Heliyon 10 (2024) e26317

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Systematic review and meta-analysis

Detecting lateral movement: A systematic survey

Christos Smiliotopoulos [a,*], Georgios Kambourakis [a], Constantinos Kolias [b]

[a] Department of Information and Communication Systems Engineering, University of the Aegean, Karlovasi 83200, Samos, Greece
[b] Department of Computer Science, University of Idaho, Idaho Falls, ID 83402, USA

ARTICLE INFO

Keywords:
Lateral movement
Advanced persistent threat
Attacks
Network security
IoT

ABSTRACT

Within both the cyber kill chain and MITRE ATT&CK frameworks, Lateral Movement (LM) is defined as any activity that allows adversaries to progressively move deeper into a system in seek of high-value assets. Although this timely subject has been studied in the cybersecurity literature to a significant degree, so far, no work provides a comprehensive survey regarding the identification of LM from mainly an Intrusion Detection System (IDS) viewpoint. To cover this noticeable gap, this work provides a systematic, holistic overview of the topic, not neglecting new communication paradigms, such as the Internet of Things (IoT). The survey part, spanning a time window of eight years and 53 articles, is split into three focus areas, namely, Endpoint Detection and Response (EDR) schemes, machine learning oriented solutions, and graph-based strategies. On top of that, we bring to light interrelations, mapping the progress in this field over time, and offer key observations that may propel LM research forward.

1. Introduction

Based on an initial point of compromise, typically through dropping malware or exploiting a vulnerability in a device or application, Lateral Movement (LM) involves moving deeper in terms of data or upwards in regard to access. Oftentimes, the attacker's goal is to remain in the system as an Advanced Persistent Threat (APT), attempting to gain as much loot as possible. What is more, in the Internet of Things (IoT) era, where more and more IoT devices penetrate critical sectors of our society, LM is gaining increased attention as attackers can exploit a plethora of IoT devices and transform them to attack vectors. That is, due to their provable security inefficiencies, IoT devices represent a particularly alluring target for a variety of threat actors aiming to move laterally and create persistence within any network, especially enterprise ones.

IoT seems ideal not only for obtaining initial foothold and persistent remote access, but also for gaining new permissions and user privileges in the breached environment; such devices are typically undersecured, they operate in a 24/7 basis, they are omnipresent and frequently in sensitive parts of the network, and, in many instances, not sufficiently updated or monitored, following an install-and-forget mentality [1]. In a typical scenario, a malicious binary can incorporate a scanner module to perform LM; in the presence of a vulnerability, the malware will propagate to, say, any discovered local printer, Wi-Fi bulb, or smart humidifier that has been identified to have minimal built-in protections. A high-level representation of the five-staged life-cycle of modern LM incidents is given in Fig. 1. The figure depicts not only the main stages through which an LM actor may escalate their privileges towards the exfiltration of data, but also the key tools and techniques as those are leveraged per stage.

* Corresponding author.
E-mail address: csmiliotopoulos@aegean.gr (C. Smiliotopoulos).

https://doi.org/10.1016/j.heliyon.2024.e26317
Received 14 October 2023; Received in revised form 25 January 2024; Accepted 9 February 2024

Available online 15 February 2024

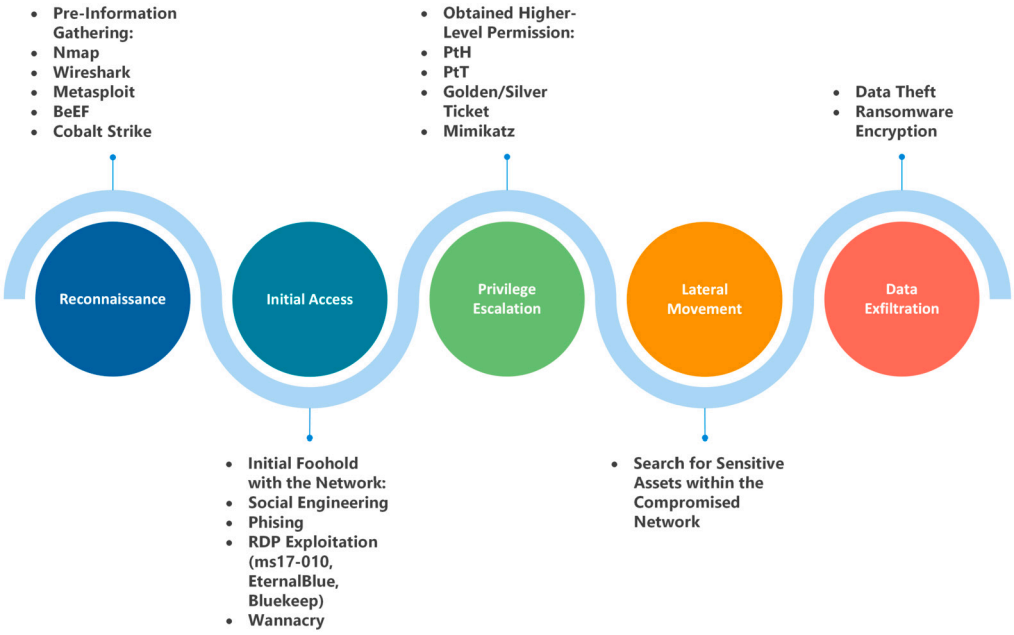

Fig. 1. Bird's eye view of an LM incident's life-cycle. Nmap: Network Mapper, BeEF: Browser Exploitation Framework, PtH: Pass the hash, PtT: Pass the Ticket, RDP Exploitation: Remote Desktop Exploitation.

LM is a key tactic in the context of modern threat modeling and associated cyberattack methodologies; the MITRE ATT&CK matrix for enterprise classifies LM under ID TA0008, identifying nine basic techniques. Taking into account the ever-growing presence of LM malevolent incidents and the outcomes of real-life cyber warfare operations, MITRE's database is continuously enriched with contemporary LM tactics. The Feb. 2022 attack against Viasat's KA-SAT network [2] is a prominent example of this situation. This attack led to a major discontinuance or total loss of the broadband satellite channels lengthwise the Ukraine terrain and several other bordering European countries. Furthermore, it is worth mentioning the references to the powerful Stuxnet [3,4] (S0603) and Mirai [5] (T1583.005) malware. The interested reader is also referred to the work of Makrakis et al. [6], which elaborates on the most impactful threats, including LM, related to industrial control systems and critical infrastructures.

However, even though a significance mass of contemporary research works address LM from either a defensive or offensive viewpoint, the literature is deprived of a concise, contemporary survey of schemes proposed to identify and track LM, mainly in the context of an Intrusion Detection System (IDS). In fact, although some studies [7–10] addressed LM as part of the general discussion on APT, they only did so in an abstract manner. In brief, Stojanovic et al. [7] focused on the description of the disposable feature selection and engineering techniques and the detailed presentation of the datasets leveraged by APT-dedicated IDS schemes. Tatam et al. [8] contributed a Systematic Literature Review (SLR) on the effectiveness of threat modelling frameworks regarding APT, while Talib et al. [9] presented a comparative study regarding the strengths and weaknesses of ML APT-related IDS schemes relied on the identification of APT's beaconing behavioral patterns. Among these studies, only Chen et al. [10] addressed the subject of APT models under the concept of IoT interconnected devices. In further detail, they summarized the most impactful models on the area in terms of anomaly detection, signature-based, and hybrid ML identification of threats. All in all, although the aforesaid studies addressed the subject of APT identification and detection, they only touched upon IDS destined for LM, following popular practices applied to IDS security concepts as those are presented in [11–16].

Given the above, the work at hand seeks to address this noteworthy literature gap by offering a systematic, complete overview of this field of research. From a methodological standpoint, the survey spans eight years, from 2015 to 2023, covering all major bibliographic databases. With reference to the LM detection methodology used, our analysis regarding the included literature works is split into three parts; Endpoint Detection and Response (EDR) log-based policy schemes, Machine Learning (ML)-powered schemes, and graph-based (GB) schemes. Additionally, given that IoT devices represent an enduring favorite among attackers attempting to move laterally, each of the above-mentioned parts includes a separate subsection dedicated to schemes applied to the IoT ecosystem. Equally important, we provide extensive discussions for each category of schemes identifying key methodologies and techniques, benchmark datasets, challenges, and shortcomings as well as interdependencies among the various studies.

The remainder of this paper is organized as follows. The next section details the methodology. Sections 3 to 5 discuss the included works per category of schemes, namely EDR log-based policy-based, ML-based, and GB, respectively. Section 6 offers key observations on the surveyed works, including prevailing methodologies, inadequacies and challenges. Section 7 wraps up and points out the main directions for future research. For easier guidance throughout the manuscript, a list of abbreviations is included at the end of the article.

Table 1
List of inclusion and preclusion criteria.

	Inclusion-Exclusion	Description
Inclusion	EDR policies	The EDR policies need to focus on LM detection. Works referencing pivoting and general APT identification are also included, as the criteria addressing them could find applicability to LM incidents too.
	Supervised ML models	Consider works that only describe the implementation of ML algorithms towards the exclusive identification of LM adversary events.
	Unsupervised ML models	Due to the limited variety of studies upon the specific category, studies related to LM events will be examined in conjunction with the corresponding ones affiliated to APT and pivoting incidents.
	GB models	Only the ones dedicated to LM identification literature are considered.
	LM-oriented testbeds	Any case-study scenario that involves a testbed along with the utilized dataset(s), if any, is examined and analyzed.
	Published papers written in English language.	-
Preclusion	Genre of literature	Book reviews - chapters, conference abstract - information chapters, mini blog reviews, editorials and online discussions, blog discussions, news.
	Not exclusively related to LM	Any other paper that could be categorized to the general category of generic defensive schemes or generalized IDS concepts has been excluded.

2. Methodology

As already mentioned in Section 1, the current work contributes an SLR, discussing the various available techniques concentrating on the detection of LM incidents. Such techniques are basically analyzed under the prism of small scale or corporate EDR policies, up to the more complicated advanced ML and GB algorithmic models. To this direction, equally important are the various testbeds, which most of the time rely on one or more benchmark datasets comprising either system logs or network traffic or both. That is, in most of the analyzed works in this survey, datasets are used to test the identification, evaluation and classification standards of pertinent threats for each LM detection model under the simulation of real-life scenarios.

As already outlined in Section 1, the scope of the SLR at hand can be outlined as follows:

- To outline the various EDR, ML algorithms, GB techniques, and datasets that have been incorporated in IDS schemes towards the identification of LM adversary incidents.
- To identify and classify the applicability and effectiveness of the various embodied IDS schemes per utilized identification technique, namely EDR policies and log-based concepts, supervised or unsupervised ML algorithms and GB models.
- To point out the types of the utilized testbeds, listing briefly the hardware and software tools, the equipment, and finally the various metrics through which the evaluation of the model's effectiveness has been conducted.
- To identify shortfalls, unattended issues and future potentials upon the field of LM incidents identification.

The SLR conforms to the methodology presented in [17], outlining the basic steps that should be used towards the design and execution of an SLR. Specifically, the steps that were followed at first stage under the present work have as follows:

- A variety of the most popular literature databases, namely Scopus, ACM, IEEE Xplore, Science Direct, and Springer Nature, were scrutinized in search of the related to LM works.
- The aforementioned databases were queried based on the combination of the following dedicated to the studied subject keywords: "lateral movement" **AND** ("endpoint detection and response policy" **OR** "EDR policy" **OR** "machine learning" **OR** "supervised machine learning" **OR** "unsupervised machine learning" **OR** "graph-based analysis" **OR** "graph based analysis") **AND** "security" **AND** ("intrusion detection systems" **OR** "IDS" **OR** "anomaly detection"). An additional query was used to scan the literature for relevant works focused on the IoT and Industrial IoT (IIoT) ecosystems: "lateral movement" **AND** "IoT" **AND** "IoT vulnerabilities" **OR** "IoT-dedicated datasets" **OR** "IoT IDS applicable schemes".
- The examined literature spans a period of nine years, i.e. from 2015 to 2023.
- The SLR took seven months to complete, i.e. from April to October 2023.

Second, as presented in Table 1, the selection process was aggregated into the list of the finally selected papers through a set of core inclusion and preclusion criteria.

3. Endpoint detection and response log-based policy schemes

3.1. EDR policy schemes

The current section briefly reviews the key pertinent literature on the subject of endpoint and log-based detection of LM events. We focus on the methodology presented in the corresponding literature, as that is related to event-oriented EDR reconnaissance of LM practices. Specifically, we concentrate on (a) the gathering of system and network related logs to the designated studied attacks, (b) the implemented EDR rules, (c) the success rate and imprint upon the efficient EDR procedure. To ease the parsing of the related literature, Table 2 summarizes the pertinent keystone characteristics in accordance with the studies embodied in this section. The various studies are chronologically arranged in ascending order. Finally, Table 3 recapitulates all the public benchmark datasets that were leveraged as a Proof-of-Concept (PoC) per case-study scenario.

Ki et al. [18] envisaged the disclosure of malware Application Programming Interface (API) calls norms under the concept of dynamic analysis and identification of anomaly behavioral signatures that may indicate LM activity. They deployed sequence alignment algorithms in an effort to extract common API call sequence patterns for malevolent functions generated from diverse categories of malware. According to the authors, what made ideal sequence alignment algorithms for malware's behavioral identification was their convenience in extracting similarities and patterns from diverse incrementally executed sequences, such as malware's API calls. To experiment upon their proposed methodology, the authors created a virtual MS Windows-based environment of hosts, while for hooking the various executables during runtime and monitor API calls, they relied on *Detour* hooking library. As it concerns sequence alignment algorithms, the ClustalX freeware library was imported to trace a malware dataset comprising more than 23 K samples. Approximately 2.7 K different API calls were identified during the experiment; these were grouped under 26 common categories based on the Microsoft Development Network (MSDN) list. To contribute with the extraction of the common API call sequence patterns among the executed malware, they introduced a custom "Longest Common Subsequences (LCSs)" formula, through which a signature based on the longest API call is created and attributed to each piece of malware. Despite the effectiveness of the proposed scheme, the formula was able to identify and categorize only user-level APIs and not the kernel-level ones.

The pioneering work of JPCERT/CC [19,20], comprises a full-blown research upon the classes of log files assembled and collected during the prosecution of LM techniques. Such techniques start with reconnaissance and identification of potential vulnerabilities on the targeted host, they continue with the infection of the target, to end up to the exploitation of critical information for malevolent purposes. JPCERT/CC first identified the standard schemes that adversaries follow in most of the cases during their targeting of network facilities. The first scheme is used to infiltrate fundamental network information, as those were collected from the infected network infrastructure using tools as *ipconfig* and *systeminfo*, or MS Windows core applications such as *Windows Event Viewer* and *Sysmon*. The second focused on the identification of valuable information regarding the network hosts, namely, OS version, account-domain characteristics, open ports, and many others, with a variety of tools like *net*. Finally, the most vulnerable host was targeted by means of a credential exploitation procedure with tools like *Mimikatz* and *pwdump*, towards spreading the infection to all network users. Such adversary patterns leave a far from negligible variety of diverse log files, which can be used by audit teams towards the identification of the existence of LM. To evaluate the aforesaid log files, the most prominent at that time LM practices, namely exploitation of remote services, password exfiltration, privilege escalation, capturing windows active directory database, and more, were applied on both the exposed servers and exploited clients affiliated to each offensive practice. Log-based files were gathered and classified via Sysmon [21] and MS Windows audit-policy. In Dec 2017, the work was updated with the recommendation of an MS Windows EDR audit policy, allowing the best for exploitation and research assortment of utile information associated to potentially impactfull LM events [20].

Mavroeidis et al. [22] experimented with the creation of a data-oriented risk categorization approach, that is relied on the extended aggregation and analysis of voluminous log files collected from Microsoft's Sysmon Security Information and Event Management (SIEM) tool. As a first step, they highlighted the importance of the notions of threat intelligence and threat information sharing, as part of a security ontology. The latter forms a collective cognitive library on which all the related to cyberthreat detection elements are imported, based on well-defined semantic concepts and their relationships. Recall that the creation of an ontology, allows the aggregation of information derived from various multipurpose sources in a singled knowledge database. According to the authors, that database may evolve over time for supporting as reasoning evidence the logical sequence of identifying and recording malevolent inconsistencies and events. Overall, the authors' work is twofold. First, they concentrated on the deployment of their presented Cyber Threat Intelligence Ontology (CTIO), which was established upon the Cyber Threat Intelligence (CTI) model, presented in [23]. CTIO may act as an ensemble of information from numerous multifarious sources, from low-level technical log-based events to more high-level observations and threat actors. This composite information grid aims to support the decision-making procedure under the concept of a security policy. To evaluate the robustness of the proposed ontology, CTIO was incorporated alongside a four-level threat assessment system based on Sysmon collected logs.

Berady et al. [24] engaged with the analysis of the distinct elements that contribute to the success or failure during the adversaries "Threat Hunting (TH)" EDR operation. Namely, the early identification of malevolent actor's areas of applicability and the qualification of their level of effectiveness in the shortest possible time are included among the most impactful factors for a successful EDR policy. Towards this direction, the authors presented an element-based TH model capitalizing on the collective notion that both offensive and defensive actors are expected to reciprocally apprehend one another. The analysis was done through a generally accepted intruder and defender standpoint, allowing the collation of the two contradictory perceptions regarding the same attack vector. The proposed model contributes in a twofold way to both adversaries and defending EDR Blue teams. That is, the opponents become more aware of the traces left during the execution of their attacking techniques, while Blue teaming is improved though the

identification and elimination of False Positives (FP) and the implementation of a realistic and calibrated log-based policy oriented around forensic and cybersecurity investigations. The authors' model was evaluated through experiments based on offensive and defensive perspectives. First, offensive practices were emulated experimentally upon the APT29 dataset, that is part of the Mordor Project of precollected data-driven event log-files of malevolent activities [25]. Note that the Mordor Project is based on the real-world APT29 threat, and was conducted according to a scenario designed and presented by MITRE's ATT&CK evaluation tactics list. The emulated offensive scenario comprised two stages; reconnaissance and compromise of the target via an injected toolkit, followed by the extraction of sensitive information from the targeted host. As it concerns Blue teams, they used Sysmon to monitor the targeted host's network activity and collect a two-days logset or both normal and attack events. The paper concludes with the proposition of an "Indicators of Compromise" list of elements. This contributes to the enhancement of the defender's base knowledge of the adversary's practices, enhancing seemingly the identification of threats in a preventive manner. We argue that, although this work examined only APT attacks against the dedicated APT29 subset, the underlying ideas are generic and can be applied to LM techniques as well.

Matsuda et al. [26] presented a Dynamic Link Library (DLL)-oriented methodology for detecting malevolent files upon logsets collected via Microsoft's Sysmon tool. To create a rich dataset, a great mass of DLL log samples were gathered and classified through different tools, namely, China Chopper, Mimikatz, PowerShell Empire, and HUC Packet Transmitter. The systematic observation of the aforesaid tools' special characteristics and traffic, revealed the presence of critical differential patterns within the pre-collected samples embodied within various versions of the MS Windows Operating System (OS). The proposed methodology concluded with the extraction of the "common DLL list", which comprises a collection of the most commonly loaded per malevolent tool DLL files, not dependent on the MS Windows OS version. Moreover, as an extension to that list of DLL files, a DLL logs-oriented detection method based on the open-source ELK Stack SIEM was introduced. According to the authors, the use of the ElasticSearch engine, allows real-time monitoring of the DLL information loaded on different hosts, leveraging the detection of potentially malicious files through their comparison with the aforesaid common list. The detection accuracy of the proposed methodology was evaluated by means of the same four tools. The generated logs were filtered via ELK Stack in a real-time manner, revealing a promising detection rate. Overall, it can be said that the enrichment of an EDR policy with DLL collected particles specifically focused on LM related Sysmon log samples, expands the detection accuracy and reduces false positives.

The authors in [27,28] also took advantage of the ELK Stack for the analysis of massive log records and the identification of malicious behavior. Precisely, Jaim et al. [27] deployed various LM techniques on a sandbox Windows-based networking virtual lab, producing massive log-based traffic related to LM events. The latter were generated from the execution of 24 LM case studies, namely PSExec, Mimikatz (Golden/Silver ticket, Pass the Hash, Pass the Ticket, etc.), PWDump, vssadmin, etc. The traffic was captured and filtered via ELK stack towards the final identification and detection of the behavioral characteristics of the tools exploited by LM techniques. The authors concluded with a presentation of a PoC of the most prominent EventID characteristics, as those were derived from each of the 24 executed LM tools and captured via ELK stack. On the other hand, Rajesh et al. [28] deployed an ELK stack-based massive data processing pipeline to collect, analyze, and identify anomalies on voluminous log-based data structures. The massive logsets were collected through *logstash*, while ElasticSearch was used for distinguishing and filtering any malicious traffic among the collected samples. The authors' experiments include the examination of Local Outlier Factor (LOF) for MySQL queries, numerical, and Boolean values, classification and regression detection. Although that paper incorporated ML notions that should be categorized in the cases of Section 4.1, the abstract representation of the reasoning upon the results renders it a good fit for the EDR category.

El Hadidi et al. [29] dealt with the detection of LM and APTs, as those are executed through any version of the Mimikatz tool, and via the identification of Mutex objects and loading DLL files through the numerous generated logs. According to the authors, APT groups follow five basic stages, namely infection, compromise, reconnaissance, credential theft, and LM. The authors concentrated on the fourth stage of the credential heist, as during its execution, the adversaries exploit the various dedicated for that purpose tools, such as Mimikatz, leading to the production of Mutex objects and DLL as evidences. To evaluate the robustness of their proposed scheme, they executed four different versions of Mimikatz on three different MS Windows Server editions, namely, 2008 R2, 2012 R2, and 2016 Standard Edition. Their findings, through the persistent analysis of the collected logs, revealed the presence of dedicated to Mimikatz Mutex objects, such as *conhost.exe*, that permitted increased accuracy levels towards the identification of LM.

Agarwal et al. [30] presented a custom-built LM and pivoting EDR tool dedicated to the Linux platform. Their tool expands on the *Osquery* and *Elastic* open-source ones, being capable to aggregate endpoint logs at a pre-configured common server. In this way, the log-oriented correlation between events stemming from various endpoint devices permits the investigation and detection of LM and pivoting incidents. According to the authors, the reason for developing their Linux EDR tool was derived by the limited variety of tools and policies available to enable monitoring and threat detection on Linux endpoint devices and servers. In general, the collection of logs in Linux environment may be accomplished through the use of *system logs* or *audit logs*, respectively; however, both these methods present disadvantages. The former cannot perceive user information for identifying the threat actor who caused the malicious event; as in case of privilege escalation, the attacker will always be captured as the root user. On the other hand, although the audit logs allow the identification of the perpetrator and provide relevant alerting, it still falls short in recognizing the endpoint device through which the attack was executed. In this respect, the major contributions of the EDR tool assemble to continuous monitoring and information gathering and analysis of the collected information in a historical archiving manner towards knowledge gaining and proactive detection of malicious actors through logs iteration. Nevertheless, the proposed tool is incapable to provide real-time incremental information regarding the events captured during the monitoring procedure. To evaluate the vigor of their contribution, the authors executed CVE-2019-2725 vulnerability of Oracle Weblogic server; through it, an adversary may gain control of a targeted host via the execution of arbitrary OS commands.

Niakanlahiji et al. [31] presented *ShadowMove*, a corroboration strategy for APT and LM techniques execution and host compromising. Although such techniques tend to be omnipresent and hard to be encountered, due to the consecutive evolution of attack tools and the variety of threat actors, they are limited to incorporate a set of legacy unique features related to their core functionality. More precisely, the execution of APT's LM has as prerequisites the creation of multiple new connections, the performing of authentications, or the requirement for process injections. Via such procedures, the adversary may stealthily move laterally within a networking facility, incrementally through privilege escalation. According to the authors, *ShadowMove* is designed to work stealthily, overcoming the aforementioned APT's shortcomings. Namely, the proposed methodology neglects the legacy features of APT's functioning in favor of a novel approach related to "socket duplication", through which a malicious process silently leverages TCP connections of benign equivalents. Additionally, the presented LM strategy permits APT threat actors to move stealthily among hosts in enterprise networks without being identified or detected by host-based, network level, or IDS systems. *ShadowMove* does not inject in any manner arbitrary malevolent code or commands on the benign processes. Instead, it passively monitors the targeted networking host's traffic to identify established connections. By this, the attacker avoids the creation of new connections, and therefore the need for new authentication. On top of that, *ShadowMove* is not restricted to the established connections only, but also to application protocols such as *WinRM* and *FTP*, both allowing the injection of remote commands on a compromised remote server. *ShadowMove* was successfully evaluated on MS Windows 10 and Ubuntu 18.0.4 with the majority of the stealthiness being presented on Windows systems instead of Linux. They finally deployed five top-notch antivirus products (McAfee, Norton, Webroot, Bitdefender, and Windows Defender) and two EDR systems (CrowdStrike Falcon Prevent and Cisco AMP) to confirm that the proposed methodology evades detection. Although this work is dedicated to implement an offensive methodology for compromising enterprise networks via APTs and LM techniques, we opt to include such schemes given that new offensive techniques do propel the corresponding defensive systems.

Smiliotopoulos et al. [32] presented a novel initialization rule-based EDR policy for the Sysmon SIEM dedicated to the detection of LM events within the MS Windows ecosystem. They experimentally elaborated on ordinal identified patterns related to LM events, following the execution of several state-of-the-art LM practices [33]. Specifically, regardless the complexity of any LM method, the consecutive experimentation with LM tactics in [33] revealed that, during the occurrence of such conventional LM incidents, the aggressor tends to repeatedly manipulate a small collection of penetration testing tools. Such tools strive for the reconnaissance, scanning and infiltration of the targeted host and the extraction of crucial information related to OS characteristics; typical tools to this end are ipconfig, systeminfo, Mimikatz, etc. Moreover, the adversary seeks the acquisition of credential information related to the clearance level of the targeted host with tools such as Mimikatz, pwdump, LazagneProject, or even its infection with dedicated to credential theft malware. To identify the fundamental technical particles of the most common LM attacks, the authors exploited a testbed with nine LM techniques of diverse targeting subject, including four variant of the "Exploitation of Remote Services" (T1210, T1021), "Pass the Hash" (T1550.002), "Pass the Ticket" (T1550.003), "Golden Ticket" (T1558.001), "Silver Ticket" (T1558.002), and post exploitation on stored passwords with Lazagne Project (T1555, T1003, T1552). Experimentation over the aforementioned LM techniques revealed numerous interrelated features that were incorporated as custom rules in Sysmon's config.xml file. Through the proposed policy oriented initialization, Sysmon acts more than a dedicated to LM SIEM, than just a log monitoring tool. Above that, the authors contributed the LM-oriented log-based LMD-2022 [34] corpus, comprising more than 870 K Sysmon event logs. To evaluate the proposed rule-based policy, an extensible Python .evtx file analyzer, dubbed PeX, was developed and assessed in terms of TP and FP rates over the LMD-2022 logset. The PeX tool was designed to serve as a parsing automatization tool for the handling of voluminous log files in .evtx format.

Noor et al. [35] addressed the subject of massive sets of data manipulation as part of raw logs auditing during forensic evidence content approximation and defense of information computing facilities. They provided a renewed pool of metrics that acts as a catalyst towards the quantification of audit log forensic evidence validity. These metrics can be used to measure the usefulness of logs under different test-case scenarios. In addition, they identified independently a preliminary entry point in the approximation design space techniques related to the elimination of, typically to system, activity related logs. More precisely, any log-oriented event that is related to typical system activity may be forced to approximation, while events identified as malicious are preserved with lossless fidelity for further analysis. This entry point approach was incorporated in *LogApprox* technique, which aims at the preservation of attack-related forensic evidence through logs. This is achieved through the creation of an extensive record comprising "process-to-process" and "process-to-network" dependencies, through which the forensic causative analysis process is aggregated. The efficiency of the proposed scheme was evaluated over the DARPA Transparent Computing Engagement 5 Data Release corpus [36], revealing a promising equilibrium among the reduction of redundant logs and the preservation of attack-oriented information. Moreover, *LogApprox's* applicability to identify and reconstruct patterns of adversary components was tested against the *Webmin* exploit APT case-study; the latter comprises a web-based Linux configuration tool that can be leveraged for LM over networking facilities.

Guri et al. [37] addressed the wider thematic area of LM techniques through the study of *USBCulprit*, which forms a USB-oriented APT, exclusively designed to breach corporate or governmental air-gapped networks. Namely, *USBCulprit* is classified in the malware category that incorporates LM, spreading, and data exfiltration characteristics, and acts through the exploitation of USB thumb drives. The authors experimentally evaluated the performance of the malware over various case study scenarios. Moreover, the numerous interrelated features of the APT were identified and isolated through reverse engineering techniques. Specifically, the source code for the malware's data collection and the air-gap exfiltration mechanisms were extracted; these include encrypted payloads, mutex objects, registry records, API calls, DLL interrelated files, etc. The authors concluded with the presentation of a number of proposed countermeasures, including policy, software, and hardware mitigation measures.

Mundt et al. [38] focused on the ever rising impact of network based attacks through infamous ransomware, such as WannaCry and NotPetya, including their elevated double extortion versions that in addition exfiltrate valuable data prior to being encrypted. First, they presented an automated ransomware mitigation concept that acts both as a first line of defense for the protection of private

and corporate networking facilities, and as a measure to cope with the constantly evolving techniques of adversaries to leverage new victims. Second, they implement in Python and evaluate a simulation tool that can dissimulate the most impactful ransomware attacks in favor of knowledge and experience acquisition in advance of the actual occurrence of an incident. To achieve their goal and identify the most current practices regarding data exfiltration, the authors relied on the MITRE ATT&CK [39] adversary tactics knowledge database. The finally presented automated mitigation concept was combined with an Extended Detection and Response (XDR) scheme and continuous Security Orchestrated Automation and Response (SOAR) practices, towards the automation and improvement of protective measures.

Mahmoud et al. [40] presented *APTHunter*, a detection system dedicated to the identification of APTs during the initial stages of a system's breach. APTs typically target impactful corporate and governmental targets of high value. The difference between *APTHunter* and other data analysis approaches of common provenance relies on the fact that the former pays attention to identifying the most prevalent to each APT threat indicators and inbound correlations on the very early stages of their existence. In this respect, it allows the identification of adversaries with precision and sensitivity on a real-time basis. The aforesaid characteristics seem rather effective compared to the already existing works, which considered APTs as short period threats that should be examined as a whole after all the enclosed stages are completed. *APTHunter* implements kernel audit logs as a reliable source of information related to system activities that could reveal signs of adversarial activity. Based on this information, *APTHunter* creates a flowchart of causal relationships, through which the already recognized indicators of each APT are used to unveil abnormal activity. *APTHunter* was evaluated via the "DARPA Transparent Computing" dataset [36] comprising the most important types of cyberthreats, particularly APTs. The methodology was applied over different OS platforms, presenting in all cases consistent and reliable detection rates in the early stages of an APT's appearance.

Park et al. [41] also recapped the subject of the timely identification and detection of APTs during the first stages of their manifestation. According to them, the traditional signature-based endpoint detection methods lack the adequate level of efficiency to adapt to the ever-changing environment of APT-oriented attacks. In this context, the authors focused on the introduction of a rule-based methodology that leverages well-known open-source tools to succeed the identification and detection of an APT adversary at the early stages of its appearance, all these under the concept of an EDR-oriented methodology. The proposed EDR scheme incorporates Google Rapid Response (GRR) and Auditbeat of Elasticsearch, as two open-source live incident response framework and logging tools, respectively. A dedicated to the dissimulation of an APT attacking environment simulator was also implemented for the analysis of the attack's stages based on the established EDR rulesets. Their EDR scheme was evaluated against the well-known APT29 dataset [25], comprising real-life APT scenarios.

Bajpai et al. [42] readdressed the domain of LM EDR through the scrutinization of the behavior of ransomware. To this end, they presented a ransomware detection and response framework that can be leveraged by organizations as a first barricade against ransomware. Through the examination of numerous ransomware test-cases, the authors extracted the most crucial for each malware's functioning characteristics, and incorporated them under a procedural basis to the policy of their proposed framework. Particularly, 25 notorious ransomware variants were examined through both static and dynamic behavioral analysis, including *REvil*, *LockBit*, *IDAFree*, *Cutter*, and *Binary Ninja*. All these samples were initially mapped through CISA's Decider tool for Time-Triggered Protocol (TTP) calls. That is, the samples were unpacked through static behavioral reverse engineering to reveal the source code of their payload and the pertinent intricacies of the different enclosed classes. On the downside, the finally proposed framework was presented theoretically without revealing any results of its efficiency based on a real-life scenario.

3.2. EDR schemes applied to the IoT ecosystem

Marquez et al. [43] presented a modern detection scheme for pivoting attacks, namely *APIVADS*. The latter analyzes APT pivoting tactics based on the flow of traffic in Small Office Home Office (SOHO) and corporate facilities, where IoT interconnected devices find great applicability. The presented scheme aims to overcome deficiencies that permit adversaries to infiltrate the targeted environment, and breach and gain access through parallel obfuscation of their pivoting traces. Towards that, the authors focused on the flow-based statistical analysis of the generated traffic during pivoting incidents, in an effort to identify anomaly indicators produced by malicious events. *APIVDAS's* core functions can be considered as a crossbreed of decentralized host-based pivoting detection with centralized practices towards the ensemble consideration of the extracted results. The efficiency and scalability of the proposed scheme was evaluated through empirical experimentation, presenting a promising rate of 98.54% successful separation of the examined traffic as normal or malicious, including the identification of TLS, HTTPS, DNS, and P2P events.

Xiao et al. [44] presented *SoK*, an EDR scheme oriented to secure emerging technological assets related to cloud/edge computing and IoT interconnected devices. The introduced policy converges two contemporary security practices, namely Zero Trust (ZT) systems and context-based access control. The authors examined these security practices for potential overlapping sectors and collaboration during their applicability, revealing the existence of an extended list of interrelated elements that can be used as incorporated features within a security EDR policy. However, the actual development and presentation of such a policy, as well as any evaluation based on a real-life attacking scenario, was left for future work.

Sarfaraz et al. [45] centered on the detection of LM APT malicious incidents over interconnected IoT devices via Linux Web servers. Towards this goal, they amended four out of six layers of the so-called "List of Pain" [46], namely Tactics - Techniques - Procedures (TTPs), tools, network, and host artifacts. Particularly, the aforesaid list forms a relational presentation of the indicators through which malicious activities can be identified. The list is presented in the form of a pyramid-like diagram, each level of which represents a number of delimiters dedicated to the identification of a specific gender of adversaries tactics. Above that, each level denotes the effort that needs to be paid ("pain") by an adversary to overcome the aforementioned delimiters. Towards the

quantization of the aforementioned policy of "pain", the authors leveraged the popular Unix-based OS instrumentation framework, called *osquery*. They, implemented the framework on a Linux OS machine to develop a dedicated OS heuristic-based log-query script that seeks for anomalies based on the aforementioned criteria, as those are related to initial connections, escalation of privileges, and above all, LM events.

The work of Weisman et al. [47] concerned with the improvement of Security Operation Center's (SOC) performance towards the identification of potentially aggressive incidents in SOHO and corporate networking environments that incorporate IoT interconnected devices through ad-hoc connectivity. Specifically, they revisited the concept of IoT integration in modern attack vectors (namely hardware, APT, LM, etc.), through the classification of the most contemporary adversary tactics that target IoT devices and relevant protocols. The specified characteristics of each attack were thoroughly recorded along with the related in each case-study scenario countermeasures. The work ends with the presentation of an EDR framework targeted to the enhancement of the automation security mechanisms of monitoring and detection.

Ricardo dos Santos et al. [48] addressed the subject of the ever-evolving cybersecurity concerns in Smart Building (SB) facilities. According to them, the integration of smart technology with building's management operations, such as automated power distribution and energy saving systems, namely Building Automation Systems (BAS), despite its great potential, enlarges the attack surface, leaving each resident exposed to adversaries. The article concentrates on BAS interconnected smart subsystems, including smart surveillance, lightning, and automated heating actuators. Above that, the authors examine the most substantial factors that make building smart devices an impactful attack vector for adversaries. The characteristics of several known and 0-day attacks were examined and analyzed within the paper under the concept of vulnerabilities exploitation of IoT specified operational protocols. The results were aggregated and provided as PoC with the presentation of an IoT devices-oriented malware. The malware is deliberately tuned to remain inactive for as long as the attacker wishes within the IoT network, and being activated as 0-day to compromise targeted devices.

Süren et al. [49] contributed a four-part vulnerability assessment methodology, namely *PatrIoT*, dedicated to the examination of IoT smart devices. Their methodology expands over four pillars of security evaluation: (a) the logical partitioning of the attack surface, (b) the aggregation and leveraging of the most distinct characteristics of the top 100 Common Vulnerabilities Exposures (CVEs) and Common Weakness Enumeration (CWEs) metrics for risk assessment, (c) an overall risk scoring system, and (d) a complete guideline for the execution of penetration testing over IoT devices and related software. The presented methodology underwent a two-year evaluation via the penetration testing of over 30 IoT devices. The authors demonstrated the penetration assessment results for seven of these devices, utilizing top-notch IoT software, such as Xiaomi Mi Home Security Camera, AI robot, and others. The acquired results revealed that the proposed evaluation finds applicability in a great range of interconnected smart devices, with quite promising results for security experts in the field.

4. Machine learning IDS schemes

The current section provides a condensed summary of the key pertinent literature on the subject of LM-specific ML IDS concepts. The concentration is on the methodology of each work regarding the implementation of ML models, either via supervised (SML) or unsupervised (UML) data manipulation techniques. Particularly, we focus on the identification of the core ML algorithm's function in terms of shallow and DNN applicability, the implementation of feature selection processes, and the leverage of publicly available benchmark datasets, if any, towards the evaluation of the effectiveness of each proposed IDS. To facilitate the parsing of the relevant literature, Table 4 recaps the chief features of every work included in this section. The various works are chronologically arranged in ascending order. Following the same chronological structure, Table 6 summarizes the public datasets leveraged as a PoC per case-study scenario.

4.1. Supervised learning based schemes

The work of Kaiafas et al. [51] was pioneering in the field of log-based identification and anomaly detection. The authors focused on the creation of an adaptable and efficient anomaly IDS methodology that leverages the combination of 10 generic log-based features and other eight custom-made, respectively. Both these set of features were extracted from the popular, publicly available Los Alamos National Laboratory (LANL) log-based corpus that was gathered between 1996 and 2005 [52]. The collected subsets were manipulated via various sampling techniques towards the facilitation of preprocessing and computational power issues with such voluminous data volumes. The authors followed a binary classification scheme exploiting popular SML techniques, namely Random Forest (RF), LogitBoost (LB) and Logistic Regression (LoR). The results were evaluated under False Positive (FPR) and False Negative (FNR) rates, while the malicious classified predictions were re-fed to the ensemble Majority Voting uniform weighted algorithm and reevaluated.

Bian et al. [53] presented a hybrid anomaly detection methodology, dedicated to the detection of LM-related vulnerabilities. Specifically, their scheme aims to protect targeted networking hosts during the early stages of their exposure to the threat. They leveraged the open access LANL dataset [52] towards the extraction of 35 custom composite features related to authentication events flow traffic. Both set of features were aggregated into a composite graphical representation of the log authentication traffic. SML techniques, namely Decision Tree (DT), RF, Linear Regression (LiR), Gaussian Naive Bayes (GNB) and Label Binarizer (LaBi), were exploited for assessing the proposed anomaly detection scheme. The aforesaid ML techniques were enhanced with under and oversampling techniques to improve their applicability due to the highly imbalanced nature of the LANL logs.

Table 2
Synopsis of the key features of the literature presented in Section 3 in a chronologically ascending format. The literature works preceded with a star (⋆) are exclusively related to EDR schemes destined to IoT.

Year	Dataset	Summary
2015 [18]	Malicia [50]	Reconnaissance and identification of malware API patterns specifically dedicated to call services related to anomaly behavioral signatures through the implementation of sequence alignment algorithms and Malicia-Project malware dataset.
2017 [19,20]	-	Examination of LM scenarios via the exploitation of prominent penetration testing tools. Compilation of data-driven log samples through Sysmon and MS Windows Audit tool, reconnaissance and detection of LM malevolent activity and proposal of a custom LM infiltration policy.
2018 [22]	-	Event-oriented LM identification, detection, and classification threat analysis framework established upon CTI ontology theory.
2018 [27]	-	Custom methodology dedicated to the generation of data-driven datasets via the aggregation of LM collected samples through Sysmon. Utilization of EKL stack towards the classification of LM oriented logs.
2019 [26]	-	Presentation of a DLL-based methodology for detecting LM infected samples collected by Sysmon [21]. Compilation of DLL specified delimiters list per attack tool, as those act independently of any MS Windows OS version.
2020 [29]	-	Employment of Mutex memory features alongside DDL delimiters towards APT exploitation of Mimikatz tool in LM scenarios.
2020 [31]	-	A strategy dedicated to the stealth execution of APT and LM techniques towards Windows-based and Linux host compromising over enterprise networks. The presented approach is related to “socket duplication”, which permits APTs to move “stealthily” among hosts in enterprise networks.
2020 [35]	DARPA Transparent Computing Engagement 5 [36]	Creation of a renewed pool of metrics towards the quantification of audit log forensic evidence validity and measure of the usefulness of logs under different test-case scenarios. Presentation of the “LogApprox” technique.
2020 ⋆ [47]	-	An EDR framework aiming at the enhancement of the automation security mechanisms of monitoring and detection under the concept of SOK operation.
2021 [24]	APT29 [25]	A dual-view threat identification and classification model built upon Sysmon log-samples.
2021 [28]	-	Data-driven forensic evaluation framework dedicated to RDP LM techniques. The framework is expanded upon samples collected with *Logstash*, utilizing ElasticSearch towards the classification of LM oriented logs.
2021 [37]	-	Introduced “USBCulprit”, an usb-oriented APT designed to breach corporate or governmental air-gapped networks. Interrelated features of the APT were identified and isolated through extensively persistent reverse engineering techniques. A number of countermeasures is given as well.
2021 ⋆ [48]	-	Focused on BAS interconnected smart subsystems, including smart surveillance, lightning, and automated heating actuators. An IoT-oriented malware was presented as PoC that acts as a 0-day, being able to leverage smart devices vulnerabilities.
2022 [30]	-	Custom-built LM and pivoting EDR tool dedicated to Linux OS platforms. It expands on the “Osquery” and “Elastic” tools, being capable of aggregating endpoint logs at a pre-configured common server.
2022 [38]	-	Automated ransomware mitigation concept that acts in parallel as a first line of defense towards the protection of private and corporate networking facilities. Implementation of a Python simulation tool of the most impactful ransomware attacks.
2022 [32]	LMD-2022 [34]	A rule-based, LM-dedicated EDR policy for the Sysmon tool. Elaboration on legacy LM patterns incorporated as policy features in Sysmon’s config.xml file. Introduced the LMD-2022 log-based corpus [34], comprising more than 870 K Sysmon event logs.
2022 ⋆ [43]	-	A detection scheme for pivoting attacks through the implementation of a hybrid methodology that crossbreeds decentralized host-based detection with centralized practices.
2022 ⋆ [44]	-	Presented *SoK*, an EDR scheme oriented to secure emerging technological assets related to cloud or edge computing and IoT interconnected devices.
2022 ⋆ [45]	-	Focuses on the detection of LM APT malicious incidents over interconnected IoT devices via Linux web servers. They implemented a heuristic-based log-query script that seeks for anomalies based on criteria extracted by [46], as those are related to initial connections, escalation of privileges, and LM activity.
2023 [40]	DARPA Transparent Computing [36]	Introduced *APTHunter*, a “kernel” audit logs based detection system dedicated to the identification of APTs during the initial stages of a system’s breach in real-time.
2023 [41]	APT29 [25]	Introduction of a rule-based, EDR focused methodology that leverages open-source tools to detect an APT adversary at the early stages of its appearance. The EDR scheme incorporates Google Rapid Response (GRR), Auditbeat of Elasticsearch, and an open-source simulator, and it was tested over the APT29 dataset [25].
2023 [42]	-	Ransomware detection and response framework. It incorporates the most impactful malware’s characteristics, as those were derived from the static and dynamic source code examination of 25 real-life ransomware samples. Time-Triggered Protocol (TTP) calls were also mapped through the CISA’s decider tool.
2023 ⋆ [49]	-	Presentation of a four-parted vulnerability assessment methodology dedicated to the examination of IoT smart devices. The methodology underwent a two-year evaluation via the penetration testing of over 30 IoT devices.

Table 3
Identified datasets incorporated in LM identification and detection EDR schemes. The works preceded with the ⋄ symbol utilize the APT29 and LMD-2022 datasets, which comprise real-life examples, as those are presented in Mitre's ATT&CK database [33].

Year	Organization	Dataset	Description
2015 [18]	IMDEA Software Institute	Malicia [50]	Malware-oriented, comprising both malicious and normal traffic as well as API calls.
2020 [35]	U.S. Department of Defense - Defense Advanced Research Projects Agency (DARPA)	DARPA Transparent Computing Engagement 5 [36]	APTs related malicious traffic combined with normal logs.
2021 ⋄ [24], 2023 ⋄ [42]	Mordor Intelligence	APT29 [25]	APT threats' evaluation logset, created based on Mitre's ATT&CK real-life examples.
2022 ⋄ [32]	Univ. of the Aegean	LMD-2022 [34]	Sysmon logset dedicated to LM.

Interestingly enough, during the same year (2019), Bian et al. revisited [53] with the work presented in [54]. This was done under a realistic case study scenario that emphasized on RDP-based LM techniques. Following the same methodology as in [53], the authors extracted two MS Windows host-based RDP-related event logs subsets from the LANL dataset [52], namely, *Comprehensive* and *Unified*, respectively. The two subsets were evaluated under supervised classification algorithms. The concept of identification and detection of LM malicious incidents in authentication logs via SML techniques was also revisited by Bian et al. in [55]. Moreover, the two above-mentioned RDP-based subsets were also leveraged under the same scheme presented in [53,54], and one year later through the same research group in [56]. The model's classification efficiency was evaluated against the two aforesaid LANL-originated subsets in terms of LM detection and overhead. Additionally, the attack frequency of the evaluated LM patterns was tampered by importing artificially generated noise and traffic variations towards the model's evaluation against adversarial LM scenarios.

Chen et al. [57] introduced a log-based anomaly IDS scheme, dedicated to the examination of Sysmon event logs via Supervised Shallow Classification (SSC) and Deep Neural Networks (DNN) techniques. In more detail, three popular algorithms, namely Support Vector Machines (SVM), Long Short-Term Memory (LSTM) and Recurrent Neural Network (RNN), were assessed against a set of experimentally collected Sysmon logs. On top of that, a generic set of custom features were presented based on the transformation of Sysmon *EventIDs* and evaluated in terms of TP and TN rates.

Narouei et al. [58] presented *DLLMiner*, a heuristic DLL-oriented malware detection methodology, which is also applicable to the general area of the identification of APT and LM activity. This scheme was constructed upon the results of the static analysis of portable executable's DLL features. As evident from the LM EDR policy presented in [32], *DLLMiner* incorporates significant characteristics regarding potentially malevolent behavior, without even executing the files.

Juwono et al. [59] examined the effectiveness of various ML models under the concept of intrusion detection over voluminous log files related to real-life malware infections. The authors created two sandbox environments, namely *Cuckoo* and *Anubis*, within which the *Weka* ML framework was exploited for assessing four popular shallow classification algorithms, namely SVM, DT, Nearest Neighbor (NNeighbor), and RF, over the legacy Malheur dataset [60].

Smiliotopoulos et al. [61], contributed a detailed methodology specifically dedicated to the identification of LM via the implementation of SML algorithmic models. They detailed the significant importance that human-driven feature selection may impose to the effectiveness of ML models, especially when those are combined with elevated preprocessing and feature importance processes. Several SML techniques were evaluated over an exclusively created for that purpose LM logset, namely *LMD-2023* [34]. The latter is unbalanced, comprising more than 1.8M (both normal and infected traffic) log samples collected through multiple, virtual and physical, MS Windows terminals by means of Sysmon SIEM. A variety of 10 base shallow estimators, one ensemble meta-estimator, and five DNN models were evaluated upon the multiclass classification of *LMD-2023's* traffic, yielding an F1-score of 99.41%. The same work additionally contributed an open-source tool called *ETCExp* for converting *EVTX* monitor log files to a *CSV* equivalent format.

He et al. [62] dealt with the presentation of a multidimensional methodology, designed to detect the LM behavioral stages of APT malware threats targeting the MS Windows Server Message Block (SMB) sharing protocol. The authors extracted and analyzed the most impactful "honeytoken" nodes related to the LM techniques executed after a malware's initial access to a targeted system in terms of an SMB-dedicated honeypot. Further, feature generation and engineering techniques were applied for creating a dataset to de used for assessing the performance of the core Neural Network (NN) supervised classification scheme. Specifically, the NN model enclosed a multilayered combination of hidden and convolution NN, namely TextCNN, LSTM, and FastText layers. The created dataset comprised by more than 10 K publicly available malicious malware APT files.

4.2. Unsupervised learning based schemes

So far, only a limited number of works considered unsupervised ML towards the classification of a vast diversity of collected logs exclusively related to LM. In general, the incorporated features were either included as generic to the originally analyzed logsets or aggregated from numerous interrelated nodes and edges included in the network topology in a custom-made manner.

Bohara et al. [63] examined the composition of an anomaly detection scheme that performs on top of ensemble UML techniques towards the identification of LM event traces on infected hosts. The LANL dataset [52] was amended through the graphical representation of the various communication nodes of the targeted hosts and via a GB model. This was done to conclude to the extraction of the related to the classification experiments features. Additionally, the authors exploited three ensembles of UML models, namely

Principal Component Analysis (PCA), k-means clustering, and Median Absolute Deviation-based outlier (MADO), evaluating the proposed scheme under a trace-related simulation case study.

Le et al. [64] manipulated four UML methods, namely Autoencoder (AE), Isolation Forest (IF), Lightweight On-line detection of anomalies (LODA), and Local Outlier Factor (LOF), for creating an IDS that targets the identification of LM insider attacks. The data were manipulated via preprocessing techniques to fit with Deep Learning (DL) models and contribute to reveal anomaly behavior. Various UML ensembles were evaluated against several state-of-the-art works exploiting popular benchmark datasets, namely CERT [65], LANL [52], and TWOS [66].

Inspired by the hybrid approaches presented in [63] and [64], Chen et al. [67] aggregated the existed theories of a network's nodes and vectors graphical representation mapping via network embedding with feature manipulation and preprocessing techniques towards the creation of composite features. The finally selected features were evaluated under a semi-supervised classification model via a Denoising Autoencoder DAE) algorithm. To this direction, the authors leveraged a balanced subset of the LANL dataset [52], namely *"The Comprehensive, Multi-Source, Cyber-Security Event"*. The experimental results were evaluated via FPR, TPR, Accuracy, and Precision metrics.

4.3. ML schemes applied on the IoT ecosystem

Noor et al. [68] introduced an IDS framework, that leverages shallow and DL techniques combined with semantic networking representation practices to build a cyber-threat identification scheme, applicable to APT targeted devices including IoT ones. Their concept relies on the contemporary notion of sharing cyberthreat Incident Reports (CTIR) towards en effective proactive countermeasure against the ever-evolving adversary tactics. In more detail, the proposed framework leverages the Mitre's ATT&CK taxonomy to search an entire network for the existence of malicious TTPs, which are then semantically interrelated through a TTP-Detection diagram. The recognized attack patterns are imported as features and evaluated though ML techniques, namely DT, RF, DL, SVM, and Bayesian Belief Networks (BBNs). The achieved accuracy of the aggregated models reaches an average of 92%, presenting a low FPR at the same time.

We observed that the relevant literature contains several works similar to [68]. All of them suggested a shallow or DNN classification scheme, originally proposed by others, with the goal to make them more efficient in terms of the identification and detection of aggressive APT or LM incidents. For instance, the work of Powel [69] presented an unsupervised learning model of LM detection, based on the role-based approach of clustering the system connections to remote hosts into distinct roles. That is, the successful identification of unusual process sequences or generic connections to remote hosts may reveal the existence of an adversary. Moreover, for discerning APT and LM activity, Imran et al. [70] evaluated multiple SSC and DNN models, namely RF, SVM, AdaBoost (AD), Stochastic-Gradient Descent (SGDC), Gradient-Boosting (GBC), MLP and LSTM. For balancing their dataset, the authors relied on the Synthetic Minority Oversampling (SMOTE) technique. A similar approach was presented by González-Manzano [71], in which numerous APT-related pieces of malware were firstly analyzed towards feature extraction, and secondly evaluated under a SSC scheme in an effort to distinguish them from regular malware. Although not specifically defined, the aforementioned works were evaluated on logs extracted from IoT or IIoT devices. For reasons of completeness, we opt to include such schemes in Table 4.

Arifeen et al. [72] contributed an automated micro-segmentation ML model destined to IIoT. Their model is dedicated to the identification and detection of LM events derived from an aggressor or malware over IIoT devices. This is achieved through the generation of micro-segments via the fragmentation of the various network blocks of traffic into normal or malicious. The performance of the presented model was evaluated over two IoT-oriented datasets, namely UNSW-ND15 [73] and IoTD20 [74], revealing a promising effectiveness to curb malicious actions due to LM or malware.

Koroniotis et al. [75] proposed a DNN forensic framework, dubbed as *Intelligent Satellite Deep Learning Network Forensic (INSAT-DLNF)*. Their scheme aims at the timely and efficient identification and detection of adversarial LM incidents targeting satellite smart networks. This is done through the consecutive training of a hybrid NN consisted from a Long Short-term Memory Recurrent Neural Network (LSTM-RNN) and a Gated Recurrent Unit (GRU) models. The efficiency of the presented network was assessed through the implementation of three popular IoT-oriented benchmark datasets, i.e., NSL-KDD [76], UNSW-ND15 [73], and Bot-IoT [77]. The derived evaluation results were compared with three SML models (ANN, NB, Association Rule Mining (ARM)) and two unsupervised algorithms (k-Means, Expectation-maximization).

The focus of Altunay et al. [78] was on the creation of an IDS scheme devoted to the protection and security of IIoT interconnected devices. They assessed three distinct IDS DNN models, namely CNN, LSTM, and the hybrid combination of both CNN+LSTM. Each model was evaluated against two datasets, i.e., UNSW-NB15 [73] and X-IIoTID [79], predicting normal and malicious records with a rate of 93.21% and 92.9% regarding binary and multiclass classification, respectively. The effectiveness of the conducted experiments via each of the three aforementioned models were compared both with each other and with other relevant studies in the field.

A similar approach with [78] was presented by Sarhan et al. [80]. The authors centered on the design of an IDS scheme over heterogeneous network data samples that targets the identification and detection of LM and APT. In detail, they aggregated cyberthreat intelligence practices from various independent organizations to build a collaborative federated learning IDS scheme, namely Threat Intelligence Sharing Scheme (TISC), towards the design and effective training of DNN ML models. Their proposal was evaluated over the UNSW-NB15 [73] and Bot-IoT [77], by means of three different models, namely federated, centralized, and localized. The federated model outperformed the localized in terms of F1-score and Accuracy, however still remained behind the centralized one.

Table 4
Summary of the key aspects of the works included in Section 4. The works are presented in chronological ascending order. The paper's titles preceded with the † or ‡ symbols represent SML or UML techniques, respectively. The literature works preceded with a star (★) are exclusively related to ML schemes destined to the IoT or IIoT.

Year	Dataset	Method	Summary
2015 † [58]	Malicia [50]	SSC (RF, NB [WEKA])	Heuristic DLL-oriented malware detection framework destined to the identification of LM and APT.
2015 † [59]	Malheur [60]	SSC (SVM, DT, RF, NNeighbor [WEKA])	ML IDS scheme integrated with *Cuckoo* and *Anubis* sandbox environments.
2017 ‡ [63]	LANL [52]	UML (PCA, k-means, MADO, Ensemble ML)	Automated unsupervised anomaly detection ensemble method for identifying LM traces on infected hosts.
2018 ‡ [67]	LANL [52]	UML (Network Embedding, Denoising Autoencoders)	Semi-supervised classification through denoising autoencoder of features stemming from network embedding and feature aggregation techniques.
2018 † [51]	LANL [52]	SSC (RF, LB, LoR, MV)	A log-based anomaly detection approach applied on generic and custom-made/engineered log features extracted from the LANL dataset.
2019 † [53]	-	SSC (DT, RF, LiR, GNB, LaBi)	Hybrid anomaly detection perspective regarding the identification of LM techniques on hosts during the early stages of their threat exposure. They extracted 35 composite log-based features from the LANL dataset, which were classified under SSC ML algorithms.
2019 † [54]	LANL [52]	SSC (DT, RF, LiR, GNB, LaBi)	Re-examination of the work presented in [53] under the prism of RDP-based LM.
2019 ★ [68]	-	SSC (DT, RF, SVM, BBN)	An IDS framework leveraging shallow and deep learning techniques. It is combined with semantic networking representation practices to build a cyberthreat identification scheme applicable to APT targeted devices, including IoT ones.
2020 † [57]	-	SSC (SVM) - NN	Shallow and DNN ML classification analysis on Sysmon log files.
2021 † [55]	LANL [52]	SSC (DT, RF, LiR, GNB, LaBi)	The same authors of [53] and [54] revisit the subject of LM detection through the classification of LM-related authentication logs with shallow ML techniques.
2021 ‡ [64]	CERT [65], LANL [52], TWOS [66]	UML (AE, IF, LODA, LOF, unsupervised ensembles)	Unsupervised ML detection of anomalies on user behavioral habits. Creation of unsupervised ML ensembles to evaluate the performance of the proposed scheme under various algorithmic combinations.
2021 ★ [72]	UNSW-ND15 [73], IoTD20 [74]	SSC (DT)	Automated micro-segmentation ML model destined to IoT devices integrated on industrial operational networks.
2022 ★ [75]	NSL-KDD, UNSW-ND15 [73], BoT-IoT [77]	DNN (LSTM, RNN, GRU), SSC (ANN, NB, Association Rule Mining (ARM), BBN), UML (k-Means, Expectation-maximization)	A DNN forensic framework centered on the detection of cyberattack incidents targeting satellite networks. This is done through the consecutive training of a hybrid NN consisted from LSTM-RNN and GRU models.
2022 ★ [69]	-	UML (SVC)	UML model based on the role-based approach of clustering the system connections to remote hosts into distinct roles.
2023 †‡ [61]	LMD-2023 [34]	SSC, DNN (MLP, CNN, LSTM, RNN, AE)	An SML-based methodology to detect LM. Human-driven feature selection combined with elevated preprocessing and feature importance processes. Unbalanced dataset of ≈1.8M Sysmon log samples. A publicly available tool to easily transform *EVTX* monitor log files to a CSV equivalent format.
2023 ★ [70]	-	SSC (RF, SVM, AD, SGDC, GBC), DDN (MLP, LSTM)	Performance evaluation of multiple SSC and DNN models to identify APT and LM infections via the SMOTE technique upon logsets.
2023 ★ [71]	-	SSC (RF, KNN), DDN (MLP)	Analysis of numerous APT-related pieces of malware towards feature extraction and evaluation under a SSC scheme. The goal is to tell apart legacy from APT-related malware.
2023 ★ [78]	UNSW-ND15 [73], X-IIoTD [79]	DNN (CNN, LSTM, CNN+LSTM)	A DDN-based IDS scheme targeting the protection of IIoT interconnected devices.
2023 ★ [80]	UNSW-NB15 [73], Bot-IoT [77]	DNN, LSTM, Federated Learning	A federated learning IDS scheme, aiming at the design and effective training of DNN ML models. The presented framework is applicable to the identification of LM and APT.
2023 ★ [81]	NSL-KDD [76], UNSW-NB15 [73] and Bot-IoT [77]	DNN (AE, CFBPNN)	A comprehensive DNN 0-day focused IDS framework, which combines existing feature mapping techniques with cascading models.
2023 ‡ [62]	-	DNN (TextCNN, LSTM, FastText NN)	Supervised DNN LM multidimensional framework targeting the identification of LM behavioral stages of APT malware related to the MS Windows SMB protocol.

Jayalaxmi et al. [81] also examined the heterogeneous nature of IIoT interconnected device's traffic, as a key cause that influences the effectiveness of IDS schemes against 0-days. To this end, they proposed a DNN 0-day IDS framework, called *PINGUS*, which combines existing feature mapping techniques with cascading models. Feature selection was conducted through the Denoising Autoencoder (AE) algorithm, while the classification and attack detection itself was accomplished via Cascade Forward Back Propagation Neural Network (CFBPNN). *PINGUS* was evaluated through five open access datasets, namely the Natural Gas Pipeline [82], WA Water Tank [83], NSL-KDD [76], UNSW-NB15 [73], and Bot-IoT [77]. The derived results were compared to the related literature on the subject, exceeding by 25% on average similar models.

5. Graph-based schemes

Although the literature is plentiful of works addressing GB IDS schemes, only few of them are dedicated to the identification and detection of LM. Table 5 summarizes the key elements of the most prominent published studies, while Table 6 recap the publicly available datasets leveraged as PoC per case-study scenario.

The work of Purvine et al. [84], described a LM detection methodology dubbed "dynamic GB reachability model (DGBR)". The scheme builds upon the definition of an impact-oriented metric on GB techniques. Particularly, the authors developed a custom algorithm towards defining the evolution of the multiple paths that an adversary could follow while moving laterally around the network nodes following the exploitation of a critical vulnerability. This model is the core of a network-level impact score, which is quantified based on the value and reachability score assigned to each network node that could be brached by adversaries. The effectiveness of the presented DGBR model was tested along with a case study scenario related to "Pass-the-Hash" technique over the LANL dataset [52]. The impact metric model was implemented in C++ code that was, however, not made publicly available.

A similar approach [84] was presented by Liu et al. [85]. Namely, the authors describe a GB IDS scheme, called *Latte*, which subserves two purposes. First, it handles the multilayered nature of voluminous data samples stemming from LM incidents, and secondly, it addresses the lack of knowledge regarding the tactics that an adversary might use. On top of that, *Latte* contributes in a twofold way to the identification and detection of LM attacks. That is, initially, the presented methodology identifies and marks host and user accounts and their various interconnections as nodes and edges, respectively. After an initially infected node is identified, it proceeds with the reconnaissance and detection of any other potentially compromised element(s). The identification and detection process continues with an algorithmic approach that leverages a remote file execution detector towards recognizing suspicious path anomalies caused by unknown LM attempts.

The work of Liu et al. [85] is regarded a milestone in LM identification, inspiring two more similar approaches on the same subject. More precisely, Ho et al. [86] presented *Hopper*, an LM identification tool that is fed by real-life generated log-based traffic. *Hopper*, tracks user's login activities and outlines their correlations among hosts on a GB algorithmic representation. The effectiveness of *Hopper* was tested on a 15-month custom dataset that was specifically injected with LM events. The results revealed that *Hopper* contributes to the detection of abnormalities among multiple logins, related to LM attacks.

Also inspired by the work in [85], Fang et al. [87] dealt with the efficiency of the existing IDS LM identification models. In this endeavor, they presented *LMTracker*, a LM detection scheme that leverages two custom GB algorithmic models, supported by advanced graph NN theory. The two algorithms are dedicated to the graphical representation of the LM-related paths and the unsupervised anomaly path detection based on a predefined threshold, respectively. The robustness of the presented NN graph model is supported via the implementation of features of various elements included in the captured log-based traffic, including users, computers, processes, etc. The features were pre-processed as nodes for constructing heterogeneous graphs that depict the various relationships among them. *LMTracker* was evaluated on two benchmark datasets, namely LANL [52] and CERT 6.2 [65].

Chen et al. [88] engaged with the fundamentals of how a Blue team may design, implement, and execute ML algorithmic models towards the perpetual hunting of APTs. Particularly, the provided a step-by-step tutorial of how security models should be built so that security teams avoid the challenges of threat's low signature footprint, the imbalanced nature of the tested datasets, and the lack of knowledge when a 0-day emerges. These guidelines were presented through two case study models, namely *Fuchikoma* and *APTEmu*. The former comprises an example of how autonomous threat hunting via Natural Language Process (NLP) NN and GB algorithms could be accomplished, while the latter is an emulator for APT3, as those are presented in Mitre's ATT&CK vulnerability list [33]. The presented models were evaluated and discussed over APT3 malware dataset, that was originally created by Haddadpajouh et al. [89].

5.1. Graph-based algorithmic schemes applied on the IoT ecosystem

Agmon et al. [90] dealt with the vectors that make IoT interrelated devices prone to LM techniques. These include the low-end nature of the device, the great diversity of vendors, the low security standards on which the device's firmware is developed, the device's location within the network, and its communication capabilities and supported protocols. In more detail, they presented a GB network-level LM risk quantification methodology, which can be evaluated via an incrementally constructed attack graph model comprising nodes and vectors derived from aspects related to the location and key communication features of the IoT device. This comes in the form of a depth-first branch and bound (DFBnB) heuristic search algorithm, aiming at the optimization of the structure of interconnected IoT devices through the mitigation of risk regarding full deployment and maximum utility.

Yang et al. [91] contributed a hybrid IoT-IDS model targeting the protection of the numerous data transferred frequently via IoT interconnected devices. Emerged on top of the inabilities of the already presented solutions in terms of robustness and real-time detection, the authors' work concentrated on the combination of data fusion practices on feature semantics level with the CNN bidirectional LSTM (BiLSTM) DNN algorithm towards the effective identification of anomalies within device's requests. In particular, the IoT extracted traffic is statistically analyzed to produce informational-enriched features related to semantic relationships. Moreover, multi-view feature fusion and alignment practices transform the extracted features into word vectors that in turn are injected into the CNN-BiLSTM DNN model to be classified. The presented model was evaluated on the NSL-KDD dataset [76] over 43 extracted features derived from four distinct attack categories, namely *Intrinsic*, *Content*, *Time-based*, and *Host-based*.

An alternative forensic investigation and traceability approach destined to LM and APT was presented by Wang et al. [92]. The authors presented a GB reconstruction methodology of APT malicious incidents in large networks, including IoT and mobile ones. They composed an APT alert correlation model targeting the elimination of FP indicators as those are generated by known

Table 5
Summary of the key aspects of the works included in Section 5. The works are presented in chronological ascending order. The literature works preceded with a ⋆ are exclusively related to GB schemes destined to the IoT or IIoT.

Year	Dataset	Method	Summary
2016 [84]	LANL [52]	DGBR model	Introduces a DGBR model that keeps track of the various adversarial paths that may be followed during the exploitation of vulnerabilities with LM techniques. Through it, the authors calculate a network-level impact score and conduct a PtH case study on the LANL dataset.
2018 [85]	-	GB model	Identification of an infected host as an anchor point to reveal other compromised hosts through forensic GB algorithms. Possible exposure of anomalies on rare paths through the detection of remote file execution.
2019 ⋆ [90]	-	DFBnB GB model	A GB, network-level LM risk quantification methodology. Through an augmented attack graph model, the proposed methodology can benchmark the location and communication characteristics of IoT devices.
2021 [86]	-	GB model	Graph-based framework for LM detection capitalizing on real-time generated logs. Tracking of login activity through graph depiction of implemented hosts, which eventually may expose LM activity.
2022 [87]	LANL [52], CERT [65]	GB model - NN	Introduces the *LMTracker* custom LM identification algorithm. This is a mixture of LM paths representation via heterogeneous graphs construction and anomaly detection through GB NN.
2022 [88]	APT3 [89]	NLP NN - GB models	A step-by-step methodology, regarding how security models should be built so that security teams avoid the challenges of APT threat's referring to low signature footprint, the imbalanced nature of the tested datasets, and the lack of knowledge when a 0-day is exploited.
2022 ⋆ [91]	NSL-KDD [76]	Hybrid (GB model - CNN-BiLSTM)	A hybrid IoT-IDS model combining data fusion practices on feature semantics level and the BiLSTM algorithm towards the identification of anomalies within requests produced by network devices.
2022 ⋆ [92]	CSE-CIC-IDS2018 [94]	GB model - Monte Carlo Tree Search (MCTS) heuristic algorithm	A GB reconstruction methodology of APT malicious incidents in large-scale networks, including IoT and mobile ones.
2023 ⋆ [95]	DAPT2020 [96], Edge-IIoT [97]	GB (GAN) model - CNN	A GAN NN APT behavioral analysis GB model comprising a feature analysis multidimensional algorithm.
2023 ⋆ [98]	-	GB model, eKNN, Blockchain authentication	A unified hybrid IDS model that aggregates the features of the bi-level optimization theory, the data privacy and security of Blockchain technology, GB attack's features reconstruction, and the classification robustness of adversary paths via the eKNN algorithm.
2023 ⋆ [99]	-	GB model, SML (SVM, RF)	An IIoT GB and SML IDS framework dedicated to the early detection of APT campaigns, including LM techniques.

or unknown APT scenarios identified by open-source or proprietary IDS tools. For the needs of the alert generation process, the authors incorporated the *Zeek* IDS and firewall, injected with various customized configuration scripts, such as those available by Mitre ATT&CK framework [93]. The presented model mined the aforementioned voluminous logset history and leveraged the Monte Carlo Tree Search (MCTS) heuristic algorithm to reveal the existence of advanced APT existence via incident's reconstruction. Their model was evaluated on the CSE-CIC-IDS2018 dataset [94], revealing promising results related to the elimination of FP alerts.

A similar approach to [92] was presented by Javed et al. [95], as part of their behavioral analysis effort to identify complex hidden APT scenarios over IIoT interconnected devices. According to the authors, while traditional ML methods present significantly promising results, it struggles to recognize in a real-time manner APT adversaries efforts to exploit cyber-physical IIoT systems. The proposed methodology leverages the robustness of novel GB schemes, specifically, the Graph Attention Neural Networks (GAN NN), that comprises a feature analysis multidimensional algorithm. Put simply, the GAN algorithm is combined with Convolutional NN (CNN) in an effort to improve the early detection of APT. The final model was evaluated on the DAPT2020 [96] and Edge-IIoT [97] datasets.

Sharadqh et al. [98] presented, *HybridChain-IDS*, an IDS model that aggregates the powerful features of the bi-level optimization theory, the data privacy and security of Blockchain technology, GB attack's features reconstruction, and the classification robustness of adversary paths via the Enhanced KNN (eKNN) algorithm, under a unified hybrid IDS concept. The presented model was assessed in terms of its capacity to identify three contemporary attacks, namely brute force, SYN flood, and phishing, over an IoT network comprised of 5% of malicious nodes. Although promising, the results still need to be appraised against voluminous corporate sets of data to prove the model's robustness.

An IIoT IDS framework named *RAPTOR* was presented by Kumar et al. [99]. This framework addresses the early stages identification and detection of APT adversarial campaigns, including LM, over IIoT corporate networks. Specifically, RAPTOR correlates data from multiple open-source origins for creating a multilevel APT campaign graph. This graph is followed by feature's vector selection and preprocessing processes for concluding to the final classification of the analyzed traffic via SML schemes. For the classification task, RAPTOR exploits two ML models, namely SVM and RF.

6. Analysis and discussion

With reference to Sections 3- 5, Tables 2- 6, and Fig. 3 it becomes apparent that the literature works on EDR-, ML-, or GB-based IDS solutions towards the identification, detection and elimination of LM follows an increasing trend from 2015 onwards. Naturally,

Table 6
Identified datasets incorporated in LM intrusion detection ML and Graph-based schemes.

Year	Organization	Dataset	Description
2015 [58]	IMDEA Software Institute	Malicia [50]	Malware-oriented. It includes both malicious and normal traffic, as well as the respective/matching API calls.
2015 [59]	Univ. of Mannheim, Germany	Malheur [60]	Legacy dataset that contains the recorded behavior of malware. It can be used for classifying and clustering malware behavior.
2016 [84], 2018 [51,67], 2019 [53,54], 2021 [55,64], 2022 [87]	U.S. Los Alamos National Lab (LANL)	LANL [52]	Publicly available logset comprising numerous collected normal and malicious MS Windows Event Viewer logs.
2021 [64], 2022 [87]	Carnegie Mellon University - Community Emergency Response Team (CERT)	CERT [65]	Synthetic insider threat dataset that includes both background and malicious actor's synthetic data.
2021 [64]	Harilal et al.	TWOS [66]	Publicly available dataset including both normal and malicious user interactions with each other. It was created during a double-role simulation gamified competition, specifically conducted to obtain normal and adversaries instances of insider threats.
2021 [72], 2022 [75], 2023 [78,80,81]	Univ. of South Wales Sydney	UNSW-ND15 [73]	Collection of 100 GB raw source files in pcap, BRO, Argus, and CSV format, including the relevant reports per file. It was created by the IXIA PerfectStorm tool at UNSW, aimed to provide a stable source of normal and malicious log files for security experts and researchers.
2021 [72]	Ullaf et al.	IoTD20 [74]	IoT botnet dataset comprising recorded raw normal and malicious traffic over IoT devices. It is provided as a reference point for the creation and benchmarking of IoT IDS related schemes.
2022 [88]	Haddadpajouh et al. - Cyber Security Lab	APT3 [89]	It contains more than 12 K samples of APT malware derived from the APT1, APT3, APT28, APT33, and APT37 APT groups.
2022 [92]	Univ. of New Brunswick - [Communications Security Establishment (CSE) & the Canadian Institute for Cybersecurity (CIC)]	CSE-CIC-IDS2018 [94]	Multipurpose IDS benchmarking dataset generated through the collection of numerous events and behaviors representations captured during the user profile's creation processes. It contains seven attack scenarios, namely Brute-force, Heartbleed, Botnet, DoS, DDoS, Web attacks, and insider's network infiltration.
2022 [75,91], 2023 [81]	Univ. of New Brunswick - Canadian Institute of Cyber-defense	NSL-KDD [76]	Popular benchmark dataset in the field of CIS and IoT security. It was produced as an upgraded version of KDD'99, however, it suffers from the lack of public related events.
2022 [75], 2023 [80,81]	Univ. of South Wales Sydney	Bot-IoT [77]	Benchmarking dataset comprising a combination of normal and botnet traffic logs. It is offered in both pcap (69.3 GB) and CSV (16.7 GB) format, incorporating logs from DDoS, DoS, OS and services scan, keylogging, and data exfiltration attacks.
2023 [78]	Muna Al-Hawawreh et al.	X-IIoTD [79]	It represents a collection of adversary's activities related to IIoT devices. The list of the included malicious traffic events includes generic scanning, vulnerability scanning, webSocket fuzzing, discovering Constrained Application Protocol (CoAP) resources, brute force attacks, reverse shell, man-in-the-middle, Message Queuing Telemetry Transport (MQTT) protocol cloud broker-subscription, data exfiltration, ransom distributed DoS, and others.
2023 [95]	Myneni & Chowdhary et al.	DAPT2020 [96]	It incorporates normal and APT-oriented malicious traffic. It was created via the recording of five days of continuous simulated network traffic.
2023 [95]	Ferrag et al.	Edge-IIoT [97]	It encloses IoT and IIoT application traffic derived from 10 IoT physical devices. It is designed to serve ML experimental concepts, both centralized and federated. It incorporates seven distinguished layers: cloud computing, network functions virtualization, Blockchain network, fog computing, software-defined networking, edge computing, and (I)IoT perception. It includes 61 features derived from diverse sources, including alerts, system resources, logs, and network traffic.

this reflects the augmenting increase in the complexity and frequency of cyberattacks, including APT, LM, pivoting, malware, and others. Inter alia, since 2020, the drift brings progressively the experimentation with benchmark corpora derived from IoT or IIoT interrelated devices, as those pose an immensely alluring attack vector for current and future threat actors.

A brief explanation for this ascendant trajectory stems from the extensive observation of how the works presenting LM-oriented detection policies [32] may be implemented as base knowledge to others dedicated to the presentation of LM IDS schemes [61]. As a characteristic example, the dedicated to LM events identification rules by means of Sysmon policies presented as Appendix "A" in [32] were implemented as the basic criteria criterion for labeling the LMD-2023 dataset, and also as fundamental during the data manipulation processes and setting of the SML models (Shallow and DNN) in [61]. Nevertheless, as already pointed out, the effectiveness of LM ML or GB IDS schemes is intertwined with the existence of robust log-based EDR procedures that will guarantee the in-time identification and alerting during potentially cybersecurity events in terms of EDR teams. The identified elements will be forwarded in turn as features to Blue teams for the second stage of the ML or GB algorithmic analysis. Especially for ML techniques, three different pools of available data manipulation techniques, namely supervised, unsupervised, and semi-supervised, together with the numerous algorithmic models available enlarge drastically the response vector's possibilities during the construction of IDS schemes, either shallow, DNN, GB, or some combination of them. To this end, the availability as public of IDS-destined security

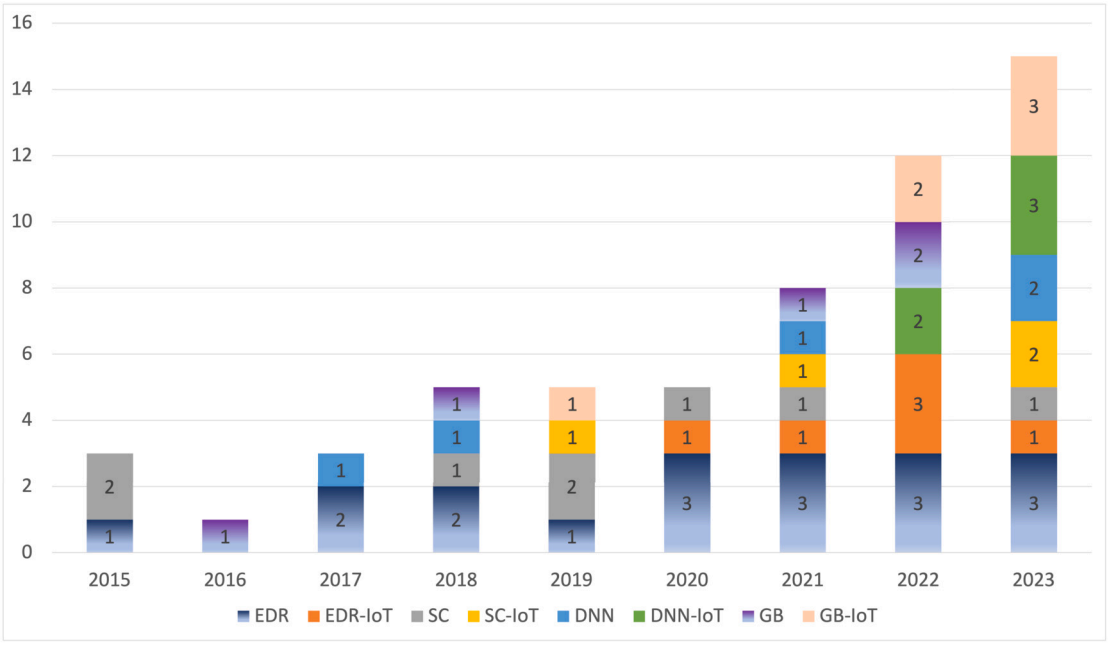

Fig. 2. Per year and category distribution of the published literature related to the subject of LM detection. Each of the four major categories, namely EDR, SSC, DNN, and GB, is equally depicted through its IoT equivalent subcategory, i.e., EDR-IoT, SC-IoT, DNN-IoT, and GB-IoT.

datasets is key to assessing the robustness and efficiency of the corresponding schemes, in terms of a sandbox environment, without jeopardizing the continuity of, say, critical infrastructures during a real-time evaluation.

The following three subsections offer a deeper discussion regarding the identified EDR, ML (SML or UML), and GB frameworks and models given in Sections 3 to 5, respectively. To further aid the reader in tracing the various works per respective section, Fig. 2 offers an aggregated, per year and category distribution of the published literature. The reader can also refer to Fig. 3 for obtaining an all-embracing view of each category, along with the possible interrelations among the works included in each category.

6.1. EDR log-based policy schemes

As observed from the first and second columns of Table 2, a great variety of 23 multipurpose EDR-related works were identified; all of them proposed some log-based and policy specific technique against the first stages of the existence of LM malevolent incidents. Specifically, all 23 works centered on the presentation of an EDR threat hunting framework, either dedicated exclusively to examine LM incidents or to APT threats, which among others enclose malware LM, pivoting, and other relevant techniques. No less important, five out of 23 works are dedicated to the endpoint analysis of threats applied on IoT or IIoT realms.

With reference to the fourth column of Table 2, six different subcategories of studies were identified among the general theme of EDR threat hunting: data-driven identification, vulnerability assessment through characteristics found in DLL directories of malware executable, APT's API calls, APT malware, APT and LM threats, and threat identification in the IoT or IIoT ecosystem. That is, a significant number of works from the presented literature emphasized the analysis of general APT malware [29,24,41,42] or LM equivalent techniques [31,37,32,40]. What is more, several researchers coped with the identification of APT and LM adversarial events through data-driven procedures and policies towards incident forensic analysis [19,20,22,27,35,28,30,38].

On the other hand, a DLL-oriented LM identification method targeting the detection of infected pieces of executable code, as those are evidenced in log-traffic collected through Sysmon, was presented in [26]. The subject of LM identification through the analysis of DLL files as part of executable snippets of code is currently limited in terms of dedicated works. To this direction, the work in [32] expanded [26] by proposing a richer set of DLL features towards the presentation of an improved LM log-based detection Sysmon policy. This substantially ameliorates the framework's effectiveness and robustness in terms of FPR and TPR. Above that, API-calls patterns were identified and analyzed in the methodology presented in [18]. With reference to the aforementioned 17 papers, it should be noted that only four of them centered on the presentation of a complete EDR framework; these are *ShadowMove* [31], *USBCulprint* [37], *PeX* [32] and *APTHunter* [40]. The 13 rest were dedicated to the presentation of general purpose APT and LM characteristics that could be implemented as key feature elements in audit-log SIEM tools and policies.

The six remaining contributions [47,48,43–45] included in the last six bottom lines of Table 2 tackle either the detection of intrusion attempts or the general endpoint security of IoT and IIoT interrelated devices implemented as core to SOHO or corporate networking environments. As opposed to the five aforementioned EDR categories, in the case of IoT, more than half of these works presented an autonomous and complete concept that could be leveraged as a general purpose framework for IoT device's security. These are *APIVDAS* [43], *SoK* [44], and *PatrIoT* [49]. Finally, a characteristic common to most works is that they neither construct

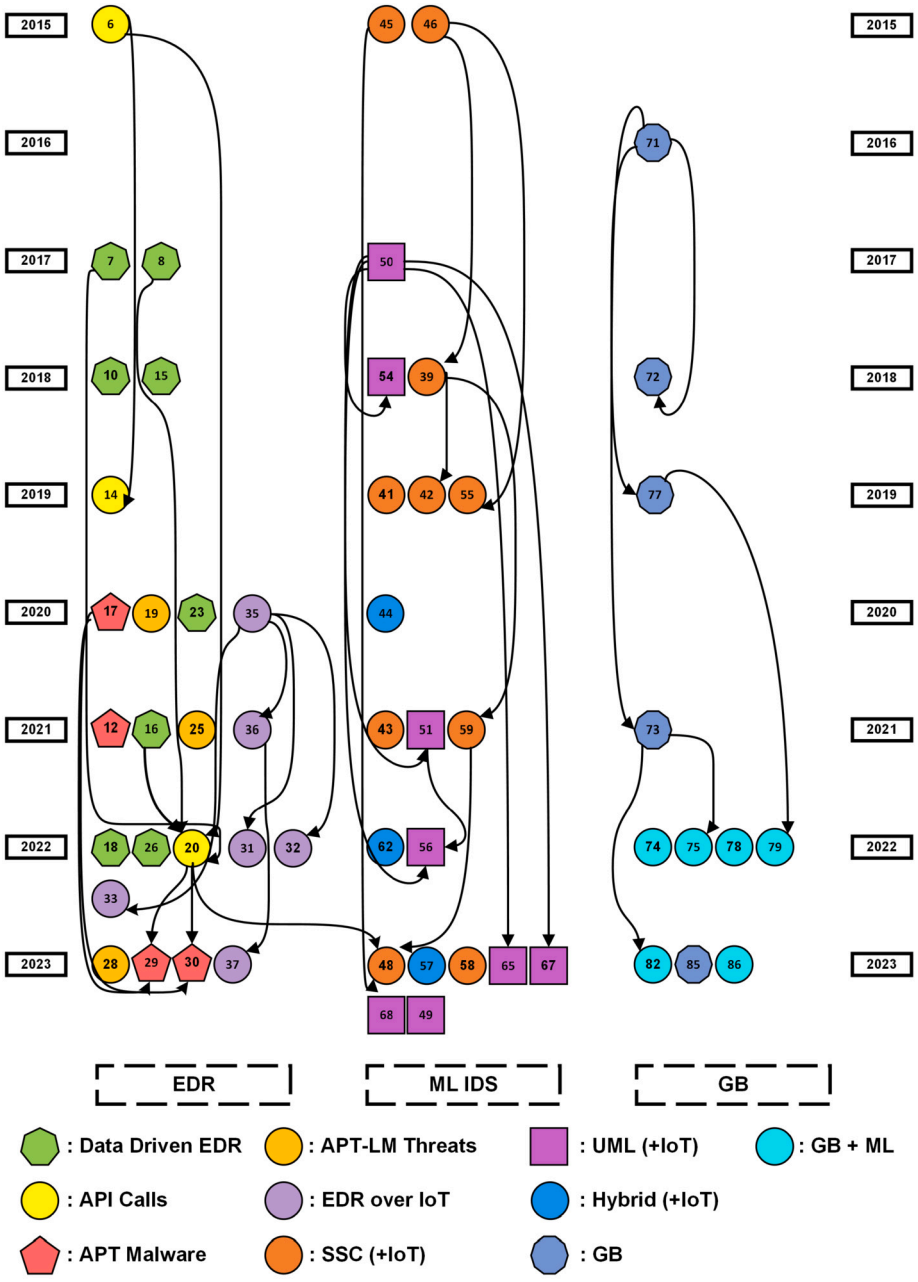

Fig. 3. Major EDR, ML (SML, UML) and GB approaches per category in chronological order. Arrows indicate other methods that possibly influenced each detection model. The IoT category is also enclosed as depicted in the index at the bottom of the Figure.

their own set of data logs and samples, nor provide adequate reference to regularization techniques and hyperparameter optimization steps, in case they were evaluated through some ML model.

The contributions in [18,35,24,32,40,41] are an exception to the aforesaid rule, creating or using an existing benchmark dataset for evaluation purposes, as presented in Table 3. Particularly, four datasets were identified; in chronological order, these are: Malicia [50], DARPA Engagement 5 [36], APT29 [25], and LMD-2022 [34]. Regarding these four datasets, the following observations can be made. Malicia is nowadays discontinued, the DARPA project, although still quite popular, is considered largely outdated. APT29 on the other hand is regarded as a descent benchmark dataset for the sandbox evaluation of EDR concepts. Finally, as its name preludes, the LMD-2022 collection is the first to our knowledge contemporary dataset that was created as a whole from LM Windows logs collected via the Sysmon SIEM tool; this renders LMD-2022 ideal for LM scenarios. Interestingly, LMD-2022 is publicly available in three different versions (in terms of the number of data samples) and is delivered in both .evtx and .csv format. Last but not least, all the works included in Table 2 but two have been published from 2018 onwards.

6.2. Machine learning IDS schemes

6.2.1. General IDS ML models

With reference to Section 4 and Tables 4 and 6, half of the presented works (10 out of 20) introduced a general purpose LM IDS model aiming at the identification and classification of LM adversarial events via contemporary ML algorithms. The remaining 10 works contributed ML-based schemes destined to the IoT and IIoT ecosystem [68,72,75,69–71,78,80,81]. As it concerns the former general LM IDS category, six contributions exploited for detection purposes SSC classifiers [58,59,51,53,54,57,55], whereas the rest three utilized UML techniques (either NN, DNN, or a combination of them) [63,67,64]. Still in the ML category, as it was also observed in subsection 6.1, most of the presented in Table 4 studies provide insufficient knowledge of the data samples upon which their ML models have been tested; naturally, this does not work in favor of replicability. As detailed next, the majority of these works utilized legacy datasets (such as LANL, KDD, Bot-IoT, etc.) or even created their own custom one for evaluation purposes. No adequate references are also provided regarding the regularization and hyperparameter optimization of the aforesaid ML models.

As already briefly mentioned, the great majority of the literature works referenced in Table 4 up to 2021, utilized either their own custom solutions related to logs collected via the MS Windows Event Viewer tool or their public equivalent *LANL* [52] dataset of multisource cybersecurity events. As it concerns logs gathered through a custom testbed, all the contributions except [61] did not deposit them in a public repository. However, the LANL [52] dataset is nowadays considered an almost outdated corpus due to the many years, since 2015, that it has to be updated with contemporary attacks, including novel LM techniques. More precisely, although exploited so far in six works [67,51,54,55,64,63] for evaluation purposes of the presented LM identification ML schemes, the small proportion of the malicious traffic enclosed, reasonably compels the majority of the authors to conduct artificial reproduction of malevolent samples for balancing the dataset. However, this manipulation of the dataset samples through over- or under-sampling methods, creates significant concerns regarding the validity of the conducted experiments. We argue that those techniques, say, *SMOTE*, *RandomUnderSampler*, and others, should be treated with great concern. That is, although the initially presented results may be rather promising, in the majority of the cases, the system logs or network traffic under analysis correspond to artificial, potentially unrealistic, samples. This in turn may yield false results regarding the benchmarking process and prediction rates, especially those related to FP classified events. This concern is rather obvious in the works [53–55,57], in which the authors, although presented an alternative dataset based on *LANL* [52], they disregarded to release it as public, impeding replicability.

Another outdated legacy dataset that has been introduced in two different versions, namely Malicia [50] and Malheur [60], was utilized in [58,59]; however, from 2017 onwards no other reference to it has been detected. As depicted in Table 4, since 2021, the ML experimentation's evaluation practices seem to be bettered, with the creation of datasets enriched with modern APT and LM techniques. Among the aforesaid, most of the works [72,75,78,80,81] exploited the UNSW-ND15 [73] dataset, while a few relied on the CERT and TWOS corpora, respectively. There is also the case of [61], in which the authors presented a dedicated to LM techniques enhanced version of the LMD-2022 dataset, coined LMD-2023 [34].

Another noteworthy aspect is that most of the presented works in Table 4, either SSC or DNN, except those in [63,51,53,57,55,61], omitted to properly justify their ML model's contribution via the presentation of their selected for the SSC or UML classification experiments features or the distribution of any repository with, say, *Python* or *R* scripts. Another drawback, which also hinders replicability, is that the majority of the authors neglect to mention the hyperparameters on which their ML models were built.

6.2.2. IDS ML models for the IoT or IIoT ecosystem

As already pointed out in subsection 6.2.1, nine out of 19 collected ML LM-related works contributed or evaluated ML-powered IDS schemes targeting the IoT or I IoT realm [68,72,75,69–71,78,80,81]. Nearly one-fourth of them (2 out of 9) presented SSC models [68,72], four more [75,71,70,78] utilized DNN algorithms for the binary or multiclass classification of the analyzed logs or traffic, while the rest three [69,80,81] presented hybrid multiplexed solutions of both SSC and DNN algorithmic models. From their thorough review in Section 4 and Table 4, almost half of them (4 out of 9) omitted to provide significant details regarding their feature selection or ML model construction processes or any hyperparameter tuning or optimization being applied. This is despite the fact that these proposals did exploit for the evaluation of their models prominent but still not LM-focused benchmark datasets, including UNSW-ND15 [73] and Bot-IoT [77]. On the other hand, the works in [75,70,78] presented in a rather analytical and documented way both their selected features and the algorithmic notion behind the hyperparameter's implementation in their ML models.

Moreover, two more papers categorized as IoT-dedicated [80,81], presented in full details their DNN model's hyperparameters initialization, omitting however any reference regarding feature selection. Overall, from the extended review of the nine aforesaid IoT or IIoT related studies, it is emphasized that the improvement in terms of the selection of ML models (namely the majority of the papers implemented novel DNN classification algorithms over multiclass schemes), the feature selection process, and the presentation of model's initialization is rather noticeable. It can be argued that this situation reflects how important the cybersecurity community considers the implementation of ML IDS knowledge into schemes which are intended for safeguarding IoT devices.

Finally yet importantly, regarding any datasets created for evaluation purposes of the presented ML schemes, almost half of the IoT-focused studies [72,75,78,80,81] (5 out of 9) employed contemporary corpora as those are presented in Table 6. In more detail, among the most recent IoT-oriented datasets stand out the IoTD20 [74], the Bot-IoT [77], the X-IIoTD [79], and the Edge-IIoT [97], all created from 2019 onwards. A noteworthy observation is that the work of Jayalaxmi et al. [81] was the first in which the evaluation of the proposed IoT LM IDS was conducted against five datasets, two of which were derived from corporate IIoT devices implemented in gas pipeline [82] and water tank [83] facilities.

6.2.3. Overall remarks

Given the above discussion regarding the various ML-powered LM IDS studies reviewed, it can be argued that their majority has been designed and evaluated on datasets that do not conform to a number of key aspects. Specifically, all the datasets up to 2018 (except those mentioned in the last part of subsection 6.2.2 related to the evaluation of IoT or IIoT LM IDS frameworks) do not meet important criteria related to (a) contemporary LM or general purpose APT and malware schemes, (b) public disposal of the selected for the experimentation process features, (c) the regularization and hyperparameter optimization and (d) multiclass labeling of the implemented samples instead of the straightforward binary one. As already point out, this situation deprives any chance for replicability in the same field.

It should also be mentioned that only three ([57,32,61]) out of the total 20 reviewed works in this category were conducted on the basis of Sysmon SIEM's collected log-based traffic, as a means to take advantage of its descriptive header's and event-oriented structure. The remaining ones either omitted to do so or were bounded to the limited legacy equivalent of MS Windows Event Viewer, limiting significantly the quantity and quality of the collected information. We consider that this shortcoming is mainly due to the lack (at least up to 2022) of a publicly available, open-source converter able to transform the extracted from Sysmon .evtx files to a .csv unlabeled equivalent, compatible for ML algorithmic experimentation. This justifies to an extent why most of the works up to 2018 relied on custom identification of LM events, as those were collected via the MS Windows Event Viewer. Therefore, most studies resorted to the pre-processed in comma-separated format legacy *LANL* [52] dataset.

Towards this extent, the work in [32] was the first to introduce an EDR policy solution dedicated to the endpoint identification and detection of LM techniques upon Sysmon raw log traffic at an EDR team level. In more detail, they evaluated the proposed EDR policy through the presentation of the *PeX* EDR tool [100], which parses raw .evtx Sysmon files and iterates over them based on the criteria imposed by its incorporated LM policy rules. Further, the work in [32] transitioned from EDR to an ML IDS equivalent [61] through the presentation of a dataset creation tool dubbed *ETCExp* [61]. Briefly, *ETCExp* serves researchers in the field to transform voluminous .evtx log files into compatible with ML algorithms unlabeled datasets in .csv format. For more details about the ETCExp tool, the reader is referred to [101].

6.3. Graph-based algorithmic schemes

With reference to subsection 5 and Tables 5 and 6 it can be argued that although the literature abounds of works dedicated to GB algorithms under general IDS schemes, only a limited number of them is devoted to the identification of LM techniques, not to mention that those presenting hybrid models for LM incident detection over IoT interrelated incidents outnumber the former. Particularly, a dozen of works have been recognized as the most related to the subject of GB model's presentation that classify incoming traffic based on the exploitation of model multidimensional feature analysis techniques over the construction of heterogeneous algorithmic graphs. Almost half of them [87,88,84,85] presented general LM identification concepts, while the rest were [95,98,99,90–92] dedicated to the hunting of LM incidents over IoT or IIoT interrelated devices.

What is more, 6 out of 12 works, namely [87,88,91,95,98,99], presented a hybrid framework, which leverages the powerful features of GB algorithmic techniques with the robustness of modern DNN models. This is rather obvious in the work that presented *LMTracker* [87], a custom LM and APTs identification and detection algorithm. Specifically, LMTracker's functionality hinges on the graphical representation of features as nodes and vectors through the construction of heterogeneous graphs, finally classifying the graphically represented features in a multidimensional way via advanced multilayered Autoencoders NN. Moreover, it is worth to be mentioned that the work in [98] was the only that mixed four pillars of contemporary technology paradigms, namely bi-level optimization theory, Blockchain, feature reconstruction into nodes and vectors via GB algorithms, and traffic anomaly detection and classification via the eKNN model. In terms of the identified datasets, the works in [84,87] exploited again the legacy LANL dataset [52] towards the evaluation of their general GB IDS schemes. Above that, only the authors in [88] employed the enhanced contemporary APT3 [89] corpus, comprising more than 12k of APT samples. Once again, since all the LM-related IoT or IIoT specified works were created from 2022 forward (except [90] in 2019) utilize contemporary datasets for their model's evaluation. A prominent example of this situation is the multipurpose CSE-CIC-IDS2018 dataset [94], comprising traffic related to more than seven attack scenarios over 420 infected corporate terminal, 30 servers, and all these organized ready for ML manipulation into 80 imbalanced features.

7. Conclusion

In the era of Internet of Everything (IoV), LM has evolved as a game-changing tactic in the quiver of cybercriminals, especially when it comes to APT. In this volatile ecosystem, the present article offers the first to our knowledge full-fledged, systematic review of literature works about schemes designed to timely identify and possibly counteract LM in the network perimeter or deeper in the network, mainly in the form of an IDS. We differentiate among three kinds of such defensive solutions; those that hinge on either graph algorithmic schemes, ML classification models, or EDR log-based policy strategies. Interestingly, for each category of solutions, an additional distinction is made between schemes proposed for general network domains, especially enterprise intranets, and IoT or IIoT ones. Just as important, an extensive and thorough array of essential observations and discussions is given, highlighting the utilized methodologies and datasets, as well as certain deficiencies and challenges.

With reference to Section 6, several issues exist that are addressed only partly or not at all, leading to the following key takeaways. First, all the published works up to 2022, although built on prominent EDR, ML, or GB solutions, largely neglected to present adequate information regarding the technical characteristics of each model's evaluation, the feature engineering and selection techniques

followed, and the dataset upon which the proposed framework's effectiveness was tested. Second, most of the studies relied on legacy dataset solutions, such as LANL [52] and UNSW-NB15 [73], which however are not LM-oriented. This observation largely applies to contributions from 2022 onwards, which leaned on contemporary, yet not-so-relevant datasets, namely Bot-IoT [77] and Edge-IIoT [97]. An exception to this rule is the works in [61,32]. Third, an essential yet disregarded subject in LM identification is the creation of unsupervised and federated learning models. That is, UML detection methods will provide to current studies an adequate level of applicability in real-life IDs scenarios, where specialized labeled datasets are scarce. On the other hand, federated models are expected to decentralize LM detection, also increasing privacy in terms of data transfer between the participating clients and ML server. Altogether, we anticipate that the work at hand will provide fundamental insight into this rapidly changing and interesting research branch, and fulfill the needs of a solid reference point for the interested readers.

Abbreviations List - The following abbreviations are used in this manuscript:

AD	AdaBoost
AE	Autoencoder
API	Application Programming Interface
APTs	Advanced Persistent Threats
BAS	Building Automation Systems
BBN	Bayesian Belief Networks
BeEF	Browser Exploitation Framework
CB	CatBoost Classifier
CERT	Computer Emergency Response Team
CFBPNN	Cascade Forward Back Propagation Neural Network
CIC	Canadian Institute for Cybersecurity
CNB	Categorical Naive Bayes
CNN	Convolutional Neural Networks
CoAP	Constrained Application Protocol
CSE	Communications Security Establishment
CSIRT	Computer Security Incident Response Team
CSV	Comma-Separated Values
CTIO	Cyber Threat Intelligence Ontology
CTI	Cyber Threat Intelligence
CTIR	Cyber Threat Incident Reports
CVE	Common Vulnerabilities Exposures
CWE	Common Weakness Enumeration
DARPA	U.S. Department of Defense - Defense Advanced Research Projects Agency
DFBnB	Depth-first branch and bound heuristic search algorithm
DGBR	Dynamic Graph-based Reachability Model
DL	Deep Learning Algorithms
DLL	Dynamic Link Library
DNN	Deep Neural Networks
DT	Decision Tree Algorithm
EDRPolicy	End-point Detection and Response Policy
eKNN	Enhanced K-Nearest Neighbor Algorithm
EoHT	Exploitation of Hashing LM Techniques (PtH, PtT, GT, ST via Mimikatz)
EoRS	Exploitation of Remote Services LM Techniques
ERS	Exploitation of Remote Services
ETCExp	evtx_To_CSV_Export Tool
ET	Extra-Trees
EVTX	Windows XML EventLog
FF	Feed-Forward
FNR	False Negative Rates
FPR	False Positive Rates
GAN	Graph Attention Networks
GB	Graph-based Model
GBC	Gradient-Boosting Classifier
GNB	Gaussian Naive Bayes Algorithm
GRR	Google Rapid Response
GRUs	Gated Recurrent Units
GT	Golden Ticket attack
IDS	Intrusion Detection System (IDS)

IF	Isolation Forest
IIoT	Industrial Internet of Things
IoT	Internet of Things
IoV	Internet of Everything
JD	Jackard Distance
KNN	K-Nearest Neighbors Algorithm
LaBi	Label Binarizer
LANL	Los Alamos National Laboratory Dataset
LB	LogitBoost Algorithm
L-based	Linear-based Algorithms
LCSs	Longest Common Subsequences
LGBM	Light Gradient Boosting Model Algorithm
LiR	Linear Regression
LM	Lateral Movement Techniques
LODA	Lightweight On-Line Anomaly Detection
LOF	Local Outlier Factor
LoR	Logistic Regression Algorithm
LSTM	Long Short-Term Memory
MADO	Median Absolute Deviation-based outlier detection
MCTS	Monte Carlo Tree Search
Min-Max	Min-Max Scaler
ML	Machine Learning
MLP	Multilayer Perceptron
MQTT	Message Queuing Telemetry Transport
MSDN	Microsoft Development Network
MV	Majority Voting Algorithm
NB	Naive Bayes Algorithm
Nmap	Network Mapper
NLP	Natural Language Process
NNeighbor	Nearest Neighbor
NN	Neural Networks
NT	Network Traffic
OHE	One-Hot Encoding
OS	Operating System
PeX - v1	Python_Evtx_Analyzer (PeX - v1)
PCA	Principal Component Analysis
PoC	Proof-of-Concept
PtH	Pass the Hash attack
PtT	Pass the Ticket attack
RDP Exploitation	Remote Desktop Exploitation
RNN	Recurrent Neural Networks
RF	Random Forest Algorithm
SB	Smart Buildings
SGDC	Stochastic Gradient Descent Classification Algorithm
SIEM	Security Information and Event Management logging tools
SLR	Systematic Literature Review
SML	Supervised Machine Learning
SMOTE	Synthetic Minority Oversampling Technique
SOAR	Security Orchestrated Automation and Response
SOC	Security Operation Center's
SOHO	Small Office Home Office
Softmax	Softmax activation function
SSC	Supervised shallow Classification
ST	Silver Ticket attack
ST-based	Stochastic - Based algorithms
STD	Standard Deviation
SVC	Support Vector Classification
SVM	Support Vector Machines
TH Model	Threat Hunting Model
TISC	Threat Intelligence Sharing scheme

TTPs Tactics - Techniques - Procedures
TXT Standard Text Document (Contains Plain Text)
UML Unsupervised Machine Learning
WEV Windows Event Viewer Application
XDR Extended Detection and Response scheme
XML Extensible Markup Language
ZT Zero Trust

CRediT authorship contribution statement

Christos Smiliotopoulos: Writing – review & editing, Writing – original draft, Methodology, Investigation, Data curation. **Georgios Kambourakis:** Writing – review & editing, Supervision, Project administration, Methodology, Data curation. **Constantinos Kolias:** Writing – review & editing, Supervision, Project administration, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] G. Kambourakis, C. Kolias, A. Stavrou, The mirai botnet and the iot zombie armies, in: 2017 IEEE Military Communications Conference, MILCOM 2017, Baltimore, MD, USA, October 23-25, 2017, IEEE, 2017, pp. 267–272.
[2] Viasat, Kasat - network cyber attack overview, 2022.
[3] J.P. Farwell, R. Rohozinski, Stuxnet and the future of cyber war, Survival 53 (1) (2011) 23–40.
[4] D. Kushner, The real story of stuxnet, IEEE Spectr. 50 (3) (2013) 48–53.
[5] C. Kolias, G. Kambourakis, A. Stavrou, J. Voas, Ddos in the iot: mirai and other botnets, Computer 50 (7) (2017) 80–84.
[6] G.M. Makrakis, C. Kolias, G. Kambourakis, C. Rieger, J. Benjamin, Industrial and critical infrastructure security: technical analysis of real-life security incidents, IEEE Access 9 (2021) 165295–165325, https://doi.org/10.1109/ACCESS.2021.3133348.
[7] B. Stojanović, K. Hofer-Schmitz, U. Kleb, Apt datasets and attack modeling for automated detection methods: a review, Comput. Secur. 92 (2020) 101734, https://doi.org/10.1016/j.cose.2020.101734.
[8] M. Tatam, B. Shanmugam, S. Azam, K. Kannoorpatti, A review of threat modelling approaches for apt-style attacks, Heliyon 7 (1) (2021) e05969, https://doi.org/10.1016/j.heliyon.2021.e05969.
[9] M. Abu Talib, Q. Nasir, A. Bou Nassif, T. Mokhamed, N. Ahmed, B. Mahfood, Apt beaconing detection: a systematic review, Comput. Secur. 122 (2022) 102875, https://doi.org/10.1016/j.cose.2022.102875.
[10] Z. Chen, J. Liu, Y. Shen, M. Simsek, B. Kantarci, H.T. Mouftah, P. Djukic, Machine learning-enabled iot security: open issues and challenges under advanced persistent threats, ACM Comput. Surv. 55 (5) (2022), https://doi.org/10.1145/3530812.
[11] M.A. Talukder, K.F. Hasan, M.M. Islam, M.A. Uddin, A. Akhter, M.A. Yousuf, F. Alharbi, M.A. Moni, A dependable hybrid machine learning model for network intrusion detection, J. Inf. Secur. Appl. 72 (2023) 103405, https://doi.org/10.1016/j.jisa.2022.103405.
[12] M.P. Novaes, L.F. Carvalho, J. Lloret, M.L. Proença, Adversarial deep learning approach detection and defense against ddos attacks in sdn environments, Future Gener. Comput. Syst. 125 (2021) 156–167, https://doi.org/10.1016/j.future.2021.06.047.
[13] S.I. Imtiaz, S. ur Rehman, A.R. Javed, Z. Jalil, X. Liu, W.S. Alnumay, Deepamd: detection and identification of Android malware using high-efficient deep artificial neural network, Future Gener. Comput. Syst. 115 (2021) 844–856, https://doi.org/10.1016/j.future.2020.10.008.
[14] L. Cui, Y. Qu, L. Gao, G. Xie, S. Yu, Detecting false data attacks using machine learning techniques in smart grid: a survey, J. Netw. Comput. Appl. 170 (2020) 102808, https://doi.org/10.1016/j.jnca.2020.102808.
[15] N. Faruqui, M.A. Yousuf, M. Whaiduzzaman, A. Azad, S.A. Alyami, P. Liò, M.A. Kabir, M.A. Moni, Safetymed: a novel iomt intrusion detection system using cnn-lstm hybridization, Electronics 12 (17) (2023) 3541, https://doi.org/10.3390/electronics12173541, https://www.mdpi.com/2079-9292/12/17/3541.
[16] M.N. Uddin, A.H.M.A. Hasnat, S. Nasrin, M.S. Alam, M.A. Yousuf, Secure file sharing system using blockchain, ipfs and pki technologies, in: 2021 5th International Conference on Electrical Information and Communication Technology (EICT), 2021, pp. 1–5.
[17] V. Kampourakis, V. Gkioulos, S. Katsikas, A systematic literature review on wireless security testbeds in the cyber-physical realm, Comput. Secur. 133 (2023) 103383, https://doi.org/10.1016/j.cose.2023.103383.
[18] Y. Ki, E. Kim, H.K. Kim, A novel approach to detect malware based on api call sequence analysis, Int. J. Distrib. Sens. Netw. 2015 (6) (2015) 1–9.
[19] J. Coordination, Detecting lateral movement through tracking event logs, https://www.jpcert.or.jp/english/pub/sr/20170612ac-ir_research_en.pdf, June 2017.
[20] J. Coordination, Detecting lateral movement through tracking event logs (version 2), https://blogs.jpcert.or.jp/en/2017/12/research-report-released-detecting-lateral-movement-through-tracking-event-logs-version-2.html, December 2017.
[21] M. Russinovich, T. Garnier, Sysmon v13.22, https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon, 2021, Retrieved June 28, 2021.
[22] V. Mavroeidis, A. Jøsang, Data-driven threat hunting using sysmon, in: Proceedings of the 2nd International Conference on Cryptography, Security and Privacy, 2018, pp. 82–88.
[23] V. Mavroeidis, S. Bromander, Cyber threat intelligence model: an evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence, in: 2017 European Intelligence and Security Informatics Conference (EISIC), 2017, pp. 91–98.
[24] A. Berady, M. Jaume, V.V.T. Tong, G. Guette, From ttp to ioc: advanced persistent graphs for threat hunting, IEEE Trans. Netw. Serv. Manag. 18 (2) (2021) 1321–1333, https://doi.org/10.1109/TNSM.2021.3056999.
[25] M. Labs, “apt29” mordor labs dataset collections, https://github.com/OTRF/detection-hackathon-apt29, 2020.
[26] W. Matsuda, M. Fujimoto, T. Mitsunaga, Real-time detection system against malicious tools by monitoring dll on client computers, in: 2019 IEEE Conference on Application, Information and Network Security (AINS), 2019, pp. 36–41.

[27] U. Jain, et al., Lateral movement detection using elk stack, Ph.D. thesis, University of Houston, 2018.
[28] P. Rajesh, M. Ismail, B. Ismail, M. Alam, M. Tahernezhadi, Network forensics investigation in virtual data centers using elk, in: 2021 International Symposium on Electrical, Electronics and Information Engineering, 2021, pp. 175–179.
[29] M.G. El-Hadidi, M.A. Azer, Detecting mimikatz in lateral movements using mutex, in: 2020 15th International Conference on Computer Engineering and Systems (ICCES), 2020, pp. 1–6.
[30] S. Agarwal, A. Sable, D. Sawant, S. Kahalekar, M.K. Hanawal, Threat detection and response in Linux endpoints, in: 2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS), 2022, pp. 447–449.
[31] A. Niakanlahiji, J. Wei, M.R. Alam, Q. Wang, B.-T. Chu, ShadowMove: a stealthy lateral movement strategy, in: 29th USENIX Security Symposium (USENIX Security 20), USENIX Association, 2020, pp. 559–576, https://www.usenix.org/conference/usenixsecurity20/presentation/niakanlahiji.
[32] C. Smiliotopoulos, K. Barmpatsalou, G. Kambourakis, Revisiting the detection of lateral movement through sysmon, Appl. Sci. 12 (15) (2022) 7746, https://doi.org/10.3390/app12157746, https://www.mdpi.com/2076-3417/12/15/7746.
[33] MITRE, Lateral movement - the adversary is trying to move through your environment, https://attack.mitre.org/tactics/TA0008/, July 2019.
[34] C. Smiliotopoulos, G. Kambourakis, “lmd” sysmon dataset collections, https://github.com/ChristosSmiliotopoulos/Lateral-Movement-Dataset–LMD_Collections.git, 2023.
[35] N. Michael, J. Mink, J. Liu, S. Gaur, W.U. Hassan, A. Bates, On the forensic validity of approximated audit logs, in: Annual Computer Security Applications Conference, ACSAC ’20, Association for Computing Machinery, New York, NY, USA, 2020, pp. 189–202.
[36] D.A.R.P. Agency, “darpa” transparent computing engagement 5 data release, https://github.com/darpa-i2o/Transparent-Computing, 2023.
[37] M. Guri, Usbculprit: usb-borne air-gap malware, in: Proceedings of the 2021 European Interdisciplinary Cybersecurity Conference, EICC ’21, Association for Computing Machinery, New York, NY, USA, 2021, pp. 7–13.
[38] M. Mundt, H. Baier, Threat-based simulation of data exfiltration towards mitigating multiple ransomware extortions, Digit. Threats Res. Pract. 4 (4) (2023) 54, https://doi.org/10.1145/3568993.
[39] MITRE, Mitre att&ck - the adversary is trying to move through your environment, https://attack.mitre.org/, July 2019.
[40] M. Mahmoud, M. Mannan, A. Youssef, Apthunter: detecting advanced persistent threats in early stages, Digit. Threats Res. Pract. 4 (1) (2023) 11, https://doi.org/10.1145/3559768.
[41] N.-E. Park, Y.-R. Lee, S. Joo, S.-Y. Kim, S.-H. Kim, J.-Y. Park, S.-Y. Kim, I.-G. Lee, Performance evaluation of a fast and efficient intrusion detection framework for advanced persistent threat-based cyberattacks, Comput. Electr. Eng. 105 (2023) 108548, https://doi.org/10.1016/j.compeleceng.2022.108548.
[42] P. Bajpai, R. Enbody, Know thy ransomware response: a detailed framework for devising effective ransomware response strategies, Digit. Threats Res. Pract. 4 (4) (2023) 57, https://doi.org/10.1145/3606022.
[43] R.S. Marques, H.M. Al-Khateeb, G. Epiphaniou, C. Maple, APIVADS: a novel privacy-preserving pivot attack detection scheme based on statistical pattern recognition, IEEE Trans. Inf. Forensics Secur. 17 (2022) 700–715, https://doi.org/10.1109/TIFS.2022.3146076.
[44] S. Xiao, Y. Ye, N. Kanwal, T. Newe, B. Lee, Sok: context and risk aware access control for zero trust systems, Secur. Commun. Netw. (2022), https://doi.org/10.1155/2022/7026779.
[45] S. Ahamed, L. Ramanathan, Real-time heuristic-based detection of attacks performed on a Linux machine using osquery, SN Comput. Sci. 3 (5) (2022) 405, https://doi.org/10.1007/s42979-022-01288-6.
[46] D.J. Bianco, Enterprise detection and response: “the pyramid of pain”, http://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html, 2014.
[47] D. Weissman, A. Jayasumana, Integrating iot monitoring for security operation center, in: 2020 Global Internet of Things Summit (GIoTS), 2020, pp. 1–6.
[48] D.R. dos Santos, M. Dagrada, E. Costante, Leveraging operational technology and the Internet of things to attack smart buildings, J. Comput. Virol. Hacking Tech. 17 (1) (2021) 1–20, https://doi.org/10.1007/s11416-020-00358-8.
[49] E. Süren, F. Heiding, J. Olegård, R. Lagerström, Patriot: practical and agile threat research for iot, Int. J. Inf. Secur. 22 (1) (2023) 213–233, https://doi.org/10.1007/s10207-022-00633-3.
[50] A. Nappa, M.Z. Rafique, J. Caballero, The malicia dataset: identification and analysis of drive-by download operations, Int. J. Inf. Secur. 14 (1) (2015) 15–33, https://doi.org/10.1007/s10207-014-0248-7.
[51] G. Kaiafas, G. Varisteas, S. Lagraa, R. State, C.D. Nguyen, T. Ries, M. Ourdane, Detecting malicious authentication events trustfully, in: NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium, 2018, pp. 1–6.
[52] A.D. Kent, Cybersecurity data sources for dynamic network research, in: Dynamic Networks in Cybersecurity, Imperial College Press, 2015.
[53] H. Bian, T. Bai, M.A. Salahuddin, N. Limam, A.A. Daya, R. Boutaba, Host in danger? Detecting network intrusions from authentication logs, in: 2019 15th International Conference on Network and Service Management (CNSM), 2019, pp. 1–9.
[54] T. Bai, H. Bian, A.A. Daya, M.A. Salahuddin, N. Limam, R. Boutaba, A machine learning approach for rdp-based lateral movement detection, in: 2019 IEEE 44th Conference on Local Computer Networks (LCN), 2019, pp. 242–245.
[55] H. Bian, T. Bai, M.A. Salahuddin, N. Limam, A.A. Daya, R. Boutaba, Uncovering lateral movement using authentication logs, IEEE Trans. Netw. Serv. Manag. 18 (1) (2021) 1049–1063, https://doi.org/10.1109/TNSM.2021.3054356.
[56] T. Bai, H. Bian, M.A. Salahuddin, A. Abou Daya, N. Limam, R. Boutaba, Rdp-based lateral movement detection using machine learning, Comput. Commun. 165 (2021) 9–19, https://doi.org/10.1016/j.comcom.2020.10.013.
[57] C.-M. Chen, G.-H. Syu, Z.-X. Cai, Analyzing system log based on machine learning model, Int. J. Netw. Secur. 22 (6) (2020) 925–933.
[58] M. Narouei, M. Ahmadi, G. Giacinto, H. Takabi, A. Sami, Dllminer: structural mining for malware detection, Secur. Commun. Netw. 8 (18) (2015) 3311–3322.
[59] J.T. Juwono, C. Lim, A. Erwin, A comparative study of behavior analysis sandboxes in malware detection, in: International Conference on New, Media (CONMEDIA), 2015, p. 73.
[60] K. Rieck, P. Trinius, C. Willems, T. Holzaff, Automatic analysis of malware behavior using machine learning, J. Comput. Secur. 19 (4) (2011) 639–668.
[61] C. Smiliotopoulos, G. Kambourakis, K. Barbatsalou, On the detection of lateral movement through supervised machine learning and an open-source tool to create turnkey datasets from sysmon logs, Int. J. Inf. Secur. 22 (2023) 1893–1919, https://doi.org/10.1007/s10207-023-00725-8.
[62] D. He, H. Gu, S. Zhu, S. Chan, M. Guizani, A comprehensive detection method for the lateral movement stage of apt attacks, IEEE Int. Things J. (2023) 1–1, https://doi.org/10.1109/JIOT.2023.3322412.
[63] A. Bohara, M.A. Noureddine, A. Fawaz, W.H. Sanders, An unsupervised multi-detector approach for identifying malicious lateral movement, in: 2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS), 2017, pp. 224–233.
[64] D.C. Le, N. Zincir-Heywood, Anomaly detection for insider threats using unsupervised ensembles, IEEE Trans. Netw. Serv. Manag. 18 (2) (2021) 1152–1164, https://doi.org/10.1109/TNSM.2021.3071928.
[65] C.R. Trzeciak, The CERT Insider Threat Database, Carnegie Mellon University’s Software Engineering Institute Blog, 2011.
[66] A. Harilal, F. Toffalini, J. Castellanos, J. Guarnizo, I. Homoliak, M. Ochoa, Twos: a dataset of malicious insider threat behavior based on a gamified competition, in: Proceedings of the 2017 International Workshop on Managing Insider Security Threats, MIST ’17, Association for Computing Machinery, New York, NY, USA, 2017, pp. 45–56.
[67] M. Chen, Y. Yao, J. Liu, B. Jiang, L. Su, Z. Lu, A novel approach for identifying lateral movement attacks based on network embedding, in: 2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom), 2018, pp. 708–715.
[68] U. Noor, Z. Anwar, A.W. Malik, S. Khan, S. Saleem, A machine learning framework for investigating data breaches based on semantic analysis of adversary’s attack patterns in threat intelligence repositories, Future Gener. Comput. Syst. 95 (2019) 467–487, https://doi.org/10.1016/j.future.2019.01.022.

[69] B.A. Powell, Role-based lateral movement detection with unsupervised learning, Intell. Syst. Appl. 16 (2022) 200106, https://doi.org/10.1016/j.iswa.2022.200106.
[70] M. Imran, H. Sidd, A. Raza, M. Raza, F. Rustam, I. Ashraf, A performance overview of machine learning-based defense strategies for advanced persistent threats in industrial control systems, Comput. Secur. 134 (2023) 103445, https://doi.org/10.1016/j.cose.2023.103445.
[71] L. González-Manzano, J.M. de Fuentes, F. Lombardi, C. Ramos, A technical characterization of apts by leveraging public resources, Int. J. Inf. Secur. (2023) 1–18, https://doi.org/10.1007/s10207-023-00706-x.
[72] M. Arifeen, A. Petrovski, S. Petrovski, Automated microsegmentation for lateral movement prevention in industrial Internet of things (iiot), in: 2021 14th International Conference on Security of Information and Networks (SIN), vol. 1, 2021, pp. 1–6.
[73] N. Moustafa, J. Slay, Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set), in: 2015 Military Communications and Information Systems Conference (MilCIS), 2015, pp. 1–6.
[74] I. Ullah, Q.H. Mahmoud, A scheme for generating a dataset for anomalous activity detection in IoT networks, in: C. Goutte, X. Zhu (Eds.), Advances in Artificial Intelligence, in: Lecture Notes in Computer Science, vol. 12109, Springer, Cham, Switzerland, pp. 508–520, 2020.
[75] N. Koroniotis, N. Moustafa, J. Slay, A new intelligent satellite deep learning network forensic framework for smart satellite networks, Comput. Electr. Eng. 99 (2022) 107745, https://doi.org/10.1016/j.compeleceng.2022.107745.
[76] C. I. of Cyber Security, NSL-KDD dataset, https://www.unb.ca/cic/datasets/nsl.html, 2020.
[77] N. Moustafa, The bot-iot dataset, https://doi.org/10.21227/r7v2-x988, 2019.
[78] H.C. Altunay, Z. Albayrak, A hybrid cnn+lstm-based intrusion detection system for industrial iot networks, Int. J. Eng. Sci. Technol. 38 (2023) 101322, https://doi.org/10.1016/j.jestch.2022.101322.
[79] M. Al-Hawawreh, E. Sitnikova, N. Aboutorab, X-iiotid: a connectivity- and device-agnostic intrusion dataset for industrial Internet of things, https://doi.org/10.21227/mpb6-py55, 2021.
[80] M. Sarhan, S. Layeghy, N. Moustafa, M. Portmann, Cyber threat intelligence sharing scheme based on federated learning for network intrusion detection, J. Netw. Syst. Manag. 31 (1) (2023) 3, https://doi.org/10.1007/s10922-022-09691-3.
[81] P. Jayalaxmi, R. Saha, G. Kumar, M. Alazab, M. Conti, X. Cheng, Pignus: a deep learning model for ids in industrial Internet-of-things, Comput. Secur. 132 (2023) 103315, https://doi.org/10.1016/j.cose.2023.103315.
[82] H. I. F.-L. D. (HIFLD), Natural gas pipelines dataset, https://hifld-geoplatform.opendata.arcgis.com/datasets/geoplatform::natural-gas-pipelines/about, 2019.
[83] W.W. Corporation, Water tank (wcorp-076), https://catalogue.data.wa.gov.au/it/dataset/water-tank, 2023.
[84] E. Purvine, J.R. Johnson, C. Lo, A graph-based impact metric for mitigating lateral movement cyber attacks, in: Proceedings of the 2016 ACM Workshop on Automated Decision Making for Active Cyber Defense, SafeConfig '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 45–52.
[85] Q. Liu, J.W. Stokes, R. Mead, T. Burrell, I. Hellen, J. Lambert, A. Marochko, W. Cui, Latte: Large-scale lateral movement detection, in: MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM), 2018, pp. 1–6.
[86] G. Ho, M. Dhiman, D. Akhawe, V. Paxson, S. Savage, G.M. Voelker, D. Wagner, Hopper: Modeling and Detecting Lateral Movement, 30th USENIX Security Symposium (USENIX Security, vol. 21, USENIX Association, 2021, pp. 3093–3110, https://www.usenix.org/conference/usenixsecurity21/presentation/ho.
[87] Y. Fang, C. Wang, Z. Fang, C. Huang Lmtracker, Lateral movement path detection based on heterogeneous graph embedding, Neurocomputing 474 (2022) 37–47, https://doi.org/10.1016/j.neucom.2021.12.026.
[88] C.-K. Chen, S.-C. Lin, S.-C. Huang, Y.-T. Chu, C.-L. Lei, C.-Y. Huang, Building machine learning-based threat hunting system from scratch, Digit. Threats Res. Pract. 3 (3) (2022) 20, https://doi.org/10.1145/3491260.
[89] H. Haddadpajouh, A. Azmoodeh, A. Dehghantanha, R.M. Parizi, Mvfcc: a multi-view fuzzy consensus clustering model for malware threat attribution, https://cybersciencelab.org/advanced-persistent-threat-apt-malware-dataset/, 2020.
[90] N. Agmon, A. Shabtai, R. Puzis, Deployment optimization of iot devices through attack graph analysis, in: Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, WiSec '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 192–202.
[91] X. Yang, G. Peng, D. Zhang, Y. Lv, et al., An enhanced intrusion detection system for iot networks based on deep learning and knowledge graph, Secur. Commun. Netw. (2022), https://doi.org/10.1155/2022/4748528, https://www.hindawi.com/journals/scn/2022/4748528/.
[92] Y. Wang, Y. Guo, C. Fang, An end-to-end method for advanced persistent threats reconstruction in large-scale networks based on alert and log correlation, J. Inf. Secur. Appl. 71 (2022) 103373, https://doi.org/10.1016/j.jisa.2022.103373.
[93] MITRE, Lateral movement - the adversary is trying to move through your environment, July 2019.
[94] CSE & the Canadian Institute for Cybersecurity (CIC), Cse-cic-ids2018 dataset, https://www.unb.ca/cic/datasets/ids-2018.html, 2018.
[95] S.H. Javed, M.B. Ahmad, M. Asif, W. Akram, K. Mahmood, A.K. Das, S. Shetty, Apt adversarial defence mechanism for industrial iot enabled cyber-physical system, IEEE Access 11 (2023) 74000–74020, https://doi.org/10.1109/ACCESS.2023.3291599.
[96] S. Myneni, A. Chowdhary, A. Sabur, S. Sengupta, G. Agrawal, D. Huang, M. Kang, Dapt 2020 - constructing a benchmark dataset for advanced persistent threats, in: G. Wang, A. Ciptadi, A. Ahmadzadeh (Eds.), Deployable Machine Learning for Security Defense, Springer International Publishing, Cham, 2020, pp. 138–163.
[97] M.A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, H. Janicke, Edge-iiotset: a new comprehensive realistic cyber security dataset of iot and iiot applications for centralized and federated learning, IEEE Access 10 (2022) 40281–40306, https://doi.org/10.1109/ACCESS.2022.3165809.
[98] A.A.M. Sharadqh, H.A.M. Hatamleh, A.M.A. Alnaser, S.S. Saloum, T.A. Alawneh, Hybrid chain: blockchain enabled framework for bi-level intrusion detection and graph-based mitigation for security provisioning in edge assisted iot environment, IEEE Access 11 (2023) 27433–27449, https://doi.org/10.1109/ACCESS.2023.3256277.
[99] A. Kumar, V.L.L. Thing, Raptor: advanced persistent threat detection in industrial iot via attack stage correlation, arXiv:2301.11524.
[100] C. Smiliotopoulos, K. Barbatsalou, G. Kambourakis, Python_evtx_analyzer (pex - v1), https://github.com/ChristosSmiliotopoulos/Python_Evtx_Analyzer.git, 2022.
[101] C. Smiliotopoulos, G. Kambourakis, evtx_to_csv_export tool (etcexp), https://github.com/ChristosSmiliotopoulos/Python-Projects-Repository.git, 2023.