process *j* on target *d*.

[0032] The intensity of system event sequences between the process *i* and the target *b* is captured by three separate terms. The first term captures the tendency of the process *I* to access the target *d*, the second term captures the influence of past events from the process *I* toward the current event, and the third term captures the influence of events from processes other than *i* toward the occurrence tendency of current events. The inter-process interactions may be carried out via some shared targets. As a

result, the number of events from processes other than i that can potentially influence the current event can be narrowed down.

[0033] If $\mathcal{O}_{jd}$ is the sequence of events carried out by host process j on target d, the negative log-likelihood for such observation is denoted as:

$$\mathcal{L}(\mathcal{O}_{id}) = -\sum_{v\in\mathcal{O}_{id}} \log\lambda_{id}(t_v) + T_{id}\cdot\delta_{id} + \sum_{p\in\mathcal{O}_{id}} a_i G\big(T-t_p\big) + \sum_{j\in E\setminus i}\sum_{q\in\mathcal{O}_{jd}} b_{ij} G\big(T-t_q\big)$$

where T_{id} is the timespan of this sequence and $G(t) = \int_0^t e^{-s}ds$. By the summation of all possible $\mathcal{O}_{id}$, the negative log-likelihood of all observations can be determined.

[0034] Adding structural regularization terms to the negative log likelihood function, the following optimization is performed:

$$\min_{a,B,\Delta} \sum_{i=1}^{M}\sum_{d=1}^{N} \mathcal{L}(\mathcal{O}_{id}) + \alpha\sum_{i=1}^{M}(a_i)^2 + \beta\sum_{i=1}^{M}\sum_{j=1}^{M}\big(b_{ij}\big)^2 + \gamma\|\Delta\|_*$$

Such that $a \geq 0, B \geq 0, \Delta \geq 0$. The terms α, β, and γ are hyper-parameters that are predetermined based on previous experience. There are furthermore three normalization functions, and the hyper-parameters control the weight of each. Δ is the matrix formed by δ_{ij}.

[0035] To solve the optimization problem, an alternating direction method of multipliers (ADMM) framework is used to convert the original optimization problem into simpler sub-problems. The optimization problem is rewritten as an equivalent formulation using the auxiliary variable $Z_1 = \Delta$ as:

$$\min_{a,B,\Delta,Z_1} \sum_{i=1}^{M}\sum_{d=1}^{N} \mathcal{L}(\mathcal{O}_{id}) + \alpha\sum_{i=1}^{M}(a_i)^2 + \beta\sum_{i=1}^{M}\sum_{j=1}^{M}\big(b_{ij}\big)^2 + \gamma\|Z_1\|_* + \rho\mathrm{tr}\big(X_1^T(\Delta - Z_1)\big) + \frac{\rho}{2}\|\Delta - Z_1\|_F^2$$

such that $a \geq 0, B \geq 0, \Delta \geq 0$, where ρ is a penalty coefficient and X_1 is a dual variable associated with the constraints $\Delta = Z_1$, which can be solved iteratively by the following steps.

[0036] A first step updates a, B, and Δ. As $\mathcal{L}(\mathcal{O}_{id})$ includes a sum of logarithm fractions, which do not have succinct derivative results, a surrogate approximation is introduced using Jensen's inequality to produce the following closed-form solutions:

$$\delta_{id} = \frac{-B_1 + \sqrt{B_1^2 - 4\rho C_1}}{2\rho}$$

$$a_i = \frac{-B_2 + \sqrt{B_2^2 - 8\alpha C_2}}{4\alpha}$$

$$b_{ij} = \frac{-B_3 + \sqrt{B_3^2 - 8\beta C_3}}{4\beta}$$

where:

$$B_1 = \rho X_{1,id}^k - \rho Z_{1,id}^k + T_{id}$$

$$C_1 = -\sum_{v \in \mathcal{O}_{id}} p_{id}^{\Delta}$$

$$B_2 = \sum_{d=1}^{N} \sum_{p \in \mathcal{O}_{id}} G\left(T_{id} - t_p\right)$$

$$C_2 = -\sum_{d=1}^{N} \sum_{v \in \mathcal{O}_{id}} \sum_{\ell \in \mathcal{O}_{id}}^{t_\ell < t_v} p_{ii}^{a}$$

$$B_3 = \sum_{d=1}^{N} \sum_{q \in \mathcal{O}_{jd}} G\left(T_{id} - t_q\right)$$

$$C_3 = -\sum_{d=1}^{N} \sum_{v \in \mathcal{O}_{id}} \sum_{\ell \in \mathcal{O}_{jd}}^{t_\ell < t_v} p_{ij}^{b}$$

and a is the vector of a_i and B is the matrix formed by b_{ij}.

[0037] The second step updates Z_1 using the updated parameters from the first step. This is performed by solving:

$$\min_{Z_1} \beta \|Z_1\|_* + \frac{\rho}{2} \left\|\Delta - Z_1^k + X_1^k\right\|_F^2$$

A closed form solution can be obtained by soft-thresholding:

$$Z_1^{k+1} = S_{\beta/\rho}(\Delta^{k+1} + X_1^k)$$

where $S(\cdot)$ is a soft-thresholding function defined as:

$$S_\alpha(X) = U\text{diag}((\sigma_i - \alpha)_+)V^T$$

where U and V are two square matrices and $diag(\cdot)$ is a matrix that has zeros for every element off the diagonal.

[0038] The third step updates the dual variable X_1 according to the ADMM framework: $X_1^{k+1} \leftarrow X_1^k + (\Delta^{k+1} - Z_1^{k+1})$. The following pseudo-code summarizes the optimization:

[0039] Randomly initialize a, B, Δ and set $X_1 = 0$.

[0040] while (a, B, Δ not converged) do

[0041] update a, B, Δ by iteratively optimizing as described in the first step above

[0042] update Z^{k+1} as described in the second step above

[0043] update $X_1^{k+1} \leftarrow X_1^k + (\Delta^{k+1} - Z_1^{k+1})$

[0044] end while

[0045] return a, B, Δ

[0046] Through event sequence modeling, a, B, Δ are obtained that capture the inner-process and inter-process triggering patterns of every event. The correlation matrix $R = [r_{mn}] \in \mathbb{R}_+^{n \times x}$, where r_{mn} represents how likely it is that the n^{th} event will trigger the m^{th} event. For a pair of events carried out by the same process, the strength

of event triggering tendency are correlated to the self-influence factor of that process and the timespan between them. For a pair of events carried out by different processes, their interactions should be build upon accessing some shared targets historically. As such:

$$r_{mn} = \begin{cases} a_{i_m}\kappa(\mathrm{t_n} - \mathrm{t_m}) & \text{If } i_m = i_n \\ b_{i_m,t_m}\kappa(t_n - t_m) & \text{If } i_m \neq i_n \text{ and } d_m = d_n \\ 0 & \text{Otherwise} \end{cases}$$

[0047] By thresholding continuous event correlations, a directed event correlation graph $G = \{V, E, w\}$ can be constructed where each vertex $v \in V$ in the graph corresponds to an observed system event and each edge $e \in E$ corresponds to an event triggering correlation, which can be inferred from the system events. $w: V \to \mathbb{R}$ denotes a function that assigns a non-negative value $w(v)$ to each event v based on anomaly labels.

[0048] Block 302 thereby generates the event correlation graph G, which may include multiple connected components $\{G_1, G_2, ...\}$. If the malicious value of each connected component is written as $\{|w|_1, |w|_2, ...\}$, alerts that are not in the top-K connected components are treated as false positive alerts. Those connected components may still not be precise, however, so after generating the kill-chains, block 316 polishes them.

[0049] As noted above, block 312 discovers dense sub-subgraphs by finding a subset of correlated events $S \subset V$, for each connected component, that has a large maliciousness value and is sufficiently compact. More formally, the subset S maximizes the objective function:

$$Q(S) = \eta \sum_{v \in S} w(v) - \frac{1}{2} \sum_{m \in S} \sum_{n \in S} d(m, n)$$

where η is a trade-off normalization coefficient and $d(m,n)$ stands for the distance of a shortest path through G from m to n, with $d(m,n)=\infty$ if there is no path from m to n through G.

[0050] Finding $S \subset V$ that minimizes $Q(S)$ in general graphs is an NP-hard problem. The present embodiments therefore approximate the optimization using a linear-time ½ approximation based on a random double-greedy search. Because such an approximation needs the objective function to be non-negative, $Q(S)$ is rewritten as follows:

$$Q^{+}(S)=\eta\sum_{v\in S}w(v)-\frac{1}{2}\left(\sum_{m\in S}\sum_{n\in S}d(m,n)+\sum_{p\in V}\sum_{q\in V}d(p,q)\right)$$

[0051] Finding a suitable trade-off parameter η is important. The approximation is therefore extended by adding a strategy to automatically search for the optimal η. The following pseudo-code provides approximation details, where $h(X)$ denotes the number of events with alert labels in event set X.

[0052] for $k = 1,2,\ldots$ do

[0053] $X_0 \leftarrow \Phi,\ Y_0 \leftarrow V$

[0054] for $i = 1$ to $|V|$ do

[0055] $a_i \leftarrow \max\{Q^{+}(X_{i-1}\cup\{v_i\})-Q^{+}(X_{i-1}),0\}$

[0056] $b_i \leftarrow \max\{Q^{+}(Y_{i-1}\cup\{v_i\})-Q^{+}(Y_{i-1}),0\}$

[0057] Draw $\delta \sim \text{Bernoulli}\left(\frac{a_i}{a_i+b_i},\frac{b_i}{a_i+b_i}\right)$

[0058] if $\delta = 0$ then

[0059] $X_i \leftarrow X_{i-1}\cup\{v_i\},\ \ Y_i \leftarrow Y_{i-1}$

[0060] else

[0061] $X_i \leftarrow X_{i-1},\ \ Y_i \leftarrow Y_{i-1}\backslash\{v_i\}$

[0062] end if

[0063] end for

[0064] if $h(X) < h(V)$ then

[0065] Increase η by $\eta^{k+1} \leftarrow \eta^k \cdot 2$

[0066] else

[0067] Decrease η by $\eta^{k+1} \leftarrow \eta^k / 2$

[0068] end if

[0069] end for

[0070] return $X_{|V|}$

[0071] The kill chains inferred by this process may be too chaotic for interpretation by end users. The goal of block 316 is to keep the longest possible kill chain. Block 316 sorts the system events $v \in V$ in ascending order of timestamp. For each event v, block 316 finds the last event l_v that may trigger v and deletes edges other than (l_v, v) from G that point to v. The resulting polished kill chain is output by block 310 for use in subsequent analysis.

[0072] Embodiments described herein may be entirely hardware, entirely software or including both hardware and software elements. In a preferred embodiment, the present invention is implemented in software, which includes but is not limited to firmware, resident software, microcode, etc.

[0073] Embodiments may include a computer program product accessible from a computer-usable or computer-readable medium providing program code for use by or in connection with a computer or any instruction execution system. A computer-usable or computer readable medium may include any apparatus that stores, communicates, propagates, or transports the program for use by or in connection with the instruction execution system, apparatus, or device. The medium can be magnetic,

optical, electronic, electromagnetic, infrared, or semiconductor system (or apparatus or device) or a propagation medium. The medium may include a computer-readable storage medium such as a semiconductor or solid state memory, magnetic tape, a removable computer diskette, a random access memory (RAM), a read-only memory (ROM), a rigid magnetic disk and an optical disk, etc.

[0074] Each computer program may be tangibly stored in a machine-readable storage media or device (e.g., program memory or magnetic disk) readable by a general or special purpose programmable computer, for configuring and controlling operation of a computer when the storage media or device is read by the computer to perform the procedures described herein. The inventive system may also be considered to be embodied in a computer-readable storage medium, configured with a computer program, where the storage medium so configured causes a computer to operate in a specific and predefined manner to perform the functions described herein.

[0075] A data processing system suitable for storing and/or executing program code may include at least one processor coupled directly or indirectly to memory elements through a system bus. The memory elements can include local memory employed during actual execution of the program code, bulk storage, and cache memories which provide temporary storage of at least some program code to reduce the number of times code is retrieved from bulk storage during execution. Input/output or I/O devices (including but not limited to keyboards, displays, pointing devices, etc.) may be coupled to the system either directly or through intervening I/O controllers.

[0076] Network adapters may also be coupled to the system to enable the data processing system to become coupled to other data processing systems or remote printers or storage devices through intervening private or public networks. Modems,

cable modem and Ethernet cards are just a few of the currently available types of network adapters.

[0077] Referring now to FIG. 4, an intrusion detection system 400 is shown. The intrusion detection system 400 includes a hardware processor 402 and memory 404. The intrusion detection system 400 further includes one or more functional modules that, in some embodiments, may be implemented as software that is executed by the hardware processor 402 and stored in the memory 404. In other embodiments, the functional modules may be implemented as one or more discrete hardware components in the form of, e.g., application specific integrated chips or field programmable gate arrays.

[0078] An anomaly detection module 406 analyzes collected host-level and network-level events and finds events that appear to be anomalous, generating an alert associated with the event. Kill chain module 410 correlates anomalous events and creates a graph representation, then generates and polishes kill chains from the graph.

[0079] A security module 412 performs manual or automated security actions in response to the kill chains. In particular, the security module 412 may have rules and policies that trigger when kill chains indicate certain kinds of attacker behavior. Upon such triggers, the security module 412 may automatically trigger security management actions such as, e.g., shutting down devices, stopping or restricting certain types of network communication, raising alerts to system administrators, changing a security policy level, and so forth. The security module 412 may also accept instructions from a human operator to manually trigger certain security actions in view of analysis of the kill chains.

[0080] A visualization module 414 presents kill chain and security management information to a user. The visualization module 414 in particular shows the events of

the kill chain over time, relating them to one another in a manner that provides an easy-to-understand progression and potentially suggesting future attack paths. Based on the display of the kill chains at the visualization module 414, the user can trigger security management actions using the security module 412.

[0081] Referring now to FIG. 5, an exemplary processing system 500 is shown which may represent the transmitting device 100 or the receiving device 120. The processing system 500 includes at least one processor (CPU) 504 operatively coupled to other components via a system bus 502. A cache 506, a Read Only Memory (ROM) 508, a Random Access Memory (RAM) 510, an input/output (I/O) adapter 520, a sound adapter 530, a network adapter 540, a user interface adapter 550, and a display adapter 560, are operatively coupled to the system bus 502.

[0082] A first storage device 522 and a second storage device 524 are operatively coupled to system bus 502 by the I/O adapter 520. The storage devices 522 and 524 can be any of a disk storage device (e.g., a magnetic or optical disk storage device), a solid state magnetic device, and so forth. The storage devices 522 and 524 can be the same type of storage device or different types of storage devices.

[0083] A speaker 532 is operatively coupled to system bus 502 by the sound adapter 530. A transceiver 542 is operatively coupled to system bus 502 by network adapter 540. A display device 562 is operatively coupled to system bus 502 by display adapter 560.

[0084] A first user input device 552, a second user input device 554, and a third user input device 556 are operatively coupled to system bus 502 by user interface adapter 550. The user input devices 552, 554, and 556 can be any of a keyboard, a mouse, a keypad, an image capture device, a motion sensing device, a microphone, a device incorporating the functionality of at least two of the preceding devices, and so

forth. Of course, other types of input devices can also be used, while maintaining the spirit of the present principles. The user input devices 552, 554, and 556 can be the same type of user input device or different types of user input devices. The user input devices 552, 554, and 556 are used to input and output information to and from system 500.

[0085] Of course, the processing system 500 may also include other elements (not shown), as readily contemplated by one of skill in the art, as well as omit certain elements. For example, various other input devices and/or output devices can be included in processing system 500, depending upon the particular implementation of the same, as readily understood by one of ordinary skill in the art. For example, various types of wireless and/or wired input and/or output devices can be used. Moreover, additional processors, controllers, memories, and so forth, in various configurations can also be utilized as readily appreciated by one of ordinary skill in the art. These and other variations of the processing system 500 are readily contemplated by one of ordinary skill in the art given the teachings of the present principles provided herein.

[0086] The foregoing is to be understood as being in every respect illustrative and exemplary, but not restrictive, and the scope of the invention disclosed herein is not to be determined from the Detailed Description, but rather from the claims as interpreted according to the full breadth permitted by the patent laws. It is to be understood that the embodiments shown and described herein are only illustrative of the principles of the present invention and that those skilled in the art may implement various modifications without departing from the scope and spirit of the invention. Those skilled in the art could implement various other feature combinations without departing from the scope and spirit of the invention. Having thus described aspects of

the invention, with the details and particularity required by the patent laws, what is claimed and desired protected by Letters Patent is set forth in the appended claims.

WHAT IS CLAIMED IS:

1. A method for detecting anomalous events, comprising:

detecting anomalous events (42, 43) in monitored system data;

generating an event correlation graph (302) based on the monitored system data that characterizes the tendency of processes to access system targets; and

generating kill chains (310) that connect malicious events, using a processor, over a span of time from the event correlation graph that characterize events in an attack path over time by sorting events according to a maliciousness value and determining at least one sub-graph within the event correlation graph with an above-threshold maliciousness rank; and

performing a security management action (412) based on the kill chains.

2. The method of claim 1, wherein determining the at least one sub-graph comprises maximizing the objective function:

$$Q(S) = \eta \sum_{v \in S} w(v) - \frac{1}{2} \sum_{m \in S} \sum_{n \in S} d(m,n)$$

where S is a subset of correlated events that maximizes the objective function, η is a trade-off normalization coefficient, and $d(m,n)$ is a distance of a shortest path through the event correlation graph from node m to n.

3. The method of claim 1, wherein determining the at least one sub-graph comprises maximizing the approximated objective function:

$$Q^{+}(S) = \eta \sum_{v \in S} w(v) - \frac{1}{2} \left(\sum_{m \in S} \sum_{n \in S} d(m,n) + \sum_{p \in V} \sum_{q \in V} d(p,q) \right)$$

where S is a subset of correlated events that maximizes the objective function, η is a trade-off normalization coefficient, and $d(m, n)$ is a distance of a shortest path through the event correlation graph from node m to n.

4. The method of claim 3, wherein maximizing the approximated objective function comprises searching for a value for the trade-off normalization coefficient η that maximizes the approximated objective function.

5. The method of claim 1, further comprising pruning the generated kill chains, keeping a longest kill chain leading to a given event.

6. The method of claim 1, wherein performing the security action further comprises automatically performing at least one security action selected from the group consisting of shutting down devices, stopping or restricting certain types of network communication, raising alerts to system administrators, and changing a security policy level.

7. The method of claim 1, further comprising ranking the anomalous events according to a degree of suspicion.

8. The method of claim 7, further comprising removing anomalous events below a threshold rank.

9. The method of claim 1, further comprising displaying the kill chains on a graphical user interface for review by a user.

10. A system for detecting anomalous events, comprising:

an anomaly detection module (406) configured to detect anomalous events in monitored system data;

a kill chain module (410) comprising a processor configured to generate an event correlation graph based on the monitored system data that characterizes the tendency of processes to access system targets and to generate kill chains that connect malicious events over a span of time from the event correlation graph that characterize events in an attack path over time by sorting events according to a maliciousness value and determining at least one sub-graph within the event correlation graph with an above-threshold maliciousness rank; and

a security module (412) configured to perform a security management action based on the kill chains.

11. The system of claim 10, wherein the kill chain module is further configured to maximize the objective function:

$$Q(S) = \eta \sum_{v \in S} w(v) - \frac{1}{2} \sum_{m \in S} \sum_{n \in S} d(m, n)$$

where S is a subset of correlated events that maximizes the objective function, η is a trade-off normalization coefficient, and $d(m, n)$ is a distance of a shortest path through the event correlation graph from node m to n.

12. The system of claim 10, wherein the kill chain module is further configured to maximize the approximated objective function:

$$Q^{+}(S) = \eta \sum_{v \in S} w(v) - \frac{1}{2}\left(\sum_{m \in S} \sum_{n \in S} d(m,n) + \sum_{p \in V} \sum_{q \in V} d(p,q) \right)$$

where S is a subset of correlated events that maximizes the objective function, η is a trade-off normalization coefficient, and $d(m,n)$ is a distance of a shortest path through the event correlation graph from node m to n.

13. The system of claim 12, wherein the kill chain module is further configured to search for a value for the trade-off normalization coefficient η that maximizes the approximated objective function.

14. The system of claim 10, wherein the kill chain module is further configured to prune the generated kill chains, keeping a longest kill chain leading to a given event.

15. The system of claim 10, wherein the security module is further configured to automatically perform at least one security action selected from the group consisting of shutting down devices, stopping or restricting certain types of network communication, raising alerts to system administrators, and changing a security policy level.

16. The system of claim 10, further comprising an alert ranking module configured to rank the anomalous events according to a degree of suspicion.

17. The system of claim 16, wherein the alert ranking module is further configured to remove anomalous events below a threshold rank.

18. The system of claim 10, further comprising a visualization module configured to display the kill chains on a graphical user interface for review by a user.

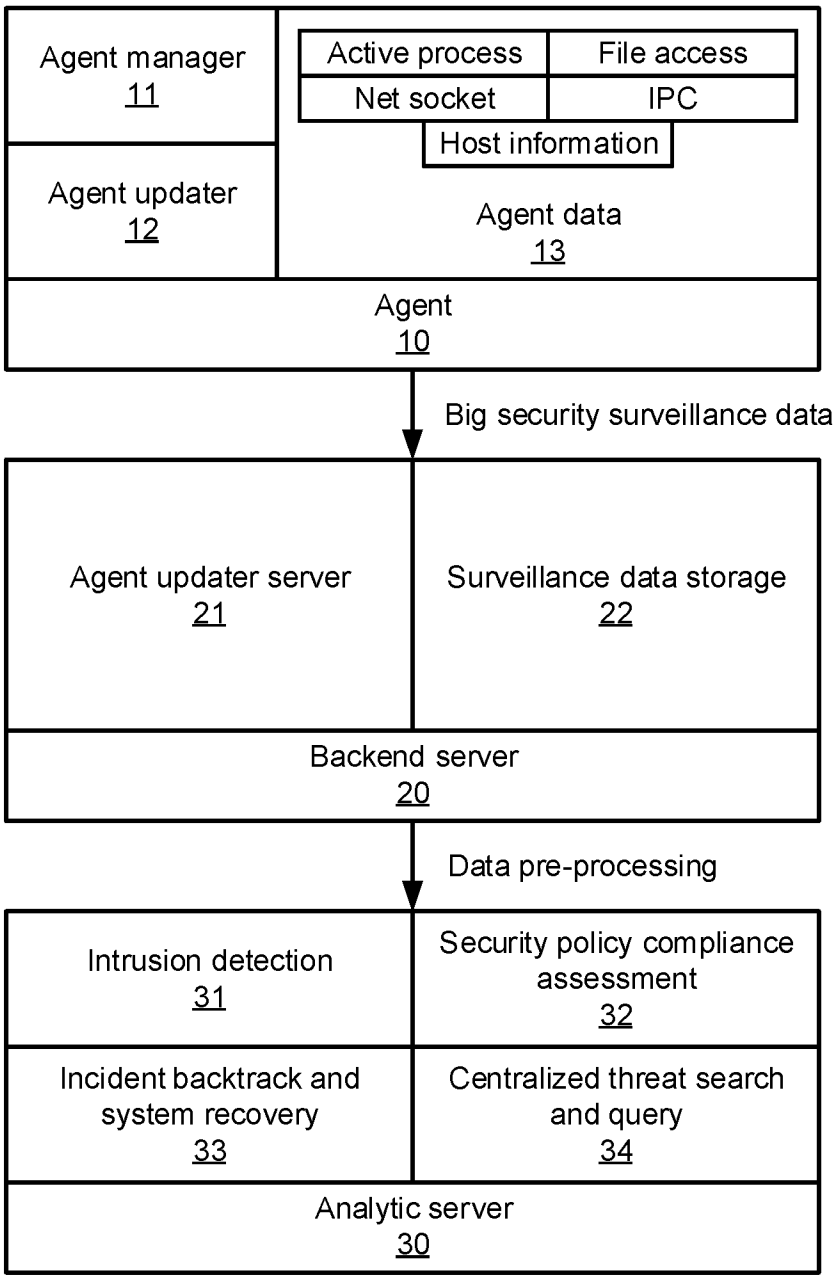
Agent manager
11
Agent updater
12
Active process
File access
Net socket
IPC
Host information
Agent data
13
Agent
10
Big security surveillance data
Agent updater server
21
Surveillance data storage
22
Backend server
20
Data pre-processing
Intrusion detection
31
Security policy compliance assessment
32
Incident backtrack and system recovery
33
Centralized threat search and query
34
Analytic server
30

FIG. 1

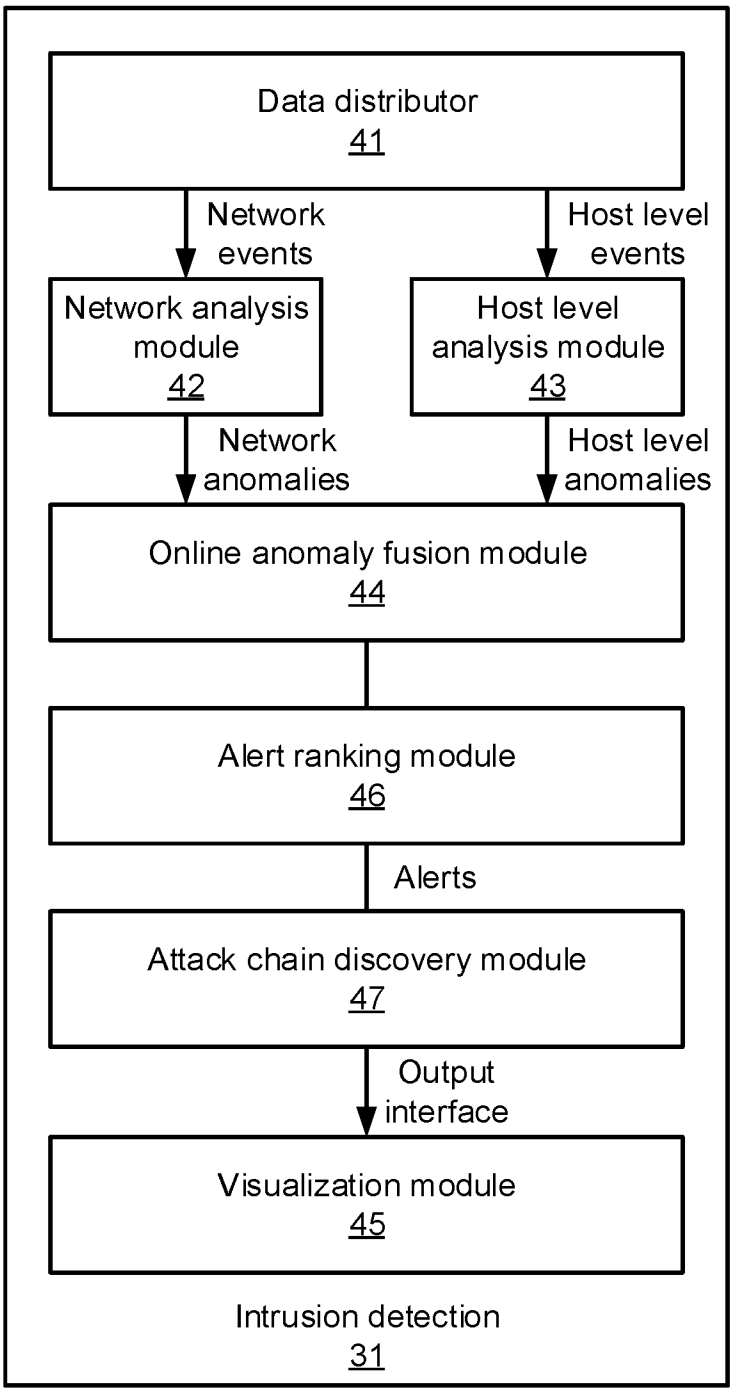
Data distributor
41
Network events
Host level events
Network analysis module
42
Host level analysis module
43
Network anomalies
Host level anomalies
Online anomaly fusion module
44
Alert ranking module
46
Alerts
Attack chain discovery module
47
Output interface
Visualization module
45
Intrusion detection
31

FIG. 2

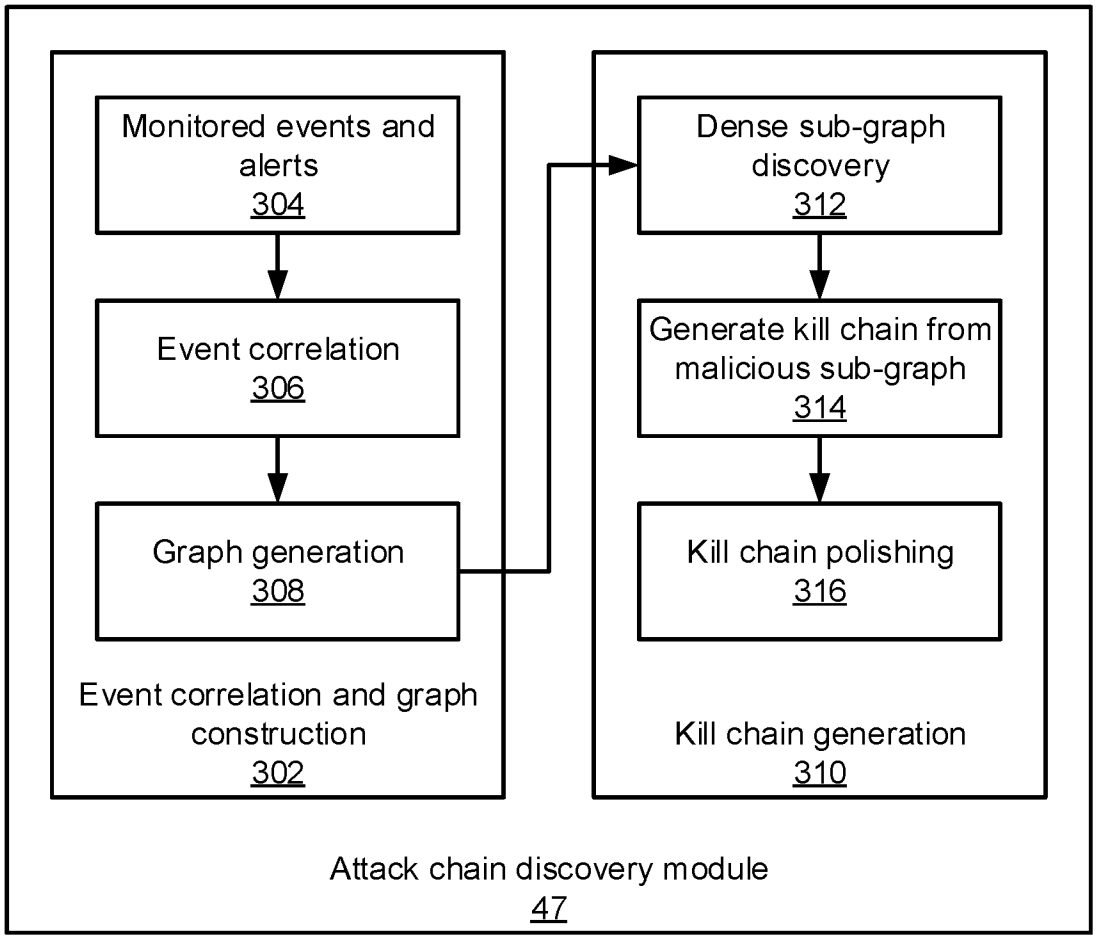
Monitored events and alerts
304
Event correlation
306
Graph generation
308
Event correlation and graph construction
302
Dense sub-graph discovery
312
Generate kill chain from malicious sub-graph
314
Kill chain polishing
316
Kill chain generation
310
Attack chain discovery module
47

FIG. 3

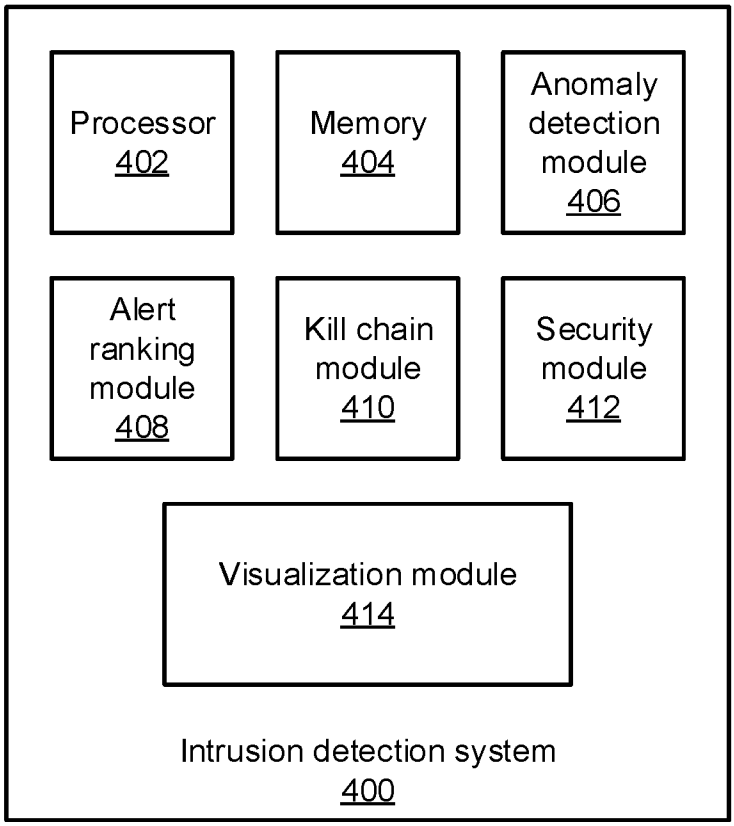
Processor
402
Memory
404
Anomaly detection module
406
Alert ranking module
408
Kill chain module
410
Security module
412
Visualization module
414
Intrusion detection system
400

FIG. 4

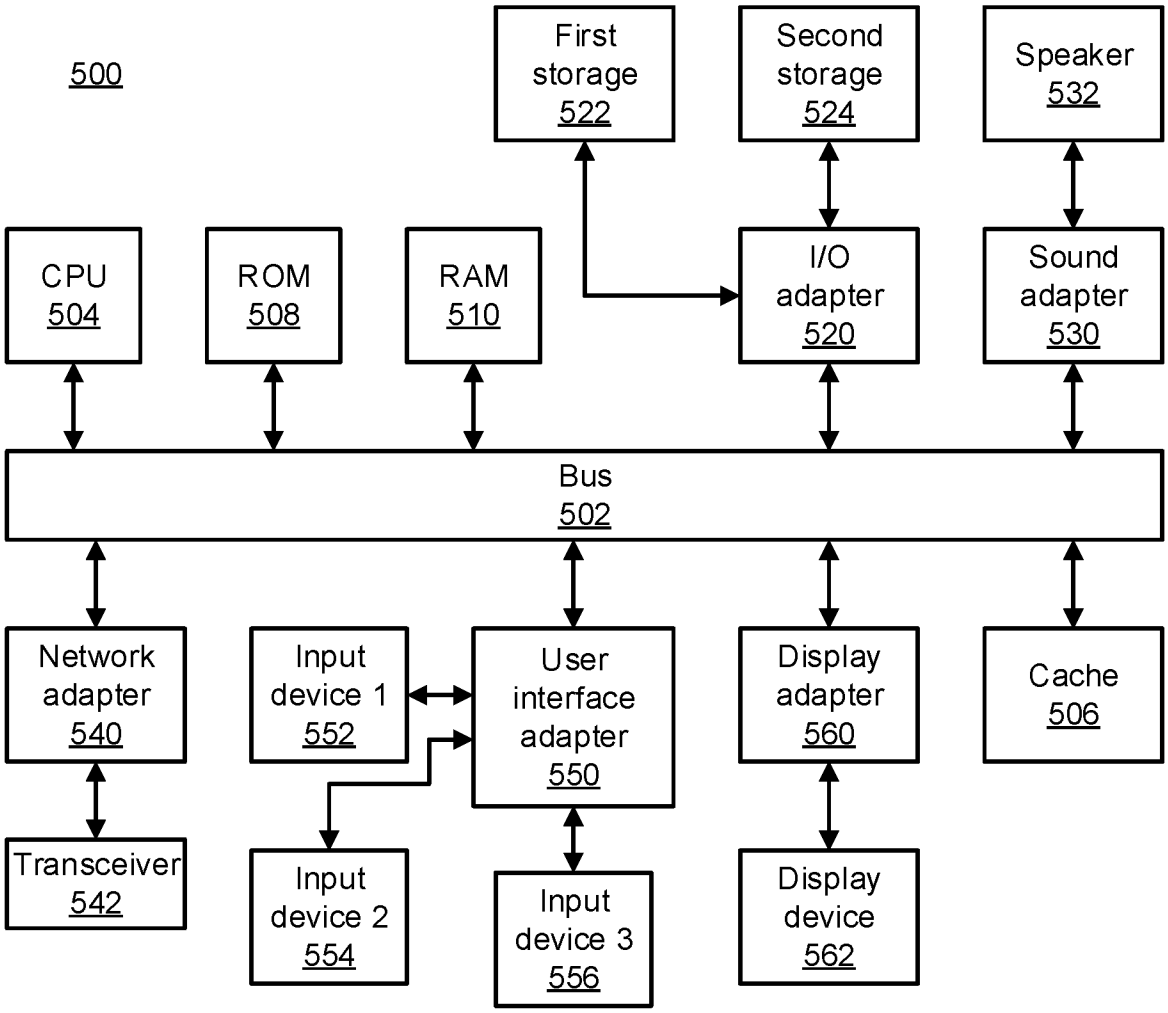
500
First storage 522
Second storage 524
Speaker 532
CPU 504
ROM 508
RAM 510
I/O adapter 520
Sound adapter 530
Bus 502
Network adapter 540
Input device 1 552
User interface adapter 550
Display adapter 560
Cache 506
Transceiver 542
Input device 2 554
Input device 3 556
Display device 562

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2017/055826

A. CLASSIFICATION OF SUBJECT MATTER

G06F 21/55(2013.01)i, G06F 21/60(2013.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F 21/55; H04L 12/24; G06F 17/30; H04L 29/06; G06F 21/60

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & keywords: network, security, graph, kill, chain

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 9363149 B1 (SPLUNK INC.) 07 June 2016 See column 12, lines 21, 22, column 22, lines 22-24, 38-40, column 23, lines 11-13, column 42, line 62 - column 43, line 2; claims 1, 6; and figure 33.	1-18
Y	LI QIANG et al., `A Reasoning Method of Cyber-Attack Attribution Based on Threat Intelligence`, World Academy of Science, Engineering and Technology. 31 May 2016, pp. 920-924. See pages 920-924; and figures 1-4.	1-18
A	US 9256739 B1 (SYMANTEC CORPORATION) 09 February 2016 See claims 1-15; and figures 3-10.	1-18
A	US 2015-0047026 A1 (LOS ALAMOS NATIONAL SECURITY, LLC) 12 February 2015 See paragraphs [0030]-[0039]; claims 1-7; and figures 3-5.	1-18
A	US 2016-0226893 A1 (WIPRO LIMITED) 04 August 2016 See paragraphs [0065]-[0086]; claims 1-7; and figures 5-7.	1-18

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
22 January 2018 (22.01.2018)	**22 January 2018 (22.01.2018)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea
Facsimile No. +82-42-481-8578

Authorized officer
CHIN, Sang Bum
Telephone No. +82-42-481-8398

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2017/055826

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 9363149 B1	07/06/2016	US 2017-0034196 A1	02/02/2017
US 9256739 B1	09/02/2016	None	
US 2015-0047026 A1	12/02/2015	AU 2013-272211 A1	02/10/2014
		AU 2013-272211 B2	24/11/2016
		AU 2013-272215 A1	09/10/2014
		AU 2013-272215 B2	12/10/2017
		AU 2016-234999 A1	20/10/2016
		AU 2017-200969 A1	02/03/2017
		CA 2868054 A1	12/12/2013
		CA 2868076 A1	12/12/2013
		CN 103764314 A	30/04/2014
		CN 103764314 B	09/09/2015
		CN 104303152 A	21/01/2015
		CN 104303152 B	13/06/2017
		CN 104303153 A	21/01/2015
		CN 104303153 B	13/06/2017
		CN 105033201 A	11/11/2015
		CN 105033201 B	05/04/2017
		EP 2741873 A1	18/06/2014
		EP 2741873 B1	20/01/2016
		EP 2828752 A2	28/01/2015
		EP 2828753 A2	28/01/2015
		JP 2014-530763 A	20/11/2014
		JP 2015-511101 A	13/04/2015
		JP 2015-514356 A	18/05/2015
		JP 2017-143578 A	17/08/2017
		JP 5793250 B2	14/10/2015
		JP 6139656 B2	31/05/2017
		JP 6148323 B2	14/06/2017
		KR 10-1440760 B1	17/09/2014
		KR 10-2014-0037975 A	27/03/2014
		US 2013-0248137 A1	26/09/2013
		US 2013-0254895 A1	26/09/2013
		US 2014-0068769 A1	06/03/2014
		US 2014-0096929 A1	10/04/2014
		US 2015-0020199 A1	15/01/2015
		US 2015-0180889 A1	25/06/2015
		US 2016-0277433 A1	22/09/2016
		US 2017-0163668 A1	08/06/2017
		US 8662145 B2	04/03/2014
		US 8813826 B2	26/08/2014
		US 9038180 B2	19/05/2015
		US 9374380 B2	21/06/2016
		US 9560065 B2	31/01/2017
		US 9699206 B2	04/07/2017
		WO 2013-138925 A1	26/09/2013
		WO 2013-184206 A2	12/12/2013

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2017/055826

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		WO 2013-184206 A3	20/02/2014
		WO 2013-184211 A2	12/12/2013
		WO 2013-184211 A3	13/03/2014
US 2016-0226893 A1	04/08/2016	None	