2022 7th IEEE International Conference on Data Science in Cyberspace (DSC) | 978-1-6654-7480-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/DSC55868.2022.00075

A Review of Provenance Graph based APT Attack Detection:Applications and Developments

1st Yang Lv
Cyberspace Institue of Advanced Technology
Guangzhou University
Guangzhou,China
2112106187@e.gzhu.edu.cn

2nd Shaona Qin
Zhaoyuan NO.1 Secondary Vocational School
Yantai, China
357886201@qq.com

3rd Zifeng Zhu
Cyberspace Institue of Advanced Technology
Guangzhou University
Guangzhou,China
841850909@qq.com

4st Zhuocheng Yu
Cyberspace Institue of Advanced Technology
Guangzhou University
Guangzhou,China
gzhu_yzc@163.com

5st Shudong Li*
1st Cyberspace Institue of Advanced Technology, Guangzhou University
Guangzhou,China
2nd Peng Cheng Laboratory
Shenzhen,China
lishudong@gzhu.edu.cn

6st Weihong Han*
1st Cyberspace Institue of Advanced Technology, Guangzhou University
Guangzhou,China
2nd Peng Cheng Laboratory
Shenzhen,China
hanweihong@gzhu.edu.cn

***Abstract*—With the development of information technology, the cyberspace also derives an increasing number of security risks and threats. There are more and more advanced cyber-attacks of which the APT is one of the most sophisticated attacks and is commonly adopted by modern attackers. Traditional detection methods are challenging to cope with complicated and persistent APT-style attacks. Provenance graph models the interactions between system-level entities within a specific system, it possesses powerful semantic expression ability and correlation analysis ability and many works have been conducted using provenance graph to detect APT-style multi-step attacks. This article mainly discusses the current research situation and tendency of APT attack detection based on provenance graph. We firstly introduce basic concepts of APT and provenance graph, and then review the existing representative works and conclude a general framework design for provenance graph based attack detection system. Finally, we also provide several promising research directions in the future.**

Keywords—cyber security, advanced persistent threat (APT), cyber attack detection, provenance graph

I. INTRODUCTION

With the rapid developments and widespread applications of information technologies, the newly-developing Internet technology makes human life more convenient and intelligent. However, the development of information technology has also derived a lot of internal and external security risks and threats at the same time [1]. The definition of cyber space security has been proposed and widely enriched in recent years and the current security situations of cyber space faces severe challenges. According to the analysis report [2] released by the China National Internet Emergency Center (CNCERT/CC), our country has suffered increasing cyberattacks at home and abroad, especially threats aimed at specific organizations, governmental institutions, commercial and technology enterprise, and critical information infrastructures. On one hand, The quantities and frequencies of cyberattack events are increasing continuously, and the stealthiness and advance of cyberattack technology make that how to detect it and improve the cyber defenses a key and urgent issue and there have been many works focus on the key issues like malware [3,4,5] or malicious code detection [6,7] and so on. On the other hand, the ultimate goal of attackers in cyberspace is to seek for practical benefits in the real world. Therefore, there exist a strong connection between security in cyberspace and in reality.

In recent years, sophisticated multi-step cyberattacks are becoming increasing prominent in modern cyberspace where Advanced Persistent Threat(APT) is the most representative attack campaign and it has become one of the most threatening cyberattacks [8]. For instance, the Google Aurora [9] is a famous APT attack occurred in 2010 which was permeated just only clicking a malicious link when surfing Internet. Besides, the well-known Stuxnet [10] leveraged personal computers and mobile devices as bridges, and then gradually spread and implemented destructions activities. In 2011, Duqu [11] which is evolved based on Stuxnet was carried out in order to collect industry control system's information. Furthermore, the influential APT attacks such as Ocean Lotus, Flame and so on [12] have been active using different attack vectors so far and pose great threats whether in cyberspace or in the real world.

In the era of big data, the cyber attack and defense scenes also show new characteristics, which are generally reflected in the larger data volume, multiple data dimensions, complicated and diverse attack vectors, passivity of defense and so on. Therefore, traditional defense strategies like intrusion detection, firewall, vulnerability scanning are severely limited [13]. It is also worth mentioning that there exists extreme asymmetries between attackers and defenders at present [14]. Therefore, on

978-1-6654-7480-1/22/$31.00 ©2022 IEEE
DOI 10.1109/DSC55868.2022.00075

the basis of traditional detection methods and technologies, more researchers begin to use provenance graph to conduct cyberattack detection researches. Provenance graph uses graph structure to model cyber security data which clearly reveals the whole implementation process of a specific attack. Compared with previous methods, provenance graph-based method has better semantic expression ability and attack events correlation ability. Using provenance graph, security analysts can find out attack events, trace attack process, identify the intention of attackers and even predict the attack tendency in the future.

This article conducts a review of existing representative provenance graph-based APT attack detection methods, the rest of the article is organized as follows: the second section introduces the background knowledge of APT and provenance graph. The third section reviews the existing representative provenance-based APT detection models and systems. In fourth section, we conclude a general system design paradigm using provenance graph for cyber attack detection based on the previous contents. And we also provide several research directions in fifth section and the last is the conclusion.

II. BACKGROUND

A. What is APT?

APT is the abbreviation for **A**dvanced **P**ersistent **T**hreat, as the name itself implies, it is significantly different compared to previous regular cyberattacks. Under the framework guide for information security risk management [15] introduced by National Institute of Standards and Technology(NIST), APT attacks refer to the form of attacks adversary with a wealth of expertise and resources that can create opportunities to achieve its goals by using multiple attack methods.

Cyber Attack Chaim is a kind of model describing the attack steps based on the whole process. It translates the whole attack scenario into a series of ongoing, connecting steps and then it can empower cyber defenses form the perspective of attackers. There have been several representative attack chain models such as Kill Chain, Mandiant [16], Diamond Model [17], ATT&CK [18] and so on, of which the Lockheed Martin Kill Chain [19]is the most common one, which can be represented as Fig. 1. shown, it defines the whole life cycle of an attack as seven steps: reconnaissance, weaponization, delivery, exploitation, installation, C2, and actions.

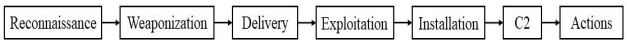

Fig. 1. Lockheed Martin Kill Chain

As a result, compared to regular cyber threats, APT shows three striking differences, namely Advanced, Persistent and Threat. Among them, Advanced refers to attacker are usually equipped with advanced attack tools and technologies and are high organized. Persistent means that attacker will not give up a campaign until achieve their purposes, so the period of an APT attack will last for a long time. The main differences between traditional attacks and APT attack are illustrated by the following Table [20].

TABLE I. DIFFERENECS BETWEEN TRADITIONAL AND APT ATTACK

	Traditional Attacks	**APT Attacks**
Attacker	Mostly single person	Highly organized, sophisticated, determined, and well-resourced group
Target	Unsepecified, mostly individual systems	Specific organizations, government institutions, commercial enterprises
Purpose	Financial benefits, demonstrating abilities	Competitive advantages, strategic benefits
Approach	Single-run, “smash and grab”, short period	Repeated attempts, stays low and slow, adapts to resist defenses, long term

B. Limitations of Traditional APT Detection Methods

Based on the characteristics of APT mentioned above, traditional intrusion detection is not suitable for the detection of APT-style long-time running attacks. On one hand, Pattern matching-based detection methods [21] are powerless when facing advanced attack tactics such as zero-day vulnerability exploits because it is highly dependent on the pre-defined rules or patterns which are used to match anomalous or malicious behavior. On the other hand, anomaly-based detection systems [22] are also limited to model the long-time running attack events because of the resource constraints. In conclusion, the limitations of traditional detection methods are as follows:

- Traditional detection methods lack effective information correlation to reconstruct attack chain due to the APT’s long-time persistent process.
- It is difficult to implement real-time detection and find out the most possible malicious behavior based on traditional detection methods.
- Results of traditional detections cannot be well leveraged by security analysts for further inference on attack-related events to figure out more information about an attack event.
- Attackers can leverage various evasion techniques to bypass the traditional detection systems because these methods only inspect short sequences of system events while lack of efficiency and robustness.

Recently, there have been some works [24-33,35-38] suggested that system-level provenance graph can make a great difference in APT attack detection. Compared to traditional cyberattack detection techniques, system-level provenance graph have appropriate granularity to model all control flows and information flows as graphs which contains rich contextual semantic information and correlations that will benefit for attack detection and thus helps improve accuracy and explainability of detection systems.

C. Definition of Provenance Graph

At the level of a single host, system-level provenance graph treats all system entities such as processes, files, sockets as vertices and operations between these entities as edges. As a result, provenance graph clearly represents system events or executions as control flows and data flows between subject entity and object entity. Fig. 2. shows two sub-graphs from a

system-level provenance graph, and some necessary definitions are given as follows:

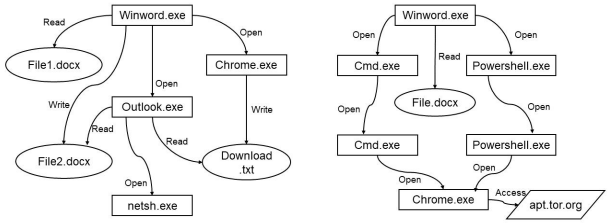

Fig. 2. Two sub-graphs from a provenance graph, left shows a benign scenario while right shows a malicious scenario [23]

1) Entities, Relations and Events: As we have mentioned before, provenance graph models control flows and data flows between system entities, so the nodes in the graph represent the practical entities such as processes, files and so on while edges represent the operations between these system entities, such as a process opened a file or an process opened another process. Futhermore, subject entity refer to an system entity which performs an specific opreation like read, open, write, fork.etc on another system entity which we call it object entity, and the specific operation between subject entity and object entity is defined as a event, denoted by a triple <subject, relation, object>. For example, in Figure. 2. a Winword.exe(subject) reads the File1.docx(object), and <Winword.exe, read, File1,docx> is an file read event in the provenance graph.

2) Backward and Forward Tracking: computer system is just like the real world, where everything has temporal and spatial attributes, so different events in a provenance graph has certain logical relationships which then imply the causality relation between events. As a result, if we start from a specific node in provenance graph, such as the malicious file node or process node, the backward tracking allows us to find out all nodes may causally influence the malicious node. And this is same as the trace of attack which benefits the in-depth analysis for an attack event. Similarily, the forward tracking tries to find out all system entities that depend on the starting node, it can be used for analysing the impacts of this attack event and implement evaluation of defence capability.

In summary, provenance graph is an effective data form and tool for APT attack detection, it represent the relations between running system entities and is able to not only connect events causally related, but also provides rich contextual information about it.

III. Provenance Graph for APT Attack Detection

This section mainly reviews recent advances on APT attack detection with the use of provenance graph. We firstly introduce several representative methods, models or systems in recent years, and then compare the methodologies and applications of these methods, and finally give our analysis of advantages and challenges of these works.

A. Representative Methods, Models, or Systems

BackTracker [24] is a classical method based on provenance graph and the following works are almost implemented based on it. BackTracker mainly figured out the causality relations of processes, files and filenames. In recent few years, SLEUTH [25] was the pioneering work which used provenance graph to reconstruct APT attack scenario, the main framework of this system is shown as Fig.3. It proposed a kind of system applied in enterprise-wide hosts to achieve real-time attack scenario reconstructions and developed a platform-neutral and main-memory based audit logs data dependence graph abstract method. SLEUTH performed much more efficient and accurate than previous methods.

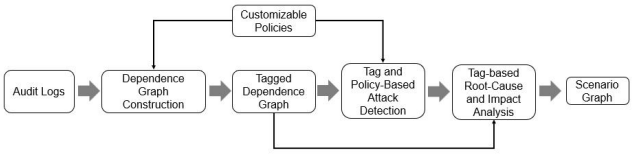

Fig. 3. Basic Framework of SLEUTH [25]

After SLEUTH introduced the provenance graph into the field of APT detection, there were increasing works focusing on this approach. Milajerdi S M et al. [26] proposed to leverage the correlation analysis of Cyber Threat Indicator(CTI) help to detect APT attacks, they developed a system called POIROT, which constructed system behavior provenance graph based on the audit logs generated within a single host. POIROT transforms the problem form of threat detection to the problem of imprecise pattern matching in the graph that is to search a sub-graph in provenance graph which is matched with the predefined graph pattern. As a result, POIROT has a better robustness and reliability with the help of CTI. In order to capture the correlations of attack events and suspicious information flows, HOLMES [27] designed a new kind of graph named high-level scenario graph (HSG) which realized the transformation of low-level information like system logs and threat alerts to high-level attack stages in a cyber kill chain like initial compromise, privilege escalation in APT stages. This system has strong ability to effectively distinguish attack scenarios from benign scenarios and mitigate the problem of semantic gap between low level and high-level information. The architecture of HOLMES is shown in Fig. 4. and it is worth noting that this is the first work that uses provenance graph to map malicious systems events to the corresponding high-level attack steps in a kill chain-style model, which points out a new direction of future's research work.

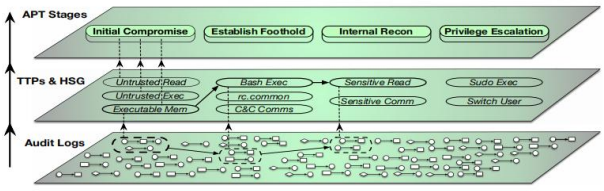

Fig. 4. Architecture of HOLMES system [27]

In addition, there exists a common problem that many intrusion detection systems suffer from the trouble of threat

alert fatigue, which can simply be understood as the high false positive rate so that it maybe confuse the attack events and normal events. Hassan W U et al. [28] proposed a system named NODOZE to mitigate this challenge. NODOZE firstly generates the causality graph of alert events based on the contextual and historical information of threat alerts, and then assigns every event an anomalous score based their previous occurrence frequencies, finally it applies a novel network fusion algorithm to generate the aggregated anomaly score of an specific alert. NODOZE can effectively decrease the false-positive rate and time cost, and it has excellent scalability that can easily been deployed in any threat detection platform.

Besides, monitoring for hosts' status is of great significance in enterprise environment. Han et al. [29] elaborately designed a system called UNICORN, an anomaly-based APT detector which has the ability to detect attacks without the prior knowledge. It can effectively models long-term system behavior as time evolves. Based on these features, UNICORN was the first APT intrusion detection system focusing on the running process of whole system and had a high accuracy and low false positive rate. Wang et al. [30] presented a stealthy malware detection method based on provenance graph. The insight of this method is that the malicious behaviors of malware inevitably interact with the operating system and such sensitive behavior can be captured by provenance graph. Hassen W et al. [31] firstly introduced the benefits of provenance graph into commercial EDR tools, it proposed the definition of Tactical Provenance Graph used to represent the causality dependency relation between threat alerts, and the experiments results indicated that it could observably improve the performance of existing EDR tools.

Furthermore, considering the sequence-based APT detection, Alsaheel A et al. [32] proposed ATLAS, an attack detection method based on the representation of attack events sequences within a graph. ATLAS integrates the advantages of causality correlation analysis, natural language process, and machine learning to model the attack behavior and benign behavior. Wang et al. [33] considered to figure out the role of system entities in provenance graph in node-level, they improved the GraphSAGE [34] in order to learn the comprehensive features of every node so that it can implement node-level detections. The core insight of their system is that nodes related with malicious behavior will always have different patterns that can be captured by machine learning algorithms. Similar with THREATRACE, Lv et al. [35] proposed a cyberattack detection model based on the heterogeneous graph learning, they believed that most of detection methods are lack of intelligence and generalization, which limits the power of data itself and the applications of AI technology, so they represented the provenance data as the form of heterogeneous graph, and then embed the nodes under the guide of a series of pre-defined meta-paths and consequently distinguish the malicious process nodes from benign process nodes. However, this method still relies on pre-defined meta-paths and quality of origin data. Based on these methodologies, Li et al. [36] proposed a hierarchical approach for APT detection with attention-based graph neural network, as shown in Fig. 5. It starts from the intrahost provenance graph within a single host trying to find the malicious events, and then builds correlations between hosts in the level of network to construct interhost provenance graph. Finally, the host-level and network-level anomalies can be associated and mapped into the corresponding APT stages.

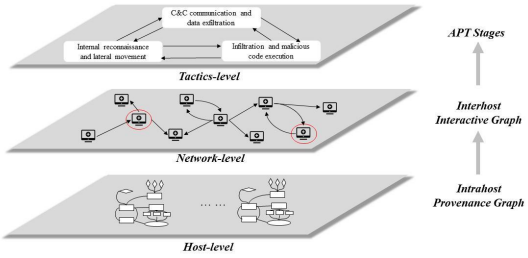

Fig. 5. The hierarchical detection framework [36]

In real world, sophisticated attacks like APT attacks have brought us serious threats, it is also of great significance to study the strategies of attackers and make reasonable predictions for the potential attacks. In addition to detecting cyberattacks using provenance graph, several works have been conducted to make cyber threat predictions using provenance graph. DeepAG [37] is a model used for attack graph construction and threats prediction developed by Li et al. DeepAG firstly leverages transformer model to capture semantic of system logs and then detects the possible attack sequences, what's more, it can reconstruct the attack graph based on the previous historical attack sequences and paths and then make predictions based on what it have learned. Moreover, there also has works modeling security-related data such as CVE, ExploitDB and so on using graph-structure in order to predict the cyber threats or reveal the evolutionary patterns of a specific attack event. Zhao et al. [38] proposed a model using dynamic heterogeneous graph learning to make cyber threats prediction named CTP-DHGL, the overview of this method is shown as Fig. 6. This model characterizes different types of security-related data with a heterogeneous graph which evolves as time goes, it essentially transform the problem of threat detection to the problem of link prediction during a series of dynamic heterogeneous graphs. And finally, the model learns the dynamic evolutionary patterns of graph and has ability to infer potential links based on the patterns have been learned from the history.

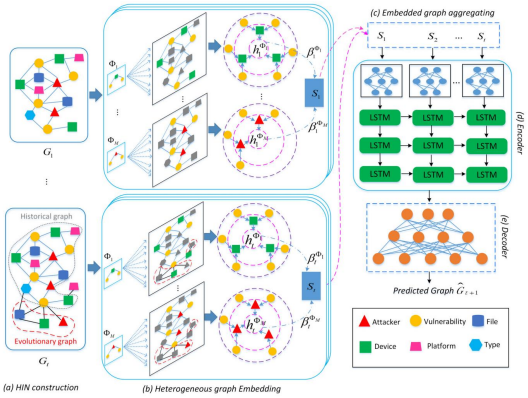

Fig. 6. The overview framework of CTP-DHGL which consists of five components (a) - (e) [38]

B. Advantages and Challenges

As described in the previous section, many works have been conducted using provenance graph in the field of cyber security, such as threat intelligence modeling, cyberattack detection, threat prediction and so on. In this section, we will give a summary of this kind of method and discuss their advantages and challenges.

As shown in Table Ⅱ, we give a brief summary of methods we have mentioned in the previous section, we conclude and compare the core methodologies used in these models and their specific applications. Furthermore, we summarize advantages and challenges of APT attack detection methods based on provenance graph as follows:

1) Advantages:

a) Provenance graph explicitly represents control flows and data flows between system entities with rich semantics and powerful representation abilities, thus security analysts can conduct more effective and comprehensive investigation of an attack campaign using provenance graph.

b) Provenance graph keeps all necessary and useful system execution records and thus is naturally suitable for the detection of long-time persistent running and stealthy attacks like APT-style attacks.

c) Provenance graph not only provides more effecitve detection performance but also gives certain explainability of the given result. What's more, analysts can implement further inference based on given results using expert experience or deep learning technology, which contribute to a more intelligence cyberattack detection and response detection system.

d) Graph representation improves the robustness of detection model as graph-structure are naturally more adversarially robust compared to traditional ML-based models. As a result, detection methods based on provenance graph are more robust to distinguish malicious behavior from benign ones in the real world's applications.

2) Challenges:

a) The first challenge is that it is difficult to establish a well-built pattern for provenance graph, that is to say what data we need and how we organize them, and this factor will influence the data's collection, storage and mangement even cause the common dependency explosion problem.

b) It is necessary to find a ideal balance between the space efficiency of data storage and time efficiency of data acquirement, so that it can meet the requirement of online real-time attack detection.

c) Provenance graph-based attack detection should have a higher true-positive rate and lower false-positive rate, this requires us to design an efficient and robust enough detection algorithm.

d) Online attack detection systems in real environment usually require efficient computation and shortest response time, this is also a great challenge for cyebrattack detection methods based on provenance graph.

TABLE II. SUMMARY AND COMPARISON OF DIFFERENT MODELS

Model	Methods	Applications
SLEUTH [25]	Provenance graph, Tag and Policy-based, Dependence graph, Cause and Impact analysis	Real-time reconstruction of attack scenario, Attack detection
POIROT [26]	Provenance graph, Cyber threat indicator(CTI), Graph pattern matching	Alignment of audit logs and attack behavior, Detection of APT attack with CTI, Threat hunting
HOLMES [27]	Provenance graph, High-level scenario graph, ATT&CK	Real-time attack detection, APT stage mapping
NODOZE [28]	Provenance graph, Alerts dependency graph, Network diffusion	Anomaly detection, Limitation of threat alert fatigue
UNICORN [29]	Provenance graph, Runtime in-memory histogram, Anomaly detection	Identify stealthy anomalous activities and long-term attack detection
ProvDetector [30]	Provenance graph, Impersonation techniques, Path selection,	Detection of stealthy malware and malicious process
RapSheet 31]	Provenance graph, Tactical provenance graph, Endpoint Detection and Response	Threat evaluation, Unknown attack detection, Causality analysis of alerts
ATLAS [32]	Causality correlation graph, Sequence-based representation learning	Attack investigation, Pattern summarization, Reconstruction of end-to-end cyberattack stories
THREATRACE [33]	Provenance graph, Graph neural network	Node-level threat detection, Early-stage detection, Anomaly tracing
Hierarchical [36]	Provenance graph, Heterogeneous graph neural network	System-level and Network-level attack detection, APT stages mapping
DeepAG [37]	Attack graph, Transformer, LSTM	Cyber threat detection, Attack paths prediction
CTP-DHGL [38]	Cyber threat graph, Dynamic heterogeneous graph	Cyber threat prediction, revealing evolutionary patterns of cyber threats

IV. GENERAL DESIGN OF PROVENANCE GRAPH BASED ATTACK DETECTION SYSTEM

In this section, we conclude a general framework of provenance-graph based attack detection system based on what we have reviewed above, and we mainly focus on the system-level provenance graph which can be further generalized to the combination with process-level and network-level provenance graphs. As Fig. 7. shows, the general framework consists of five modules including data collection, construction of

provenance graph, data management, representation learning of provenance graph, and final specific downstream applications. We will then briefly introduce the purposes of each module and corresponding necessary technologies.

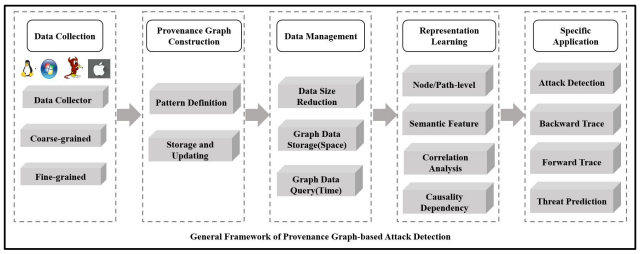

Fig. 7. General framework of system-level provenance graph-based cyberattack detection system

A. Data Collection

The construction of system-level provenance graph mainly relies on the audit logs generated by system, and as a result the data collection module should be firstly deployed in the target system. The collection of provenance data can be further divided into two aspect: coarse-grained collection and fine-grained collection. The former refers to data collection focusing on the interaction behaviors between system entities such as a process read a file and so on, and there have well-built tools such as ETW for windows, CamFlow for Linux, SPADE [39] and so on. While the latter fine-grained data collection aims to improve the quality of collected data in order to mitigate common problem like dependency explosion when constructing a graph.

B. Construction of Provenance Graph

Based on the definitions and provenance data as below, we then represent the subject and object of an event in audit log as nodes while interaction between them as edge connected these two nodes. Generally speaking, the pattern of provenance graph defined in current works are much the same [22], namely the types of nodes usually include process, file, socket while types of edges include common operations between these objects such as open, read, fork, close and so on. We give a simple example as Fig. 8 [24]. The left are system events extracted based on audit logs, while the right is corresponding provenance graph build based on left events. Furthermore, it is necessary to use technologies like graph pruning and graph merge to improve efficiency of computation and storage.

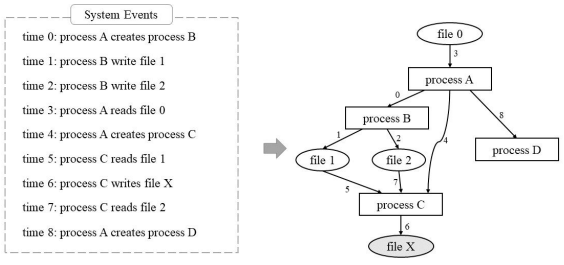

Fig. 8. Example of construction of provenance graph, left are systems events from audit logs while right is provenance graph based on the set of left events [24]

C. Data Management

Data management includes storage and query. Nowadays, NO-SQL database like graph database has been widely used and we can use it to store the provenance graph, and there are also some works [25,26,27,29] proposed to store graph data in the form of cache-based streaming graphs. Besides, as the large size of provenance data which will brings much I/O and storage overload, there is also a need of data reduction methods to minimize the size of data while preserving maximum semantic. In a word, in order to make provenance graph works best, it is of great significance to find an ideal balance between the space cost of data storage and time cost of data query.

D. Representation Learning of Provenance Graph

The purpose of representation learning is to transform provenance graph to computable form that then can be input to machine learning algorithms or deep learning algorithms. What is noteworthy is that the principle of representation learning is to not only preserve the semantic of event itself in a provenance graph but also the correlation and causality of different events, which is shown in Fig. 9 [40], and this kind of causality correlation analysis offered by provenance graph is of crucial importance in sophisticated APT attack detection. On this condition that it can better benefit the following applications when applied in different downstream tasks.

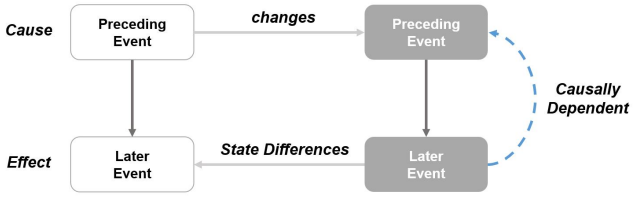

Fig. 9. Causality illustration between preceding event and later event [40]

E. Specific Application

After obtaining the computable system-level provenance graph, security analysts can apply it in different kinds of security-related applications. On one hand, it can be used to detect cyber threats or attacks either in supervised methods or unsupervised methods to find anomaly nodes, edges, paths or even sub-graphs. What's more, external knowledge bases such as attack models or threat intelligence models can also be integrated into provenance graph which can be comprehensively used to implement attack story reconstruction, attack causality inference, attacker intention investigation, threat evaluation or prediction and other tasks, which will finally improve the overall defense capability of detection system.

V. Future Directions

As we have mentioned below, many efforts and meaningful works have been conducted to tackle challenges of using provenance graph to detect APT attacks in cyberspace. However, there are still remaining several potential challenges to overcome and promising future research directions.

A. Data Collection, Storage and Management

It is a universal fact that the scale of data like audit logs generated by computer systems is extremely enormous in today's complex large-scale information systems. It is meaningless that to gather all possible data and then construct the provenance graph. It is worth noting that the first significant stage to construct and use provenance graph is exactly the effective and rational methods of data collection, storage and management. As a result, the future research targets could include three sub-targets. The first is effective and efficient data collection approach which aims to collect necessary data as completely as possible under the limited resources. The second is about efficient data access algorithms, so that users can acquire data which are useful for their works quickly and simply. The last is effective data managements, as the size of provenance data increases over time, it is of fundamental importance to decide which data will be deleted and which to be preserved.

B. Effective and Efficient Graph Mining Algorithm

Provenance graph consists of multiple kinds of nodes and edges, and it provides abundant semantic representations in the level of nodes, edges, paths, and sub-graphs. Previous graph mining algorithms are aimed at obtaining topological features within a graph through using shallow models such as Deepwalk [41], node2vec [42] and so on. However, thanks to the rapid developments of graph neural network especially heterogeneous GNN [43], it would be a noteworthy way to use these GNN-based deep learning methods to capture more useful high-level structure and semantic features in provenance graph in the future works.

C. Temporal and Spatial Attributes

In the real world, everything has its own unique temporal and spatial attributes, the same is true of system events in provenance graph. One straightforward way of describing these attributes is to add timestamp attached to specific edge which indicates time when this event occurred and leverage IP address to denotes the spatial attribute. Provenance graph has a more powerful information representation capabilities after being added with temporal and spatial attributes, and there already have some cyber-security related works focusing on such as temporal and spatial knowledge graph [44], dynamic heterogeneous graph and so on, where these methods empower the reasoning and inference ability of system so that furtherly strengthen our cyber security defense capabilities to a certain extent.

D. Hierarchical Down-to-Top Detection Framework

In fact, many works in the field of APT detection only focus on attack behaviors or events detection or anomaly detection which only reflect on specific kind of attack technology or a specific stage in cyberattack model like Kill Chain. As we all known, APT-style sophisticated attack has to go through multiple stages of attacks in different forms, which includes 14 tactics and more than 150 technologies based on ATT&CK [18]. As a result, in the future work, we are looking forward to realize the valid mapping from low-level single-host based detection to high-level network-based detection, and finally try to map malicious system events to its corresponding correct APT attack stages [27,36]. We believe that this problem paradigm of hierarchical down-to-top attack detection will play a significant role in future's cyberattack detection and provide us with better explainability.

VI. Conclusion

Traditional detection methods for APT attacks could not work efficiently due to the persistence, stealthiness, advancement and other unique characterizes of APT. Provenance graph has strong ability of modeling data flows and control flows between system entities and thus are gradually applied in cyberattack detection. This article conducts a review of APT attack detection methods based on provenance graph from the following four scopes: 1) definition of APT attack and its distinctive characteristics compared with regular attacks; 2) review of existing representative works in APT attack detection based on provenance graph; 3) introduction of general framework design of provenance graph based cyberattack detection system and corresponding key technologies used in different modules; 4) discussions on future promising research directions and challenges. In conclusion, provenance graph is powerful and non-trivial for APT attack detection and it is also potential to play an important role in other cybersecurity-related fields, how to make full use of provenance graph is worthy of more investigation.

Acknowledgment

This research was funded by NSFC (No. 62072131, 61972106), the Major Key Project of PCL (Grant No. PCL2022A03， PCL2021A02， PCL2021A09), Key R&D Program of Guangdong Province (No.2019B010136003), National Key Research and Development Program of China (No. 2019QY1406), Guangdong Basic and Applied Basic Research Foundation (No.2022A1515011401） and Science and Technology Projects in Guangzhou (No.202102010442).

References

[1] Symantec internet security threat report, 2020.

[2] http://www.cac.gov.cn/2020-09/26/c_1602682854845452.htm

[3] Xu N, Li S, Wu X, et al. An APT Malware Classification Method Based on Adaboost Feature Selection and LightGBM[C]//2021 IEEE Sixth International Conference on Data Science in Cyberspace (DSC). IEEE, 2021: 635-639.

[4] Shudong Li,Qianqing Zhang, Xiaobo Wu, Weihong Han, Zhihong Tian. Attribution classification method of APT malware in IoT using machine learning techniques. Security and Communication Networks, vol.2021, Article ID 9396141,2021.

[5] Meihua Fan, Shudong Li, Xiaobo Wu, Weihong Han, Zhaoquan Gu and Zhihong Tian. A Novel Malware Detection Framework Based on Weighted Heterograph. CIAT 2020: 2020 International Conference on Cyberspace Innovation of Advanced Technologies Guangzhou China. December 4-6, 2020,PP:39–43.

[6] Shudong Li, Laiyuan Jiang, Qianqing Zhang, Zhen Wang, Zhihong Tian and Mohsen Guizani. A Malicious Mining Code Detection Method

Based on Multi-Features Fusion. IEEE Transactions on Network Science and Engineering, 2022.

[7] Shudong Li, Yuan Li, Weihong Han, Xiaojiang Du, Mohsen,Guizani, Zhihong Tian. Malicious mining code detection based on ensemble learning in cloud computing environment. Simulation Modelling Practice and Theory, Volume 113, Article ID 102391, 2021.

[8] FireEye-Mandiant. 2020. FireEye Mandiant M-Trends Report. https://content.fireeye.com/m-trends/rpt-m-trends 2021.[Online;accessed 12-July-2021].

[9] Ghafir I, Prenosil V. Advanced persistent threat attack detection: an overview[J]. Int J Adv Comput Netw Secur, 2014, 4(4): 5054.

[10] Langner R. Stuxnet: Dissecting a cyberwarfare weapon[J]. IEEE Security & Privacy, 2011, 9(3): 49-51.

[11] Virvilis N, Gritzalis D. The big four-what we did wrong in advanced persistent threat detection?[C]//2013 international conference on availability, reliability and security. IEEE, 2013: 248-254.

[12] Mei Y, Han W, Li S, et al. A Survey of Advanced Persistent Threats Attack and Defense[C]//2021 IEEE Sixth International Conference on Data Science in Cyberspace (DSC). IEEE, 2021: 608-613.

[13] Han X, Pasquier T, Seltzer M. Provenance-based intrusion detection: opportunities and challenges[C]//10th USENIX Workshop on the Theory and Practice of Provenance (TaPP 2018). 2018.

[14] Ren J, Xu Y. A compartmental model to explore the interplay between virus epidemics and honeynet potency[J]. Applied Mathematical Modelling, 2018, 59: 86-99.

[15] Ross R S. Managing information security risk: Organization, mission, and information system view[J]. 2011.

[16] Cyber Attack Lifestyle[EB/OL].https://www.iacpcybercenter.org/resource-center/what-is-cyber-crime/cyber-attack-lifecycle/.

[17] Caltagirone S, Pendergast A, Betz C. The diamond model of intrusion analysis[R]. Center For Cyber Intelligence Analysis and Threat Research Hanover Md, 2013.

[18] Strom B E, Applebaum A, Miller D P, et al. Mitre att&ck: Design and philosophy[J]. Technical report, 2018.

[19] Hutchins E M, Cloppert M J, Amin R M. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains[J]. Leading Issues in Information Warfare & Security Research, 2011, 1(1): 80.

[20] Alshamrani A, Myneni S, Chowdhary A, et al. A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities[J]. IEEE Communications Surveys & Tutorials, 2019, 21(2): 1851-1877.

[21] Bhuyan M H, Bhattacharyya D K, Kalita J K. Network anomaly detection: methods, systems and tools[J]. Ieee communications surveys & tutorials, 2013, 16(1): 303-336.

[22] Li Z, Chen Q A, Yang R, et al. Threat detection and investigation with system-level provenance graphs: a survey[J]. Computers & Security, 2021, 106: 102282.

[23] Anjum M, Iqbal S, Hamelin B. ANUBIS: A Provenance Graph-Based Framework for Advanced Persistent Threat Detection[J]. arXiv preprint arXiv:2112.11032, 2021.

[24] King S T , Chen P M . Backtracking Intrusions[J]. Operating Systems Review, 2003, 37(5):p.223-236.

[25] Hossain M N, Milajerdi S M, Wang J, et al. {SLEUTH}: Real-time attack scenario reconstruction from {COTS} audit data[C]//26th USENIX Security Symposium (USENIX Security 17). 2017: 487-504.

[26] Milajerdi S M, Eshete B, Gjomemo R, et al. Poirot: Aligning attack behavior with kernel audit records for cyber threat hunting[C]//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. 2019: 1795-1812.

[27] Milajerdi S M, Gjomemo R, Eshete B, et al. Holmes: real-time apt detection through correlation of suspicious information flows[C]//2019 IEEE Symposium on Security and Privacy (SP). IEEE, 2019: 1137-1152.

[28] Hassan W U, Guo S, Li D, et al. Nodoze: Combatting threat alert fatigue with automated provenance triage[C]//Network and Distributed System Security Symposium. 2019.

[29] Han X, Pasquier T, Bates A, et al. Unicorn: Runtime provenance-based detector for advanced persistent threats[J]. arXiv preprint arXiv:2001.01525, 2020.

[30] Wang Q, Hassan W U, Li D, et al. You Are What You Do: Hunting Stealthy Malware via Data Provenance Analysis[C]//NDSS. 2020.

[31] Hassan W U, Bates A, Marino D. Tactical provenance analysis for endpoint detection and response systems[C]//2020 IEEE Symposium on Security and Privacy (SP). IEEE, 2020: 1172-1189.

[32] Alsaheel A, Nan Y, Ma S, et al. {ATLAS}: A Sequence-based Learning Approach for Attack Investigation[C]//30th USENIX Security Symposium (USENIX Security 21). 2021: 3005-3022.

[33] Wang S, Wang Z, Zhou T, et al. threaTrace: Detecting and Tracing Host-based Threats in Node Level Through Provenance Graph Learning[J]. arXiv preprint arXiv:2111.04333, 2021.

[34] Hamilton, Will, Zhitao Ying, and Jure Leskovec. "Inductive representation learning on large graphs." Advances in neural information processing systems. 2017.

[35] Lv M, Dong C, Chen T, et al. A Heterogeneous Graph Learning Model for Cyber-Attack Detection[J]. arXiv preprint arXiv:2112.08986, 2021.

[36] Li Z, Cheng X, Sun L, et al. A hierarchical approach for advanced persistent threat detection with attention-based graph neural networks[J]. Security and Communication Networks, 2021, 2021.

[37] Li T, Jiang Y, Lin C, et al. DeepAG: Attack Graph Construction and Threats Prediction with Bi-directional Deep Learning[J]. IEEE Transactions on Dependable and Secure Computing, 2022.

[38] Zhao J, Shao M, Wang H, et al. Cyber threat prediction using dynamic heterogeneous graph learning[J]. Knowledge-Based Systems, 2022: 108086.

[39] Gehani A, Tariq D. SPADE: Support for provenance auditing in distributed environments[C]//ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing. Springer, Berlin, Heidelberg, 2012: 101-120.

[40] Kwon Y , Kim D , Sumner W N , et al. LDX: Causality Inference by Lightweight Dual Execution[J]. Acm Sigarch Computer Architecture News, 2016, 51(2):503-515.

[41] Perozzi B, Al-Rfou R, Skiena S. DeepWalk: Online Learning of Social Representations[C]. Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '14, 2014: 701-710.

[42] Grover A, Leskovec J. node2vec: Scalable feature learning for networks[C]//Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining. 2016: 855-864.

[43] Zhou J, Cui G, Hu S, et al. Graph neural networks: A review of methods and applications[J]. AI Open, 2020, 1: 57-81.

[44] Jia Y, Gu Z, Li A. MDATA: A New Knowledge Representation Model[J]. Springer International Publishing. doi, 2021, 10: 978-3.