IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 9 August 2022, accepted 4 October 2022, date of publication 13 October 2022, date of current version 21 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3214506

SURVEY

Application of Artificial Intelligence to Network Forensics: Survey, Challenges and Future Directions

SYED RIZVI[1], MARK SCANLON[2], (Senior Member, IEEE), JIMMY MCGIBNEY[1], (Member, IEEE), AND JOHN SHEPPARD[1], (Member, IEEE)
[1]Department of Computing and Mathematics, South East Technological University, Waterford, X91 HE36 Ireland
[2]School of Computer Science, University College Dublin, Dublin, D04 V1W8 Ireland

Corresponding author: Mark Scanlon (mark.scanlon@ucd.ie)

This work was supported by the SETU PhD Scholarship Programme.

ABSTRACT Network forensics focuses on the identification and investigation of internal and external network attacks, the reverse engineering of network protocols, and the uninstrumented investigation of networked devices. It lies at the intersection of digital forensics, incident response and network security. Network attacks exploit software and hardware vulnerabilities and communication protocols. The scope of a network forensic investigation can range from Internet-wide down to a single device's network traffic. Network analysis tools (NATs) aid security professionals and law enforcement in the capturing, identification and analysis of network traffic. However, in most instances, the sheer volume of data to be analyzed is enormous and, despite some built-in NAT automation, the investigation of network traffic is often an arduous process. Furthermore, significant expert time remains wasted in the investigation of a high frequency of false positive alerting from automated systems. To address this globally impacting problem, artificial intelligence based approaches are becoming increasingly employed to automatically detect attacks and increase network traffic classification accuracy. This paper provides a comprehensive survey of the state-of-the-art in network forensics and the application of expert systems, machine learning, deep learning, and ensemble/hybrid approaches to a range of application areas in the field. These include network traffic analysis, intrusion detection systems, Internet-of-Things devices, cloud forensics, DNS tunneling, smart grid forensics, and vehicle forensics. In addition, the current challenges and future research directions for each of the aforementioned application areas is discussed.

INDEX TERMS Network forensics, artificial intelligence, cybersecurity, digital forensics.

I. INTRODUCTION

Recent advances in artificial intelligence (AI) have aided in its adoption by a wide spectrum of organizations and technologies. The pervasiveness of AI can be seen in everyday devices such as smartphones, automobiles, smartwatches, and televisions [1], [2] as well as in various sectors such as healthcare, manufacturing industries, logistics, finance, entertainment, and smart cities [3], [4], [5], [6]. The area of cybersecurity and digital forensics has also been a major adopter of AI technologies for the detection and analysis of cybersecurity incidents.

The growth in cybercrime, along with the increasing relevance of digital devices to ''traditional'' crime investigation, has led to an increased demand for digital forensics. Digital forensics involves the investigation of digital data and devices in a manner that is legally acceptable in a court of law. It includes the processes of identification, collection, verification, analysis, interpretation, documentation, and presentation of digital evidence [7], [8], [9]. According to Cisco's latest ''Annual Internet Report'', by the end of 2023, the total number of network-connected devices will be 29.3 billion [10], and the average traffic volume handled by a system

The associate editor coordinating the review of this manuscript and approving it for publication was Bilal Alatas.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

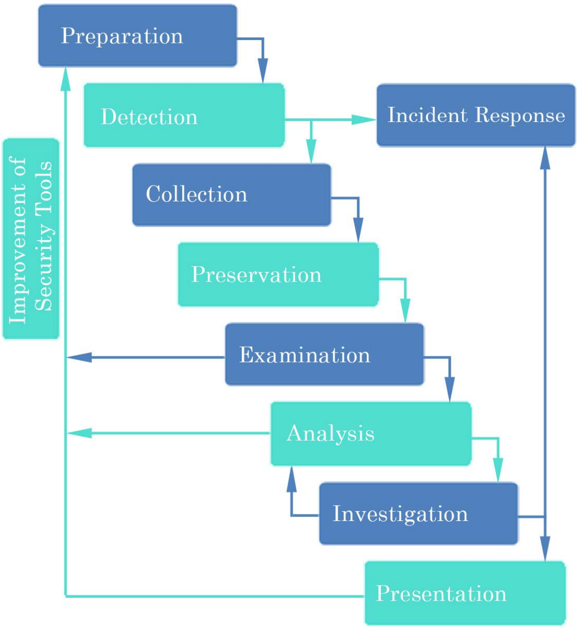

FIGURE 1. Network forensics process model (adapted from [16]).

will be approximately 50 GB per month [11]. In a multi-layered security model, AI can assist in the securing, monitoring and, when necessary, the investigation of a network. Network investigation often goes hand-in-hand with issues associated with big data [12]. Proactively monitoring security events through technologies such as AI-based Intrusion Detection Systems (IDS) can play a crucial role in the recovery of critical data [13]. This makes the use of AI a natural fit in assisting investigators in processing large volumes of data to find the pieces relevant to an investigation.

Digital evidence can be acquired from several sources, including locally on the device under investigation, in transit on the network, and from connected cloud environments [14]. Filesystems provide access to low level device data and deleted evidence. Operating systems record machine activity into files (such as system, application and event logs). These logs provide investigators with information on application use and facilitate the inference of how the device has been used. Network traffic can provide statistical, session, and alert data to the investigator. Cloud environments can hold evidence not stored locally on a machine, or shared data that can be correlated across multiple devices [14], [15].

Network forensics is a fundamental branch of digital forensics. The analysis of network traffic to investigate security incidents, data breaches and security policy violations is known as network forensics [17]. Network forensics evidence can be collected when communication is intercepted at the packet level. A process model for network forensic investigations can be seen in Figure 1. Network packets contain more than simply the routing information necessary for communication, in some cases network packet streams can also be used to recreate files that have been sent and received [18].

Network forensic systems are often used by organizations during the course of a digital investigation [7]. The network traffic data used in these systems can be collected in two ways:

1) *Catch it as you can* – this is a proactive approach where network traffic is continuously monitored and analysis is performed on-the-fly. This option is computationally intensive.
2) *Stop, look, and listen* – this is a reactive approach where following the detection of an attack, network traffic is captured for subsequent offline analysis. This option requires a significantly smaller memory footprint.

Network forensics enables self-protecting systems to analyze and comprehend the factors and effects of an unknown, previously encountered attack. In order to improve the performance of network forensic classifications, the monitoring/capturing system must keep a record of all traffic that passes through the network [19]. Examples of the most commonly encountered attacks resulting in a network compromise are botnets, buffer overflow attacks, business email compromise, cross-site scripting, cryptojacking, distributed denial of service (DDoS), DNS tunneling, Internet worms, man-in-the-middle, phishing, ransomware, and SQL injection. Further information on these various types of attacks and their classifications is discussed in several survey papers [20], [21] and the open web application security project (OWASP) maintains a list of the most prevalent web application exploits [22].

The paper is organized into 3 primary sections. The datasets publicly available for network forensics are presented in Section II. Section III provides an overview of the state-of-the-art of existing AI applications in network forensics. Section IV summarizes the current challenges and potential future directions in network forensics.

A. METHODOLOGY

This work employed an extensive combination of four bibliographic approaches to undertake state-of-the-art analysis for each of the phases of the study presented in this work: 1) The Snowball technique [23] to find the most relevant sources, 2) Pearl Growing [24] to capitalize on significant research gatherings, 3) Citation Searching [25] to locate further articles that cite popular pathways, and 4) PRISMA [26] to limit the number of publications to the most relevant for the selected topic. Highly cited publications were selected for inclusion in this paper as they outlined influential strategies. However, it was acknowledged that this concentration may lead to the omission of major new and developing approaches, thus recently published works were also included.

B. CONTRIBUTION

This paper is expected to serve researchers in the digital forensics and artificial intelligence communities to appreciate the state-of-the-art and the current challenges in network forensics. It is anticipated that this work will also facilitate researchers to further explore effective and efficient AI approaches to solve these emerging challenges.

This paper's primary contribution is a comprehensive survey of AI approaches for network forensics. It encompasses expert systems, deep learning (DL), ensemble learning, and hybrid learning. This can be used by network forensic researchers and practitioners to identify the most recent applications of AI in domains that employ network forensics, including IDS, vehicular networks, and smart grids. This paper will aid researchers to identify trending AI approaches adopted by researchers for network forensics with reference to time. Furthermore, this work provides an outline of current network forensic challenges and the future scope of research.

II. DATASETS FOR NETWORK FORENSICS

The data required by expert analysts varies greatly based on the task they are focusing on. High-quality, sufficiently sized data is always a requirement; whether it is a library of surveillance videos, diagnostic devices, text, financial data, or network traffic data. High-quality from a computational perspective also necessitates that it is adequately labeled, well-organized, and machine-readable. However, depending on the approaches being utilized, they will require varying amounts of data.

To utilize the maximum benefit of AI in any domain, the availability of public datasets is often the starting point – as AI model training and their accuracy highly depend upon realistic datasets [27]. Table 1, adapted from [28], shows publicly available datasets utilized for network forensics and are discussed in further detail below.

1) CSE-CIC-IDS2018 DATASET

The CSE-CIC-IDS2018 [34] dataset was introduced by Canada's Communications Security Establishment (CSE) and the Canadian Institute for Cybersecurity (CIC) for IDS in 2018. The 2018 dataset is much larger when compared to CICIDS-2017 dataset. Network traffic was collected for 10 days to create a dataset of 16,233,002 packets. 17% of the data contains attack traffic including DDoS (7.786%), DoS (4.031%), brute-force (2.347%), botnet (1.763%), infiltration (0.997%) and web attacks (0.006%). An extensive analysis on the dataset was done by Leevy and Khoshgoftaar [72].

2) CICIDS-2017 DATASET

This dataset [39], provided by the Canadian Institute of Cybersecurity, comprises information recorded from Monday, July 3, 2017, until Friday, July 7, 2017, and is stored in eight files. It incorporates sophisticated attacks such as brute-force SSH, DoS, Heartbleed, web attacks, infiltration, botnet, DDoS, and brute-force FTP. CICIDS-2017 meets all the characteristics of real-world attacks. The CICFlowMeter utility was used to extract 83 network flow characteristics from produced network traffic, consisting of 15 distinct classes with 2,830,540 distinct instances in total. Furthermore, the CICIDS-2017 dataset isolates the subjective behavior of 25 users depending on protocols like FTP and HTTPS. However, class imbalance is one of the significant downsides with this. An extensive analysis of CICIDS-2017 is presented by Sharafaldin et al. [73].

3) BOT-IoT DATASET

This dataset contains 73,370,443 instances, including a large number of attack categories: DoS, DDoS, reconnaissance (OS fingerprinting, service scanning), and information theft (data exfiltration, keylogging). Koroniotis et al. [71] provides an overview of the Bot-IoT dataset, which contains 29 features. This provides novelty in the context of IoT when compared to earlier datasets. To replicate the network behavior of Internet of Things (IoT) devices, the researchers used the *Argus* security tool. The MQTT protocol, which is quite popular in IoT, is used to connect machine-to-machine communications. The testing platform is deployed based on five different IoT scenarios.

4) TON_IoT DATASET

TON_IoT [69] is one of the most recent datasets created by UNSW Canberra IoT Labs and their Cyber Range specifically for IoT networks. A medium-scale IoT network provided a heterogeneous dataset. The primary goal of TON_IoT is telemetry data and characteristics of industrial IoT (IIoT)/IoT services. The label feature of TON_IoT specifies whether an observation is normal or malicious, while the type feature identifies the attack subclasses for multi-class classification issues. Scanning, data injection, DoS, DDoS, ransomware, backdoor, password cracking attack, cross-site scripting, and meet-in-the-middle are among the captured attacks.

5) CIC-DDoS2019 DATASET

The CIC-DDoS2019 dataset [29] is the most recently constructed dataset released by the Canada Cyber Security Institute in 2019. The information was collected over two days to construct an appropriate dataset. In total, it consists of 50,063,112 instances, where 50,006,249 instances are DDoS attacks, along with 80 features. An adequate test context was built with limitations of earlier datasets in mind. CIC-DDoS2019 includes the results of network traffic analysis (NTA) in addition to regular and recent DDoS attacks that are similar to genuine data (*PCAP*). CICFlowMeter-V3 is used to analyze network traffic since it has labeled traffic. There are several types of DDoS attack, such as port-map, NetBIOS, LDAP, MSSQL, and so on.

6) CIDDS-001 DATASET

CIDDS-001, a flow-based dataset, was released in 2017 by Ring et al. [74]. The data was collected through *OpenStack* and external servers for four weeks. It is quite realistic since it better reflects corporation cycles and working hours. The dataset consists of 14 features, out of which 10 features are from Netflow and the other 4, namely class, AttackID, AttackType, and AttackDescription, are added during the

TABLE 1. Publicly available datasets relevant to network forensics.

Network traffic based datasets				Electrical network/Smart-Grid datasets		Internet traffic based datasets		Android based datasets	
Year	**Dataset**	**Year**	**Dataset**	**Year**	**Dataset**	**Year**	**Dataset**	**Year**	**Dataset**
2019	CIC-DDoS2019 [29]	2013	ADFA2013 [30]	2019	IEC 61850 GOOSE [31]	2018	DeepCorr [32]	2020	CICMalDroid 2020 [33]
2018	CSE-CIC-IDS 2018 [34]	2013	CTU-13 [35]	2016	LBNL Power Data [36]	2016	Tor-nonTor ISCXTor2016 [37]	2019	CIC-InvesAndMal2019 [38]
2017	CIC-IDS2017 [39]	2012	ISCXIDS2012 [40]	2015	Virtual Gas pipeline [41]	2016	URL [42]	2018	CIC-AndMal2017 [43]
2017	CIC DoS [44]	2009	NSL-KDD [45]	2014	Power system [46]	2016	UGR'16 [47]	2017	CIC-AAGM2017 [48]
2017	CAIDA [49]	2006	KYOTO [50]	2014	Remote Terminal Unit Communications [51]	2011	MAWI [52]	2014	UNB ISCX Android Validation [53]
2015	UNSW-NB15 [54]	2000	DEFCON	2014	Gas Pipeline data [41]	2010	Heritrix [55]		
2014	TWENTE [56]	1999	KDD Cup 1999 [57]	1993	IEEE 300-bus power test system [58]	2008	ISOT [59]		
2013	CDX [60]	1998	1998 DARPA [61]						

IoT traffic based datasets		DNS related datasets		Vehicular related datasets		VPN based dataset		Internet connected devices based dataset	
Year	**Dataset**	**Year**	**Dataset**	**Year**	**Dataset**	**Year**	**Dataset**	**Year**	**Dataset**
2022	CIC IoT [62]	2021	CIC-Bell-DNS-EXF [63]	2019	Car-Hacking [64]	2016	VPN-nonVPN (ISCXVPN2016) [65]	2014	Botnet [66]
2021	Enriching IoTs dataset [67]	2020	CIRA-CIC-DoHBrw [68]	2018	VeReMi [69]				
2019	TON_IoT [70]	2019	TI-2016 DNS [71]						
2018	Bot-IoT [72]								

labeling process. It encompasses both benign and malicious traffic, including ping scans, port scans, brute-force, and DoS. CIDDS-001 has 146,500 instances, with the normal class representing 91.6% of network traffic.

7) UNSW-NB15 DATASET

In 2015, the Australian Center for Cyber Security (ACCS) used tools such as *IXIA PerfectStorm*, *Tcpdump*, *Argus*, and *Bro-IDS* to create the UNSW-NB15 dataset [53]. The IXIA PerfectStorm tool, which is used to generate both normal and anomalous traffic, is constructed on three virtual servers. ACCS collected data for 15 and 16 hours that consisted of nine different attack categories, including fuzzers, analysis, backdoors, DoS, exploits, generic, reconnaissance, shellcode, and worms. The collected data is based on different protocol types such as TCP, UDP, ICMP, etc. The gathered data was separated into 49 features.

8) ISCXIDS2012 DATASET

In 2012, the Information Security Centre of Excellence at the University of New Brunswick prepared a dataset, namely ISCXIDS2012 [40]. It was created through the use of a systematic strategy to minimize validity concerns in existing datasets. The complete dataset, 78.6 GB in size, consists of 2,450,324 network traffic packets with 20 features that span seven days of network activity (i.e., normal and intrusion). It consists of four different attack types, including brute-force SSH, infiltrating, HTTP DoS, and DDoS. Intrusion data represents around 2% of the entire dataset.

9) NSL-KDD DATASET

The NSL-KDD dataset is an enhanced version of the KDD Cup 99 dataset [56] generated by Tavallaee et al. [76] in 2009. The authors address the KDD Cup 99 dataset's inherent duplicate record concerns while simultaneously lowering the level of complexity. The data collection contains 41 feature records. There are five main classes in it; one is a normal class and the other four are attack classes, namely DoS, probe, remote to local attack (R2L), and user to root attack (U2R). The dataset consists of 125,973 instances of the training dataset and 22,544 instances of the test dataset.

10) KDD CUP 99 DATASET

In 1999, DARPA's [60] *tcpdump* files were modified and analyzed by University of California researchers to generate the KDD Cup 99 dataset [56]. The simulated attacks are categorized into four groups, i.e., DoS, R2L, U2R, and probing attacks. With KDD Cup 99, 41 features are divided into three classes, i.e., basic features: extracted trough TCP/IP connection; traffic features: further divide into two groups namely same host and same services; and content features: malicious behavior. KDD Cup 99's training set had 22 attack types, whereas the test data contained an additional 15 attack types. Researchers have mostly used KDD Cup 99 for the evaluation of intrusion detection models. This dataset was critiqued by McHugh [77].

III. STATE OF THE ART

Most common implementations of AI are based on machine learning (ML), deep learning, or ensemble learning. AI has been posited as part of the solution to the ever-increasing number of cases requiring expert digital forensic investigation [78]. This section reviews how a variety of AI techniques have been applied to a selection of different applications of network forensics.

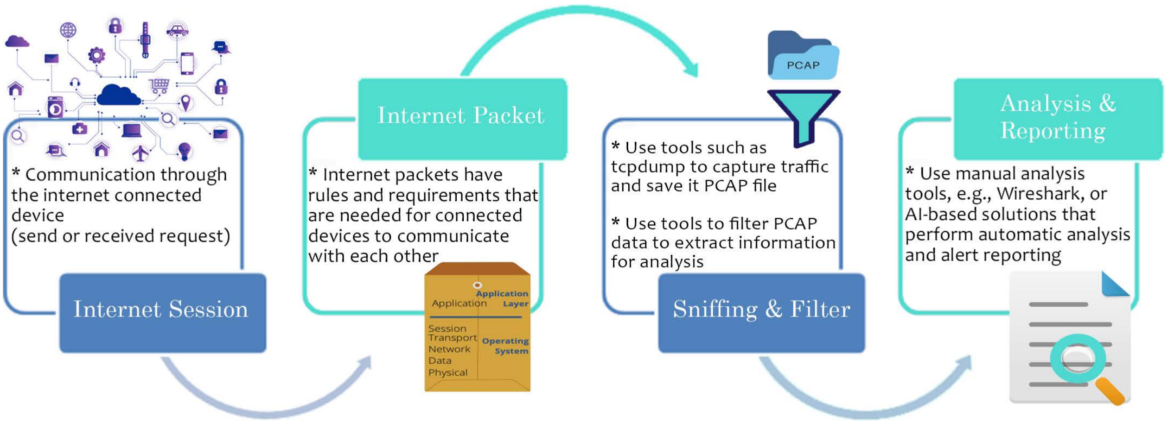
* Communication through the internet connected device (send or received request)
Internet Session
Internet Packet
* Internet packets have rules and requirements that are needed for connected devices to communicate with each other
Application Layer
Application
Session
Transport
Network
Data
Physical
Operating System
PCAP
* Use tools such as tcpdump to capture traffic and save it PCAP file
* Use tools to filter PCAP data to extract information for analysis
Sniffing & Filter
Analysis & Reporting
* Use manual analysis tools, e.g., Wireshark, or AI-based solutions that perform automatic analysis and alert reporting

FIGURE 2. Network traffic analysis process (adapted from [75]).

A. NETWORK TRAFFIC ANALYSIS

Network traffic analysis (NTA) detects, identifies, and analyses security threats and potential operational difficulties by utilizing network communications and associated protocols. NTA is a critical security method that moves threat hunting away from security perimeters and endpoints and onto network flows. An overview of the NTA process is shown in Figure 2. NTA employs a combination of ML, advanced analytics, and rule-based detection to construct (or improve) a baseline model of typical network activity and to send highly contextualized warnings when anomalous patterns are found. Increases in network traffic volumes have motivated several studies on new NTA techniques.

The popularity of expert systems was at its peak in the 1990s due to their effectiveness in analyzing data about the situation in a specific environment and drawing logical conclusions from them. Several studies from that time utilized expert systems for NTA. Stern and Chemouil [79] used an event-driven network simulator to model a management system and utilized an expert system on a French long-distance network by setting up different rules with some threshold to detect and diagnose the events. Lindqvist and Porras [80] discussed the efficacy of the Production Based Expert System Toolset (P-BEST), which was used for decades to monitor, control, and identify misuse. A library of runtime procedures (rule translator) and garbage collection algorithms make up the P-BEST toolset. Due to its general-purpose inference engine, P-BEST can be applied to various problem domains, including analysis of traffic streams, detection of TCP/IP layer attacks, and application layer attacks. Expert systems provide significant benefits for analyzing firewall rules and also support signature analysis of network traffic. ‘‘SEMACS’’, a real-time monitor and control system implemented on the Universal Floor Device Controller (UFDC) system by Dunning and Switlik [81] in 1988, which was designed to detect problems before they become serious issues and assist in corrective decision-making. It updates its knowledge every 20 seconds to detect network hardware and software issues.

In the late 1990s, fuzzy-based expert systems were utilized for asynchronous transfer mode (ATM) networks [82]. These were used to prevent network resources from overloading in a scenario where the connection exceeds the negotiated traffic parameters. Experiments show the effectiveness of the fuzzy policer in terms of responsiveness and selectivity.

A significant change occurred after 2000 with the adoption of ML models. A major reason behind this was the availability of datasets such as 1998 DARPA Intrusion Detection Evaluation, KDD Cup 99, NSL-KDD, etc. Various different statistical approaches, including linear and non-linear principal component analysis (PCA) and genetic algorithms (GA), were used for dimensionality reduction. For ML algorithms, dimensionality reduction is considered an essential step as it helps to remove multicollinearity, thereby enhancing comprehension of ML model parameters and reducing model training time. Various ML approaches, including but not limited to, Support Vector Machine (SVM), Decision Tree (DT), Linear Regression (LR), Random Forest (RF), Naive Bayes (NB), and K-Nearest Neighbors (k-NN), have been utilized to address the NTA domain. Knapińska et al. [83] have recently utilized ML to examine numerous time-series predictions for traffic of multiple frame sizes to address modeling and prediction of long term network traffic patterns. They explain the acquired real network traffic statistics and investigate periodicity and traffic type relationships. An extensive experiment was performed for traffic prediction using Fourier transform and ML-based prediction utilizing Multilayer Perceptron (MLP) regressor by changing the parameters to increase the prediction quality and time.

Millán [84] investigates how many time-series points from a high-speed traffic network are necessary to properly predict the *Hurst exponent*. The Hurst exponent is used to calculate the long-term memory of a time series. The process involves planning an experiment that employs time-series estimation methods and then addresses the smallest number of points necessary to produce reliable Hurst exponent estimations in real-time. An experiment shows different behavior depending upon time-series length, where Whittle’s estimator performs well for both short-term and long-term time series. Dong [85] suggests a modified SVM technique called cost-sensitive SVM (CS-SVM) that forecasts the kind of traffic generated by an application to avoid the imbalance problem in network traffic detection, which is a burden on the model's performance. CS-SVM uses an active learning, multi-class SVM algorithm to dynamically allocate weight to applications. The proposed model achieves 70% of the geometric mean and the multi-class area under the curve (MAUC), both typically used to evaluate the solution to the data imbalance problem.

The dramatic increase in the number of network-connected devices together with increased network speeds results in huge network traffic volumes, which encourages researchers to deploy DL models and ensemble learning for NTA. Identifying useful information from encrypted traffic is a challenging task [15]. Lotfollahi et al. [86] introduced Deep Packets, which is a DL-based traffic classification approach to identify encrypted applications such as BitTorrent, Skype, and others, and also distinguish traffic types such as FTP, P2P, and others. Instead of inspecting packet content for keywords or usage patterns, as deep packet inspection techniques do, the methodology uses DL architecture to learn new features for each application. The authors employed a one-dimensional convolutional neural network (1D-CNN) and stacked autoencoders on network traffic for automated feature extraction and classification to achieve both application identification and traffic characterization in encrypted or unencrypted traffic. Experimental results demonstrate that the proposed model outperformed the general ML-based methods.

The adoption of wireless mesh networks is increasing due to their adaptability, flexibility, and efficiency in terms of cost and time. Contributing to traffic prediction problems for mesh networks, Mahajan et al. [87] proposes a unique architecture based on CNN and long short-term memory (LSTM). An experiment was performed on sensors that formed a network, a mesh network. Extensive experiments show that the proposed unique Convo-LSTM model provides improved performance for network traffic prediction. Resource optimized ML models are an essential requirement, especially in IoT security. A recent study by Gandhi and Ribeiro [88] examines the influence of network packet clustering on the combination of performance metrics (accuracy, F1 score) and system resources (CPU and memory) required by traditionally used ML algorithms, including LR, RF, k-NN, SVM, XGBoost, and Deep Neural Network (DNN) in the scope of botnet detection in IoT networks. The paper concentrates on the system resources used by these algorithms, rather than optimizing ML algorithms for resource limitations or application workloads.

A meta-learning approach can assist in reducing false positive rates caused by non-malicious activity during the attack detection phase. The attack detection system requires meta-learning to integrate several classifiers and apply an integration strategy to decrease false positives. Possebon et al. [89] performed experiments to classify network traffic using meta-learning approaches including voting, stacking, bagging and boosting and to evaluate the results with conventional models. The results demonstrate that bagging got better scores when compared to other meta-learners in terms of accuracy and false positives, whereas other meta-learners yielded scores equivalent to non-meta algorithms, with no discernible enhancements. Several surveys have been conducted to summarize ML and DL approaches for NTA [75], [90], [91], [92].

1) CURRENT CHALLENGES AND FUTURE DIRECTIONS

The characteristics of networks vary depending on their architectures, equipment, scale, applications, and so on. It creates significant challenges in which ML approaches must be trained for every network independently. However, ML-trained models may reduce their accuracy on different network topologies. ML and DL are well known for resolving complex problems, and both of them have been utilized for NTA. One inadvertent benefit of AI-based NTA approaches is that it can help reduce access to privileged information [94]. The latest publicly available datasets, such as CICIDS-2017 and CSE-CIC-IDS2018, are still vulnerable to excessive imbalance problems [95], which can lead to low accuracy and a high false-positive rate. A dataset's files may be combined to contain all the attack descriptions for analysis. However, merging examples of each attack type expands the dataset, resulting in higher computation and processing time. Due to the vast utilization of internet-connected devices, classifier analyzers must deal with an increase in volume and transmission rates.

Multi-layer DL models, due to their complex architecture, require a long processing time. To resolve this issue, lightweight algorithms with low computing costs are sought that can solve complex problems. The expanding trend of data encryption and protocol tunneling introduces new obstacles for security specialists. There is a distinct absence of studies on fault management, and a significant proportion of the examined publications employ DL for other objectives, such as traffic flow categorization and forecasting. Traditional solutions for fault management, such as rule-based systems and algorithmic approaches, have significant drawbacks. On the one hand, where attackers may generate sophisticated attacks or tricks to bypass systems to harming users or organizations; on the other hand, security experts try to improve the monitoring and analysis system continuously for corrective action before any major incident.

B. INTRUSION DETECTION SYSTEMS

A network-based intrusion detection system (NIDS) is a detection and prevention mechanism that monitors network traffic for hostile and suspicious behavior. NIDS can help traditional corporate systems and organizations strengthen their security controls and secure their network environment.

To identify attacks, NIDS use a signature-based or anomaly-based approach. Signature-based approaches require a library of known attacks to compare network traffic. Signature detection uses a rule-based system to

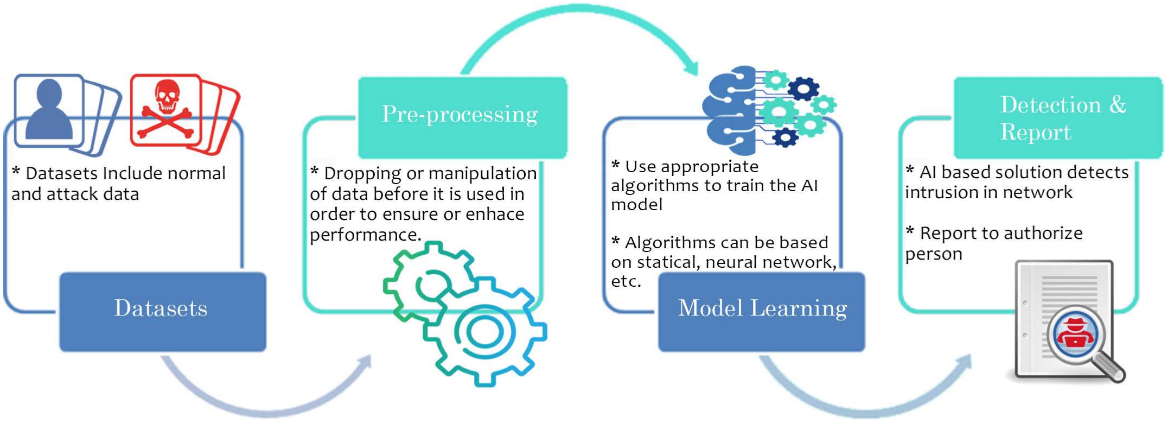
* Datasets Include normal and attack data
Datasets
Pre-processing
* Dropping or manipulation of data before it is used in order to ensure or enhace performance.
* Use appropriate algorithms to train the AI model
* Algorithms can be based on statical, neural network, etc.
Model Learning
Detection & Report
* AI based solution detects intrusion in network
* Report to authorize person

FIGURE 3. **The architecture of AI based IDS (adapted from [93]).**

detect malicious activities by comparing the network traffic against a library of known vulnerabilities. Anomaly-based approaches have been developed using AI techniques to detect abnormal behavior. The architecture of IDS based on AI is shown in Figure 3.

Even the most secure systems may be abused by outsiders who attempt to breach the network for a purpose, or by insiders who abuse their privileges. To address insider abuse and other challenges, Denning [96] introduced an Intrusion Detection Expert System (IDES) that is based on the principles of anomaly detection, which is not dependent on any specific target system. IDES represents profiles for describing the behavior of individuals from the perspective of metrics and statistical models, and rules for learning about such behavior from audit logs to identify aberrant activity. The research focused on activity profiles based on subject and object, along with metrics such as resources and time, to generate low false alarms. Lunt et al. [97] enhanced the work of [96] and proposed a solution to characterize the behavior of subjects based on a combination of statistical and expert-based approaches. Wisdom and Sense (W&S) [98] is an anomaly detection system that generates rules automatically using historical data. Its goals are to identify breaches, harmful or erroneous user behavior, Trojan horses, and infections.

However, W&S rule bases have a huge number of instantiations, many of which are irrelevant. In the Wisdom part, a heuristic is designed for dealing with categorical data and adapting skewed, multi-modal continuous data to categorical data. It works well on computer audit logs and developed rules that are human-readable, allowing for the incorporation of human and machine rule bases into a unified rule base. It assigns grades, which is a measure of the historical accuracy of the rule. At Sense, the anomaly detection module calculates the transaction score of the event and checks whether it exceeds the set limit to detect anomalies. However, W&S is not capable of detecting anomalies in real-time or may be feasible with lower detection sensitivity. Signature-based IDS offers a low false alarm rate, but are unable to detect new or previously unseen attacks.

ML has been used to address the challenge of detecting novel attacks. This was based on detecting unexpected behavioral patterns on the network and alerting users to any detected deviations from normal behavior. Publicly available datasets were used to evaluate ML algorithms. A major difficulty in abuse detection is determining how to create signatures that encompass all conceivable attacks in order to avoid false negatives, as well as how to create signatures that do not match non-intrusive actions in order to avoid false positives. Though false negatives are usually regarded as more dangerous, the setting of threshold levels is critical to ensure that none of the aforementioned issues are exaggerated unnecessarily. To address the challenge of detecting anomalies and misuses, Mukkamala et al. [99] developed two IDSs by utilizing SVM and neural networks on the KDD99 dataset to train models with normal user activities and attack patterns. The selection of the SVM model is based on scalability and speed. SVM IDS was built based on 41 input features that included both normal and attack classes. An IDS based on a neural network was trained using MLP with *gradient descent*, since it is computationally efficient and produces a stable error gradient and convergence. To improve the effectiveness and results of ML models, feature selection was investigated while deploying ML for IDS. The feature selection method determines which characteristics are more discriminative than others. It has been demonstrated that both neural networks and SVM generate quite accurate results. SVM, on the other hand, can only conduct binary classification, which is a severe disadvantage when the IDS requires multiple-class identifications.

Many IDSs have low detection rates and high false alarm rates due to the massive volume of network data and the imbalanced distribution of regular and anomalous actions. Ren et al. [100] proposed a data-optimized IDS solution to handle the unbalanced distribution of normal and anomalous behaviors. They deployed a hybrid data optimization method based on sampling and feature selection using multiple ML algorithms. For data sampling, Isolation Forest is used, which is a tree-based outlier identification approach with linear time complexity and strong precision that is ideal for high-dimensional and large-scale data sets. For feature selection, they used genetic algorithms. RF is used for classification and for optimization of features and sampling ratio. The model performed well, especially in detecting anomalous behaviors with fewer records, such as DoS, analysis, backdoors, and worms. However, there are still enhancements that may be made, such as reducing the time spent on data optimization and providing support for online processing.

Recently, Chen et al. [101] also proposed an improved RF-based model by employing ADAptive SYNthetic sampling (ADASYN) to balance the dataset and applied it to detect network attacks accurately and efficiently. They merged eight different datasets into the CIC-IDS2017 dataset in order to simulate benign data-flow and the latest common attacks. An experiment shows that the proposed model has higher prediction performance, efficiency, and robustness compared to the traditional ML algorithms. However, the result shows that false positives are still one of the major concerns. Modern-day network traffic requires an IDS with optimal efficacy. Furthermore, false positives may need more system resources and false negatives may render the entire system inoperable. Thus, latency is one of the key evaluation metrics along with accuracy to evaluate the performance of an IDS because IDSs make predictions in real-time. Seth et al. [102] proposed a time-efficient model focused on latency that did not impact the performance of attack detection. The feature selection phase was done through a hybrid approach that includes RF and PCA. PCA is applied to the selected essential features, and the implemented approach reduces the prediction latency by reducing the complexity of the model. The model was trained using the light gradient boosting machine (LightGBM) algorithm, which is a DT-based gradient boosting system that is fast, distributed, and high-performance. The latest CIC-IDS-2018 dataset is used in order to identify the vast majority of modern-day attacks. An experiment compared the results of the proposed models with five ML algorithms, namely RF, Extra Trees, XGBoost, k-NN, and Histogram Gradient Boosting.

One way to handle the false positive problem in IDS is to use the optimum number of features. Megantara and Ahmad [103], [104] suggested a hybrid ML technique that combines the feature selection and data reduction methods. It works by employing a feature significance DT-based approach with recursive feature elimination to pick critical and essential features, as well as the Local Outlier Factor (LOF) method to discover outlier data. Experimental findings reveal that the suggested technique detects R2L with higher accuracy and maintains better precision for other attack types than previous studies on the NSL-KDD dataset. As a result, it performs more steadily than others. However, the authors faced some challenges while comparing the result of UNSW-NB15 with binary classes. There is still a lot of room for improvement in terms of accuracy and efficiency.

Many researchers have used ML methods on the topic of intrusion detection, such as DT, k-NN, SVM, and DNN, and have obtained some preliminary results. Each algorithm model may be superior in some aspects while being deficient in others, and resolving such deficiency is one of the current challenges. In addition, strengthening the detection ability of small-scale samples is also a major concern. Tang et al. [105] proposed an integrated learning solution to solve deficiencies such as limited adaptability, latency in detection, inadequate detection accuracy, and so on, in ML algorithms. It belongs to ensemble learning to blend the benefits of different ML algorithms and improve the detection rate. An experiment was done on the NSL-KDD dataset using the deep-stacked technique based on different algorithms. The undersampling method is used to process the training samples of the dataset in order to handle the unbalanced training samples problem to avoid biases. The authors used *cross-validation* to determine the hyperparameters and discovered that four models, DT, k-NN, DNN, and RF have superior detection performance and fulfil the demands of diverse classification impacts. The deep stacking network increased its classification impact.

DL has been used for IDS in recent years, due to its automatic feature generation and scalability. It has the ability to extract better representations from data in order to develop better models. On CIC-IDS2017, Sun et al. [107] used two DL approaches, CNN and LSTM, to extract features and categorize network data. CNN was used to extract spatial features, whereas LSTM was utilized to detect temporal information. To overcome the class imbalance issue, the authors further performed weight optimization on the training dataset. 1D-CNN is becoming increasingly popular in comparison to other ML approaches due to its superior feature extraction capabilities. Azizjon et al. [108] employed 1D-CNN for supervised learning on time-series data with 42 features by serializing TCP/IP traffic in a predefined time period as an invasion internet traffic model for the IDS. A max-pooling layer deals with the CNN layer to optimize output size, feature count, and computational complexity. For both balanced and imbalanced training datasets, the proposed 1D-CNN performance is compared to SVM, RF, and the combined architecture of 1D-CNN and LSTM. Extensive experiments were carried out using the publicly available dataset UNSW-NB15, and random over-sampling was applied to handle the unbalanced data problem. Vinayakumar et al. [109] also combined 1D-CNN with RNN, LSTM, and Gated Recurrent Unit (GRU). The authors analyzed the impact of the number of layers using different modeling architectures.

Yin et al. [110] proposed an RNN-based system to detect both binary and multi-class intrusion. Exclusive experiments were performed to evaluate the effect of neurons and learning rate on the accuracy of the model. They used the NSL-KDD dataset to create a map from 41 to 122-dimensional extracted features by transforming non-numeric characteristics into numeric values for preprocessing. The logarithmic scaling

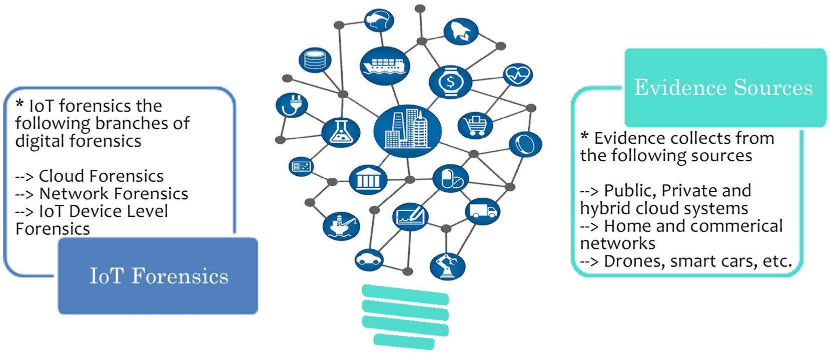
* IoT forensics the following branches of digital forensics
--> Cloud Forensics
--> Network Forensics
--> IoT Device Level Forensics
IoT Forensics
Evidence Sources
* Evidence collects from the following sources
--> Public, Private and hybrid cloud systems
--> Home and commerical networks
--> Drones, smart cars, etc.

FIGURE 4. IoT forensics components (adapted from [106]).

and linear normalization approaches are then utilized to bridge the gap between the maximum and minimum values for certain features, enhancing accuracy. RNN-IDS beats the traditional classification approach in binary and multi-class classification on the NSL-KDD dataset.

DL has yielded excellent results, but requires large, centralized datasets for better performance of the model. The centralized collection of sensitive network traffic data raises concerns about privacy. There has also been a focus on federated learning due to its ability to train across several decentralized fog devices or servers. The server does not gather data, but it can collect model parameters. A federated learning-based NIDS is presented in [111] to address the problems of insufficient NIDS datasets and privacy protection. The authors utilized GRU on the CIC-IDS2017 dataset for experiments, and their results proved that the federated learning approach achieves more accuracy as compared to the traditional centralized training approach, and is capable of providing privacy protection. Federated learning requires more communication, but its performance is strikingly similar to that of centralized learning.

1) CURRENT CHALLENGES AND FUTURE DIRECTIONS

AI-based IDS is mostly performed using publicly available datasets. However, there is no universal way to verify the extent to which these datasets represent real-world network traffic. The performance of computational intelligence systems with intermediate datasets in dynamic environments has yet to be examined. ML algorithms are commonly used on publicly accessible datasets and have shown significant results in terms of detection rate and accuracy; nevertheless, a comparison of effectiveness with private datasets must be examined. Despite the fact that most of the techniques have high alert rates, unfortunately they often also have high false alarm rates. The categorization challenge in a multifactor environment is complicated; hence, false alerts are inherent with any IDS. To limit false alerts, preventive measures should be adopted, such as network behavioral analysis, to help more accurately detect previously unencountered attack types. Furthermore, a high false-positive rate results in a high cost, since considerable resources are used in analyzing the detected activity, which ultimately turns out to be typical network traffic. Along with detecting and preventing intrusions, IDS requires an intelligent response mechanism that can notify and conduct early action as an intrusion is detected, as well as notify the support team. Because attacks are directed at different layers of the network communication model, the security aspects of those layers should be explored by identifying and locating separate attacks at those layers. Ensemble and hybrid strategies should be developed to boost attack detection and classification rates by syntactically and semantically examining the operation and understanding of functional characteristics in respect to present methodologies.

It is extremely difficult to create an online and real-time, anomaly-based IDS for IoT networks. This is due to the fact that such an IDS would have to first acquire regular behavior in order to identify anomalous or malicious activity. ML and DL-based IDS, increase computing complexity. Creating an efficient IDS with low computing needs is thus another issue and topic for future study.

C. IoT FORENSICS

The Internet-of-Things (IoT) is a relatively new category of consumer and industrial electronics, and IoT forensics is still very much in its infancy. Of course, the objective of IoT forensics mirrors that of digital forensics; namely, to detect and retrieve digital information in an ethical and forensically reliable manner [112]. In addition to an IoT device's local storage, evidence can be collected via a local network or from the associated cloud service back-end [113]. The components of IoT forensics are shown in Figure 4.

With IoT, the focus has been on the benefits and uses of the technology, as well as on security and privacy risks that may arise. Within the IoT domain, there is little in the way of a specialized incident response technique for digital forensics responders. To fill the gap, Oriwoh et al. [114] explored theoretical forensic models for IoT forensics to facilitate the investigation process. The authors suggested a high-level incident response framework based on zones, namely internal, middle, and external networks, for dealing with IoT-related cybercrime incidents. Attacks and abnormalities are more likely in an IoT system since it must operate 24/7 on the internet or a local network.

Detecting threats and attacks in an IoT platform necessitates extensive data analysis and computational intelligence. Adversarial threats are the deliberate actions of an entity with the objective of interfering with corporate IT systems in such a way that the organization suffers failure or loss. Shakeel et al. [115] presented a blockchain-assisted shared audit architecture for determining the source of data from attackers accessing virtual resources in IoT platforms. The use of blockchain with AI in IoT can provide distributed trust, minimize computational and complexity in security, and allow quick transactions with scalability and flexibility. For audit analysis and access restrictions, the suggested framework incorporates virtual resources, infrastructure units, and end-users. For detection and verification, the system employs two layers of virtual resource log analysis. To find an acceptable detection, log data analysis is concatenated at the first level using logical regression ML. In the second level, data is cross-validated to verify the cause of scavenging. The proposed solution first detects the adversary by leveraging requests mapped to the virtual resource, and then filters the detection for improved verification by making use of the density of IoT devices. To monitor events between virtual resources and end-users, blockchain and data analytics for audits are used in combination. More recently, in 2022, Mukherjee et al. [116] used a supervised ML model to detect anomalies in smart devices and IoT systems, which may then be used in real-world settings to prevent future abnormalities and attacks. An experiment employed ML algorithms over *DS2oS* traffic trace data [117] and evaluated its effectiveness against the state of the art. The authors found that DT and RF performed best in their experiments where binary values from the feature "value" were removed.

The IoT concept rapidly increases the number of devices. The classification of IoT devices is necessary for numerous purposes, including but not limited to identifying illegitimate devices and unwanted devices. Cvitić et al. [118] investigates the possibility of employing attributes to identify devices in diverse environments, regardless of function or purpose. This study made use of 41 IoT devices in total. A classification model was created using logistic regression and then enhanced with supervised ML (*logitboost*). To develop the multi-class classification model, 13 network traffic characteristics generated by IoT devices were employed. Based on the traffic flow properties of such devices, research has shown that it is feasible to categorize devices into four groups with excellent performance and accuracy. Specially in IoT applications, one of the most crucial challenges is resourceful analytic procedures with low energy consumption. Saba et al. [119] combines the Q-learning approach to the built energy-efficient and fault-tolerant routes, including a cryptography algorithm, to ensure security protection of confidentiality and authentication against maliciously elements in a wireless sensor network.

DL has been leveraged by several researchers in the IoT network security domain to address various challenges. Data heterogeneity and learning from unlabeled data have emerged as critical research topics in the IoT ecosystem. Abdel-Basset et al. [121] deployed DL in a semi-supervised technique (SS-Deep-ID) to identify IoT intrusions. Furthermore, the approach makes use of LSTM and CNN for feature extraction from the spatio-temporal dataset. The traffic attention layer is also used to quantify the relevance of features and enhance feature extraction prior to actually determining the final traffic class. An experiment that uses the CIC-IDS2017 and CSE-CIC-IDS2018 datasets demonstrates that SS-Deep-ID achieves outstanding results in intrusion detection for IoT contexts. The suggested model is straightforward to incorporate into a fog-enabled IoT network. Abdel-Basset et al. [122] also addresses CNN's failure to capture the long-term properties of IIoT traffic data and RNN's problems of gradient expansion and vanishing by proposing Deep-IFS based on DL models, which are used for detecting intrusions in IIoT communications. Deep-IFS captures local representations using a local gated recurrent unit (LocalGRU) and global representations using multihead attention (MHA). The MHA layer allows the capture of connected positional information and provides a flexible flow of information without suffering any loss. The addition of two autoregressive units improves the Deep-IFS model's robustness for intrusion detection on IIoT traffic in a fog computing environment. The effectiveness of the proposed model is tested on two different datasets, namely BoT-IoT and UNSW-NB15.

Although DL algorithm-based intrusion detection systems have advantages over traditional techniques, they suffer from over-fitting issues as the number of attacks grows. Scalable IDS is required to handle such issues. Jothi and Pushpalatha [123] investigated the convergence of DL with metaheuristics. Metaheuristics are search techniques that help to direct the search process. Essentially, the Whale optimizer was combined with LSTM to do automated weight and bias selection. Their approach was evaluated by utilizing various benchmark datasets, including CIDDS-001, UNSWNB15, and KDD. The analysis revealed that an accuracy of more than 99% was maintained in all datasets examined, and the performance of the proposed model illustrated their suitability for an IoT network.

A single classifier is often inadequate to design an effective IDS, pushing researchers to propose a classifier ensemble model. Rashid et al. [124] discovered multi-classification cyberattacks at fog nodes in a distributed rather than a centralized system to track network traffic with ensemble approaches for IoT-based smart cities. Before creating the model, an information gain-based feature selection approach is used to find the most essential features. ML models are strengthened using bagging, boosting, and stacking techniques. On datasets, they had the most success using the stacking strategy.

1) CURRENT CHALLENGES AND FUTURE DIRECTIONS

Recognizing the network architecture of the endpoints in an IoT system forensic investigation is a challenging problem. There is still the possibility that a sensor transmitting data to

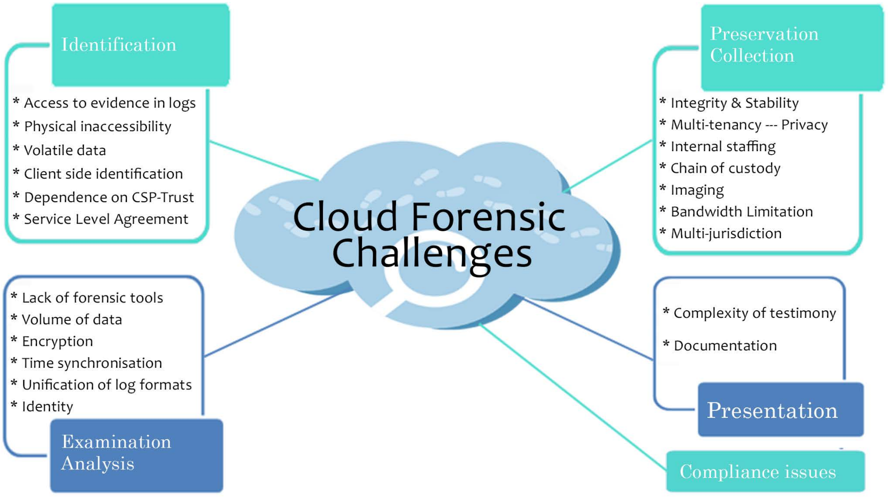

FIGURE 5. Cloud forensics challenges based on their stages (adapted from [120]).

an IoT gadget is located in an unknown place on the crime scene. IoT forensics presents certain difficulties in addition to the potential it provides. New devices, new interfaces, new storage medium, new file systems, new network protocols, distributed cloud storage, and ambiguous authority and jurisdiction are just a few examples. The volume of data that must be maintained, stored, and analyzed is enormous. Even presenting the results might be difficult. The evidence-gathering phase is among the most critical parts of the forensic technique, since any inaccuracy might render the evidence material incorrect and damage the entire investigative process. Evidence of cybercrime is difficult to acquire in the case of IoT devices that are part of huge networks, due to a lack of equipment and professional expertise, as well as improper or insufficient documentation. To gather some proof of illegal conduct, the forensic investigator should attempt to evaluate the logs that contribute substantially to this procedure. Pertinent materials including network logs, process logs, and application logs from multiple resources may aid investigators in gaining a better understanding of the overall device's activity. Standardization of log formats and ingestion from various systems is required to address this challenge. The use of ontologies and semantics have been explored as an approach to developing a standardized baseline. This approach can be used to reveal the degree of interdependencies among the various devices. Behavioral analysis at the device-level can also aid in the detection of previously unencountered approaches.

The findings also revealed that data encryption is a critical topic that needs to be addressed in a future study. Data encryption is an anti-forensic tactic that has previously been identified as a barrier in digital forensics. Throughout the investigation process, the majority of the solutions have overlooked data privacy. Furthermore, authenticating IoT devices is essential to ensure that no unauthorized access is permitted. During the cybercrime investigation process, the IoT-based authentication system may be used to authenticate access to IoT devices and identify IoT users. Forensic readiness seeks to provide a specific organization with the administrative, technological, and physical control necessary to conduct an efficient investigation. Forensics readiness in IoT systems is a difficult problem that requires more study to make IoT networks forensically suitable. Furthermore, in the context of IoT, laws and regulations must keep up with the engagement of technology and forensic procedures.

D. CLOUD FORENSICS

A cloud forensics investigation is described as an examination of cybercrime that requires evidence from any of the cloud computing platforms or services [14], [125]. The most essential aspect is that, in a virtual environment, evidence may be stored anywhere. Early investigations into cloud systems relied on traditional digital forensics methodologies and technologies. Rapid improvements in cloud computing required the development of new approaches, frameworks, and tools for conducting forensics in cloud systems [113]. Early articles focused on retained data for cloud forensics. From 2010 onwards, cloud forensic frameworks emerged [126].

Most of the publications on cloud forensics have covered evidence collection, network concerns, privacy issues, and frameworks to facilitate cloud forensics. Cloud forensics challenges, based on their stages, are shown in Figure 5.

Khorshed et al. [127] discussed the challenges in cloud computing related to trust, threats, risks, and other issues that slow down adoption. The authors also addressed malicious insider attacks, where an authorized employee uses their privileges to purposefully or unintentionally harm an organization by stealing, disclosing, or deleting its data. The proposed model aims to identify an attack when it begins, or at the very least while it is underway, and also provides information about the attack type, if cloud providers attempt to conceal attack information from consumers. NB, MLP, SVM, DT, and PART were implemented on WEKA [128], a Java-based ML tool that includes a range of data visualization tools and algorithms, as well as a graphical interface for easy access to these features. Attackers usually seek to disrupt cloud computing performance. A DDoS attack is used to make services unavailable to their users by flooding the environment with fake requests. A statistical technique for detecting DDoS attacks was presented by Gaurav et al. [129]. To distinguish malicious traffic from regular communication, the authors applied the concepts of cluster entropy and packet score. Each incoming packet's packet score is computed and compared to a predefined threshold. The proposed approach is reactive in nature, so it begins filtering attack packets at the start of the DDoS attack to mitigate damage. The trained model was evaluated using the *OMNET++* simulator.

Attackers do reconnaissance and scanning to detect weaknesses and initiate attacks on a network. To secure the network, real-time IDS is required with effective accuracy. Alshammari and Aldribi [130] presented a lightweight detection approach for network traffic abnormalities that includes an ML model for feeding IDS in real-time. This detection technique makes use of a dataset comprising malicious and benign data. Wireshark is used to extract features from the ISOT-CID network traffic dataset [131], which is used to train six different ML models. The authors added six new features to the dataset, namely; *T-IN*, *T-Out*, *APL*, *PV*, *TBP*, and novel *Rambling* that computes the traffic data connection interval time. During rambling, they extract the packet payload length and compute the length diversion around the mean of all packet lengths. Experiments confirmed that it improves detection accuracy. They compare the effectiveness of cross-validation (5-, 10-, and 15-folds) and split-validation (90-10%, 80-20%, and 70-30%) on accuracy. To deal with ICMP attacks, TCP Sync attacks, UDP attacks, log analysis, and pattern finding difficulties, Sachdeva and Ali [132] presented a novel solution based on a GA suitable for large datasets. However, handling such a large volume of data through a search algorithm makes the process slow. Therefore, k-NN and MLP are used for preprocessing and finding duplicates in the data. The suggested method optimizes the subsets and parameters to achieve higher accuracy.

The pay-as-you-go model of cloud computing encourages adoption, but can be exploited by an attack known as Economic Denial of Sustainability (EDoS), which forces customers to pay for extra services triggered by the attacker. Dinh and Park [133] proposed a solution that consists of online and offline stages to tackle sophisticated attacks and satisfies both resource usage and detection performance requirements. Considering that network traffic has a sequential relationship in the time dimension, a multivariate time-series data-based methodology based on GRU is presented to identify and resolve EDoS intrusions in each network flow. These methods achieve great accuracy by employing a dynamic threshold, which helps to lower the high false-alarm rate.

Execution of malicious code, for example for mining cryptocurrency, is also one of the threats to the cloud environment. Such malicious code typically has multiple ways of transmission and attempts to conceal profit-seeking activity that is destructive and constantly updated. A technique for identifying suspicious mining code on cloud platforms is presented by Li et al. [134], which combines ensemble learning approaches like bagging and boosting to construct a detection model. The authors propose a static method to detect malware based on *n-gram* string features. The variance of model detection may be significantly decreased by randomly collecting samples and allowing models to vote together to determine the result. The suggested technique outperforms standard classifiers with regard to accuracy and robustness.

In part due to the Covid-19 pandemic, the health care sector has paid increased attention to secure cloud-based systems. When compared to large models in the core cloud, smaller models in the edge clouds require significantly minimal time to train. MUSE, a deep hierarchical stacked neural network based on DNN, is proposed by Gupta et al. [136] for detection of malicious behavior that causes changes in meta-information through the IoT gateway, edge, and core clouds. This MUSE system integrates and consolidates layers of learned edge cloud models to produce a partially pre-trained core cloud model. It enhances the efficiency of training and the accuracy of detection of large core cloud models. In contrast to edge clouds, the suggested model in the core cloud takes significantly less time.

1) CURRENT CHALLENGES AND FUTURE DIRECTIONS

Structural differences in cloud architectures poses several problems in cloud forensics at different stages of investigation, including identification, preservation, collection, analysis, and reporting.

Unification of log formats is one of the challenges in cloud forensics, which impedes the investigative process since the evidence acquired will be in many formats. A distributed location and a large number of servers are the characteristics of cloud computing that bring the problems of synchronization and timestamping. It is challenging to find evidence in the cloud infrastructure since service models are deployed differently. Furthermore, the seizure of machines containing possible pieces of evidence is limited. Volatile data is a concern for investigators throughout the preservation and collection stages since critical data such as processes, registry entries, and temporary files are erased when a virtual machine is turned off or restarted under the IaaS service model.

For the investigation process, there can be a single point of failure in the cloud forensics investigation process. Client-side evidence identification is critical in investigations, yet is sometimes difficult to obtain due to multiple jurisdictions. The terms agreed upon inside the cloud service level agreement (SLA) may offer information on how a forensic investigation would be conducted. In many circumstances, critical phrases for inquiry are not mentioned in the SLA between the cloud service provider (CSP) and the client. In cross-national data access and sharing, there is a lack of international collaboration and regulatory mechanisms,

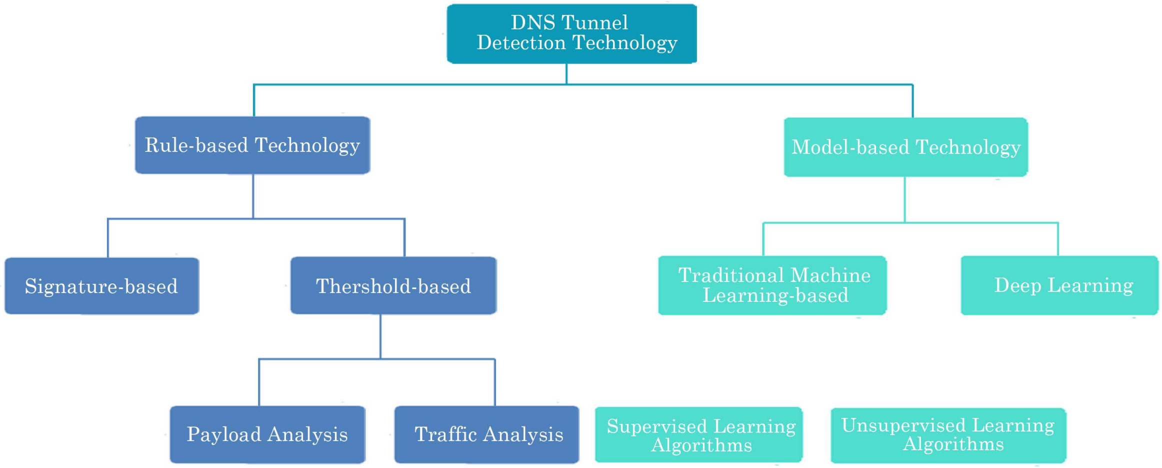
DNS Tunnel Detection Technology
Rule-based Technology
Model-based Technology
Signature-based
Thershold-based
Traditional Machine Learning-based
Deep Learning
Payload Analysis
Traffic Analysis
Supervised Learning Algorithms
Unsupervised Learning Algorithms

FIGURE 6. Overview of AI based DNS detection technology (adapted from [135]).

especially when cloud forensics relies on gathering evidence through servers located in multiple locations. Furthermore, there is significant potential for future work in the application of AI to specific cloud service delivery options, such as Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS) and Infrastructure-as-a-Service (IaaS) [15].

E. DNS TUNNELING

Domain Name System (DNS) tunneling is an attack technique that attempts to evade detection by using DNS requests and responses to transmit data. Although DNS tunnels only provide a limited amount of transport capacity, this data often includes malware payloads as well as command and control messaging [137], [138], [139], [140]. DNS tunneling is considered here in this paper as an example of a kind of attack that AI techniques have been deployed to address. The AI-based DNS detection technology is shown in Figure 6.

DNS tunnel detection approaches can be considered as belonging to either of two different categories, namely rule-based detection, often based on signatures, or model-based detection. With the former, rules are manually defined based on an examination of pertinent features.

Signature-based detection can usually detect DNS tunnels with high accuracy and few false positives. Horenbeeck [141], using the Snort IDS, showed that the early DNS tunneling tool *NSTX* [137] included a unique hard-coded value in DNS packet headers which could be designated as an NSTX signature. The covert channel attack is used to transfer information that is not allowed by the security policy. Sheridan and Keane [142] investigated the detection of DNS-based covert channels attacks. They used the differences in average packet size among active and passive DNS tunnel traffic as a signature for the open-source DNS tunneling tool *Iodine* [143]. The signature associated with certain DNS tunnel techniques or malware activity data suggests weak generality and significant consumption of resources. Various model-based detection approaches address this issue.

Model-based detection develops identification rules automatically stemming from different features using an ML model. Its accuracy highly depends upon the extracted features; Sammour et al. [144] extensively discussed the payload analysis and traffic analysis features. A combination of two different ML algorithms, RF and DT, has been used to detect DNS tunneling. The proposed classifiers have been trained on encrypted flows sufficiently to facilitate a statistics modeling approach on the inner protocol carried [145]. Each flow is examined in terms of its unique characteristics, as well as packet size and inter-arrival latency. The difficulty in identifying DNS tunneling activity in general stems from the fact that each malware family behaves differently. In this regard, Preston [146] utilized the recommendation proposed by Nadler et al. [147] to classify malicious domains instead of malicious queries. An experiment with six different algorithms has been performed to achieve higher accuracy and low false alarms, which is discussed in detail for insight.

Some technologies create tunnel traffic with a character distribution comparable to standard DNS queries, which increases the false alarm rate. Liu et al. [148] addresses this issue by focusing on DNS record type and query length. It makes a model to analyze recursive DNS. The model was trained on dns2tcp, DNScat2, Iodine, and OzymanDns based datasets. Among the deployed versions of ML for binary classification, SVM showed the best detection performance in terms of accuracy, precision, and recall. With the increasing usage of cryptographic protocols and the use of numerous protocols to construct tunnels, the flow in the tunnel cannot be detected directly, rendering the identification method for a single protocol ineffective. To bridge this gap, instead of aiming at a particular protocol, Bai et al. [149] seeks to discover pairwise mixing of three protocols: Simple Mail Transfer Protocol (SMTP), SSH, and HTTP. Regardless of whether the detection is for a single protocol or a combination of protocols contained in DNS tunnels, the experiment is based on an exploratory investigation utilizing regression analysis.

DL-based approaches may fully employ data structure and sequence information and automatically extract essential characteristics. Lai et al. [150] applied this to a feed-forward neural network without normalizing the dataset. The proposed model has neither specified features nor known malicious samples to train the model. To address the complexity of real-world data, the detection process uses packet bytes as features. However, no feature selection leads to high false positives, making it infeasible. To avoid inadequate feature selection, Liu et al. [151] utilized CNN, which can realize automatic feature extraction to construct the detection model. The suggested technique can also learn sequential and structural information in a single DNS packet, something typical ML algorithms cannot do. The authors take full advantage of the information by converting DNS packets into bytes and applying byte-level CNN on the dataset collected through Iodine, Dns2tcp, Dnscat2, OzymanDNS, and ReverseDNShell and comparing existing ML algorithms such as SVM, LR, and neural networks with the proposed model.

Zhang et al. [152] demonstrated the efficacy of identifying a covert channel prior to data exfiltration so that the network security system may instantly stop DNS tunneling. The proposed model utilizes DNS query payloads as the predictive variable instead of query length or query ratio. They created features such as domain name length, character proportion, a randomness feature, and semantic feature composition. The system is developed using models included a Dense Neural Network, 1D-CNN, RNN-LSTM, and RNN-GRU. Based on the scoring algorithm's combination rule, the 1D-CNN model with the best Matthews Correlation Coefficient (MCC) value was chosen as the one-model detection decision maker, and the best three models were chosen as the multimodel detection decision maker. 1D-CNN was investigated by Palau et al. [153] for its potential use in lexicographical DNS tunnel discovery. Due to a scarcity of datasets for assessing DNS tunneling connections, they employ their own dataset. The dataset was created on a virtual machine infrastructure using a time injection methodology. The model achieved a false positive rate of close to 0.8%. Detecting malicious payloads from a single DNS query is critical for detecting DNS tunneling.

Sakarkar et al. [154] performed a comparative study between different DL models such as 1D-CNN, simple RNN, LSTM, and GRU over general datasets. The implementation of DL models in late 2021 [149], [155], [156], indicates the potential for improvement or enhancement in DNS tunneling detection systems through DL.

1) CURRENT CHALLENGES AND FUTURE DIRECTIONS

The issue with identifying DNS tunneling activity, in general, is that each malicious family may act differently. The highlighted features might be controlled by modifying the volume of data delivered or the length, by making small, infrequent queries. Tunneling designed to be undetected by classifiers may become more widespread in the future, and ML models may require characteristics on the non-encrypted side of the stack to become more recognizable. Research in the area suggests that models can improve with a larger dataset that includes the majority of current DNS tunneling query behaviors. Furthermore, by having a pre-existing list of recognized valid queries, the operational cost of verifying each DNS tunneling query may be decreased. A collection of known authentic and malicious queries would help the cache miss technique to detect the portion of the query that reflects known fraudulent queries. Orphan DNS queries could also be used to identify DNS tunneling since they lack a comparable request from some other service, such as HTTP. Orphan DNS queries, on the other hand, may be lawfully utilized by security devices and programs for IP address search queries. Outliers in the dataset were detected using flaggable features.

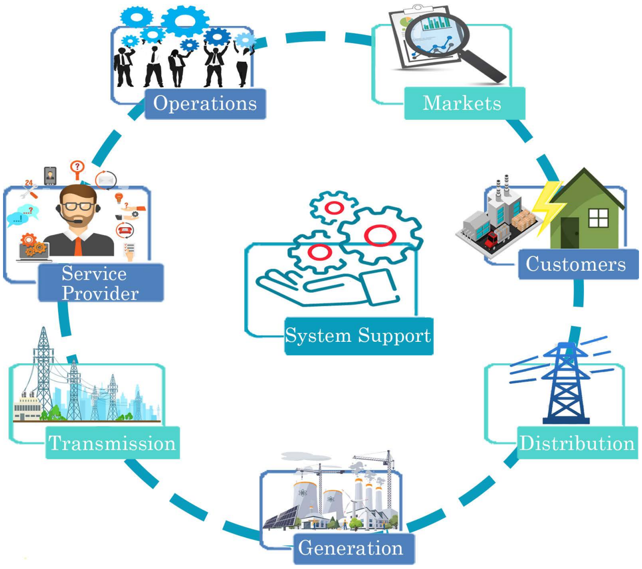

FIGURE 7. Smart grid domains (adapted from [158]).

F. SMART GRID FORENSICS

Smart Grid forensics research is mostly used to identify security issues with a smart electrical grid system. Smart grid forensics may also conduct cybercrime investigations, including hacking, data theft, and so on [157]. Different domains associated with smart grid are shown in Figure 7.

The smart grid system was proposed by Arnold et al. [159], which promises frameworks and road-maps for an efficient and intelligent approach to managing energy supply and consumption. The adoption of different AI techniques including ML and DL to resolve complex problems was already well established at that point. There are three critical factors to security in the smart grid network, namely confidentiality, integrity, and availability. Zhang et al. [160] proposed a distributed IDS and deployed it on a three-layer network. The AIS [161] algorithms (CLONALG or AIRS2Parallel) based model considers security for both physical power systems and communication systems. It is capable of efficiently and effectively analyzing network traffic in order to assess whether an attack is taking place, what form of attack it is, and where it is coming from in the network. Experiments through multiple simulations demonstrate the effectiveness of the model. The smart grid infrastructure consists of heterogeneous and homogeneous devices that may be vulnerable to different attacks, such as implant attacks, black hole attacks, and malicious handheld terminals. Baig [162] proposed a lightweight model based on Graph Neuron, which is a decentralized pattern recognition algorithm for smart grid infrastructure to detect the various type of attacks. It creates an associative memory structure by linking individual device readings in a graph-like pattern. Due to its lightweight characteristic, it is affordable and feasible for deployment with time-bound applications.

The automated generation control (AGC) loop is one of the communication-dependent systems in an electric grid, and the power modulation controller (PMC) is used as a damping controller to regulate system performance. Both are vulnerable to attacks such as false data injection, which disrupt the stability of power systems. Recently, a multi-agent system consisting of a master agent and several agents was proposed by Roy et al. [163]. For intrusion detection, a master agent with a one-class classifier (OCC) is used to examine large area signals. The model uses a unique ML training approach to fit an OCC capable of detecting data-availability and data-integrity attacks on high-voltage direct current (HVDC) systems, AGC, and PMC. Training of OCC is done with predefined parameters, where feature extraction takes place using the binary classification algorithm. The dataset used to train the proposed model is simulated, and a comparison is done against the state-of-the-art algorithms of ML & DL such as SVM, DT, k-NN, 1D-CNN, LSTM, etc.

A low detection rate and high false alarms are the current issues when employing IDS to detect malicious activities in any field. To improve the smart grid's security and reduce high false alarms, Khoei et al. [164] investigate ensemble learning methods, i.e., bagging-based, boosting-based, and stacking-based, over the CIC-DDoS2019 benchmark dataset that contains a lot of DDoS attacks for anomaly IDS. The impact of two different methods of feature selection, namely *Pearson's Correlation Coefficient*, which computes the strength of the linear relationship between two characteristics whose values are between -1 and 1 and *Extra Tree Classifier*, which only rates the relevance of characteristics and deletes unnecessary ones from the dataset [165], is investigated. The authors used the 5-fold validation and grid-search hyperparameter methods to train and validate the proposed classifiers. The stacking method outperforms other ensemble learning methods and traditional ML algorithms.

Smart grid communication data must be protected against two types of attacks: passive attacks, which affect confidentiality, and active attacks that affect availability or integrity. To complement both, Prasad et al. [167] suggested a physical layer security solution be considered as the first line of defense in their work for intrusion detection in smart grids. The presented scheme uses the ML model SVM and AdaBoost, which helps to provide an accurate classification result and is capable of identifying and tracking down a malicious communication node. To recognize and locate an active intruder, exploit PLC channel state information (CSI) that depends on the power line's physical characteristics and is naturally calculated by PLC modems. A change in the nature of one of these characteristics affects the estimated CSI. The suggested technique may be deployed as a standalone model or in conjunction with an existing IDS to detect active and passive network eavesdroppers. The entire system does not require any physical devices; instead, it communicates using the grid's existing PLC modems.

Privacy preservation is essential in network communication applications. Blockchain technology has been used in smart power networks to authenticate meter nodes. However, significant resources and time are required to process the data. A privacy-preserving architecture was developed to accomplish privacy and security at the same time by Keshk et al. [168]. The proposed framework is built on two levels: the first module is focused on verifying data integrity using proof of work blockchain and using a variational AutoEncoder to modify data, and the second is an anomaly detection module for training and evaluating the output of the first module. An AutoEncoder is used along with eight input features to form data into an encoded shape to avoid inference attacks. Using two public datasets, UNSW-NB15 and ICS power systems, the anomaly detection module trains and validates the outputs of the two-level privacy module using a DL approach through LSTM. Experiments demonstrate its competitiveness versus cutting-edge approaches for data protection and anomaly detection. Similarly, Yao et al. [169] suggested an energy theft detection approach based on *Paillier Homomorphic* algorithm and CNN to protect energy privacy. The research takes into account a network model that comprises a local area network, a control center, users, and a trusted third party. A CNN was employed to detect unusual stealing activity, whereas the *Paillier Homomorphic* algorithm used in the proposed model aims at safeguarding the confidentiality of users, which is one of the major concerns in smart grid. The State Grid Cooperation of China (SGCC) dataset was used to evaluate performance. To meet data integrity and confidentiality requirements, Jakaria et al. [170] suggested a safety inspection for AMI Networks, focusing on protection against physical manipulation and infiltration. The study presents two major defensive strategies for attack situations, such as modifying data traveling through itself and manipulating selected data to attack a particular node, naming them as ‘‘Suspicious Node Detection and Anomaly Detection Technique’’. The study's goal is to anticipate the veracity of incoming data recorded by smart meters. The suggested model is trained on the data collected from the smart meters' layout by the ‘‘Irish Social Science Data Archive Center’’.

Many other pieces of research have been done by the researchers to secure the smart grid system and detect malicious activities by using DL techniques. The survey has been done by Zhang et al. [171].

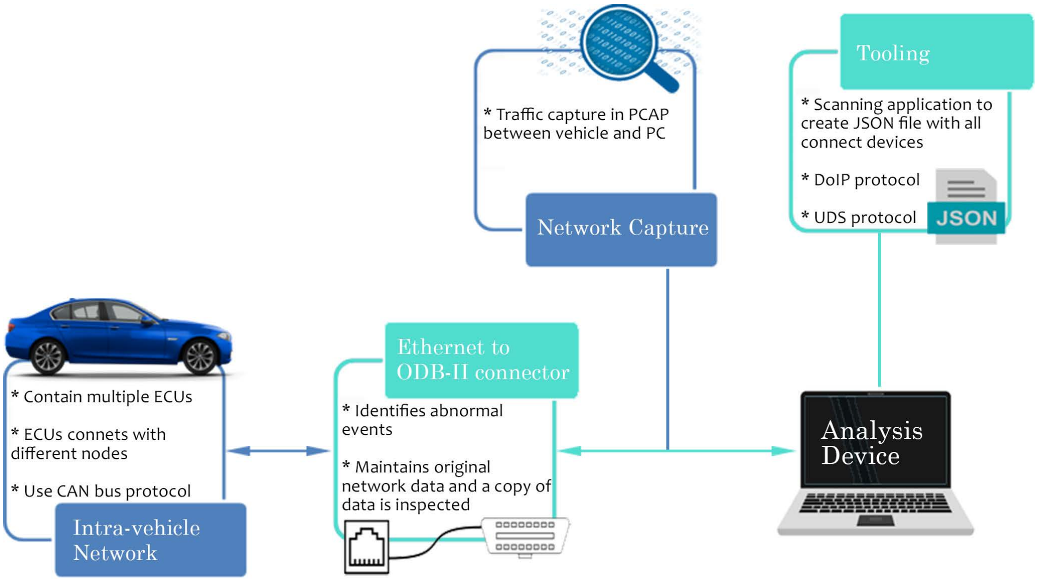
* Traffic capture in PCAP between vehicle and PC
Network Capture
Tooling
* Scanning application to create JSON file with all connect devices
* DoIP protocol
* UDS protocol
JSON
* Contain multiple ECUs
* ECUs connets with different nodes
* Use CAN bus protocol
Intra-vehicle Network
Ethernet to ODB-II connector
* Identifies abnormal events
* Maintains original network data and a copy of data is inspected
Analysis Device

FIGURE 8. Vehicular network data acquisition setup (adapted from [166]).

1) CURRENT CHALLENGES AND FUTURE DIRECTIONS

The heterogeneous nature of smart grid systems poses a significant challenge as well as a possible threat to smart grid security. Many current solutions still have significant limitations. Data aggregation and protocol translation are required for device communication. However, such aggregation might lead to incidental breaches and vulnerabilities, since a characteristic in one protocol can not be adequately translated into another. A broad range of issues with communication protocols, operating systems, and hardware in the smart grid might expose the system to a broad range of attacks. Operating systems can lack appropriate security features since they are developed for automation control components. New, more secure protocols for smart grid networks can provide for improved confidentiality, privacy, integrity, and transparency. Furthermore, the majority of physical devices are outdated, while others have insufficient memory space and low processing capability, making them incapable of supporting new security procedures.

IDS, firewalls, and encryption technologies all play important roles in safeguarding traditional networks. These technologies have significant limitations and are unsuitable for a distributed environment with varying application needs, including latency and bandwidth. IDSs also have a number of shortcomings, e.g., high false positive rates and significant manual investigation of alerts to determine their legitimacy. ML/DL approaches to increase the performance of these systems need substantially large datasets, which are not commonly shared. Though some organizations released datasets, these do not contain real-world data acquired from real attacks. As a result, there remains a need for producing and sharing datasets to train and validate AI models. These technologies are incapable of mitigating upcoming attacks. Smart grids are divided into logical domains, and security needs vary from one domain to the next. Scalable techniques that can collect sufficient evidence from the system need to be improved. Incidents that occur moments before a breakdown may provide useful information about system weaknesses.

G. VEHICULAR FORENSICS

The capture and analysis of digital evidence from motor vehicles is a part of digital vehicle forensics. This evidence can assist in the investigation of crimes involving a motor vehicle or in determining the cause of an automobile accident [172]. The data acquisition setup from the vehicle is shown in Figure 8.

The Driving Ad Hoc Networking Infrastructure (DAHNI) solution for delivering driver assistance is presented by Zarki et al. [173]. Based on three main assumptions, i.e., location awareness: vehicles equipped with GPS; ad hoc networking: vehicles equipped with computing components; and access to fixed infrastructure: to upload data from vehicles. The authors did an analysis of how they may utilize a vehicular network to track surrounding cars and alert the driver to potential risks. Security and privacy threats that vehicle networks are vulnerable to are highlighted in [174] and [173]. Based on the information given for authentication, malicious cars might track the actions of the targeted driver. Hubaux et al. [174] attempted to overcome these issues by employing anonymity techniques and temporary pseudonyms. Considering key sizes and authentication delays, Hubaux [175] investigated the challenges associated with key management for vehicle networks. The researchers proposed a vehicular public key infrastructure that includes a certificate authority (CA) that supplies cars with public/private key pairs. The authors assume that automobiles have electronic identities in the form of an electronic license plate or electronic chassis number. To protect privacy and avoid monitoring, each car receives a huge number of short-lived anonymous key certificates that do not include the vehicle ID. To avoid location monitoring, public/private key pairs must be updated on a regular basis. The authors examined the computational complexity and signature size of three public-key cryptosystems (PKCS). The proposed digital signature is based on the elliptic curve digital signature algorithm (ECDSA), which aids in packet size reduction.

A controller area network (CAN) bus connects all electronic control units (ECUs) in a vehicle to transfer messages and execute actions. However, modern vehicles increase connectivity and complexity. Security issues have become a significant concern in Vehicle-to-Everything (V2X). Smart vehicles contain information about the vehicle as well as the driver. To avoid hacking and car theft, the vehicle must ensure that the driver's identity and profile are valid. A recent study carried out by Talpur and Gurusamy [176] to categorize ML techniques based on their utilization in V2X applications and methodologies, along with the working principles of these ML techniques in solving various security concerns, including attacks, privacy, trust, intrusion detection, and driver identification/fingerprinting, were reviewed. Not long ago, Martinelli et al. [177] demonstrated how ML algorithms may help distinguish between genuine automobile owners and reprobates using features from the CAN. The intake air pressure (used to calculate air density and determine the engine's air mass flow rate) and torque of friction (defined by the authors as the frictional force when two objects come into contact) were used to build the classification models. Ten different ML models were analyzed, including k-NN, SVM, DT, NB, and RF. Security weaknesses of *CAN* were discussed and addressed by D'Angelo et al. [178] through a cluster-based multidimensional model that is used to detect the DoS, fuzzy attacks, GEAR attack, and RPM attacks. The authors mine key features from data associated with various messages traveling on the CAN bus in an unsupervised manner. Javed et al. [179] also addresses the CAN bus communication by using novel approach CANintelliDS, which is based on DL models namely CNN and GRU-based attention. Compared to conventional techniques such as RF, LR, CNN, and so on, CANintelliDS fared well in identifying monomer and hybrid attacks.

Traditional misbehavior detection approaches are successful at preventing intrusions, but they fall short of safeguarding V2V communication. [180] proposed a data-centric approach that identifies erroneous information transmitted between Internet-of-Vehicles (IoV) with a lightweight statistical approach relevant for real-time safety applications. This research is unique in that it combines location and movement plausibility tests with typical supervised ML techniques to improve the accuracy of results. In addition to detecting misbehavior, the model identifies attack types to aid in validating counter measures. The authors evaluated the effectiveness of six supervised ML techniques, including SVM, RF, k-NN, NB, and ensemble learning.

Mekki et al. [181] combined driver behavior data with a DL system to ensure driver identification while accounting for abnormalities. They provided a comprehensive driver fingerprint identification solution based on CNN and RNN. The authors consider driver personal data to be a time series, and driver identification to be a multivariate time series classification. The proposed model was trained on smartphone sensors and the vehicle's ECU. Many traditional security mechanisms are unsuitable for IVNs due to the violating timing constraints of CAN communications. Recently, Yang et al. [182] addresses internal and external network's weaknesses and proposes the first multi-tiered hybrid IDS (MTH-IDS) for the detection of known and undiscovered threats. The proposed MTH-IDS framework has both signature-based IDS and anomaly-based IDS, where the data preprocessing is performed through the K-means algorithm. For unsupervised learner optimization, two biased classifiers and a Bayesian optimization with Gaussian Process (BO-GP) technique were used. The proposed solution is evaluated using two public network data sets, namely CAN-intrusion-dataset [183] and the CIC-IDS2017 dataset. Various metrics are used to analyze the model's feasibility, performance, and efficiency.

According to Cai et al. [184], a superior traffic feature extraction approach not only lowers duplicate features but also improves network convergence performance. They suggested an effective hybrid parallel deep learning Model (HPM) for IDS. Instead of using CICFlowMeter's standard way to extract the feature, the authors propose a novel dataset creation methodology for ISCX 2012 [40] and CIC-IDS 2017, and regard data flow as a detection object. To minimize computing complexity and condense data dissemination, flow splitting, traffic cleaning, and traffic tailoring stages are used. HPM utilizes a hybrid parallel structure as a training module and a double-LSTM as a temporal feature extraction module, rather than a single model. The approach filters the beneficial local and global characteristics from processed data flow and forecasts the data flow's future changing behavior and occurrence probability from a batch of time series. The CosMargin discriminative classification algorithm is introduced, which indirectly implements a margin boundary on the feature layer to distinguish benign/malicious traffic.

1) CURRENT CHALLENGES AND FUTURE DIRECTIONS

Some vehicular agents may intentionally or unintentionally broadcast false data as a result of a malfunction in an embedded sensor, which may then be transmitted to servers for training regardless of its trustworthiness. This fictitious information may give AI models inaccurate data and make them vulnerable to making incorrect decisions. Among the most serious challenges in vehicular networks are jammer attacks, cache pollution and void announcements. A jammer can switch the attack operation mode (run/sleep) at any point. Cache pollution attacks require protective solutions, since the sole option recommended is to prevent caching for certain items. This affects/disables the fundamental operation of the networks. A void announcement attack is where a malicious vehicle announces availability of certain data, but then does not respond to requests for this. In addition to IDS at the data connection and physical levels, the more technologically advanced vehicles require an effective defensive system at the application layer to protect them from attacks. Due to the dispersed nature of the automotive environment, there are additional issues to study, such as disclosing private vehicular information or corrupting local data and federated learning models. The proposed models to handle such issues still have limitations such as computational complexity and unsatisfactory performance on a large number of vehicles. Because of the computational costs imposed by ML/DL architectures, resource management is critical to achieving usable, practical, and successful AI solutions. The offloading of heavy computational DL can lead to faster and more secure vehicular systems. One of the obstacles posed by intellectual property concerns is obtaining confidential details from the vendors of the various vehicle parts throughout the investigation. Vehicle producers also face the problem of reputational and legal risks associated with releasing intelligence. The makers might face substantial financial, legal, and reputational consequences as a result of this.

IV. CONCLUSION

This paper presents the state-of-the-art in the application of AI across several domains within network forensics. It provides an overview of AI approaches previously used on datasets relevant to network forensics, and highlights the current challenges and future directions in network forensics. Addressing the constraints and problems in tools and ecosystem implementations can assist investigators to conduct more efficient and reliable network investigations.

Undoubtedly, developments or advancements in any sector or domain, either through AI or other means, are impossible to achieve without the core ingredient – data. This is still the main constraint across many domains due to the dearth of sufficiently large, clean, labeled databases or limitations on the access of the available datasets. One contributing factor is that many organizations avoid revealing information regarding security breaches due to the heightened risk of reputational harm that could ultimately result in financial loss. One tactic for mitigating the problem is to bridge the gap between blue-chip companies and academic/research institutions by signing non-disclosure agreements (NDA) for the advancement and enhancement of the field. This could be coupled with sufficient anonymization of the data itself. Various AI approaches, such as ML, DL, ensemble learning, and others, are being used in network forensics. However, different algorithms, hybrid techniques, and approaches can be employed for future work. Furthermore, while deploying AI for network forensics, reducing false-positives is currently the most pressing issue that needs to be addressed.

Cloud computing is one of the fastest growing sectors in today's world, and it generates a large volume of different types of logs including host logs, application logs, API logs, and many more. The gathering of crucial logs in IDS and standardization of all created logs in a centralized repository will assist in enhancing cloud forensics using AI, which will undoubtedly help to speed up the process and effectiveness of manual processes. Smart cities, IoT, cloud computing, and a slew of other domains are now reliant on fog devices, posing issues in terms of complex development and resource consumption. Despite the fact that federated ML has emerged as a new wave of AI based on the decentralization of data learning at the network's edge, it still requires development through the implementation of innovative ideas. Many ransomware attacks result in financial loss and can damage the reputations of well-known organizations. This demonstrates one significant challenge. Namely the prevention and detection of ransomware attacks by improving the detection of non-trusted sources, improving or proposing strong authentication mechanisms, and applying AI for cyber hygiene.

The popularity of virtualization in the current era corresponds to a prevalence of software-defined networks (SDNs). In the initial development of software defined network architectures, increased focus was its functional benefits and the security aspect somewhat lagged behind. However, improvements in SDNs and the fact that it is a centralized controlled infrastructure increases the network threats – including the hijacking of SDN controllers. Improving the security in the existing layers or adding a security layer can be employed in the future to secure the SDN and avail its maximum benefits. The utilization of emerging blockchain and AI technologies together can help to address various security and investigatory challenges.

REFERENCES

[1] G. Palmer, "A road map for digital forensics research," Digital Forensics Res. Workshop (DFRWS), Utica, NY, USA, Tech. Rep. DTR-T001-01, 2001.

[2] J. R. Vacca, *Computer and Information Security Handbook*. San Mateo, CA, USA: Morgan Kaufmann, 2012.

[3] D. W. McKee, S. J. Clement, J. Almutairi, and J. Xu, "Survey of advances and challenges in intelligent autonomy for distributed cyber-physical systems," *CAAI Trans. Intell. Technol.*, vol. 3, no. 2, pp. 75–82, Jun. 2018.

[4] F. Jiang, Y. Jiang, H. Zhi, Y. Dong, H. Li, S. Ma, Y. Wang, Q. Dong, H. Shen, and Y. Wang, "Artificial intelligence in healthcare: Past, present and future," *Stroke Vascular Neurol.*, vol. 2, no. 4, pp. 1–14, 2017.

[5] X.-K. Liao, K. Lu, C.-Q. Yang, J.-W. Li, Y. Yuan, M.-C. Lai, L.-B. Huang, P.-J. Lu, J.-B. Fang, J. Ren, and J. Shen, "Moving from exascale to zettascale computing: Challenges and techniques," *Frontiers Inf. Technol. Electron. Eng.*, vol. 19, no. 10, pp. 1236–1244, Oct. 2018, doi: 10.1631/FITEE.1800494.

[6] J. Flynn and C. Giannetti, "Using convolutional neural networks to map houses suitable for electric vehicle home charging," *AI*, vol. 2, no. 1, pp. 135–149, Mar. 2021.

[7] M. Reith, C. Carr, and G. Gunsch, "An examination of digital forensic models," *Int. J. Digit. Evidence*, vol. 1, no. 3, pp. 1–12, 2002.

[8] K. Kent, S. Chevalier, T. Grance, and H. Dang, "Guide to integrating forensic techniques into incident response," *Nat. Inst. Standards Technol.*, vol. 10, no. 14, pp. 86–800, 2006.

[9] X. Du, N.-A. Le-Khac, and M. Scanlon, "Evaluation of digital forensic process models with respect to digital forensics as a service," in *Proc. 16th Eur. Conf. Cyber Warfare Secur. (ECCWS)*, Dublin, Ireland: ACPI, Jun. 2017, pp. 573–581.

[10] Cisco. (2020). *Cisco Annual Internet Report (2018–2023)*. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[11] Cisco. (2018). *VNI Complete Forecast Highlights*. [Online]. Available: https://www.cisco.com/c/dam/m/en_us/solutions/service-provider/vni-forecast-highlights/pdf/Global_Device_Growth_Traffic_Profiles.pdf

[12] E. V. D. Wiel, M. Scanlon, and N.-A. Le-Khac, "Enabling non-expert analysis of large volumes of intercepted network traffic," in *Proc. IFIP Int. Conf. Digit. Forensics*. New Dehli, India: Springer, 2018, pp. 183–197.

[13] K. E. Pavlou and R. T. Snodgrass, "Forensic analysis of database tampering," *ACM Trans. Database Syst.*, vol. 33, no. 4, pp. 1–47, Nov. 2008.

[14] J. Farina, M. Scanlon, N.-A. Le-Khac, and M.-T. Kechadi, "Overview of the forensic investigation of cloud services," in *Proc. 10th Int. Conf. Availability, Rel. Secur.*, Aug. 2015, pp. 556–565.

[15] X. Du, C. Hargreaves, J. Sheppard, F. Anda, A. Sayakkara, N.-A. Le-Khac, and M. Scanlon, "SOK: Exploring the state of the art and the future potential of artificial intelligence in digital forensic investigation," in *Proc. 15th Int. Conf. Availability, Rel. Secur.* New York, NY, USA: Association for Computing Machinery, 2020, doi: 10.1145/3407023.3407068.

[16] E. S. Pilli, R. C. Joshi, and R. Niyogi, "Network forensic frameworks: Survey and research challenges," *Digit. Invest.*, vol. 7, nos. 1–2, pp. 14–27, Oct. 2010.

[17] T. Manesh, B. Brijith, T. M. Bhraguram, R. Rajaram, and V. K. Bhadran, "Network forensic investigation of HTTPS protocol," *Int. J. Modern Eng. Res.*, vol. 3, no. 5, pp. 3096–3106, Sep./Oct. 2013.

[18] R. Beverly, S. Garfinkel, and G. Cardwell, "Forensic carving of network packets and associated data structures," *Digit. Invest.*, vol. 8, pp. S78–S89, Aug. 2011.

[19] Q. Chen, "Toward realizing self-protecting healthcare information systems: Design and security challenges," in *Advances in Computers*, vol. 114. Amsterdam, The Netherlands: Elsevier, 2019.

[20] M. Uma and G. Padmavathi, "A survey on various cyber attacks and their classification," *Int. J. Netw. Secur.*, vol. 15, no. 5, pp. 390–396, Sep. 2013.

[21] T. Vaidya, "2001–2013: Survey and analysis of major cyberattacks," 2015, *arXiv:1507.06673*.

[22] OWASP. (2021). *OWASP Top 10:2021*. [Online]. Available: https://owasp.org/Top10/

[23] I. Etikan, "Comparision of snowball sampling and sequential sampling technique," *Biometrics Biostatistics Int. J.*, vol. 3, no. 1, p. 55, Jan. 2016.

[24] R. W. Schlosser, O. Wendt, S. Bhavnani, and B. Nail-Chiwetalu, "Use of information-seeking strategies for developing systematic reviews and engaging in evidence-based practice: The application of traditional and comprehensive pearl growing. A review," *Int. J. Lang. Commun. Disorders*, vol. 41, no. 5, pp. 567–582, Jan. 2006.

[25] K. Wright, S. Golder, and R. Rodriguez-Lopez, "Citation searching: A systematic review case study of multiple risk behaviour interventions," *BMC Med. Res. Methodol.*, vol. 14, no. 1, pp. 1–8, Dec. 2014.

[26] D. Moher, A. Liberati, J. Tetzlaff, D. G. Altman, and G. Prisma, "Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement," *Ann. Internal Med.*, vol. 151, no. 4, pp. 264–269, 2009.

[27] X. Du, C. Hargreaves, J. Sheppard, and M. Scanlon, "TraceGen: User activity emulation for digital forensic test image generation," *Forensic Sci. Int., Digit. Invest.*, vol. 38, Oct. 2021, Art. no. 301133.

[28] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *J. Inf. Secur. Appl.*, vol. 50, Feb. 2020, Art. no. 102419.

[29] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy," in *Proc. Int. Carnahan Conf. Secur. Technol. (ICCST)*, Oct. 2019, pp. 1–8.

[30] G. Creech, "Developing a high-accuracy cross platform host-based intrusion detection system capable of reliably detecting zero-day attacks," Ph.D. dissertation, School Eng. Inf. Technol., Univ. New South Wales, Canberra, NSW, Australia, 2014.

[31] P. P. Biswas, H. C. Tan, Q. Zhu, Y. Li, D. Mashima, and B. Chen, "A synthesized dataset for cybersecurity study of IEC 61850 based substation," in *Proc. IEEE Int. Conf. Commun., Control, Comput. Technol. for Smart Grids (SmartGridComm)*, Oct. 2019, pp. 1–7.

[32] M. Nasr, A. Bahramali, and A. Houmansadr, "DeepCorr: Strong flow correlation attacks on tor using deep learning," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2018, pp. 1962–1976.

[33] S. Mahdavifar, A. F. Abdul Kadir, R. Fatemi, D. Alhadidi, and A. A. Ghorbani, "Dynamic Android malware category classification using semi-supervised deep learning," in *Proc. IEEE Intl Conf Dependable, Autonomic Secure Comput., Intl Conf Pervasive Intell. Comput., Intl Conf Cloud Big Data Comput., Intl Conf Cyber Sci. Technol. Congr. (DASC/PiCom/CBDCom/CyberSciTech)*, Aug. 2020, pp. 515–522.

[34] Canadian Institute for Cybersecurity. *A Realistic Cyber Defense Dataset (CSE-CIC-IDS2018)*. Accessed: Jun. 2, 2022. [Online]. Available: https://registry.opendata.aws/cse-cic-ids2018

[35] S. García, M. Grill, J. Stiborek, and A. Zunino, "An empirical comparison of botnet detection methods," *Comput. Secur.*, vol. 45, pp. 100–123, Sep. 2014.

[36] S. Peisert, R. Gentz, J. Boverhof, C. McParland, S. Engle, A. Elbashandy, and D. Gunter, "LBNL open power data," LBNL, Berkeley, CA, USA, Tech. Rep., 2017. [Online]. Available: https://escholarship.org/uc/item/92f19749, doi: 10.21990/C21599.

[37] A. Habibi Lashkari, G. Draper Gil, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of tor traffic using time based features," in *Proc. 3rd Int. Conf. Inf. Syst. Secur. Privacy*, 2017, pp. 253–262.

[38] L. Taheri, A. F. A. Kadir, and A. H. Lashkari, "Extensible Android malware detection and family classification using network-flows and API-calls," in *Proc. Int. Carnahan Conf. Secur. Technol. (ICCST)*, Oct. 2019, pp. 1–8.

[39] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy*, vol. 1, Jan. 2018, pp. 108–116.

[40] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, "Toward developing a systematic approach to generate benchmark datasets for intrusion detection," *Comput. Secur.*, vol. 31, no. 3, pp. 357–374, 2012.

[41] T. H. Morris, Z. Thornton, and I. Turnipseed, "Industrial control system simulation and data logging for intrusion detection system research," in *Proc. 7th Annu. Southeastern Cyber Secur. Summit*, 2015, pp. 3–4.

[42] M. S. I. Mamun, M. A. Rathore, A. H. Lashkari, N. Stakhanova, and A. A. Ghorbani, "Detecting malicious URLs using lexical analysis," in *Proc. Int. Conf. Netw. Syst. Secur.* Taipei, Taiwan: Springer, 2016, pp. 467–482.

[43] A. H. Lashkari, A. F. A. Kadir, L. Taheri, and A. A. Ghorbani, "Toward developing a systematic approach to generate benchmark Android malware datasets and classification," in *Proc. Int. Carnahan Conf. Secur. Technol. (ICCST)*, Oct. 2018, pp. 1–7.

[44] Canadian Institute for Cybersecurity. (2017). *CIC DoS Dataset*. Accessed: Jun. 2, 2022. [Online]. Available: https://www.unb.ca/cic/datasets/dos-dataset.html

[45] Canadian Institute for Cybersecurity. (2009). *NSL-KDD Dataset*. Accessed: Jun. 2, 2022. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html

[46] S. Pan, T. Morris, and U. Adhikari, "Developing a hybrid intrusion detection system using data mining for power systems," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 3104–3113, Nov. 2015.

[47] G. Maciá-Fernández, J. Camacho, R. Magán-Carrión, P. García-Teodoro, and R. Therón, "UGR '16: A new dataset for the evaluation of cyclostationarity-based network IDSs," *Comput. Secur.*, vol. 73, pp. 411–424, Mar. 2018.

[48] A. H. Lashkari, A. F. A. Kadir, H. Gonzalez, K. F. Mbah, and A. A. Ghorbani, "Towards a network-based framework for Android malware detection and characterization," in *Proc. 15th Annu. Conf. Privacy, Secur. Trust (PST)*, Aug. 2017, pp. 233–23309.

[49] Center for Applied Internet Data Analysis (CAIDA). (2017). *CAIDA's Dataset*. Accessed: Jun. 2, 2022. [Online]. Available: https://www.caida.org/data/overview

[50] Kyoto University. (2006). *Traffic Data From Kyoto University's Honeypots*. Accessed: Jun. 2, 2022. [Online]. Available: http://www.takakura.com/Kyoto_data/

[51] MAWILab. *Mawilab Dataset*. Accessed: Jun. 2, 2022. [Online]. Available: http://www.fukuda-lab.org/mawilab/data.html

[52] Canadian Institute for Cybersecurity. *Android Validation Dataset*. Accessed: Jun. 2, 2022. [Online]. Available: https://www.unb.ca/cic/datasets/android-validation.html

[53] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, Nov. 2015, pp. 1–6.

[54] Heritrix Team. (2010). *Heritrix dataset*. Accessed: Jun. 2, 2022. [Online]. Available: http://crawler.archive.org/index.html

[55] M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. M. Havinga, "Fusion of smartphone motion sensors for physical activity recognition," *Sensors*, vol. 14, no. 6, pp. 10146–10176, 2014.

[56] KDD Cup 1999 Data. (1999). *The Third International Knowledge Discovery and Data Mining Tools Competition*. Accessed: Jun. 2, 2022. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[57] P. Hines, S. Blumsack, E. C. Sanchez, and C. Barrows, "The topological and electrical structure of power grids," in *Proc. 43rd Hawaii Int. Conf. Syst. Sci.*, 2010, pp. 1–10.

[58] G. Szabó, D. Orincsay, S. Malomsoky, and I. Szabó, "On the validation of traffic classification algorithms," in *Proc. Int. Conf. Passive Active Netw. Meas.* Cleveland, OH, USA: Springer, 2008, pp. 72–81.

[59] I. Homoliak, M. Barabas, P. Chmelar, M. Drozd, and P. Hanacek, "ASNM: Advanced security network metrics for attack vector description," in *Proc. Int. Conf. Secur. Manag. (SAM)*, in The Steering Committee of The World Congress in Computer Science, 2013, pp. 1–9.

[60] MIT Lincoln Laboratory. (Feb. 1998). *1998 DARPA Intrusion Detection Evaluation Dataset*. Accessed: Jun. 2, 2022. [Online]. Available: https://www.ll.mit.edu/r-d/datasets/1998-darpa-intrusion-detection-evaluation-dataset

[61] S. Dadkhah, H. Mahdikhani, P. K. Danso, A. Zohourian, K. A. Truong, and A. A. Ghorbani, "Towards the development of a realistic multidimensional IoT profiling dataset," in *Proc. 19th Annu. Int. Conf. Privacy, Secur. Trust (PST)*, Aug. 2022, pp. 1–11.

[62] S. Mahdavifar, N. Maleki, A. H. Lashkari, M. Broda, and A. H. Razavi, "Classifying malicious domains using DNS traffic analysis," in *Proc. IEEE Intl Conf Dependable, Autonomic Secure Comput., Intl Conf Pervasive Intell. Comput., Intl Conf Cloud Big Data Comput., Intl Conf Cyber Sci. Technol. Congr. (DASC/PiCom/CBDCom/CyberSciTech)*, Oct. 2021, pp. 60–67.

[63] Hacking and Countermeasure Research Lab. (2017). *Car-Hacking Dataset*. Accessed: Jun. 2, 2022. [Online]. Available: https://ocslab.hksecurity.net/Datasets/car-hacking-dataset

[64] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of encrypted and VPN traffic using time-related features," in *Proc. 2nd Int. Conf. Inf. Syst. Secur. Privacy*, 2016, pp. 407–414.

[65] E. Biglar Beigi, H. Hadian Jazi, N. Stakhanova, and A. A. Ghorbani, "Towards effective feature selection in machine learning-based botnet detection approaches," in *Proc. IEEE Conf. Commun. Netw. Secur.*, Oct. 2014, pp. 247–255.

[66] M. Erfani, F. Shoeleh, S. Dadkhah, B. Kaur, P. Xiong, S. Iqbal, S. Ray, and A. A. Ghorbani, "A feature exploration approach for IoT attack type classification," in *Proc. IEEE Intl Conf Dependable, Autonomic Secure Comput., Intl Conf Pervasive Intell. Comput., Intl Conf Cloud Big Data Comput., Intl Conf Cyber Sci. Technol. Congr. (DASC/PiCom/CBDCom/CyberSciTech)*, Oct. 2021, pp. 582–588.

[67] M. MontazeriShatoori, L. Davidson, G. Kaur, and A. Habibi Lashkari, "Detection of DoH tunnels using time-series classification of encrypted traffic," in *Proc. IEEE Intl Conf Dependable, Autonomic Secure Comput., Intl Conf Pervasive Intell. Comput., Intl Conf Cloud Big Data Comput., Intl Conf Cyber Sci. Technol. Congr. (DASC/PiCom/CBDCom/CyberSciTech)*, Aug. 2020, pp. 63–70.

[68] R. W. Heijden, T. Lukaseder, and F. Kargl, "VeReMi: A dataset for comparable evaluation of misbehavior detection in VANETs," in *Int. Conf. Secur. Privacy Commun. Syst.* Singapore: Springer, Jun. 2018, pp. 318–337.

[69] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustain. Cities Soc.*, vol. 72, Sep. 2021, Art. no. 102994.

[70] M. Singh, M. Singh, and S. Kaur. TI-2016 DNS Dataset. IEEE Dataport. 2019. [Online]. Available: https://dx.doi.org/10.21227/9ync-vv09

[71] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset," *Future Gener. Comput. Syst.*, vol. 100, pp. 779–796, Nov. 2019.

[72] J. L. Leevy and T. M. Khoshgoftaar, "A survey and analysis of intrusion detection models based on CSE-CIC-IDS2018 big data," *J. Big Data*, vol. 7, no. 1, pp. 1–19, Dec. 2020.

[73] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "A detailed analysis of the CICIDS2017 data set," in *Proc. Int. Conf. Inf. Syst. Secur. Privacy*. Funchal, Portugal: Springer, 2018, pp. 172–188.

[74] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, and A. Hotho, "Flow-based benchmark data sets for intrusion detection," in *Proc. 16th Eur. Conf. Cyber Warfare Security. (ACPI)*, 2017, pp. 361–369.

[75] M. Abbasi, A. Shahraki, and A. Taherkordi, "Deep learning for network traffic monitoring and analysis (NTMA): A survey," *Comput. Commun.*, vol. 170, pp. 19–41, Feb. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366421000426

[76] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl.*, Jul. 2009, pp. 1–6.

[77] J. McHugh, "Testing intrusion detection systems: A critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory," *ACM Trans. Inf. Syst. Secur.*, vol. 3, no. 4, pp. 262–294, 2000, doi: 10.1145/382912.382923.

[78] M. Scanlon, "Battling the digital forensic backlog through data deduplication," in *Proc. 6th Int. Conf. Innov. Comput. Technol. (INTECH)*, Aug. 2016, pp. 10–14.

[79] D. Stern and P. Chemouil, "A diagnosis expert system for network traffic management," in *Proc. Netw. Conf.*, 1992.

[80] U. Lindqvist and P. Porras, "Detecting computer and network misuse through the production-based expert system toolset (P-BEST)," in *Proc. IEEE Symp. Secur. Privacy*, May 1999, pp. 146–161.

[81] B. B. Dunning and J. Switlik, "A real-time expert system for computer network monitor and control," *ACM SIGMIS Database, Adv. Inf. Syst.*, vol. 19, no. 2, pp. 35–38, Aug. 1988, doi: 10.1145/54132.54136.

[82] V. Catania, G. Ficili, S. Palazzo, and D. Panno, "A fuzzy expert system for usage parameter control in ATM networks," in *Proc. GLOBECOM*, vol. 2, 1995, pp. 1338–1342.

[83] A. Knapińska, P. Lechowicz, and K. Walkowiak, "Machine-learning based prediction of multiple types of network traffic," in *Proc. Int. Conf. Comput. Sci.* Kraków, Poland: Springer, 2021, pp. 122–136.

[84] G. Millán, "Traffic flows analysis in high-speed computer networks using time series," 2021, *arXiv:2103.03984*.

[85] S. Dong, "Multi class SVM algorithm with active learning for network traffic classification," *Exp. Syst. Appl.*, vol. 176, Aug. 2021, Art. no. 114885. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417421003262

[86] M. Lotfollahi, M. J. Siavoshani, R. S. H. Zade, and M. Saberian, "Deep packet: A novel approach for encrypted traffic classification using deep learning," *Soft Comput.*, vol. 24, no. 3, pp. 1999–2012, 2020.

[87] S. Mahajan, H. R., and K. Kotecha, "Prediction of network traffic in wireless mesh networks using hybrid deep learning model," *IEEE Access*, vol. 10, pp. 7003–7015, 2022.

[88] H. Gandhi and V. Ribeiro, "Packet batching for reducing system resource consumption for botnet detection using network traffic analysis," in *Proc. 14th Int. Conf. Commun. Syst. Netw. (COMSNETS)*, Jan. 2022, pp. 1–6.

[89] I. P. Possebon, A. S. Silva, L. Z. Granville, A. Schaeffer-Filho, and A. Marnerides, "Improved network traffic classification using ensemble learning," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2019, pp. 1–6.

[90] A. Shahraki, M. Abbasi, A. Taherkordi, and A. D. Jurcut, "Active learning for network traffic classification: A technical study," *IEEE Trans. Cognit. Commun. Netw.*, vol. 8, no. 1, pp. 422–439, Mar. 2022, doi: 10.1109/TCCN.2021.3119062.

[91] C. Oh, J. Ha, and H. Roh, "A survey on TLS-encrypted malware network traffic analysis applicable to security operations centers," *Appl. Sci.*, vol. 12, no. 1, p. 155, Dec. 2021. [Online]. Available: https://www.mdpi.com/2076-3417/12/1/155

[92] A. A. Kashyap, S. Raviraj, A. Devarakonda, S. R. K. Nayak, S. Kv, and S. J. Bhat, "Traffic flow prediction models—A review of deep learning techniques," *Cogent Eng.*, vol. 9, no. 1, Dec. 2022, Art. no. 2010510.

[93] A. Aldweesh, A. Derhab, and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," *Knowl.-Based Syst.*, vol. 189, Feb. 2020, Art. no. 105124. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705119304897

[94] D. Fleurbaaij, M. Scanlon, and N.-A. Le-Khac, "Privileged data within digital evidence," in *Proc. IEEE Trustcom/BigDataSE/ICESS*, Aug. 2017, pp. 737–744.

[95] S. Rizvi, M. Scanlon, J. McGibney, and J. Sheppard, "Deep learning based network intrusion detection system for resource-constrained environments," in *Proc. 13th EAI Int. Conf. Digit. Forensics Cyber Crime (ICDFC)*. Boston, MA, USA: Springer, 2023, pp. 1–7.

[96] D. E. Denning, "An intrusion-detection model," *IEEE Trans. Softw. Eng.*, vol. SE-15, no. 2, pp. 222–232, Feb. 1987.

[97] T. F. Lunt, R. Jagannathan, R. Lee, A. Whitehurst, and S. Listgarten, "Knowledge based intrusion detection," in *Proc. Annu. AI Syst. Government Conf.*, Washington, DC, USA, 1989, pp. 1–6.

[98] H. S. Vaccaro and G. E. Liepins, "Detection of anomalous computer session activity," in *Proc. IEEE Symp. Secur. Privacy*, May 1989, pp. 1–10.

[99] S. Mukkamala, G. Janoski, and A. Sung, "Intrusion detection using neural networks and support vector machines," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, vol. 2, May 2002, pp. 1702–1707.

[100] J. Ren, J. Guo, W. Qian, H. Yuan, X. Hao, and H. Jingjing, "Building an effective intrusion detection system by using hybrid data optimization based on machine learning algorithms," *Secur. Commun. Netw.*, vol. 2019, pp. 1–11, Jun. 2019.

[101] Z. Chen, L. Zhou, and W. Yu, "ADASYN–random forest based intrusion detection model," in *Proc. 4th Int. Conf. Signal Process. Mach. Learn.*, Aug. 2021, pp. 152–159, doi: 10.1145/3483207.3483232.

[102] S. Seth, G. Singh, and K. Kaur Chahal, "A novel time efficient learning-based approach for smart intrusion detection system," *J. Big Data*, vol. 8, no. 1, Dec. 2021, Art. no. 111, doi: 10.1186/s40537-021-00498-8.

[103] A. A. Megantara and T. Ahmad, "Feature importance ranking for increasing performance of intrusion detection system," in *Proc. 3rd Int. Conf. Comput. Informat. Eng. (ICIE)*, Sep. 2020, pp. 37–42.

[104] A. A. Megantara and T. Ahmad, "A hybrid machine learning method for increasing the performance of network intrusion detection systems," *J. Big Data*, vol. 8, no. 1, pp. 1–19, Dec. 2021.

[105] Y. Tang, L. Gu, and L. Wang, "Deep stacking network for intrusion detection," *Sensors*, vol. 22, no. 1, p. 25, Dec. 2021. [Online]. Available: https://www.mdpi.com/1424-8220/22/1/25

[106] M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis, and E. K. Markakis, "A survey on the Internet of Things (IoT) forensics: Challenges, approaches, and open issues," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1191–1221, 2nd Quart., 2020.

[107] P. Sun, P. Liu, Q. Li, C. Liu, X. Lu, R. Hao, and J. Chen, "DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system," *Secur. Commun. Netw.*, vol. 2020, pp. 1–11, Aug. 2020.

[108] M. Azizjon, A. Jumabek, and W. Kim, "1D CNN based network intrusion detection with normalization on imbalanced data," in *Proc. Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, Feb. 2020, pp. 218–224.

[109] R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Applying convolutional neural network for network intrusion detection," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Sep. 2017, pp. 1222–1228.

[110] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017.

[111] Z. Tang, H. Hu, and C. Xu, "A federated learning method for network intrusion detection," *Concurrency Comput., Pract. Exper.*, vol. 34, no. 10, p. e6812, May 2022.

[112] S. Alabdulsalam, K. Schaefer, T. Kechadi, and N.-A. Le-Khac, "Internet of Things forensics–challenges and a case study," in *Proc. IFIP Int. Conf. Digit. Forensics*. New Delhi, India: Springer, 2018, pp. 35–48.

[113] H. Schut, M. Scanlon, J. Farina, and N.-A. Le-Khac, "Towards the forensic identification and investigation of cloud hosted servers through non-invasive wiretaps," in *Proc. 10th Int. Conf. Availability, Rel. Secur.*, Aug. 2015, pp. 249–257.

[114] E. Oriwoh, D. Jazani, G. Epiphaniou, and P. Sant, "Internet of Things forensics: Challenges and approaches," in *Proc. 9th IEEE Int. Conf. Collaborative Comput., Netw., Appl. Worksharing*, 2013, pp. 608–615.

[115] P. M. Shakeel, S. Baskar, H. Fouad, G. Manogaran, V. Saravanan, and C. E. Montenegro-Marin, "Internet of Things forensic data analysis using machine learning to identify roots of data scavenging," *Future Gener. Comput. Syst.*, vol. 115, pp. 756–768, Feb. 2021.

[116] I. Mukherjee, N. K. Sahu, and S. K. Sahana, "Simulation and modeling for anomaly detection in IoT network using machine learning," *Int. J. Wireless Inf. Netw.*, pp. 1–17, Jan. 2022.

[117] F.-X. Aubet, "Machine learning-based adaptive anomaly detection in smart spaces," B.Sc. Thesis, Dept. Inform., Technische Universität München, Munich, Germany, 2018.

[118] I. Cvitić, D. Peraković, M. Periša, and B. Gupta, "Ensemble machine learning approach for classification of iot devices in smart home," *Int. J. Mach. Learn. Cybern.*, vol. 12, pp. 1–24, Jan. 2021.

[119] T. Saba, K. Haseeb, A. A. Shah, A. Rehman, U. Tariq, and Z. Mehmood, "A machine-learning-based approach for autonomous IoT security," *IT Prof.*, vol. 23, no. 3, pp. 69–75, May 2021.

[120] S. Simou, C. Kalloniatis, S. Gritzalis, and H. Mouratidis, "A survey on cloud forensics challenges and solutions," *Secur. Commun. Netw.*, vol. 9, no. 18, pp. 6285–6314, Dec. 2016.

[121] M. Abdel-Basset, H. Hawash, R. K. Chakrabortty, and M. J. Ryan, "Semi-supervised spatiotemporal deep learning for intrusions detection in IoT networks," *IEEE Internet Things J.*, vol. 8, no. 15, pp. 12251–12265, Aug. 2021.

[122] M. Abdel-Basset, V. Chang, H. Hawash, R. K. Chakrabortty, and M. Ryan, "Deep-IFS: Intrusion detection approach for industrial Internet of Things traffic in fog environment," *IEEE Trans. Ind. Informat.*, vol. 17, no. 11, pp. 7704–7715, Nov. 2021.

[123] B. Jothi and M. Pushpalatha, "WILS-TRS—A novel optimized deep learning based intrusion detection framework for IoT networks," *Pers. Ubiquitous Comput.*, vol. 3, pp. 1–17, Jun. 2021.

[124] M. M. Rashid, J. Kamruzzaman, M. M. Hassan, T. Imam, and S. Gordon, "Cyberattacks detection in IoT-based smart city applications using machine learning techniques," *Int. J. Environ. Res. Public Health*, vol. 17, no. 24, p. 9347, Dec. 2020. [Online]. Available: https://www.mdpi.com/1660-4601/17/24/9347

[125] K. Ruan, I. Baggili, J. Carthy, and T. Kechadi, "Survey on cloud forensics and critical criteria for cloud forensic capability: A preliminary analysis," in *Proc. 6th Annu. Conf. ADFSL Conf. Digit. Forensics, Secur. Law*, 2011, pp. 55–70.

[126] Y. Wei and M. B. Blake, "Service-oriented computing and cloud computing: Challenges and opportunities," *IEEE Internet Comput.*, vol. 14, no. 6, pp. 72–75, Nov. 2010.

[127] M. T. Khorshed, A. B. M. S. Ali, and S. A. Wasimi, "Monitoring insiders activities in cloud computing using rule based learning," in *Proc. IEEE 10th Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Nov. 2011, pp. 757–764.

[128] S. Singhal and M. Jena, "A study on WEKA tool for data preprocessing, classification and clustering," *Int. J. Innov. Technol. Exploring Eng.*, vol. 2, no. 6, pp. 250–253, May 2013.

[129] A. Gaurav, B. B. Gupta, C.-H. Hsu, D. Perakovic, and F. J. García Peñalvo, "Filtering of distributed denial of services (DDoS) attacks in cloud computing environment," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2021, pp. 1–6.

[130] A. Alshammari and A. Aldribi, "Apply machine learning techniques to detect malicious network traffic in cloud computing," *J. Big Data*, vol. 8, no. 1, pp. 1–24, Dec. 2021.

[131] A. Aldribi, I. Traoré, B. Moa, and O. Nwamuo, "Hypervisor-based cloud intrusion detection through online multivariate statistical change tracking," *Comput. Secur.*, vol. 88, Jan. 2020, Art. no. 101646. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404819301907

[132] S. Sachdeva and A. Ali, "A hybrid approach using digital forensics for attack detection in a cloud network environment," *Int. J. Future Gener. Commun. Netw.*, vol. 14, no. 1, pp. 1536–1546, 2021.

[133] P. T. Dinh and M. Park, "R-EDoS: Robust economic denial of sustainability detection in an SDN-based cloud through stochastic recurrent neural network," *IEEE Access*, vol. 9, pp. 35057–35074, 2021.

[134] S. Li, Y. Li, W. Han, X. Du, M. Guizani, and Z. Tian, "Malicious mining code detection based on ensemble learning in cloud computing environment," *Simul. Model. Pract. Theory*, vol. 113, Dec. 2021, Art. no. 102391. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1569190X21000976

[135] N. Moustafa and J. Slay, "RCNF: Real-time collaborative network forensic scheme for evidence analysis," 2017, *arXiv:1711.02824*.

[136] L. Gupta, T. Salman, A. Ghubaish, D. Unal, A. K. Al-Ali, and R. Jain, "Cybersecurity of multi-cloud healthcare systems: A hierarchical deep learning approach," *Appl. Soft Comput.*, vol. 118, Mar. 2022, Art. no. 108439. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494622000175

[137] G. Farnham. (Mar. 2013). *Detecting DNS Tunneling*. [Online]. Available: https://www.sans.org/white-papers/34152/

[138] J. Yang and K. Wang, "The wheel-rail vibrations on the key points of planar and vertical section in high-speed railways," *J. Chongqing Univ. Technol., Natural Sci.*, vol. 27, no. 9, pp. 49–52, 2013.

[139] C. J. Dietrich, C. Rossow, F. C. Freiling, H. Bos, M. V. Steen, and N. Pohlmann, "On botnets that use DNS for command and control," in *Proc. 7th Eur. Conf. Comput. Netw. Defense*, Sep. 2011, pp. 9–16.

[140] A. M. Kara, H. Binsalleeh, M. Mannan, A. Youssef, and M. Debbabi, "Detection of malicious payload distribution channels in DNS," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 853–858.

[141] M. V. Horenbeeck. *Detection of DNS Tunneling*. Accessed: Oct. 14, 2022. [Online]. Available: https://www.daemon.be/maarten/dnstunnel.html

[142] S. Sheridan and A. Keane, "Detection of DNS based covert channels," in *Proc. Eur. Conf. Cyber Warfare Secur.* Hatfield, U.K.: Academic Conferences International Limited, 2015, p. 267.

[143] (2006). *Iodine DNS Tunneling Tool*. Accessed: Jun. 2, 2022. [Online]. Available: https://code.kryo.se/iodine/

[144] M. Sammour, B. Hussin, M. F. I. Othman, M. Doheir, B. AlShaikhdeeb, and M. S. Talib, "DNS tunneling: A review on features," *Int. J. Eng. Technol.*, vol. 7, no. 3.20, pp. 1–5, 2018.

[145] F. Allard, R. Dubois, P. Gompel, and M. Morel, "Tunneling activities detection using machine learning techniques," *J. Telecommun. Inf. Technol.*, vol. 1, pp. 37–42, Jan. 2011.

[146] R. Preston, "DNS tunneling detection with supervised learning," in *Proc. IEEE Int. Symp. Technol. Homeland Secur. (HST)*, Nov. 2019, pp. 1–6.

[147] A. Nadler, A. Aminov, and A. Shabtai, "Detection of malicious and low throughput data exfiltration over the DNS protocol," *Comput. Secur.*, vol. 80, pp. 36–53, Jan. 2019.

[148] J. Liu, S. Li, Y. Zhang, J. Xiao, P. Chang, and C. Peng, "Detecting DNS tunnel through binary-classification based on behavior features," in *Proc. IEEE Trustcom/BigDataSE/ICESS*, Aug. 2017, pp. 339–346.

[149] H. Bai, G. Liu, J. Zhai, W. Liu, X. Ji, L. Yang, and Y. Dai, "Refined identification of hybrid traffic in DNS tunnels based on regression analysis," *ETRI J.*, vol. 43, no. 1, pp. 40–52, Feb. 2021.

[150] C.-M. Lai, B.-C. Huang, S.-Y. Huang, C.-H. Mao, and H.-M. Lee, "Detection of DNS tunneling by feature-free mechanism," in *Proc. IEEE Conf. Dependable Secure Comput. (DSC)*, Dec. 2018, pp. 1–2.

[151] C. Liu, L. Dai, W. Cui, and T. Lin, "A byte-level CNN method to detect DNS tunnels," in *Proc. IEEE 38th Int. Perform. Comput. Commun. Conf. (IPCCC)*, Oct. 2019, pp. 1–8.

[152] J. Zhang, L. Yang, S. Yu, and J. Ma, "A DNS tunneling detection method based on deep learning models to prevent data exfiltration," in *Network and System Security*. Berlin, Germany: Springer-Verlag, Dec. 2019, pp. 520–535, doi: 10.1007/978-3-030-36938-5_32.

[153] F. Palau, C. Catania, J. Guerra, S. Garcia, and M. Rigaki, "DNS tunneling: A deep learning based lexicographical detection approach," 2020, *arXiv:2006.06122*.

[154] G. Sakarkar, M. K. H. Kolekar, K. P. G. Patil, P. Dutta, R. Chaturvedi, and S. Kumar "Advance approach for detection of DNS tunneling attack from network packets using deep learning algorithms," *Adv. Distrib. Comput. Artif. Intell. J.*, vol. 10, no. 3, pp. 241–266, 2021.

[155] B. Savenko, S. Lysenko, K. Bobrovnikova, O. Savenko, and G. Markowsky, "Detection DNS tunneling botnets," in *Proc. 11th IEEE Int. Conf. Intell. Data Acquisition Adv. Comput. Systems, Technol. Appl. (IDAACS)*, Sep. 2021, pp. 64–69.

[156] M. A. Altuncu, F. K. Gulagiz, H. Ozcan, O. F. Bayir, A. Gezgin, A. Niyazov, M. A. Cavuslu, and S. Sahin, "Deep learning based DNS tunneling detection and blocking system," *Adv. Electr. Comput. Eng.*, vol. 21, no. 3, pp. 39–48, 2021.

[157] H. He and J. Yan, "Cyber-physical attacks and defences in the smart grid: A survey," *IET Cyber-Phys. Syst., Theory Appl.*, vol. 1, no. 1, pp. 13–27, Dec. 2016.

[158] K. Demertzis, K. Tsiknas, D. Taketzis, D. N. Skoutas, C. Skianis, L. Iliadis, and K. E. Zoiros, "Communication network standards for smart grid infrastructures," *Network*, vol. 1, no. 2, pp. 132–145, Aug. 2021. [Online]. Available: https://www.mdpi.com/2673-8732/1/2/9

[159] G. W. Arnold, D. A. Wollman, G. FitzPatrick, D. Prochaska, D. Holmberg, D. H. Su, A. R. Hefner Jr., N. T. Golmie, T. L. Brewer, M. Bello, and P. A. Boynton, "NIST framework and roadmap for smart grid interoperability standards, release 1.0," Nat. Inst. Standards Technol., MD, USA, Tech. Rep. 1108, 2010.

[160] Y. Zhang, L. Wang, W. Sun, R. C. Green, and M. Alam, "Artificial immune system based intrusion detection in a distributed hierarchical network architecture of smart grid," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Jul. 2011, pp. 1–8.

[161] S. X. Wu and W. Banzhaf, "The use of computational intelligence in intrusion detection systems: A review," *Appl. Soft Comput.*, vol. 10, no. 1, pp. 1–35, Jan. 2010.

[162] Z. A. Baig, "On the use of pattern matching for rapid anomaly detection in smart grid infrastructures," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, Oct. 2011, pp. 214–219.

[163] S. D. Roy, S. Debbarma, and J. M. Guerrero, "Machine learning based multi-agent system for detecting and neutralizing unseen cyber-attacks in AGC and HVDC systems," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 12, no. 1, pp. 182–193, Mar. 2022.

[164] T. T. Khoei, G. Aissou, W. C. Hu, and N. Kaabouch, "Ensemble learning methods for anomaly intrusion detection system in smart grid," in *Proc. IEEE Int. Conf. Electro Inf. Technol. (EIT)*, May 2021, pp. 129–135.

[165] S. Chesney, K. Roy, and S. Khorsandroo, "Machine learning algorithms for preventing IoT cybersecurity attacks," in *Proc. SAI Intell. Syst. Conf.* Springer, 2020, pp. 679–686.

[166] K. K. Gomez Buquerin, C. Corbett, and H.-J. Hof, "A generalized approach to automotive forensics," *Forensic Sci. Int., Digit. Invest.*, vol. 36, Apr. 2021, Art. no. 301111. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666281721000056

[167] G. Prasad, Y. Huo, L. Lampe, and V. C. M. Leung, "Machine learning based physical-layer intrusion detection and location for the smart grid," in *Proc. IEEE Int. Conf. Commun., Control, Comput. Technol. Smart Grids (SmartGridComm)*, Oct. 2019, pp. 1–6.

[168] M. Keshk, B. Turnbull, N. Moustafa, D. Vatsalan, and K.-K. R. Choo, "A privacy-preserving-framework-based blockchain and deep learning for protecting smart power networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5110–5118, Aug. 2020.

[169] D. Yao, M. Wen, X. Liang, Z. Fu, K. Zhang, and B. Yang, "Energy theft detection with energy privacy preservation in the smart grid," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7659–7669, Oct. 2019.

[170] A. H. M. Jakaria, M. A. Rahman, and M. G. M. M. Hasan, "Safety analysis of AMI networks through smart fraud detection," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Washington, DC, USA, Jun. 2019, pp. 1–7.

[171] J. Zhang, L. Pan, Q.-L. Han, C. Chen, S. Wen, and Y. Xiang, "Deep learning based attack detection for cyber-physical system cybersecurity: A survey," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 3, pp. 377–391, Mar. 2022.

[172] D. Kopencova and R. Rak, "Issues of vehicle digital forensics," in *Proc. 12th Int. Sci. Tech. Conf. Automot. Saf.*, Oct. 2020, pp. 1–6.

[173] M. E. Zarki, S. Mehrotra, G. Tsudik, and N. Venkatasubramanian, "Security issues in a future vehicular network," in *Proc. Eur. Wireless Conf.*, vol. 2, 2002, pp. 1–5.

[174] J. P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *IEEE Security Privacy*, vol. 2, no. 3, pp. 49–55, May/Jun. 2004.

[175] M. Raya and J.-P. Hubaux, "The security of vehicular ad hoc networks," in *Proc. 3rd ACM Workshop Secur. Ad Hoc Sensor Netw. (SASN)*, 2005, pp. 11–21.

[176] A. Talpur and M. Gurusamy, "Machine learning for security in vehicular networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 1, pp. 346–379, 1st Quart., 2021, doi: 10.1109/COMST.2021.3129079.

[177] F. Martinelli, F. Mercaldo, and A. Santone, "Machine learning for driver detection through CAN bus," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, May 2020, pp. 1–5.

[178] G. D'Angelo, A. Castiglione, and F. Palmieri, "A cluster-based multidimensional approach for detecting attacks on connected vehicles," *IEEE Internet Things J.*, vol. 8, no. 16, pp. 12518–12527, Aug. 2021.

[179] A. R. Javed, S. U. Rehman, M. U. Khan, and M. Alazab, "CANintelliIDS: Detecting in-vehicle intrusion attacks on a controller area network using CNN and attention-based GRU," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 2, pp. 1456–1466, Apr. 2021.

[180] P. Sharma and H. Liu, "A machine-learning-based data-centric misbehavior detection model for Internet of Vehicles," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4991–4999, Mar. 2021.

[181] A. E. Mekki, A. Bouhoute, and I. Berrada, "Improving driver identification for the next-generation of in-vehicle software systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7406–7415, Aug. 2019.

[182] L. Yang, A. Moubayed, and A. Shami, "MTH-IDS: A multitiered hybrid intrusion detection system for Internet of Vehicles," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 616–632, Jan. 2022.

[183] E. Seo, H. M. Song, and H. K. Kim, "GIDS: GAN based intrusion detection system for in-vehicle network," in *Proc. 16th Annu. Conf. Privacy, Secur. Trust (PST)*, Aug. 2018, pp. 1–6.

[184] S. Cai, D. Han, X. Yin, D. Li, and C.-C. Chang, "A hybrid parallel deep learning model for efficient intrusion detection based on metric learning," *Connection Sci.*, vol. 34, no. 1, pp. 551–577, Dec. 2022.

SYED RIZVI received the B.E. degree in computer engineering from UIT University, Karachi, Pakistan, in 2014, and the M.S. degree in data science from the National University of Computer and Emerging Sciences, Karachi, in 2020. He is currently pursuing the Ph.D. degree with the School of Science and Computing, South East Technological University, Waterford, Ireland. His research interests include computer science, cybersecurity, digital forensics, the IoT, healthcare, data analysis, and artificial intelligence.

MARK SCANLON (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in remote digital forensic evidence acquisition. He is currently an Associate Professor with the School of Computer Science, University College Dublin (UCD), and the Founding Director of the Forensics and Security Research Group, UCD. He is a Fulbright Scholar in cybersecurity and cybercrime investigation. His research interests include digital forensics, artificial intelligence, computer vision, data encryption, network forensics, and digital forensics education. He is a Senior Editor of *Forensics Science International: Digital Investigation* journal (Elsevier), and is a keen editor, a reviewer, and a conference organizer in the field of digital forensics, including Digital Forensics Research Workshop (DFRWS).

JIMMY MCGIBNEY (Member, IEEE) received the B.E. degree from University College Dublin and the M.Eng. degree (by research) from Dublin City University. He is currently a Lecturer in computer science at South East Technological University, Waterford, where he has taught courses for over 20 years in the areas of information security, distributed systems, and cloud computing. His research in more recent years has mostly been in the area of distributed trust management, and before that he was active in the area of telecommunications network management. He has supervised research students and staff and has participated in several European and national collaborative projects. He has acted as a reviewer and a program committee member for a variety of conferences and publications.

JOHN SHEPPARD (Member, IEEE) received the Ph.D. degree from University College Dublin (UCD). He is currently a Lecturer and a Researcher in the area of AI techniques for cybersecurity and digital forensics with the Department of Computing and Mathematics, South East Technological University (SETU), Ireland. He is a Fulbright Cybersecurity TechImpact Scholar at the Boston College. His research interests include digital forensics and incident response, and the use of data mining/machine learning for intrusion detection, network forensic analysis, and the IoT and small device forensics. He is an Organizing Committee Member of the Digital Forensics Research Workshop EU (DFRWS-EU). He is a reviewer for a number of international journals and conferences.