1-1-2023

Cybersecurity knowledge graphs

Leslie Sikos
Edith Cowan University

10.1007/s10115-023-01860-3
Sikos, L. F. (2023). Cybersecurity knowledge graphs. Knowledge and Information Systems, 1-21. https://doi.org/10.1007/s10115-023-01860-3

Knowledge and Information Systems
https://doi.org/10.1007/s10115-023-01860-3

SURVEY PAPER

Cybersecurity knowledge graphs

Leslie F. Sikos[1]

Received: 1 November 2021 / Revised: 26 February 2023 / Accepted: 11 March 2023

Abstract
Cybersecurity knowledge graphs, which represent cyber-knowledge with a graph-based data model, provide holistic approaches for processing massive volumes of complex cybersecurity data derived from diverse sources. They can assist security analysts to obtain cyberthreat intelligence, achieve a high level of cyber-situational awareness, discover new cyber-knowledge, visualize networks, data flow, and attack paths, and understand data correlations by aggregating and fusing data. This paper reviews the most prominent graph-based data models used in this domain, along with knowledge organization systems that define concepts and properties utilized in formal cyber-knowledge representation for both background knowledge and specific expert knowledge about an actual system or attack. It is also discussed how cybersecurity knowledge graphs enable machine learning and facilitate automated reasoning over cyber-knowledge.

Keywords Cybersecurity knowledge graph · Cyber-knowledge · Cyber-situational awareness · Cyber-resilience · Attack graph

1 Introduction to cybersecurity knowledge graphs

Applying knowledge graphs in the cybersecurity domain can be used to organize, manage, and utilize massive volumes of information in cyberspace, such as via ontology-based knowledge representation, which can completely and accurately represent the complex knowledge of heterogeneous systems [69]. These are called *cybersecurity knowledge graphs* or *CKG*s for short.

Formal knowledge representation, a branch of artificial intelligence, can be used in cybersecurity to formally define concepts, properties, and the relationships between them, enabling automated software agents to categorize vulnerabilities, threats, and attacks; perform entity resolution; detect anomalies; and match attack patterns [49]. These might reveal data correlations even experienced analysts would overlook.

✉ Leslie F. Sikos
l.sikos@ecu.edu.au

[1] Security Research Institute, Edith Cowan University, 270 Joondalup Drive, Joondalup, WA 6027, Australia

There are many information security and network process features that need to be stored when working with cybersecurity knowledge graphs (usually directed, labeled graphs), and the semantics of the captured cybersecurity knowledge varies greatly depending on the graph data model used [52], typically one of the following:

- an *RDF*[1] *graph* G_R, which is a set of *RDF triples* (RDF statements) of the form $(s, p, o) \in (\mathbb{I} \cup \mathbb{B}) \times \mathbb{I} \times (\mathbb{I} \cup \mathbb{L} \cup \mathbb{B})$, where
  - $\mathbb{I}$ is a set of International Resource Identifiers (IRIs), i.e., sets of strings of Unicode characters of the form `scheme:[//[user:pwd@]host[:port][/]path[?query][#fragment]` or a valid subset of these (such as URLs);
  - $\mathbb{L}$ represent RDF literals, which are either
    - $\mathbb{L}_P$ are self-denoting plain literals of the form "$<$`string`$>$"($@<$`lang`$>$), where $<$`string`$>$ is a string and $<$`lang`$>$ is an optional language tag; or
    - typed literals $\mathbb{L}_T$ of the form "`<string>`"`^^`$<$`datatype`$>$, where $<$`datatype`$>$ is an IRI denoting a datatype according to a schema, such as the XML Schema, and $<$`string`$>$ is an element of the lexical space corresponding to the datatype; and
  - $\mathbb{B}$ is a set of blank nodes, i.e., unique but anonymous resources that are neither IRIs nor RDF literals;

  with $\mathbb{I}, \mathbb{L}, \mathbb{B}$ being pairwise disjoint infinite sets;
- a *labeled property graph* of the form $G_{LP} = (V, E, \iota, \lambda, \pi)$, where V is a finite set of graph vertices (nodes), E is a finite set of graph edges s. t. V and E are disjoint, $\iota : E \rightarrow (N \times N)$ is an incidence function that maps each edge in E into a pair of vertices in V, $\lambda : (V \cup E) \rightarrow L_S$ is a labeling function that associates an edge with a set of labels from L, and $\pi : (N \cup E) \times P \rightarrow V_S$ is a property assignment function that assigns a set of values from V to each property, the second and third of which are partial functions;
- a *hypergraph* of the form $G_H = (V, E)$ where V is a set of vertices and E is a set of hyperedges between the vertices, each of which is a set of vertices, i.e., $E \subseteq \left\{\{u, v, \ldots\} \in 2^V\right\}$; or
- a *multigraph* of the form $G_M = (V, E)$, where V is a set of vertices and E is a bag of edges.

There is an increasing number of graph databases supporting various graph data models [52]; some of the most prominent ones include the following:

[1] *Resource Description Framework,* https://www.w3.org/TR/rdf11-concepts/

- Semantic graph databases, including RDF triplestores and quadstores: *Allegrograph*,[2] *Amazon Neptune*,[3] *AnzoGraph DB*,[4] *CRAY Graph Engine*[5], *Dgraph*,[6] *GraphDB*,[7] *MarkLogic*,[8] *Openlink Virtuoso*,[9] *RDFox*,[10] *Stardog*[11]
- Property graph databases: *MEMGraph*,[12] *NebulaGraph*,[13] *Neo4j*,[14] *RedisGraph*,[15] *TigerGraph*,[16] *Trovares*[17]
- Multi-model graph databases: *AgensGraph*,[18] *ArangoDB*,[19] *DataStax*,[20] *FlockDB*,[21] *graphbase.ai*,[22] *HyperGraphDB*,[23] *JanusGraph*,[24] *Microsoft Azure Cosmos DB*,[25] *Oracle Spatial and Graph*,[26] *OrientDB*,[27] *SAP HANA*,[28] *Sparksee*,[29] *TerminusDB*,[30] *TypeDB*[31]

The various graph-based implementations come with different strengths and weaknesses [42]. For example, not all support n-ary relations, even though these can be powerful for modeling communication networks [23]. Data provenance, which can be utilized in cyber-situational awareness [55], cybersecurity decision support [14], anomaly detection [53], network forensics [48], etc., are not supported by all knowledge graphs either, although hybrid solutions exist. While the RDF data model, for example, does not have a built-in mechanism for capturing provenance, the Semantic Web research community introduced

[2] https://allegrograph.com

[3] https://aws.amazon.com/neptune/

[4] https://cambridgesemantics.com/anzograph/

[5] https://support.hpe.com/hpesc/public/docDisplay?docId=a00113912en_us&page=About_the_Cray_Graph_Engine_CGE.html

[6] https://dgraph.io

[7] https://graphdb.ontotext.com

[8] https://www.marklogic.com

[9] https://virtuoso.openlinksw.com

[10] https://www.oxfordsemantic.tech/product

[11] https://www.stardog.com

[12] https://memgraph.com

[13] https://nebula-graph.io

[14] https://neo4j.com

[15] https://oss.redis.com/redisgraph/

[16] https://www.tigergraph.com

[17] https://www.trovares.com

[18] https://bitnine.net/agensgraph/

[19] https://www.arangodb.com

[20] https://www.datastax.com

[21] https://github.com/twitter-archive/flockdb

[22] https://graphbase.ai

[23] http://www.hypergraphdb.org

[24] https://janusgraph.org

[25] https://azure.microsoft.com/en-us/services/cosmos-db/

[26] https://www.oracle.com/database/technologies/spatialandgraph.html

[27] https://orientdb.org

[28] https://www.sap.com/products/hana.html

[29] https://www.sparsity-technologies.com

[30] https://terminusdb.com

[31] https://github.com/vaticle/typedb

advanced formalisms that extend the standard RDF data model for this purpose [51]. The number of, and the timespan of introduction of, these approaches indicate the importance of justifying the utilization of a particular graph data model over others.

2 Knowledge graph-based network, CTI, and cyber-physical system models

Cybersecurity knowledge graphs can be formally written as $:KG_n \{:s_k :p_k :o_k\ t\}$, where *KG* is a named graph representing a data source (e.g., traceroute, a routing message (BGP update message or OSPF LSA)), a router configuration file, a cybersecurity dataset (such as CAIDA), a server log (AWS CloudWatch log, AWS S3 log, Apache web server log, etc.) or a system log (Windows event log, Linux `auditd` daemon log, etc.), or a packet capture), n is the data source identifier, with $c \in \mathbb{Z}^+$ and $n \leqslant i$; s_{ki} is a knowledge statement's subject representing a network concept; p_{ki} is a knowledge statement's predicate, which is either a cybersecurity term (such as from an ontology like CNTFO) or the `rdf:type` predicate (expressing an "is a" relationship); and o_{ki} is a knowledge statement's object; t is the termination of statement symbol, i.e., a semicolon if another RDF statement follows, otherwise a full stop [54].

When modeling communication networks or cyber-physical systems with knowledge organization systems, the following main scenarios can be differentiated:

Type I a graph of a knowledge base represents a network infrastructure, and depending on the granularity, the nodes represent either:

- simulated or real-world network infrastructure and network device entities and their properties, and the arcs are the physical and logical links between them [55];
- autonomous systems (ASes) and their properties, and the arcs show how they are connected to each other [54]; or
- network information flow, and the arcs represent routing [25]; or
- a cyberattack graph, where the arcs are attack paths [69].

Type II a graph of a knowledge base represents cyberthreat intelligence covering system information, system parameters, cyberthreat data, and user or malware behavior data [39];

Type III a graph represents a controlled vocabulary or an ontology:

- the nodes are cybersecurity concepts and properties, and the arcs are correlations between them [50];
- the nodes are network device types and their properties, the arcs are connections between them;
- the nodes are vulnerabilities and the arcs define properties, such as vulnerability scoring, weaknesses, and platforms [26].

Type IV there are multiple, uniquely identified graphs (such as named graphs) that are connected to each other, each of which capture data from a different data source for data amalgamation and dimensionality reduction [53].

OWL[32] ontologies provide conceptual modeling of concepts and properties for arbitrary knowledge domains, including cybersecurity, cyber-situational awareness [57], and

[32] Web Ontology Language, https://www.w3.org/TR/owl-overview/

cyberthreat intelligence, in which they can facilitate partial automation for tasks that would otherwise have to be manually conducted or would be performed using multiple software tools and would rely on human supervision [47]. For example, digital forensic investigations can be partially automated subject to adequately captured forensic investigation knowledge and associated semantics, assisting timeline creation and event reconstruction [16].

Entities (such as specific malware) derived from multiple sources, such as multiple after action reports of attacks, if identical, can be matched and defined using `owl:sameAs` in a fused cybersecurity knowledge graph, thereby providing all the available information about the entity, plus naturally merging seemingly unrelated CKGs [40].

3 Knowledge graph-based KOSes for cybersecurity applications

Knowledge organization systems (KOSes), such as taxonomies, thesauri, controlled vocabularies, datasets, and ontologies, can be utilized for the automation of data processing for CTI, keeping CTI on track, turning CTI into action, performing adaptive threat-based adversary emulation, threat-based purple teaming, security tool evaluation, and post-exploit threat modeling.

The machine-processability and machine-interpretability of cybersecurity and CTI KOSes depend on the underlying data model, the used data structure, and the level of abstraction. For example, a *matrix*, or a *circular dendrogram* based on the structure represented by a matrix, can represent data sources, offensive and defensive techniques and tactics, and the properties can represent permissions. While such representations are not mapped directly to knowledge graphs, there is a clear link between them. For example, the *MITRE ATT&CK*®[33] framework, which constitutes an industry standard knowledge base of adversary tactics and techniques based on real-world observations, is typically represented as a matrix by default; its concepts and relationships can also be represented as a graph.

Another industry standard, *STIX™ (Structured Threat Information Expression)*, is a language and serialization format that can be used in ontological modeling of cybersecurity knowledge graphs [29].

The *Situation and Threat Understanding by Correlating Contextual Observations (STUCCO)* ontology,[34] written in *JSON Schema*[35] and as such, compatible with the *GraphSON* format, defines the concepts user, account, host, software, vulnerability, malware, flow, attack, attacker, host, address, IP, address range, port, service, and domain name, and 115 properties to characterize these and their relationships [18]. The optimality of the granularity of this ontology can be disputed, considering address range to be ideally defined as a datatype property restriction instead of a concept, the actual addresses being property values rather than entities. Nevertheless, the ontology can be used, for example, in incident response tasks, such as searching through flow and IDS records by address for a particular time slot, and check whether remote addresses are on blacklists; or attempting to identify malware based on network traffic logs and system changes.

The *Cybersecurity Operations Centre Ontology for Analysis (CoCoa)* is a NIST-aligned ontology that covers cyberthreat intelligence and information sources, including events and logs; network information; unstructured, semistructured, and structured feeds; and threat intelligence [37]. Using CoCoa, cyber-incidents can be represented in knowledge graphs with

[33] https://attack.mitre.org

[34] https://stucco.github.io

[35] https://json-schema.org

concepts such as cyber-incident, collector, vulnerability, threat, and network infrastructure, which map relationships and connections of incidents for monitoring and visualization.

The cybersecurity terminology captured by KOSes might be linked even between semistructured and structured systems. For example, a core node for linking, and mediating between, cybersecurity *Linked Open Data (LOD)* KOSes in the *LOD Cloud*[36] is the *Unified Cybersecurity Ontology (UCO)*. It defines typed connections between STIX, *CAPEC*,[37] *MAEC*,[38] *CWE*,[39] *CVE*,[40] *CVSS*,[41] *Cybox*,[42] *CPE*,[43] *OpenIOC*,[44] *STUCCO*, *Mobile Access Control*, and the *Cloud User Ontology* terms [61]. *VulOntology*[45] is a vulnerability ontology that defines the relationship between vulnerabilities and applications, platforms, and weaknesses [44]. Similarly, the *SEPSES Cybersecurity Knowledge Graph (CSKG)* links and integrates vulnerabilities, weaknesses, and attack patterns from a wide range of data sources, including CAPEC, CPE, CVE, CVSS, and CWE [26]. Alignment with these de facto standard data sources is vital, as seen with mainstream cybersecurity knowledge graphs (see Table 1).

MITRE's *CyGraph*[46] can be used for both proactive and reactive cyber-resilience measures. It employs a property graph formalism and provides uniform representation of network infrastructures, cyberthreats, mission dependencies, and overall security posture [36]. CyGraph's knowledge base not only holds information to construct attack graphs and mission dependency models, but also includes potential attack-pattern relationships that provide insight to correlations between known vulnerabilities and threat indicators.

By combining a cybersecurity ontology covering network attack types and characteristics with the implementation of a cybersecurity knowledge base from knowledge acquisition, knowledge fusion/extraction, knowledge storage, knowledge inference, and knowledge update, real-time solutions can be realized [28]. The ontological representation of, and formal definition of the relationships between, devices, features, and attacks can be utilized when converting heterogeneous network data to RDF triples. These rely on extracting reliable features from industry standard file formats to be converted, for example, from PCAP packet capture files with tools such as *CICFlowMeter*,[47] ultimately resulting in structured data (derived from unstructured or semistructured data).

The *Knowledge Graph of Threat Actor (TAGraph)* is a framework consisting of a threat actor ontology and a named entity recognition system to be used for automatically extracting cybersecurity-related entities from webpages and generate a dataset and associated knowledge graph based on them [17]. This can be particularly useful if information about a threat actor is extracted from multiple sources and then subsequently fused and represented as a single knowledge graph.

36 https://lod-cloud.net

37 Common Attack Pattern Enumeration and Classification, https://capec.mitre.org

38 Malware Attribute Enumeration and Characterization, https://maecproject.github.io

39 Common Weakness Enumeration, https://cwe.mitre.org

40 Common Vulnerabilities and Exposures, https://cve.mitre.org

41 Common Vulnerability Scoring System, https://www.first.org/cvss/

42 Cyber Observable eXpression, https://cyboxproject.github.io

43 Common Platform Enumeration, https://nvd.nist.gov/products/cpe

44 Open Indicators of Compromise, https://github.com/mandiant/OpenIOC_1.1

45 https://github.com/Brian-hku/VulKG

46 https://www.mitre.org/research/technology-transfer/technology-licensing/cygraph

47 https://github.com/CanadianInstituteForCybersecurity/CICFlowMeter

Table 1 Comparison of prominent cybersecurity knowledge graphs

KG	Purpose	Data model	Implementation	Query language/software	Standard alignment
CSKG[a]	General-purpose cyber-knowledge graph	RDF	OWL	SPARQL	CAPEC, CVE, CWE
CWE-KG[b]	Twitter data analysis	Relational data model	CSV	Log Parser or similar	CAPEC, CWE
Live Cybersecurity KG[c]	Security infrastructure representation	RDF	GraphDB	SPARQL	N/A
Open-CyKG [46]	Open Cyberthreat Intelligence	Uncanonicalized KG	Neo4j	Cypher	N/A
MalKG[d]	Malware threat intelligence	Tree	JSON	jsonQuery or similar	CVE
SEPSES CKB[e]	General-purpose cyber-knowledge graph	RDF	OWL	SPARQL	CAPEC, CPE, CVE, CVSS, CWE
UCO[f]	Cybersecurity standard alignment	RDF	OWL	SPARQL	STIX, CAPEC, MAEC, CWE, CVE, CVSS, Cybox, CPE, OpenIOC
Vulnerability KG[g]	Vulnerability visualization	Labeled property graph	Neo4j	Cypher	CVE, CWE

[a] https://github.com/HoloLen/CyberSecurity_Knowledge_graph
[b] https://github.com/nansunsun/CWE-Knowledge-Graph-Based-Twitter-Data-Analysis-for-Cybersecurity
[c] https://www.ontotext.com/knowledgehub/case-studies/ontotext-graphdb-powers-global-cybersecurity-company-infrastructure/
[d] https://github.com/liujie40/MalKG-1
[e] https://sepses.ifs.tuwien.ac.at/dumps/version/102019/
[f] https://unifiedcyberontology.org
[g] https://cinnqi.github.io/Neo4j-D3-VKG/

Table 2 Primary application areas of mainstream cybersecurity KOSes

KOS	Main application area			
	CI	Cyber-resilience	IR	Digital forensics
CASE [7]	–	–	–	+
CNTFO [54]	+	–	–	–
CoCoa [37]	–	–	+	–
CyGraph [36]	+	+	–	–
MITRE D3FEND [24]	+	+	+	+
ParFor [63]	–	–	–	+
STIX [29]	+	–	–	–
STUCCO [18]	+	–	–	–
TAGraph [17]	+	–	+	–
UCO [61]	+	–	+	–

MITRE D3FEND™[48] is a knowledge graph of cybersecurity countermeasures. It categorizes concepts in five categories: harden, detect, isolate, deceive, and evict. Within each subcategory, specific techniques are defined and described. These form a matrix, which is complemented by the *Digital Artifact Ontology*[49] to represent the concepts of digital artifact and related file types, network traffic types, and software types. It captures the semantics of the concepts that link processes to digital artifacts (such as executable binary file, process code segment, user account), and concepts of MITRE's Offensive Model that modify process code segments (exploitation of remote services, exploitation for privilege escalation) as well as the process code segment verification of MITRE's defensive model, covering five tactics to classify defensive methods (harden, detect, isolate, deceive, and evict).

Jia et al. [21] proposed a framework to generate a cybersecurity knowledge base by utilizing an ontology based on vulnerabilities and by using the *Stanford Named Entity Recognizer (NER)*[50] and conditional random fields (CRFs) to extract cybersecurity entities from unstructured data. These are expressed in RDF, similar to the structured data (which is directly written in RDF). This knowledge base consists of quintuples, capturing concept, instance, relation, properties, and rule for each statement. Concepts such as OS, vulnerability, and consequences are instantiated and characterized to capture the operating systems with version number, the vulnerabilities with the associated threat type and threat level, and cyberattack types.

Table 2 summarizes popular cybersecurity knowledge organization systems by their main application areas: cyberthreat intelligence (CI), cyber-resilience, incident response (IR), and digital forensics.

Note that domain ontologies are typically too specific to be used across multiple cybersecurity fields, while upper ontologies, particularly those aligned with multiple industry standards, can be applied in many.

48 https://d3fend.mitre.org

49 https://d3fend.mitre.org/dao

50 https://nlp.stanford.edu/software/CRF-NER.html

4 Automated reasoning over cybersecurity knowledge graphs

One of the key benefits of utilizing machine-readable, and whenever available, machine-interpretable, knowledge graphs in cybersecurity is that they facilitate automated reasoning so that new facts can be inferred from explicit statements (existing data), and dynamically updated information provided on the latest vulnerabilities and threats [68].

By using RDF quadruples to model communication networks, cyber-situational awareness can be improved via automated reasoning over implicit knowledge. For example, based on CAIDA open data, BGP update messages, OSPF LSAs, and router configuration files, explicit statements can automatically be generated, such as a "peers with" relationship between two autonomous systems, or a "connected to" relationship between a network and a network interface [54].

By modeling attackers' background knowledge in a knowledge graph, the sensitive information not disclosed yet can be inferred from implicit knowledge can be approximated [43]. The four core cases are 1) an attacker can infer the relationship of two persons based on shared attributes, 2) an attacker can infer whether a user has a specific attribute based on a relationships of the person has that attribute, 3) an attacker can infer the relationship of two persons who are both connected to a third person, and 4) an attacker can infer a property of a person based on the dependency of the property on another property.

For big data analysis for cyber-situational awareness, semantic data mining can be used; however, achieving interoperability and generalization can be difficult, particularly for unordered rules. The *Subsumption Reasoning for Rule Deduction (SRDD)* method has been proposed to address this, whereby redundant semantic rules can be discovered based on the rule subsumption decided by knowledge graph reasoning.

Denoising entity extraction from cyber-knowledge graphs can assist overwhelmed security analysts to make sense of threat intelligence data [10].

Logs of cybersecurity incidents can be captured efficiently in RDF-based provenance graphs, which can be used to generate provenance graphs with alerts, and eventually conceptualized attack graphs [27]. This allows combining and integrating a range of techniques for *cyberthreat detection and alert generation*. *Attack graphs can be constructed*—and hence attacks reconstructed—by backward–forward chaining and graph querying. Contextual cyber-knowledge graphs provide provenance data for alerting, which in turn can be utilized for identifying a potential root cause of an attack, whereby the alert score is increased for each preceding alert in the path.

Attack graphs can be combined with a Bayesian network to effectively *determine the probability of attack paths*. By writing reasoning rules for vulnerabilities (that are represented as graph nodes), automated reasoning can be performed to *infer that a vulnerability can cause a particular consequence, two different vulnerability nodes have similar attributes, or that two vulnerabilities can be continuously exploited* [8].

Reasoning rules for cyberthreat information can be used to provide specific defense strategies, whereby the relationship between vulnerabilities, weaknesses, platforms, and attack patterns can be used to automatically infer a range of useful threat information [70]. Examples for what reasoning can generate include a platform having one of the vulnerabilities also has the other, a platform may be attacked using a particular attack technique or a counterattack technique for a malicious action. Moreover, *an attack pattern can be linked to a platform based on the exploit and the known CWE weakness*, and *actions can be recommended for reducing an attack risk*.

Ontology-based representation of packet analysis semantics can facilitate automated reasoning for network monitoring applications and honeypots [56]. Reasoning over ontologies describing BGP update messages can facilitate the automation of network analysis to *detect BGP hijacking* [65], such as to be used for man-in-the-middle (MITM) attacks by diverting traffic to the attacker, or for obtaining IP addresses for spamming or distributed denial-of-service (DDoS) attacks.

Reasoning for logical subsumptions between concepts and roles can ultimately be used for rule reduction after knowledge graph mining for cyber-situational awareness analysis, such as to determine which attack techniques are easier for adversaries and which ones are detected by common defense technologies [29].

Based on semantic modeling and a reasoning engine considering asset categories, relationships and input/output incident types, the impacts of complex cyber-physical attacks against critical infrastructure can be propagated and the mitigation of potential harming effects assisted [45].

5 Utilizing machine learning on cybersecurity knowledge graphs

The categorization of algorithms for graph-based anomaly detection depends on the approach being unsupervised or semi-supervised, and whether the graph is static or dynamic, and attributed (node-/edge-labeled) or plain (unlabeled) [3]. These will determine whether the detection is structure-based, community- or clustering-based, relational learning-based, decomposition-based, or window-based. This can be complemented by graph-based anomaly description, either in the form of interpretation-friendly graph anomaly detection or interactive graph querying and sense-making. In dynamic graphs, anomalies are highly flexible, and typically, there is insufficient labeled data; learning anomaly patterns can be more efficient if all hints of structural, content, and temporal features are taken into account, rather than using heuristic rules over partial features [72].

Frequent sequential patterns can be found in streaming data by considering temporal information, such as via using the *PrefixSpan* algorithm [21].

Combining analyst intuition with machine learning, as seen with the system AI^2, is capable of learning to defend against previously unseen attacks [64]. Unsupervised learning can learn a model to identify anomalies, such as extreme or rare cyber-events, which can be ranked based on a predefined metric and forwarded to human analysts, who can add labels to be used by supervised learning. The resulting model can predict from features potential attacks in the near future.

When cyber-knowledge graphs are used to represent cyber-knowledge, whether entities derived from logs or cyberthreat intelligence (which MAC address requested access to which IP or domain, an IP is in which IP address space assigned to which autonomous system, etc.), cyberthreat detection in SOC/SIEM environments can be formulated as a large-scale graph inference problem [33]. *Graph netural networks (GNNs)* can be used for graph-based network intrusion detection, capturing both edge features and a network's topological information—as seen in the example of *Graph SAmple and aggreGatE (GraphSAGE)* detecting malicious information flow in IoT networks [30]. However, graph-based inference algorithms, such as belief propagation, random walk with restart, influence and diffusion, SimRank, graph-based semi-supervised learning, and GraphSAGE, have various limitations when used for threat detection, and this is why MalRank has been introduced, with the purpose of finding a maliciousness score of a node, given a directed weighted graph, in which the vertices

are collections of entities, such as domains and IPs, and the edges are sets of relationships between these; and an a priori label and confidence over the set of vertices.

By taking an entity relationship set and asserting it in a triple-based cybersecurity knowledge graph, substantial information about various cybersecurity entities can be accessed, such as via *SPARQL*[51] queries, while the relationships between entity pairs can be predicted using deep learning [38]. For applications where the navigational programming paradigm based on graph traversal is preferred over the SPARQL query paradigm based on graph patterns, the *RDFFrames* framework offers a suitable interface [32].

Prior expert security knowledge and open threat data represented in cybersecurity knowledge graphs can be used to guide reinforcement learning to effectively identify ways to detect malware so that they can be deleted, thereby mitigating cyberattacks [41]. Such an approach can mimic how SOC analysts process data based on their background knowledge. In fact, the knowledge stored in cybersecurity knowledge graphs may provide multiple mitigation strategies when a malware is being executed. The malware features can also be used to identify the malware family to which a previously unknown malware sample belongs.

In cyber-knowledge graphs, which are inherently sparse, highly incomplete (the *open-world assumption* applies), and noisy, statistical relational learning can be applied to predict missing links and identify relationships between nodes [21]. Relational learning on cybersecurity knowledge graphs can be applied to information security monitoring and intrusion detection, whereby the context provided by rich sets of entity and relationship types can be utilized. Garrido et al. applied machine learning on cybersecurity knowledge graphs to detect unexpected activities in industrial automation systems. By training a generative graph embedding algorithm on a graph built from a training dataset, a baseline normal behavior and operating conditions of an industrial system can be learned, and subsequently, link prediction can be performed unsupervised to rank the likelihood of triple statements resulting from events observed at test time and determine whether there is a substantial deviation from the baseline [15]. This results in a qualitative evaluation of the predictions, with not only anomalies detected, but also with the option to assign severity levels manually based on available contextual information.

The *K2* machine learning algorithm has been introduced to classify cyberattacks, where dependency links between graph nodes are built to be tested against the preceding nodes in order, and a new edge is added to the graph if it improves the Bayesian measurement [1].

6 Visualization of cyber-knowledge with knowledge graphs

A serious limitation of traditional information security tools is that too much information might be displayed (from IDSes, vulnerability scanners, firewall managers, SIEM tools, and security intelligence) with too little context [35]. Cybersecurity knowledge graphs provide a viable option to represent and visualize security information, allowing timely cyber-incident detection and response, which are becoming more and more demanding for security analysts. Some examples of cybersecurity knowledge graph visualizations include, but are not limited to enabling security analysts explore aggregated log data via relationships without complex query languages (see Fig. 1), explore vulnerabilities and attack patterns with contextual information (see Fig. 2), visualizing intrusion detection with packet capture-based logs of interacting IP addresses (see Fig. 3), and visualizing an attack tree with attack goals and subgoals, and the corresponding attack medium (see Fig. 4).

[51] SPARQL Protocol and RDF Query Language, https://www.w3.org/TR/sparql11-query/

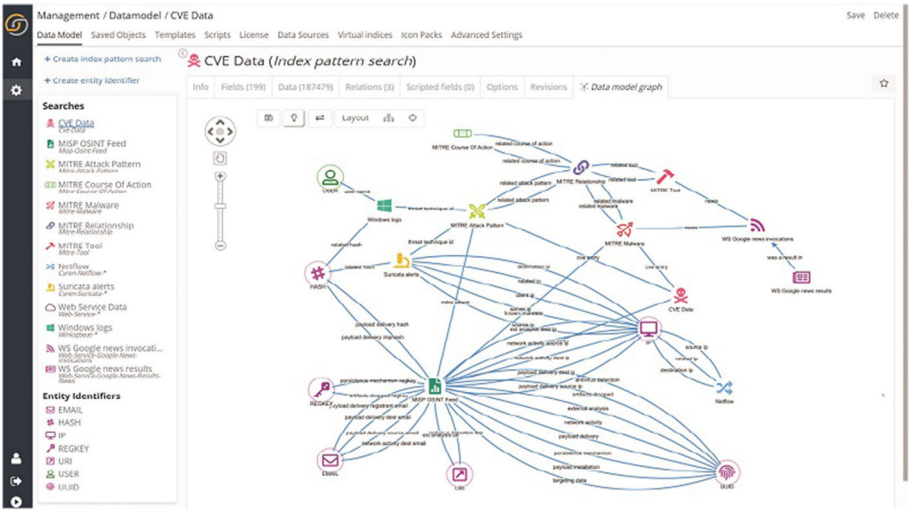

Fig. 1 MITRE ATT&CK patterns and courses of action with CVE alignment (Siren) [59]

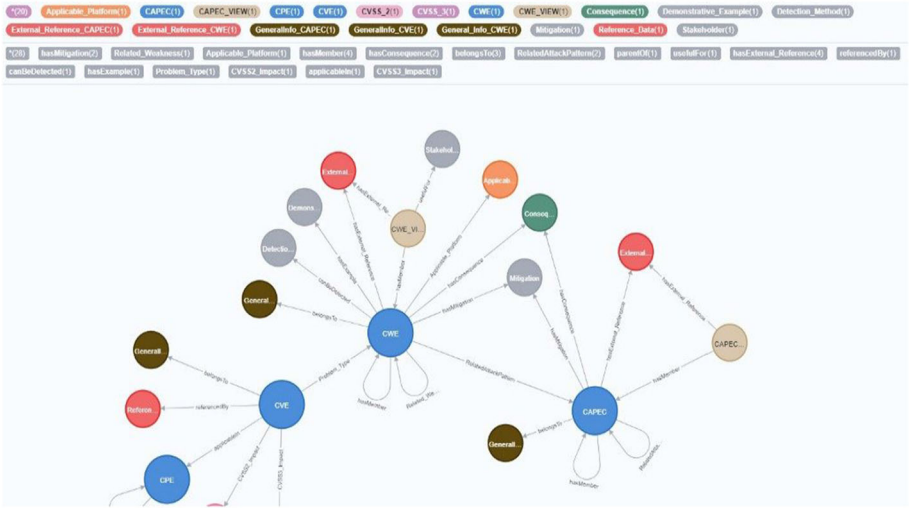

Fig. 2 CVE, CWE, CAPEC, and CPE recordes in a connected graph (GraphKer) [4]

Knowledge-driven systems can provide assistance to analysts via partial automation of analytics and visualization of complex cybersecurity data. For example, *VisAlert*, proposed as a radial display for network alert monitoring and visual correlation analysis, was designed to display a local network topology graph, surrounded by alert types, with the aim of enabling Tier 1 analysts to detect signs of potential anomalies [13]. Spam campaigns can also be efficiently visualized using graph representations, allowing the in-depth analysis of the underlying botnet ecosystem [62]. By employing hypergraphs, multi-attribute associations of the patterns extracted from large cybersecurity datasets can be displayed [22]. This is suitable for timeline creation during network monitoring and forensic analysis, and for identifying unknown attack patterns.

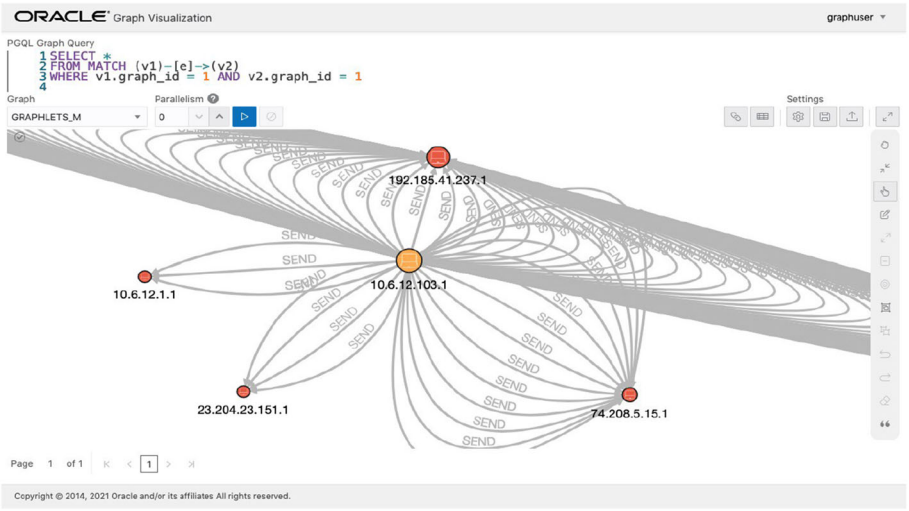

Fig. 3 Visualizing intrusion detection for Oracle Machine Learning (OPG4Py) [67]

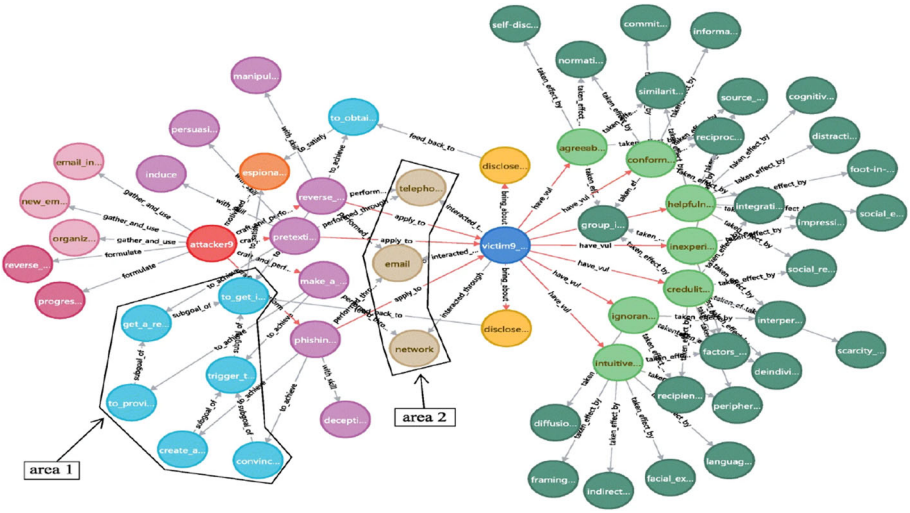

Fig. 4 Visualizing a reverse social engineering attack with an attack graph [66]

KAVAS (Knowledge-Assisted Visual Analytics for STIX), a graph-based visual analytics platform, can be used for analyzing and enriching cyberthreat intelligence data [5]. It utilizes both operational knowledge and domain knowledge of security experts for filtering, mapping, and rendering CTI in the visualization phase. The STIX alignment follows SDOs and SROs represented as links in node-link diagrams, but extends these by also displaying important relationships embedded into SDOs referencing other objects. The implementation of KAVAS also highlighted some limitations of STIX, such as the absence of a top-level element for representing specific organizational assets (e.g., IT systems affected by attacks).

Cyber-alerts can be investigated efficiently using graph-based analytics and narrative visualization [2]. By capturing complex relationships between alerts and background knowledge in knowledge graphs, security analysts can be assisted with context for interpreting cyberthreats, performing risk management, and achieving a high level of cybersituational awareness.

While link graphs have many benefits in data visualization in the cybersecurity domain, the size of a graph can have a reverse effect on analysis efficiency and might even jeopardize usability altogether. If the number of nodes and edges is too high, there are too many elements to show, resulting in unreadable and/or confusing representations. Moreover, showing additional dimensions, such as alert type or severity, might not be practical [9].

Alternate representations, such as 3D graph visualizations, can somewhat overcome these limitations. For example, *DAEDALUS-VIZ* can generate real-time 3D graphs for Darknet monitoring-based alerts displaying spheres and tori [19]. It provides the option to filter by network, protocol, port, sensor ID, alert, and filter status.

7 Data aggregation and data fusion using cybersecurity knowledge graphs

Cybersecurity knowledge graphs have a huge potential when it comes to aggregating and fusing data, which is typical in SOC and SIEM monitoring dashboards, for example. Potential data sources include, but are not limited to, network topology, IDS, firewall rules, firewall manager, routing messages, vulnerability scanner, SIEM software, security intelligence, and publicly available datasets, such as from the LOD Cloud. Aggregating data from diverse sources is particularly useful when working on the zero-day mitigation of critical vulnerabilities being exploited in the wild, such as the Apache Log4j vulnerability CVE-2021-44228 at the time of writing, which results in remote code execution. Figure 5shows an example for representing this vulnerability with data from the developer and an affected software vendor, cyberthreat intelligence, and publicly available datasets, specifically, Apache, Cisco, MITRE, NIST, and the LOD Cloud. Using an RDF-powered knowledge graph in this instance, the data sources could be represented as identifiers of named graphs, and statements can be written accordingly, e.g.,

```
:MITRE {
     :CVE-2021-44228 a :Vulnerability ;
        :accessComplexity "medium" ;
        :requiresAuthentication "false" .
}
:NIST {
     :CVE-2021-44228 :baseScore "10.0" ;
        :knownAffectedSoftwareConfiguration
        "cpe:2.3:a:apache:log4j:2.0:-:*:*:*:*:*:*"
        .
}
:CISCO {
     :CVE-2021-44228 :vulnerableProduct
        :CiscoWebexMeetingsServer ; :knownAffected
        :CVRFPID-277610 ; :baseScore "10.0" .
}
```

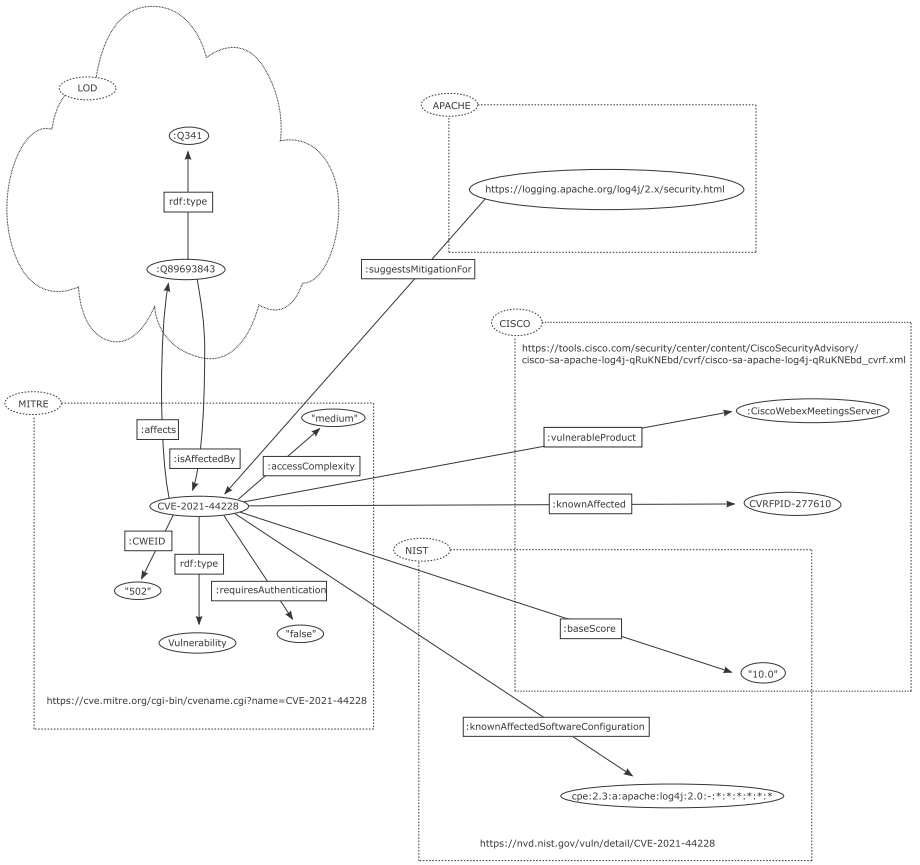

Fig. 5 Knowledge graph-based data aggregation for cyber-situational awareness and attack graph analysis

```
:APACHE :suggestsMitigationFor :CVE-2021-44228.
```

Note that the base score in this case has been confirmed by two independent sources, namely Cisco and NIST, and such matches indicate high likeliness of information correctness and trustworthiness.

The above representation also allows provenance or other metadata to be captured for each statement, making it possible to weight cyberthreat intelligence information for incomplete, non-matching, or contradictory statements typical to the cyberthreat domain.

Host-level process communication graphs are suitable for inferring network connection causations, which in turn can be aggregated into system-wide host-communication graphs. Data fusion on directed graphs, in which the set of edges represents the communication structure of data collection, transformation, and transmission agents, can be used to detect lateral movement between hosts [12].

By considering the distribution of graph edges and the maximum degree of occurrences, spoofing attacks, DoS attacks, fuzzy attacks, replay attacks, etc. can be identified, as seen in the example given by Islam et al. in controller area network (CAN) communication of self-driving cars [20]. Network information flow, when represented with graphs, can serve

for training and data evaluation for network intrusion detection systems (NIDS), whereby graph neural networks can be applied to detect intrusions using flow-based data [30].

Knowledge graph-based data aggregation and fusion can be well-utilized in IoT networks, such as by uniformly representing sensor data in medical smart home settings to facilitate automated reasoning over technology and software vulnerabilities [6]. This is useful for preventing cyberattacks targeting medical devices and sensors, and indicating the need for firmware and application updates.

By running federated queries on such a cyber-knowledge graph, entities having certain property values according to data derived from multiple data sources simultaneously can be found effectively. For example, the CVE of all the vulnerabilities that are associated with a vulnerable product (as described in one dataset) and a known affected software configuration at the same time (according to another dataset) can be identified, e.g.,

```
SELECT ?cve
FROM NAMED MITRE
FROM NAMED CISCO
FROM NAMED NIST
WHERE
{ ?v :vulnerableProduct :CiscoWebexMeetingsServer
    ;
     :knownAffectedSoftwareConfiguration
        "cpe:2.3:a:apache:log4j:2.0:-:*:*:*:*:*:*".
}
```

In turn, this can be utilized by semantic agents to infer, for example, whether a patch should be installed for a vulnerable product of an organization having a specific software configuration, which, when automated this way, can take some load off security professionals.

Vulnerability data captured in knowledge graphs can enable *CWE chain reasoning*, whereby the number of products having a particular weakness can be determined, and the knowledge graph is queried to calculate chain confidence, based on which a candidate can be selected [44]. Whether this is a possible CWE chain needs to be validated, such as via the CVE vulnerability description.

Depending on the knowledge represented, the output of such systems can be used for decision support, data analysis, task automation, and more. Such data-driven architectures can represent how network segmentation affects the placement and configuration of firewalls, and to find ways to prevent cyberattacks by pinpointing the most vulnerable services via examining firewall rules in context, in particular, the source and destination addresses. Using cybersecurity knowledge graphs, exposed vulnerabilities can be listed in order of frequency and represented before and after mitigation. Complex queries executed on a knowledge graph can be used to determine the relevance of a particular alert, such as from an intrusion detection system, by providing correlation data between a cyber-event, an exploit, and a vulnerability [31].

Graph-based IDSes (GrIDSes) are designed to detect large-scale automated attacks in communication networks, forming graphs from incident reports and network traffic logs [11]. They can aggregate cybersecurity graphs into simpler forms at higher hierarchical levels. Semantically enriched cyber-knowledge representation can be complemented by machine learning to help security analysts in collaborative frameworks utilizing data from host- and network-based sensors and security specialists alike, which is particularly useful in case of novel complex cyberattacks, such as ransomware attacks [34]. Knowledge graphs can help model cyberthreat and cyberattack trends, and understand new attack strategies ultimately

leading to new attack categories [60]. By using knowledge graphs to represent cyberthreat intelligence, malware behavior can be fused with cyber-knowledge [39]. Knowledge graphs capturing known security vulnerabilities of medical devices in hospitals can contribute to the protection of user data via augmenting data from device vendors, CISA *ICS-CERT*,[52] etc., with Linked Open Data (LOD) datasets such as *Wikidata*[53] and medical databased like FDA's *AccessGUDID*[54] [58]. Knowledge graphs can also be utilized in automated malicious repository detection [71]. A knowledge graph where nodes represent repositories and keywords, and the edges between the nodes capture whether a keyword occurs in a repository, can be used as the basis for repository representation learning using deep neural networks.

8 Conclusion

The complex correlations between cybersecurity data captured in a variety of data formats and derived from a diverse range of data sources can be efficiently modeled using knowledge graphs. The data model used determines the capabilities and limitations of a particular implementation, whether representing a computer network, interconnected devices, or cyberattack paths. The formal grounding of these graphs ensure clearly understandable computational properties and reasoning complexity for the represented background knowledge and captured expert knowledge. Cybersecurity knowledge graphs contribute to the standardization of terminology use in the cybersecurity and digital forensics domains, and the mainstream processing of security and security-related data that would otherwise be isolated and would have limited automated processing support due to proprietary data formats and content normally not accessible to software agents.

Cybersecurity knowledge graphs are suitable for network data aggregation, data integration, data fusion, data mapping, and knowledge discovery; they facilitate machine learning and can be used for efficient visualizations in ways not feasible with other technologies.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions

References

1. Abid A, Jemili F (2020) Intrusion detection based on graph oriented big data analytics. Procedia Comput Sci 176:572–581. https://doi.org/10.1016/j.procs.2020.08.059
2. AfzaliSeresht N, Miao Y, Liu Q et al (2020) Investigating cyber alerts with graph-based analytics and narrative visualization. In: Banissi E (ed) 24th International Conference on Information Visualisation. IEEE, pp 521–529. https://doi.org/10.1109/IV51561.2020.00090

[52] https://us-cert.cisa.gov/ics

[53] https://www.wikidata.org

[54] https://accessgudid.nlm.nih.gov

3. Akoglu L, Tong H, Koutra D (2015) Graph-based anomaly detection and description: a survey. Data Min Knowl Discov 29:626–688. https://doi.org/10.1007/s10618-014-0365-y
4. Berzovitis AM (2021) How to have a cybersecurity graph database on your PC. https://neo4j.com/developer-blog/how-to-have-a-cybersecurity-graph-database-on-your-pc/
5. Böhm F, Menges F, Pernul G (2018) Graph-based visual analytics for cyber threat intelligence. Cybersecurity. https://doi.org/10.1186/s42400-018-0017-4
6. Bughio KS, Sikos LF (2023) Knowledge organization systems to support cyber-resilience in medical smart home environments. In: Ahmed M, Haskell-Dowland P (eds) Cybersecurity for smart cities: advanced sciences and technologies for security applications. Springer, Cham, pp 61–69. https://doi.org/10.1007/978-3-031-24946-4_5
7. Casey E, Nelson A, Hyde J (2019) Standardization of file recovery classification and authentication. Digital Invest. https://doi.org/10.1016/j.diin.2019.06.004
8. Chen X, Shen W, Yang G (2021) Automatic generation of attack strategy for multiple vulnerabilities based on domain knowledge graph. In: 47th Annual Conference of the IEEE Industrial Electronics Society. IEEE. https://doi.org/10.1109/IECON48115.2021.9589233
9. Crémilleux D (2019) Visualization for information system security monitoring. PhD thesis, Loire Bretagne University, Rennes, France
10. Du M, Jiang J, Jiang Z et al (2019) PRTIRG: a knowledge graph for people-readable threat intelligence recommendation. In: Douligeris C, Karagiannis D, Apostolou D (eds) Knowledge science, engineering and management. Springer, Cham, pp 47–59. https://doi.org/10.1007/978-3-030-29551-6_5
11. Etoty RE, Erbacher RF (2014) A survey of visualization tools assessed for anomaly-based intrusion detection analysis. Technical report, Army Research Laboratory. https://apps.dtic.mil/sti/pdfs/ADA601590.pdf
12. Fawaz A, Bohara A, Cheh C et al (2016) Lateral movement detection using distributed data fusion. In: 35th Symposium on Reliable Distributed Systems. IEEE, Los Alamitos, pp 21–30. https://doi.org/10.1109/SRDS.2016.014
13. Foresti S, Agutter J (2007) VisAlert: from idea to product. In: Goodall JR, Conti G, Ma KL (eds) VizSEC 2007. Springer, Heidelberg, pp 159–174. https://doi.org/10.1007/978-3-540-78243-8_11
14. Garae J, Ko RKL (2017) Visualization and data provenance trends in decision support for cybersecurity. In: Carrascosa IP, Kalutarage HK, Huang Y (eds) Data analytics and decision support for cybersecurity. Springer, Cham, pp 243–270. https://doi.org/10.1007/978-3-319-59439-2_9
15. Garrido JS, Dold D, Frank J (2021) Machine learning on knowledge graphs for context-aware security monitoring. In: 2021 IEEE International Conference on Cyber Security and Resilience. IEEE, pp 55–60. https://doi.org/10.1109/CSR51186.2021.9527927
16. Grojek AE, Sikos LF (2022) Ontology-driven artificial intelligence in IoT forensics. In: Daimi K, Francia G III, Encinas LH (eds) Breakthroughs in digital biometrics and forensics. Springer, Cham, pp 257–286. https://doi.org/10.1007/978-3-031-10706-1_12
17. Hooi EKJ, Zainal A, Maarof MA et al (2019) TAGraph: knowledge graph of threat actor. In: 2019 International Conference on Cybersecurity (ICoCSec). IEEE. https://doi.org/10.1109/ICoCSec47621.2019.8970979
18. Iannacone M, Bohn S, Nakamura G et al (2015) Developing an ontology for cyber security knowledge graphs. In: Trien JP, Prowell SJ, Bridges RA et al (eds) Proceedings of the 10th Annual Cyber and Information Security Research Conference. ACM, New York. https://doi.org/10.1145/2746266.2746278
19. Inoue D, Eto M, Suzuki K et al (2012) DAEDALUS-VIZ: novel real-time 3D visualization for Darknet monitoring-based alert system. In: Ninth International Symposium on Visualization for Cyber Security. ACM, New York, pp 72–79. https://doi.org/10.1145/2379690.2379700
20. Islam R, Refat RUD, Yerram SM et al (2022) Graph-based intrusion detection system for controller area networks. IEEE Trans Intell Transp Syst 23(3):1727–1736. https://doi.org/10.1109/TITS.2020.3025685
21. Jia Y, Qi Y, Shang H et al (2018) A practical approach to constructing a knowledge graph for cybersecurity. Engineering 4(1):53–60. https://doi.org/10.1016/j.eng.2018.01.004
22. Jiang J, Chen J, Choo KKR et al (2018) A visualization scheme for network forensics based on attribute oriented induction based frequent item mining and hyper graph. In: Matoušek P, Schmiedecker M (eds) Digital forensics and cyber crime. Springer, Cham, pp 130–143. https://doi.org/10.1007/978-3-319-73697-6_10
23. Johnson JH (2016) Embracing n-ary relations in network science. In: Wierzbicki A, Brandes U, Schweitzer F et al (eds) Advances in network science. Springer, Cham, pp 147–160. https://doi.org/10.1007/978-3-319-28361-6_12
24. Kaloroumakis PE, Smith MJ (2021) Toward a knowledge graph of cybersecurity countermeasures. https://d3fend.mitre.org/resources/D3FEND.pdf

25. Kang JJ, Sikos LF, Yang W (2021) Reducing the attack surface of edge computing IoT networks via hybrid routing using dedicated nodes. In: Ahmed M, Haskell-Dowland P (eds) Secure edge computing: applications, techniques and challenges. CRC Press, Boca Raton, pp 97–111. https://doi.org/10.1201/9781003028635
26. Kiesling E, Ekelhart A, Kurniawan K et al (2019) The SEPSES knowledge graph: an integrated resource for cybersecurity. In: Ghidini C, Hartig O, Maleshkova M et al (eds) The Semantic Web—ISWC 2019. Springer, Cham, pp 198–214. https://doi.org/10.1007/978-3-030-30796-7_13
27. Kurniawan K, Ekelhart A, Kiesling E et al (2022) KRYSTAL: knowledge graph-based framework for tactical attack discovery in audit data. Comput Secur. https://doi.org/10.1016/j.cose.2022.102828
28. Li K, Zhou H, Tu Z et al (2020) CSKB: a cyber security knowledge base based on knowledge graph. In: Yu S, Mueller P, Qian J (eds) Security and privacy in digital economy. Springer, Singapore, pp 110–113. https://doi.org/10.1007/978-981-15-9129-7_8
29. Liu Z, Sun Z, Chen J et al (2020) STIX-based network security knowledge graph ontology modeling method. In: 3rd International Conference on Geoinformatics and Data Analysis. ACM, New York, pp 152–157. https://doi.org/10.1145/3397056.3397083
30. Lo WW, Layeghy S, Sarhan M et al (2022) E-GraphSAGE: a graph neural network based intrusion detection system for IoT. In: Varga P, Granville LZ, Galis A et al (eds) 2022 IEEE/IFIP Network Operations and Management Symposium. IEEE. https://doi.org/10.1109/NOMS54207.2022.9789878
31. MITRE (2016) What is the relevance of Alert X? https://dist.neo4j.com/wp-content/uploads/20160218120000/cyber-attack-alert-relevance.png
32. Mohamed A, Abuoda G, Ghanem A et al (2021) RDFFrames: knowledge graph access for machine learning tools. VLDB J. https://doi.org/10.1007/s00778-021-00690-5
33. Najafi P, Mühle A, Pünter W et al (2019) MalRank: a measure of maliciousness in SIEM-based knowledge graphs. In: Balenson D (ed) 35th Annual Computer Security Applications Conference. ACM, New York, pp 417–429. https://doi.org/10.1145/3359789.3359791
34. Narayanan S, Ganesan A, Joshi K et al (2018) Early detection of cybersecurity threats using collaborative cognition. In: 4th International Conference on Collaboration and Internet Computing. IEEE, Los Alamitos, CA, USA, pp 354–363. https://doi.org/10.1109/CIC.2018.00054
35. Noel S (2015) Building a big data architecture for cyber attack graphs. GraphConnect, San Francisco, 21 Oct 2015
36. Noel S, Bodeau D, McQuaid R (2017) Big data graph knowledge bases for cyber resilience. In: Kott A, Rodosek GD (eds) NATO IST-153/RWS-21 Workshop on Cyber Resilience. RWTH Aachen, Aachen, pp 6–21. https://ceur-ws.org/Vol-2040/paper2.pdf
37. Onwubiko C (2018) CoCoa: an ontology for cybersecurity operations centre analysis process. In: 2018 International Conference on Cyber-Situational Awareness, Data Analytics and Assessment. IEEE, https://doi.org/10.1109/CyberSA.2018.8551486
38. Pingle A, Piplai A, Mittal S et al (2019) RelExt: relation extraction using deep learning approaches for cybersecurity knowledge graph improvement. In: Spezzano F, Chen W, Xiao X (eds) Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. ACM, New York, pp 879–886. https://doi.org/10.1145/3341161.3343519
39. Piplai A, Mittal S, Abdelsalam M et al (2020) Knowledge enrichment by fusing representations for malware threat intelligence and behavior. In: 2020 IEEE International Conference on Intelligence and Security Informatics. IEEE. https://doi.org/10.1109/ISI49825.2020.9280512
40. Piplai A, Mittal S, Joshi A et al (2020) Creating cybersecurity knowledge graphs from malware after action reports. IEEE Access 8:211691–211703. https://doi.org/10.1109/ACCESS.2020.3039234
41. Piplai A, Ranade P, Kotal A et al (2020) Using knowledge graphs and reinforcement learning for malware analysis. In: 2020 IEEE International Conference on Big Data. IEEE, pp 2626–2633. https://doi.org/10.1109/BigData50022.2020.9378491
42. Pokorný J (2015) Graph databases: their power and limitations. In: Saeed K, Homenda W (eds) Computer information systems and industrial management. Springer, Cham, pp 58–69. https://doi.org/10.1007/978-3-319-24369-6_5
43. Qian J, Tang S, Liu H et al (2016) Privacy inference on knowledge graphs: hardness and approximation. In: 12th International Conference on Mobile Ad-Hoc and Sensor Networks. IEEE, Los Alamitos, CA, USA, pp 132–138. https://doi.org/10.1109/MSN.2016.030
44. Qin S, Chow KP (2019) Automatic analysis and reasoning based on vulnerability knowledge graph. In: Ning H (ed) Cyberspace data and intelligence, and cyber-living, syndrome, and health. Springer, Singapore, pp 3–19. https://doi.org/10.1007/978-981-15-1922-2_1
45. Rihany M, Hannou FZ, Mimouni N et al (2021) A semantic-based approach for assessing the impact of cyber-physical attacks: a healthcare infrastructure use case. In: Braun T, Gehrke M, Hanika T et al (eds)

Graph-based representation and reasoning. Springer, Cham, pp 208–215. https://doi.org/10.1007/978-3-030-86982-3_16
46. Sarhan I, Spruit M (2021) Open-CyKG: an open cyber threat intelligence knowledge graph. Knowl Based Syst. https://doi.org/10.1016/j.knosys.2021.107524
47. Sikos LF (2019) OWL ontologies in cybersecurity: conceptual modeling of cyber-knowledge. In: Sikos LF (ed) AI in cybersecurity. Springer, Cham, pp 1–17. https://doi.org/10.1007/978-3-319-98842-9_1
48. Sikos LF (2020) AI in digital forensics: ontology engineering for cybercrime investigations. WIREs forensic science 3:e1394. https://doi.org/10.1002/wfs2.1394
49. Sikos LF (2020) AI-powered cybersecurity: from automated threat detection to adaptive defense. CISO Mag 4(5):74–87
50. Sikos LF (2021) Contextualized knowledge graphs in communication network and cyber-physical system modeling. In: Sikos LF, Seneviratne OW, McGuinness DL (eds) Provenance in data science: from data models to context-aware knowledge graphs. Springer, Cham, pp 47–58. https://doi.org/10.1007/978-3-030-67681-0_4
51. Sikos LF, Philp D (2020) Provenance-aware knowledge representation: a survey of data models and contextualized knowledge graphs. Data Sci Eng 5:293–316. https://doi.org/10.1007/s41019-020-00118-0
52. Sikos LF, Philp D, Stumptner M et al (2018) Visualization of conceptualized dynamic network knowledge for cyber-situational awareness. In: Cañas AJ, Reiska P, Zea C et al (eds) Proceedings of the 8th International Conference on Concept Mapping, p 396
53. Sikos LF, Philp D, Voigt S et al (2018) Provenance-aware LOD datasets for detecting network inconsistencies. In: Capadisli S, Cotton F, Giménez-García JM et al (eds) CKGSemStats 2018: Contextualized Knowledge Graphs, and Semantic Statistics. RWTH Aachen University, Aachen
54. Sikos LF, Stumptner M, Mayer W et al (2018) Automated reasoning over provenance-aware communication network knowledge in support of cyber-situational awareness. In: Liu W, Giunchiglia F, Yang B (eds) Knowledge science, engineering and management. Springer, Cham, pp 132–143. https://doi.org/10.1007/978-3-319-99247-1_12
55. Sikos LF, Stumptner M, Mayer W et al (2018) Representing network knowledge using provenance-aware formalisms for cyber-situational awareness. Procedia Comput Sci 126:29–38. https://doi.org/10.1016/j.procs.2018.07.206
56. Sikos LF (2019) Knowledge representation to support partially automated honeypot analysis based on Wireshark packet capture files. In: Czarnowski I, Howlett RJ, Jain LC (eds) Intelligent decision technologies 2019. Springer, Singapore, pp 345–351. https://doi.org/10.1007/978-981-13-8311-3_30
57. Sikos LF, Philp D, Howard C et al (2019) Knowledge representation of network semantics for reasoning-powered cyber-situational awareness. Springer, Cham, pp 19–45. https://doi.org/10.1007/978-3-319-98842-9_2
58. Sills M, Ranade P, Mittal S (2020) Cybersecurity threat intelligence augmentation and embedding improvement: a healthcare usecase. In: 2020 IEEE International Conference on Intelligence and Security Informatics. IEEE. https://doi.org/10.1109/ISI49825.2020.9280482
59. Siren (2022) The siren data model and cyber investigations. https://siren.io/cyber-security/
60. Sleeman J, Finin T, Halem M (2020) Temporal understanding of cybersecurity threats. In: 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS). IEEE, Los Alamitos, CA, USA, pp 115–121. https://doi.org/10.1109/BigDataSecurity-HPSC-IDS49724.2020.00030
61. Syed Z, Padia A, Finin T et al (2016) UCO: a Unified Cybersecurity Ontology. In: AAAI Workshop on Artificial Intelligence for Cyber Security. AAAI
62. Tsigkas O, Thonnard O, Tzovaras D (2012) Visual spam campaigns analysis using abstract graphs representation. In: Ninth International Symposium on Visualization for Cyber Security. ACM, New York, pp 64–71. https://doi.org/10.1145/2379690.2379699
63. Turnbull B, Randhawa S (2015) Automated event and social network extraction from digital evidence sources with ontological mapping. Digit Invest 13:94–106. https://doi.org/10.1016/j.diin.2015.04.004
64. Veeramachaneni K, Arnaldo I, Korrapati V (2016) AI^2: training a big data machine to defend. In: Qiu M (ed) 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS). IEEE, pp 49–54. https://doi.org/10.1109/BigDataSecurity-HPSC-IDS.2016.79
65. Voigt S, Howard C, Philp D et al (2018) Representing and reasoning about logical network topologies. In: Croitoru M, Marquis P, Rudolph S et al (eds) Graph structures for knowledge representation and reasoning. Springer, Cham, pp 73–83. https://doi.org/10.1007/978-3-319-78102-0_4

66. Wang Z, Zhu H, Liu P et al (2021) Social engineering in cybersecurity: a domain ontology and knowledge graph application examples. Cybersecurity. https://doi.org/10.1186/s42400-021-00094-6
67. Yamanaka R (2021) Graphs and machine learning for cybersecurity. https://medium.com/oracledevs/graphs-and-machine-learning-for-cybersecurity-7115b9b544b5
68. Yankulov M (2020) Boosting cybersecurity efficiency with knowledge graphs. https://www.ontotext.com/blog/boosting-cybersecurity-efficiency-with-knowledge-graphs/
69. Zhang K, Liu J (2020) Review on the application of knowledge graph in cyber security assessment. In: IOP conference series: materials science and engineering. IOP Publishing https://doi.org/10.1088/1757-899X/768/5/052103
70. Zhang S (2023) Generating network security defense strategy based on cyber threat intelligence knowledge graph. In: Quan W (ed) Emerging networking architecture and technologies. Springer, Singapore, pp 507–519. https://doi.org/10.1007/978-981-19-9697-9_41
71. Zhang Y, Fan Y, Hou S et al (2020) Cyber-guided deep neural network for malicious repository detection in GitHub. In: 2020 IEEE International Conference on Knowledge Graph. IEEE, pp 458–465. https://doi.org/10.1109/ICBK50248.2020.00071
72. Zheng L, Li Z, Li J et al (2019) AddGraph: anomaly detection in dynamic graph using attention-based temporal GCN. In: Twenty-eighth International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence Organization, pp 4419–4425. https://doi.org/10.24963/ijcai.2019/614

Dr. Leslie F. Sikos is a computer scientist specializing in artificial intelligence and data science, with a focus on cybersecurity applications. He has worked on major research projects to develop knowledge organization systems, in particular, ontologies and context-aware semantic knowledge graphs, that support cyber-situational awareness and cyber-physical system modeling, and ontology-powered AI frameworks that partially automate digital forensic investigations. He holds two PhD degrees and 20+ industry certificates. He is an active member of the research community as an author, editor, reviewer, conference organizer, and speaker; a senior member of the IEEE, and a certified professional of the Australian Computer Society. Dr. Sikos published more than 20 books, including textbooks, monographs, and edited volumes.