Journal of Information Security and Applications 93 (2025) 104145

Contents lists available at ScienceDirect

Journal of Information Security and Applications

journal homepage: www.elsevier.com/locate/jisa

SDN-based solutions for malware analysis and detection: State-of-the-art, open issues and research challenges

Cristian H.M. Souza [a,c,*], Túlio Pascoal [a], Emidio P. Neto [a], Galileu B. Sousa [a,b], Francisco S.L. Filho [a], Daniel M. Batista [c], Felipe S. Dantas Silva [a]

[a] LaTARC Research Lab – Federal Institute of Education, Science, and Technology of Rio Grande do Norte (IFRN), Natal/RN, Brazil
[b] Federal Police Department, Natal/RN, Brazil
[c] University of São Paulo, São Paulo/SP, Brazil

ARTICLE INFO

Keywords:
Network security
Malware analysis
Malware detection
Software-Defined Networking
SDN
Review
Research challenges
Machine learning
Deep learning

ABSTRACT

Software-Defined Networking (SDN) has emerged as a key technology for countering evolving malware threats in 5G and Internet-of-Things (IoT) environments. This paper provides a comprehensive survey of SDN-based strategies for malware analysis and detection, consolidating several hundred candidate works and distilling a focused set of studies published up to April 2025. We examine approaches ranging from static code inspection and heuristic traffic monitoring to advanced machine learning and deep learning frameworks, demonstrating that these methods consistently achieve high detection accuracy with low false-positive rates while imposing only modest latency and resource overhead. We illustrate how SDN's centralized control and programmable data plane enable rapid policy updates and real-time mitigation of malicious flows, surpassing traditional network defense mechanisms. Our review clarifies how AI-driven techniques enhance the identification of novel and obfuscated malware, and highlights persistent challenges such as the need for standardized datasets, controller scalability, and privacy-preserving inspection. By synthesizing key insights, open issues, and future research directions, this survey underscores the essential role of SDN in fortifying contemporary cybersecurity architectures.

1. Introduction

Malware infections continue to be one of the main threats to computer systems, especially with the growth of malware families and, consequently, the lucrative business of Malware-as-a-Service (MaaS) [1–3]. These threats are responsible for causing various damages, such as compromising data integrity, stealing confidential information from users and companies, and causing financial losses to victim organizations.

Recent cybersecurity reports [4–6] highlight that MaaS has become a highly lucrative business for attackers, who can create new malware families on demand. In addition, research indicates that global ransomware damage costs are predicted to exceed $275 billion by 2031.[1] The widespread massification of Internet of Things (IoT)-enabled devices and the fifth-generation of mobile networks (5G) technology [7] has sparked numerous discussions about the security of such devices and the environments in which they operate [8–13].

The diversity of IoT devices connected to the same 5G network, allied to the high density of devices (up to one million per square kilometer [14]) hinders the effective detection of cyber threats. These devices can have different hardware and software characteristics, such as processing power, network capabilities, and operating systems [15]. This scenario becomes extremely attractive to attackers, as it is possible to compromise IoT devices connected to 5G infrastructures by exploring different attack vectors [16]. Furthermore, malicious actors can use infected devices for amplifying distributed denial-of-service (DDoS) attacks [17–20].

In the context of 5G services, the growing use of devices in critical scenarios, such as the Internet of Health Things (IoHT) [21–23] has brought several concerns regarding the protection of personal data, as well as ensuring that these devices are available and securely collecting critical data almost in a timely fashion. These concerns are entirely valid, as failures in such devices can expose users' sensitive data to attackers, enabling the execution of targeted phishing attacks [24–27] and putting lives at risk [28–30].

More recent technological enablers, such as the Software-Defined Networking (SDN) and Network Function Virtualization (NFV)

* Corresponding author at: University of São Paulo, São Paulo/SP, Brazil.
E-mail address: cristianmsbr@gmail.com (C.H.M. Souza).
URL: https://www.cristian.sh (C.H.M. Souza).

[1] https://cybersecurityventures.com/wp-content/uploads/2023/11/RansomwareCost.pdf

https://doi.org/10.1016/j.jisa.2025.104145

Available online 2 July 2025

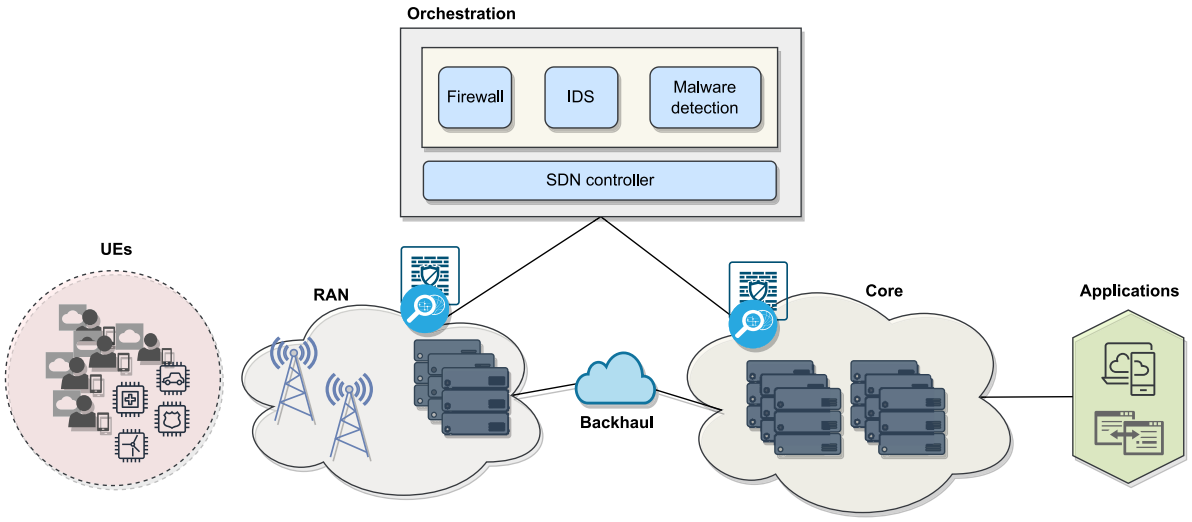

Fig. 1. Example of a scenario where malware mitigation is performed with the support of an SDN-enabled infrastructure.

paradigms, have attracted attention from the scientific community and industry as they have been supporting the creation of new adaptive solutions for 5G networks [31–33]. These paradigms allow a holistic view of the infrastructure, facilitate network orchestration, and increase programmability [34–36] of the network. Specific requirements of 5G (e.g., high bandwidth rates, ultra-low latencies, reliability, security, and stability) demand elastic, flexible, programmable, and secure services supported by the advent of SDN [9,37,38].

Since SDN technology has been largely incorporated in 5G systems [39,40], many attacks have been developed and exploiting the vulnerabilities of such infrastructures, aiming at either making them unavailable or infect connected devices with malware for subsequent use in DDoS attacks [41–43]. This complex scenario (composed of different technologies and devices) allows attackers to obtain a high computational power for the execution of distributed attacks, increasing their chances of success and raising the damage to the victims [44,45].

The use of SDN to increase and optimize security resources in computer networks has been proven feasible in several works present in the literature [17,46–48]. The high programmability of SDN and the holistic view of controllers allow for the detection and mitigation of threats with great ease and in different scenarios [49]. In addition, SDN enables the creation of controlled and adaptive environments to analyze malicious artifacts, facilitating the analyst's work. As highlighted by Ceron et al. [50], SDN allows for the rapid reconfiguration of controlled environments to analyze and identify new behaviors of malicious artifacts. Fig. 1 illustrates an example of a scenario where the detection and mitigation of malware are performed with the support of an SDN-enabled infrastructure.

As illustrated in Fig. 1, the holistic view proportioned by the SDN controller enables the deployment of defense modules for the entire infrastructure. These modules can act in an agnostic manner, detecting threats that affect different platforms and operating systems with high network visibility [51]. This scheme allows flexibility for the adoption of new solutions. In addition, SDN does not require substantial modifications to the infrastructure to deploy new tools or functions into the network. Therefore, defensive techniques can be added to SDN architectures for increased malware detection in 5G and IoT ecosystems composed of different devices with distinct characteristics. This makes it possible to mitigate threats accurately and automate the analysis process for network operators.

In addition, SDN offers unique capabilities that have the potential to enhance malware detection effectiveness compared to traditional approaches [52], such as Next-Generation Firewalls (NGFWs) or Intrusion Detection Systems (IDS). Some of these advantages include:

- Flexibility in policy management: The ability to rapidly reconfigure network policies through the SDN controller can be harnessed to adapt to emerging threats more effectively than static traditional solutions [53].
- Granular visibility and control: SDN allows for detailed visibility into network traffic and the ability to apply specific control measures in real-time, which can be beneficial in detecting suspicious activities [54].
- Contextual integration: SDN can integrate contextual and external information to enhance detection by aggregating insights from diverse sources for a more comprehensive view of network security [55].

Given the diversity and efficiency of existing solutions proposed by the research community that leverages SDN for malware detection and mitigation, a more in-depth analysis of such proposals is necessary to understand better their advantages and shortcomings as well as to export this knowledge out of the academic world, i.e., allowing the industrial and private sectors to profit from such solutions. Considering that, this work plays a key role and advances the state-of-the-art by offering a systematic review of SDN-based techniques for malware analysis and detection. We believe that with this more precise understanding and broader dissemination of technological advancements in attack protection and prevention, new mechanisms and innovations can arise, improving the defense of IT systems. Finally, to the best of our knowledge, this survey is the first to explore and review SDN-enabled malware detection and mitigation solutions. Therefore, this work also fills a gap in the literature.

Upon extensive searches on the main computing-related research libraries, we identified 485 studies directly or indirectly related to malware analysis and detection using SDN. After criterion filtering and classification, 22 papers were selected for reading and analysis to highlight the SDN paradigm's advantages in combating such cyber threats. Finally, open research challenges are presented, aiming to encourage the development of new solutions for malware analysis and mitigation.

1.1. Contributions

The main contributions of this study are as follows:

1. A comprehensive review of malware detection and mitigation solutions powered by SDN technologies;

2. A synthesis of existing solutions in the context of malware detection and analysis considering different dimensions, such as goals, operating methodology, and operationalization of the considered SDN controllers;
3. A comparative analysis of the investigated solutions in the context of malware analysis and detection using SDN technologies from the perspective of positive points that motivate their use and their limitations to guide researchers on possible operational restrictions and their consequences;
4. A discussion of open issues and research challenges in the context of malware analysis and detection using SDN technology;
5. Showing how emerging hybrid SDN solutions integrate with legacy or traditional security mechanisms, highlighting scalability limits and suggesting avenues for further enhancements in real-world deployments.

Moreover, this study incorporates several recent publications (from 2022 to April 2025) that apply AI- and ML-based approaches to malware detection [56–59], thereby offering a more up-to-date and comprehensive view of the state-of-the-art. These works underscore the growing trend of integrating deep learning architectures with SDN to achieve higher detection accuracy and reduced overhead, reflecting the ever-evolving nature of advanced threats. Our inclusion of these recent contributions helps fill the gap identified by previous surveys and ensures our review reflects the current research landscape.

1.2. Article organization

The remainder of this article is organized as follows: Section 2 reviews the existing literature on malware analysis and detection, highlighting recent works and critical gaps that motivate our research. Section 3 describes our research methodology, research questions (RQs), and search criteria. Section 4 discusses SDN-based malware solutions in detail, providing a refined taxonomy table and elaborating on key architectural components, performance metrics, and real-world applicability. Section 5 addresses open issues and research challenges, categorized into technical, operational, and ethical dimensions, and also explores the practical implications of SDN adoption. Finally, Section 6 presents concluding remarks, emphasizing our contributions to the field, outlining future research avenues, and summarizing our findings.

2. Related work

Malware analysis is of fundamental importance for information security because, through understanding its behavior and Assembly code, efficient tools for detection and mitigation are developed [60]. However, the time an analyst takes to dissect a malicious binary and deploy rules in the developed solution can be unfavorable to the end user.

As specified by [61], malware analysis generally requires that the adopted solution has an improved ability to classify new files based on previous investigations. The authors emphasize the diversity of available tools and that the wrong choice of one can delay the analysis work. Furthermore, the analysis task itself can have different purposes: one analyst may be interested only in identifying whether an artifact is malicious or not, while another analyst may delve deeper into the analysis to discover which malware family it belongs to.

Effective malware detection strategies encompass a spectrum of techniques, ranging from signature-based approaches to more advanced behavioral analysis and machine learning models [62]. Signature-based methods involve comparing files or code against a database of known malware signatures. While efficient at detecting known threats, they may fall short when dealing with 0-day or polymorphic malware variants.

To address this limitation, behavioral detection comes into play. It involves monitoring the execution of software to detect anomalous behavior patterns, which may indicate the presence of malware [63]. This dynamic approach is instrumental in identifying previously unseen threats, as it focuses on behavior rather than specific signatures.

Machine learning, particularly deep learning models, has also gained prominence in malware detection. These models can analyze vast amounts of data to discern intricate patterns and anomalies indicative of malware [64]. They adapt and evolve as new threats emerge, making them a valuable asset in the ongoing battle against evolving malware.

More recently, post-2022 works [56–58,65] have demonstrated the applicability of convolutional and recurrent neural network architectures for enhanced intrusion and malware detection, especially within IoT-driven networks. These approaches often integrate with SDN controllers, leveraging programmable data planes to facilitate near-real-time threat response. Their positive results highlight the shifting landscape toward AI-powered, SDN-centric solutions.

Several research studies have aimed at listing and classifying academic strategies for malware analysis and detection. However, despite the various existing SDN-based solutions, no published review was found at the time of writing this paper that highlights SDN-based solutions capable of analyzing or detecting malware. The following subsections present existing related reviews.

2.1. Malware analysis surveys

Or-Meir et al. [60] provides a state-of-the-art review on dynamic analysis mechanisms. The authors describe the existing methods, their strengths and weaknesses, and the resilience of proposed solutions against evasion mechanisms. Additionally, the paper highlights machine learning methods to improve dynamic analysis techniques.

Ucci et al. [61] presents a detailed examination of machine learning-based solutions for analyzing malicious artifacts in Windows environments. The authors synthesize articles based on their objectives, the information used to identify the malicious properties of an artifact (i.e., features), and which machine learning methods are employed in the classification process.

2.2. Malware detection surveys

Bazrafshan et al. [66] lists approaches that use heuristics for malicious program classification. Different resources used in the heuristic identification of malware (such as API calls and opcodes) are also highlighted. Additionally, the authors present the advantages and disadvantages of these resources.

Odusami et al. [67] presents approaches for detecting malware developed for the Android platform. The authors highlight that machine learning is one of the most promising approaches. Moreover, gaps are identified to provide a basis for guiding the development of more effective solutions for mitigating malicious programs targeting the Android operating system. Similarly, Qiu et al. [68], and Kouliaridis and Kambourakis [69] focus on machine learning-based solutions for malware identification in Android.

Ye et al. [70] presents an overview of the functioning of malicious programs and the antivirus industry's current needs in developing new identification and mitigation techniques. In addition, the authors highlight the stages of feature extraction and classification using data mining techniques. Finally, additional problems and challenges when using such methods are discussed.

Souri and Hosseini [71] presents a systematic review of data mining-based mechanisms for detecting malware. The authors classify approaches that use signature-based and behavior-based detection methods. Moreover, the advantages and disadvantages of each presented solution are discussed.

Tripathy et al. [72] reviews data mining strategies that analyze unstructured text documents to identify malicious patterns. Additionally, the advantages of using these techniques for early detection (i.e., before execution) of these threats are highlighted.

Mahdavifar and Ghorbani [73] details the use of deep learning in different areas of cybersecurity, and then focuses on malware identification and intrusion detection systems. Furthermore, possible research topics for improving deep learning techniques focused on combating cyber threats are presented.

Negera et al. [74] presents a review focused on detecting botnet attacks originating from IoT devices enabled by SDN infected with malicious programs. The authors highlight that machine learning (ML) and deep learning techniques have comparable performance in detecting botnet attacks. However, the study also emphasizes that classical ML algorithms require more feature extraction from artifacts for efficient detection and may fail to identify new threats.

Madan et al. [75] exposes various tools and techniques for collecting, detecting, and analyzing malware focused on IoT environments. Unlike other reviews, the authors' focus is on Executable and Linkable Format (ELF) files, since Linux systems operate most IoT devices. Additionally, some evasion techniques that can be used by malware in IoT scenarios, as well as industry-proposed sandboxes, are presented.

Tayyab et al. [76] reviews recent trends in malware detection based on deep learning techniques. The authors outline the hierarchical evolution of malware identification techniques and present research challenges in anti-analysis.

Gopinath and Sethuraman [77] highlights that traditional techniques are no longer sufficient for detecting modern malware threats. Subsequently, the authors review and classify solutions proposed by academia in the field of malware detection using machine learning and deep learning techniques. Finally, the study presents a series of algorithms that can detect malicious artifacts statically and dynamically on different platforms.

Janabi et al. [78] offers a comprehensive review of state-of-the-art Intrusion Detection Systems (IDS) that operate within SDN. After detailed SDN architecture and its main attack surfaces, the authors classify machine learning simulations, studies that rely on public datasets, and deep learning solutions, pointing out recurring bottlenecks such as controller overload, outdated data, and poor detection of low-rate DDoS. They then list nine open research gaps, from distributing traffic processing across switches to automating security policy management, and propose directions for adaptive multi-controller IDS frameworks. Although comprehensive, the paper does not implement a new detection scheme nor does it cover malware topics beyond IDS; its contribution is restricted to SDN-centric analysis.

Alzahrani et al. [79] distills research on ransomware detection, categorizing static, dynamic and hybrid analyses. It shows ML/DL now dominate but are hampered by small, outdated datasets, model size and adversarial evasion, while lightweight non-ML schemes still help in resource constrained or real-time settings. The authors call for a community maintained, regularly updated corpus plus explainable, privacy preserving, adversarial-robust detection models to meet the rise of double extortion and cloud-targeted ransomware.

2.3. Discussion

As presented in Table 1, the literature reveals many studies developed in recent years to promote discussion on techniques for analyzing and detecting malicious programs. However, no previous studies have committed to discussing SDN-based strategies in combating malware threats to provide a comprehensive state-of-the-art perspective. Nonetheless, these reviews analyze techniques used in SDN, such as Intrusion Detection Systems (IDS) [80–82] and combating Advanced Persistent Threats (APTs) [83–85]. Furthermore, even when the analysis is being carried out within the applications, it identifies Indicators of Compromise (IoC) that can be used in SDN controllers, for example, to confine or prevent threats. Finally, the studies also present everyday challenges, such as traffic encryption and the self-protection of malicious programs through packing and polymorphism/metamorphism [86–89].

Nevertheless, the lack of consolidation of knowledge associated with combating threats in the scope of SDN restricts researchers' ability to identify and propose new solutions for protecting computer systems against malware threats. In addition, the absence of information on the subject in focus makes it difficult to understand the effects of defense mechanisms in specific situations, resulting from the wide diversity of scenarios.

Section summary: We have surveyed existing reviews covering both malware analysis and detection methods, illustrating their strengths and gaps. While many studies offer valuable insights, most fail to address SDN-specific strategies in a comprehensive manner. In the following section, we detail our methodology to systematically explore SDN-based malware solutions.

3 Review methodology

This section outlines our research methodology, detailing the search strategies, inclusion/exclusion criteria, and specific research questions formulated to guide the analysis.

3.1 Research questions

To comprehensively frame our investigation, we define the following RQs:

- **RQ1:** What are the main SDN-based approaches for malware detection and analysis, and how do they differ in methodology (e.g., signature-based, behavioral, ML-driven)?
- **RQ2:** Which performance metrics (e.g., detection rate, false positive rate, response time) are most commonly used to evaluate SDN-based malware solutions, and how effective are these approaches in real-world or large-scale environments?
- **RQ3:** What are the open technical, operational, and ethical challenges in integrating SDN-based architectures for malware defenses, and how might future research address these gaps?

RQ1 focuses on discovering the variety and novelty of SDN-based solutions, RQ2 targets the evaluation metrics and real-world feasibility, while RQ3 guides our synthesis of research gaps and future directions.

3.2 Definition of research libraries

As presented in Fig. 2, six research libraries were used to identify the reviewed papers, namely:

1. ACM Digital Library[2]
2. IEEE Xplore[3]
3. Science Direct[4]
4. MDPI[5]
5. Springer Link[6]
6. Wiley[7]

These databases collectively ensure coverage of major publishers in computer science and network security. Our searches specifically encompassed publications from 2013 to 2025 to capture both foundational and cutting-edge approaches in SDN-based malware detection.

[2] https://dl.acm.org/
[3] https://ieeexplore.ieee.org/
[4] https://www.sciencedirect.com/
[5] https://www.mdpi.com/
[6] https://link.springer.com/
[7] https://onlinelibrary.wiley.com/

Table 1
Comparison between existing surveys and the current proposal.

Publication	Year	# Refs	Year range	Focus domain	SDN focus?	Malware focus?
[66]	2013	22	2001–2012	Malware detection heuristics	×	✓
[70]	2017	25	2001–2016	Malware behavior & data mining	×	✓
[67]	2018	22	2010–2018	Android malware detection	×	✓
[71]	2018	21	2008–2017	Data-mining for malware detection	×	✓
[72]	2018	17	1998–2015	Text-based classification	×	✓
[60]	2019	29	2006–2014	Dynamic malware analysis	×	✓
[61]	2019	26	2001–2017	ML-based Windows malware analysis	×	✓
[73]	2019	45	2011–2018	Deep learning in cybersecurity	×	×
[68]	2020	31	2014–2019	DL for Android malware	×	✓
[69]	2021	19	2014–2021	Android ML-based detection	×	✓
[74]	2022	43	2016–2021	Botnet detection in SDN-based IoT	×	✓
[75]	2022	25	1994–2021	IoT malware, ELF binaries	×	✓
[76]	2022	47	2006–2021	DL for general malware detection	×	✓
[77]	2023	199	2010–2023	ML/DL for general malware detection	×	✓
[78]	2024	31	2010–2023	IDS in SDN	✓	×
[79]	2025	45	2019–2025	Ransomware detection methods	×	✓
This work	2025	22	2013–2025	SDN for malware analysis and detection	✓	✓

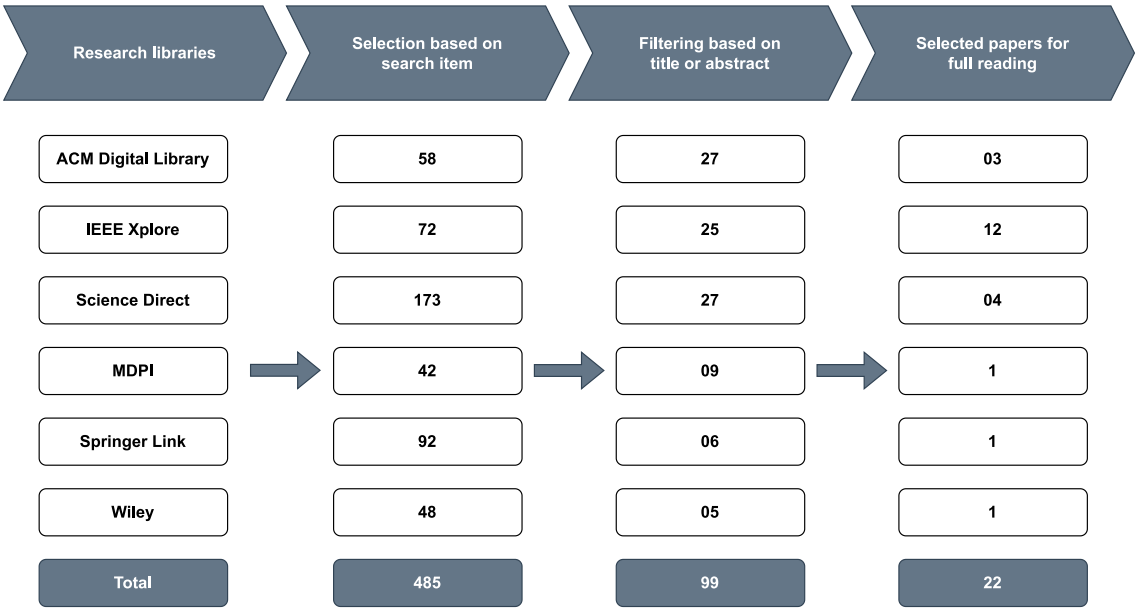

Fig. 2. Flowchart of the filtering stages for the studies identified in the selected research libraries.

3.3 Identification of studies based on search terms

The search strings were defined based on a combination of keywords related to the objects under study. The searches aimed to identify the largest possible number of papers related to the topics. For this purpose, the following terms were used:

- ("Malware Analysis" OR "Malware Detection") AND ("SDN Malware Analysis" OR "SDN Malware Detection").

We carefully selected these search strings to capture works that specifically address both malware and SDN paradigms, thereby aligning with **RQ1**. While broader search terms risked returning irrelevant documents, narrower queries might exclude novel techniques. This approach balances breadth and specificity.

3.4 Filtering based on title and abstract

The titles and abstracts of all solutions identified during the search stage were analyzed to reach SDN-based approaches. The following criteria were adopted for the inclusion and exclusion of studies:

Inclusion criteria

- Studies for malware analysis in SDN-based environments.
- Studies for malware detection in SDN-based environments.
- Studies published or made available until April 2025 that discuss either empirical or simulated environments.
- Studies presenting hybrid solutions that combine SDN capabilities with conventional security measures (e.g., firewalls, IDS).

Exclusion criteria

- Studies that could not have the text analyzed due to the unavailability of the manuscript.
- Previous versions of more complete studies.
- Studies unrelated to malware analysis or detection using SDN.
- Works focused solely on general IDS/IPS approaches without explicit consideration of malware-specific behaviors in an SDN setting.

We also sought to mitigate potential bias by including both conference papers and journal articles, provided they met the inclusion criteria. However, doctoral theses, technical reports, and pre-prints without peer review were excluded to maintain quality control.

Table 2
Taxonomy of SDN-based malware detection and analysis proposals, highlighting detection approach, platform, performance metrics, and mitigation strategies.

Proposal	Detection approach	Platform/Target	Performance metrics	Key contributions/Architecture	Mitigation strategy?
Jin and Wang [90]	Behavior-based/real-time traffic	Android (APK)	Not explicitly reported	Coupled with SDN controller, blacklists malicious IPs	Yes (blocking via blacklist)
Abaid et al. [91]	Behavior-based	–	No performance data given	Architecture with multiple detection modules and Snort IDS	Not specified
Ceron et al. [50]	Dynamic analysis with topology reconfiguration	Windows (PE)	Analysis coverage, overhead not quantitatively measured	Dynamically changes sandbox environment to elicit more malware behaviors	N/A (analysis-focused)
Lee and Shin [92]	Static code analysis	Java-based SDN apps	Avg. detection time: 45.26s; ~92% accuracy	Soot framework for detecting malicious API calls in SDN controllers	N/A (analysis-focused)
Cabaj and Mazurczyk [93]	Network traffic + blacklist	Windows (CryptoWall)	High detection rate for known addresses; overhead minimal	Controller-based blocking with updated blacklists	Yes (block victim-attacker communication)
Nguyen and Yoo [94]	Behavior-based	Generic	Latency minimal, detection thresholds set heuristically	Real-time analysis of suspicious requests	Yes (add attacker to blacklist)
Tatang et al. [95]	Reverse proxy + behavior check	SDN controller (rootkits)	Small networks feasible; overhead not detailed	Compares controller's network view with proxy's	Yes (invalidate malicious rules)
Cabaj et al. [96]	HTTP traffic dynamic analysis	Generic (Locky, CryptoWall)	97%–98% detection, 4%–5% FPR	Inspects HTTP POST payloads for anomalies	No (detection only, blocklist proposed)
Lee et al. [97]	Static analysis + AI	Java-based SDN apps	96% accuracy	Identifies suspicious API calls	N/A (analysis-focused)
Cusack et al. [98]	ML-based (Random Forest)	Generic (HTTPS)	86% accuracy; overhead due to packet-level inspection	Uses P4 switches to collect features	Not explicitly stated (detection only)
Letteri et al. [99]	MLP-based DL	Generic (botnets)	96% accuracy	HogZilla/CTU-13/ISCX datasets	Not specified
Maeda et al. [100]	DL-based (generalizable)	Generic (botnets)	99.2% accuracy	Isolates infected machines	Yes (isolation and blocking)
Akbanov et al. [101]	DNS traffic + blacklist	Windows (WannaCry)	Not explicitly measured, tested in real scenario	Blacklist-based detection	Yes (isolate infected devices)
Rouka et al. [102]	SMB/HTTP packet inspection	Windows (ExPetr)	High success rate for tested samples	SDN controller sets blocking rules	Yes (isolate infected nodes)
Alotaibi and Vassilakis [103]	Packet inspection	Windows (BadRabbit)	High efficiency; overhead on CPU	Controller-based detection modules	Not explicitly stated
Khan and Akhunzada [104]	Hybrid DL (CNN + LSTM)	IoMT devices	Better detection than single-model approaches	Combines temporal + spatial analysis	Not specified
Ahmed et al. [105]	ML-based (DNS DGA detection)	Generic	>97% accuracy, real network test	Intercepts DNS requests, identifies malicious domains	Not explicitly stated (focus on detection)
Muthanna et al. [65]	DL (cuLSTMGRU)	Generic IoT	99.23% accuracy	Uses CICIDS2017 dataset	Not specified
Chang et al. [56]	CNN-based	Generic IoT	99% accuracy, low overhead	Runs on programmable switches	Yes (blocks malicious flows)
Chaganti et al. [57]	LSTM-based	Generic IoT	97.1% accuracy	Uses SDNIoT/SDN-NF-TJ datasets	Not specified
Kumar and Kumar [58]	CNN-based (image analysis)	Generic	98.5% accuracy, 0.006s processing time	Converts malware into images	Not specified
Souza and Arima [59]	Hybrid YARA + ML (Random Forest)	IoT (P4 switches)	99.33% accuracy, ~0.02s overhead	Eliminates single controller failure	Yes (blocking in P4 data plane)
Almotiri [106]	Hybrid static + dynamic ML/DL (XGBoost, LightGBM, DNN)	IoMT/SDN healthcare	Accuracy 99.60%, F1 0.9966 (XGB); 99.32%, F1 0.9934 (LGBM)	Bi-layer framework with GAN/autoencoder augmentation; modular continuous-training design	Not specified

3.5 Potential biases and limitations

Despite our rigorous selection process, potential biases may still arise, whether from relying on specific databases, the subjective nature of keyword-based filtering (which can overlook studies using alternative terminology such as "SDN-based malicious traffic identification"), or publication bias that tends to underreport negative or null findings. Additionally, many proprietary or industry solutions remain insufficiently disclosed in academic literature. To mitigate these limitations, we cross-checked references via a snowballing technique, consulted multiple publishers, and included both conference and journal papers wherever possible, thereby broadening coverage and reducing the likelihood of missing relevant work.

3.6 Synthesis of the selected studies for full reading

After filtering by title and abstract and following the specified inclusion and exclusion criteria, the most relevant proposals were considered for full reading and analysis. During this process, meaningful information about the proposed solutions was extracted. In total, we identified 485 initial studies, of which 22 satisfied all criteria and were thus included in this review (Fig. 2).

Section summary: In this section, we introduced the refined methodology underlying our study, emphasizing transparent research questions, tailored search strategies, and explicit inclusion/exclusion criteria. We also discussed potential biases, ensuring clarity and repeatability of our approach. The next section will address RQ1 and RQ2 by describing selected SDN-based malware solutions and analyzing their architectures, detection techniques, and performance metrics.

4 SDN in the fight against malware

In this section, we address **RQ1** by examining the main SDN-based approaches for malware analysis and detection. We provide an updated taxonomy of solutions (Table 2), offering more in-depth insights into architectural components and detection methodologies. We then discuss performance metrics (detection rate, false-positive rate, response time), covering **RQ2**, and elaborate on how each proposal evaluates its effectiveness.

We critically assess each approach's strengths and limitations, especially concerning scalability, real-world deployment feasibility, and integration with existing security infrastructures. We also integrate new frameworks and algorithms being used in SDN-based malware detection, including CNN, LSTM, GRU, and hybrid ML techniques (discussed in Section 4.4).

Fig. 3 depicts a high-level taxonomy of SDN-based malware solutions, categorized by detection approach (static, dynamic, hybrid), deployment location (controller-based or data-plane-based), and evaluation metrics (accuracy, latency, overhead). This structure clarifies how different research works approach malware defense and enables a deeper understanding of the architectures and trade-offs involved.

4.1 SDN for malware analysis

Ceron et al. [50] proposes the MAlwaRe Analysis Architecture based on SDN (MARS), a solution for malware analysis that can dynamically reconfigure the SDN infrastructure based on the actions of the malicious program. Unlike other approaches, the authors emphasize that the tool

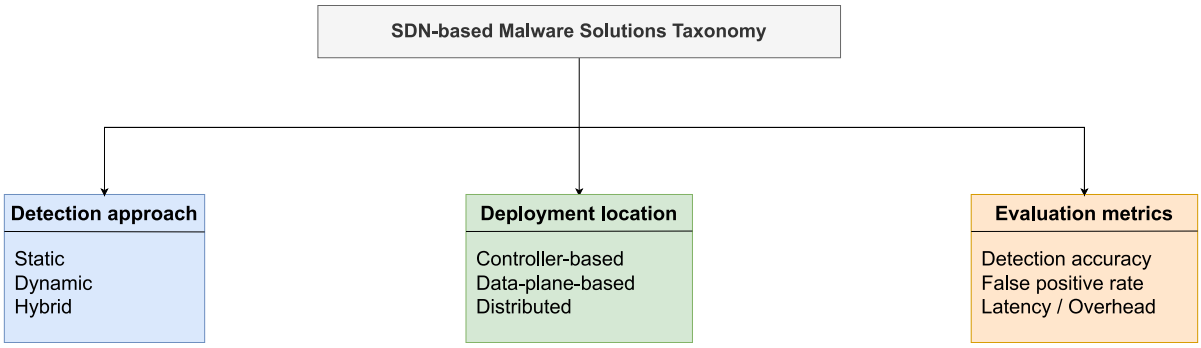

Fig. 3. High-level taxonomy of SDN-based malware solutions, illustrating different detection approaches, deployment strategies, and key evaluation metrics.

focuses on reconfiguring the environment around the sandbox, since malware can exhibit different behaviors depending on the network characteristics. Furthermore, the mechanism can monitor network traffic and modify the topology according to a set of rules defined by the analyst. Thus, the solution can identify new behaviors that would hardly be determined using a traditional sandbox.

Lee and Shin [92] proposes SHIELD, a tool to automate the static analysis of SDN applications. The mechanism can identify malicious rules based on the sequence of calls to the APIs of SDN controllers. To accomplish this, the authors use the Soot framework [107], an open-source tool for static analysis of Java applications. Unfortunately, the proposal was evaluated in only 34 applications (21 malicious and 13 benign). It can take up to 45.26 s on average to assess an artifact, which can be a limitation in the case of benign applications that need to be instantiated promptly by the controller.

4.2 SDN for malware detection

Jin and Wang [90] analyzes the behavior of different malicious artifacts and implements a system for malware detection on mobile devices using SDN. The proposed mechanism is coupled with the SDN controller and identifies suspicious activities on the network by analyzing real-time traffic. Once malicious traffic is detected, the tool adds the attacker's IP address to a blacklist.

Abaid et al. [91] proposes MalwareMonitor, a solution composed of different detection modules capable of elastically balancing traffic between them. The authors highlight the tool's ability to detect coordinated attacks on the network through the Snort IDS.[8] Although details about the different modules of the proposal are exposed, the authors do not present performance results for the solution. Additionally, suggestions for future implementations are listed.

Cabaj and Mazurczyk [93] presents a case study using SDN to mitigate the CryptoWall ransomware.[9] The authors developed a solution capable of detecting malicious traffic and blocking communication between the victim and the attacker. This accomplishment relies on a blacklist containing the addresses of the proxy servers used for communication with the C&C (Command and Control) server used by the original artifact.

Nguyen and Yoo [94] proposes a system for malware detection based on network behavior. The mechanism is coupled with the controller and detects infected devices in real-time from several requests. The proposal checks the number of connection attempts each device makes to determine whether the requests are malicious. After detection, the malicious addresses are added to a blacklist.

Tatang et al. [95] proposes SDN-Guard, a tool for detecting and mitigating rootkits in SDN controllers. The architecture consists of a reverse proxy between the controller and the switches and a decision module responsible for detecting malicious changes in the behavior of the rules installed in the controller. To identify malicious changes, the decision module compares the controller's current network view against the view observed by the proxy. If a difference is noticed, the decision module requests the proxy to invalidate the installed rules. The authors highlight an additional module to allow the adjustment of the tool for different controllers.

Cabaj et al. [96] presents a scheme for dynamic analysis of HTTP traffic using SDN architecture. More specifically, the authors offer a traffic classification method that analyzes the HTTP POST packets' payloads transmitted on the network (between possibly infected machines and the attacker). The authors present the effectiveness of the solution against the ransomware CryptoWall and Locky. Malware is detected on the network when the behavior of the HTTP POST packets diverges from the standard behavior (measured in the learning phase of the strategy). Unfortunately, the authors do not offer a mitigation strategy for the attack in this work. Still, they argue that their mechanism can be used to identify malware behavior on the network and block communications with the attacker via a blacklist.

Lee et al. [97] presents INDAGO, a framework for detecting malicious SDN applications. The tool statically analyzes the code of an SDN application and uses artificial intelligence to detect possibly harmful code snippets. Classification is based on detecting suspicious calls to controller APIs. The authors highlight an accuracy of 96% for the tested samples.

Cusack et al. [98] combines SDN with machine learning for malware detection. Firstly, the authors argue that strategies based on HTTP traffic analysis are no longer effective because recent malware uses the HTTPS protocol, which does not allow inspection of payloads in packets due to the encryption layer employed. Therefore, the authors offer a strategy based on Programmable Forwarding Engines (PFEs), which enables the data collection from each network packet individually and effectively. Furthermore, PFEs are feasible on more advanced SDN switches, such as P4 switches.[10] The presented mechanism aims to extract characteristics of the packet flow in the network and thus create a classification scheme using the Random Forest algorithm. The proposed solution achieves an accuracy of 86%.

Letteri et al. [99] utilizes deep learning methods for identifying bots in SDN-based network infrastructures. In particular, their approach relies on multi-layer perceptron (MLP) neural networks over network traffic to detect malicious behavior. The authors leverage the HogZilla,[11] CTU-13 [108] and ISCX 2012 IDS[12] datasets for training the model. Although the proposed model achieved a detection accuracy rate of 96%, it may be affected by the size of the target network and the volume of traffic, potentially requiring significant computational resources to maintain performance at scale.

[8] https://www.snort.org
[9] https://www.kaspersky.com/blog/cryptowall-3-0-an-evolution-twist/
[10] https://opennetworking.org/p4/
[11] https://ids-hogzilla.org/dataset/
[12] https://www.unb.ca/cic/datasets/ids.html

Maeda et al. [100] proposes a mechanism for detecting malicious traffic originating from infected machines in SDN using deep learning. The tool can generalize its classifications even for new malware families through artificial intelligence. After identifying a malicious artifact, the mechanism isolates infected machines and blocks external connections. In addition, the authors highlight that the algorithm's accuracy was 99.2%.

Akbanov et al. [101] focuses on using SDN for detecting and mitigating ransomware. The system was evaluated against the WannaCry ransomware as proof of concept. The proposed solution performs DNS traffic analysis to detect suspicious queries based on a blacklist, and isolates infected devices from the rest of the network.

Rouka et al. [102] aims to develop a defensive mechanism based on the study of the ExPetr ransomware.[13] To achieve this, the proposed solution inspects SMB and HTTP packets and blocks any malicious traffic based on the payload analysis of the packets. Once the threat is detected, the SDN controller installs rules in the flow tables of switches to mitigate the attack and isolate infected devices.

Alotaibi and Vassilakis [103] offers a scheme that utilizes the SDN architecture for detecting self-propagating ransomware. The authors use BadRabbit ransomware as a case study. In this proposal, various modules in the SDN execute a packet inspection system to identify typical anomalies of ransomware attacks. The proposal is shown to be scalable in terms of CPU consumption, memory, and network parameters (e.g., latency), and it has a high degree of efficiency in detecting threats.

Khan and Akhunzada [104] proposes a hybrid mechanism for malware detection that exploits vulnerabilities in IoT devices, specifically in Internet of Medical Things (IoMT) devices. The solution combines deep learning for malware classification and detection and employs the SDN architecture to facilitate the acquisition and collection of network traffic. The main contribution of the work is the increased effectiveness in malware detection due to the use of two classification models: (1) Convolution Neural Network (CNN) and (2) Long Short-Term Memory (LSTM). Furthermore, the authors demonstrate that this scheme performs better than other classification strategies.

Ahmed et al. [105] presents an SDN-based solution that uses machine learning to detect infected machines that communicate with command and control servers. The developed tool can intercept DNS requests in the network and identify possibly malicious domains generated by Domain Generation Algorithms (DGAs). Additionally, the authors developed three algorithms for detecting malicious HTTP, HTTPS, and UDP traffic, with an accuracy of over 97% in all three trained models. Finally, the tool is validated on a real network over 50 days.

Muthanna et al. [65] offers a mechanism for intrusion detection in IoT networks via deep learning. Their tool analyzes network traffic by using a Cuda Long Short-Term Memory Gated Recurrent Unit (cuLSTMGRU) classifier. The presented model is trained and evaluated using the CICIDS2017 [109] dataset, achieving an accuracy rate of 99.23%.

Chang et al. [56] proposes a tool for malware detection leveraging programmable switches and deep learning. The solution employs a Convolutional Neural Network (CNN) to classify network traffic and identify malicious flows that are later blocked. The authors trained their model with the IoT-23 [110] dataset and claimed that their approach achieves an accuracy of 99%, while imposing minimal computational overhead compared to traditional intrusion detection tools.

Chaganti et al. [57] presents a novel mechanism for intrusion detection in IoT environments based on network traffic classification. Their solution utilizes an LSTM classifier for detecting suspicious activities. The authors leveraged the SDNIoT [111] and SDN-NF-TJ [112] datasets for training and evaluation. The model achieved an accuracy of 97.1%.

Kumar and Kumar [58] presents a new approach for malware detection based on image processing. The authors implemented a CNN that is capable of classifying artifacts based on their image representation. The experimental results indicated that the algorithm was able to achieve a 98.5% detection accuracy with a processing time of 0.006s.

Souza and Arima [59] propose Heimdall, a hybrid approach for malware detection in SDN-enabled IoT scenarios. The tool integrates YARA signatures and a Random Forest machine learning classifier in programmable switches using the P4 language. The model achieved an accuracy of 99.33% against the IoT-23 dataset [110], demonstrating high detection accuracy and low processing time. The authors highlight the advantages of using programmable switches for fast malware detection and the elimination of single points of failure associated with a centralized SDN controller.

Almotiri [106] present a bi-layer (static + dynamic) AI framework aimed at safeguarding SDN-enabled IoMT environments from malware and ransomware. The double-tier design, combining hand-crafted Ember features with grayscale image representations and combining tree-based learners (XGBoost, LightGBM) with a CNN-supported DNN, demonstrates interesting headline metrics (up to 99.60% accuracy and 0.9966 F1 in blended PE datasets). Strengths include the modular architecture, extensive dataset preprocessing, and clear reporting of hyper-parameters, which together bolster reproducibility.

4.3 Taxonomy-based summary of SDN-based proposals

Table 2 provides a refined taxonomy of the analyzed SDN-based proposals. We categorize them by **(a) Detection Approach**, **(b) Platform/Target**, **(c) Key Metrics**, **(d) Key Contributions/Architecture**, and **(f) Mitigation Strategy**, where applicable.

While Table 2 offers an essential overview of SDN-based malware detection approaches, a more granular evaluation reveals context-sensitive limitations across methodologies. For instance, static analysis methods are often effective at early-stage artifact inspection but lack adaptability in dynamic threat environments, limiting their real-time applicability. Conversely, behavior-based systems relying on network traffic patterns may struggle in encrypted or obfuscated communication settings, as observed in HTTPS-based malware bypassing inspection layers. Deep learning models, though performant, typically demand substantial training data and computational overhead, which may not be feasible in resource-constrained deployments. We emphasize that accuracy alone is insufficient; factors such as update latency, deployment complexity, and integration with legacy systems significantly affect real-world usability. Future benchmarking should include multi-dimensional comparisons—detection precision, interpretability, response speed, and integration feasibility.

4.4 Novel ML/DL algorithms in SDN-based detection

Recent solutions emphasize advanced deep learning architectures tailored for SDN-based malware detection. For instance, [57] evaluated an LSTM-based intrusion detection system that leverages temporal dependency in network flows, while [104] proposed a hybrid CNN-LSTM approach to capture both spatial and sequential patterns. Additionally, [65] integrated GRU cells to reduce computational overhead while maintaining high detection rates. Such architectures highlight the ongoing trend of adopting specialized neural networks that can handle the growing volume and complexity of IoT and 5G traffic in SDN deployments.

4.5 Performance metrics

Performance assessment typically involves detection accuracy/rate, false-positive rate (FPR), latency/overhead, and in some cases, resource usage (CPU, memory). For instance, solutions like Maeda et al. [100] reported a detection rate above 99%, while Cusack et al. [98] experienced a moderate overhead due to the fine-grained packet inspection. Approaches deploying CNN or LSTM often measure training

[13] https://www.kaspersky.com/blog/new-ransomware-epidemics/17314/

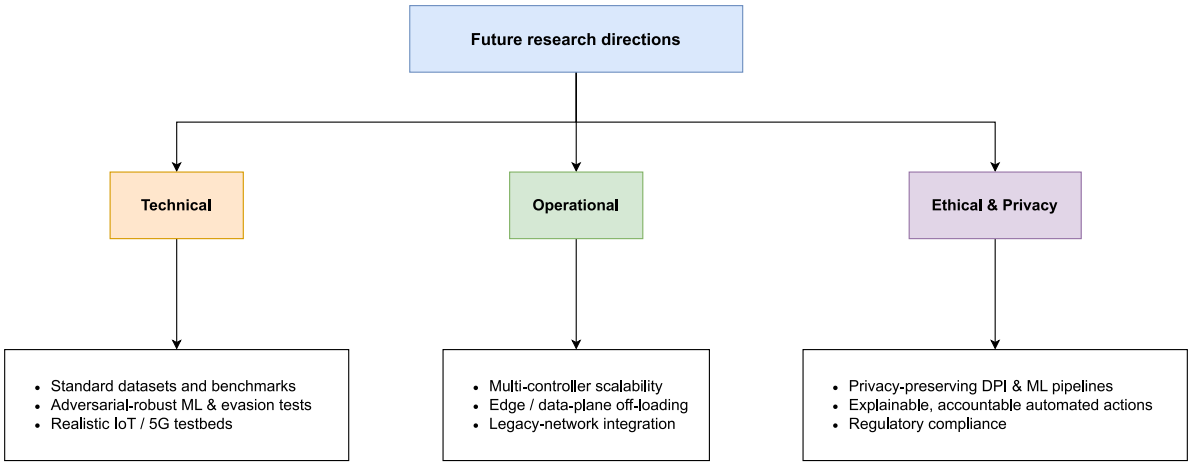

Fig. 4. High-level view of future research directions.

time and inference latency to ensure real-time feasibility, exemplified by Khan and Akhunzada [104] and Chang et al. [56]. However, uniform evaluation metrics are not consistently reported, complicating direct comparisons across studies.

Table 2 attempts to partially standardize these metrics, but variations in dataset size, network topology, and hardware configurations remain limiting factors. The table also indicates whether each proposal implements a mitigation strategy (e.g., flow rule insertion or blacklisting) or remains detection-only, highlighting a critical gap in some approaches that omit real-time remediation.

4.6 Practical implications and potential barriers

Several proposals, such as Ceron et al. [50] and Souza and Arima [59], highlight practical advantages in reconfiguring the network data plane for responsive malware detection or analysis. Nonetheless, integration barriers include hardware costs (particularly for specialized SDN-capable devices), the complexity of deploying AI models at scale, and the need to maintain operational SLAs. Ethical concerns arise when deep packet inspection or advanced telemetry may infringe on user privacy. Overall, while SDN holds great promise for enabling agile, programmable defenses, real-world deployment requires careful balancing of performance, cost, and data governance.

Section summary: We have systematically examined SDN-based approaches, classifying them by detection method and platform, and assessing their performance metrics and mitigation strategies. This addresses **RQ1** by illuminating the landscape of current methodologies, while also answering **RQ2** through discussion of detection rates and overhead. The next section delves deeper into open issues, categorized by technical, operational, and ethical dimensions, and highlights future research directions.

5 Open issues and research challenges

In this section, we address **RQ3** by presenting open challenges in SDN-based malware detection and analysis, summarized in Fig. 4. We categorize these challenges into **technical**, **operational**, and **ethical** dimensions. We also underscore practical implications, potential adoption barriers, and relevant references that shed light on these issues.

Where applicable, we provide concrete real-world examples to showcase how these challenges emerge. For instance, the rapid influx of IoT devices in industrial control systems underscores the scale-related constraints in SDN controller performance, while advanced ransomware campaigns highlight potential ethical ramifications of deep packet inspection techniques.

5.1 Technical challenges

Lack of realistic scenarios. Developing malware detection and analysis solutions is complex, as most tests are performed in controlled environments (sandboxes). Unfortunately, these environments may fail to create realistic scenarios. Furthermore, as Ceron et al. [50] demonstrated, malicious artifacts may have distinct behaviors depending on the network on which they are executed. Therefore, it is necessary to establish parameters (such as hardware and network requirements) to test the proposed solutions under conditions close to those of real environments.

Additionally, deploying large-scale testbeds with high-fidelity traffic is not trivial, especially in hybrid IoT-5G contexts. Future work could build open-access platforms similar to the *GENI* testbed [113] or the *FABRIC* infrastructure [114] to evaluate different SDN-based malware solutions in production-like environments.

Non-standardized datasets. For solutions that use machine learning, it is noteworthy that the absence of standardized databases containing benign and malicious programs makes it difficult to establish metrics for fair comparison of the tools proposed by academia. In addition, some malware programs are known to affect specific regions or countries. Therefore, evaluating whether the detection rate remains dependent on the context in which the artifact is being executed is also necessary.

While dataset initiatives like IoT-23 [110] and SDNIoT [111] are emerging, further curation is needed to ensure comprehensive coverage (e.g., including encrypted payloads, obfuscated malware variants). We propose a community driven benchmarking initiative that includes: (i) multi-format malware samples (e.g., PE, ELF), (ii) encrypted and polymorphic traffic samples, (iii) scenarios simulating SDN-enabled and traditional hybrid infrastructures, and (iv) annotations with ground-truth labels and adversarial examples. Criteria for inclusion should emphasize diversity, reproducibility, and real-world relevance. Datasets must be openly accessible and continuously updated. Collaboration between academia, government Computer Emergency Response Teams (CERTs), and cybersecurity firms is essential for broad coverage and reliability, but privacy and confidentiality challenges may limit data sharing.

Deep learning trade-offs and threats. Although several solutions adopt deep learning models (e.g., CNN, LSTM, GRU), most lack comprehensive analysis of training overhead, interpretability, and robustness. For example, LSTM-based approaches often require extensive temporal data, careful feature engineering, and hyperparameter tuning, which can be computationally expensive and limit deployment flexibility [115]. The black-box nature of deep models reduces transparency, complicating forensic audits and regulatory compliance in sensitive

sectors. Furthermore, recent studies [116,117] reveal deep learning classifiers' vulnerability to adversarial evasion, where imperceptible input perturbations cause misclassification. Defense strategies such as adversarial training, input preprocessing, and robust architecture design mitigate these risks but often increase complexity and degrade performance on clean data. Balancing efficiency, explainability, and security remains a critical challenge for deploying DL-based malware detection systems in real-world SDN environments.

Defenses against adversarial artifacts. Despite growing evidence of machine learning vulnerabilities to adversarial inputs, dedicated SDN-based defenses remain scarce. Ensemble architectures that combine multiple complementary classifiers (e.g., Random Forest with LSTM) improve robustness against evasion attacks by leveraging diverse detection perspectives [118]. Additionally, anomaly detection models that establish baseline network behavior can identify subtle adversarial shifts that evade signature-based methods. We recommend integrating adversarial training, robust feature selection, and continual learning to counteract poisoning and evasion risks, particularly in SDN edge deployments supporting critical infrastructures like healthcare and industrial control systems. Developing standardized adversarial test suites tailored to SDN contexts is crucial to evaluate and enhance defense efficacy.

Absence of standardized testing methodology. A standard dataset is necessary, but other network parameters are not considered, making comprehensive and fair benchmarking of the different tools presented difficult. Therefore, it is necessary to develop a standard methodology for testing malware detection solutions, considering detection rate, latency, overhead, and user experience.

Organizations like the IEEE SDN Initiative[14] could spearhead efforts to define such benchmarks. Creating a widely accepted reference architecture for measurement allows more consistent comparison of solutions. This standardization could parallel existing frameworks like TPC benchmarks[15] for databases or SPEC benchmarks[16] for system performance.

Limited solutions for malware analysis. Few works have committed to creating environments that aim to facilitate the creation of malware analysis mechanisms. The scheme presented in Ceron et al. [50] demonstrates that malicious programs can adopt different behaviors depending on the infrastructure in which they are executed, underscoring the need for innovative dynamic analysis tools within SDN.

Providing open-source frameworks that integrate with container-based solutions (e.g., Docker,[17] Kubernetes[18]) and SDN controllers may expedite dynamic malware analysis. This synergy can better reflect complex multi-tenant 5G/IoT environments and inform the design of more robust detection engines.

Insufficient testing against evasion techniques. One point not considered by the analyzed works is the evasion techniques that can be adopted to make it difficult for static or dynamic analysis to detect malicious artifacts that seek to attack the SDN infrastructure [119–121]. The ubiquity of encrypted traffic in network communications is also a powerfully impactful aspect that has been little considered.

Future research could incorporate adversarial machine learning, focusing on how attackers might poison datasets or adapt malware signatures to evade detection. Formal methods for verifying detection robustness could also be explored, aiming secure software design verification.

Scalability in real-world IoT expansions. With IoT device counts growing exponentially in sectors like smart cities and Industry 4.0, SDN controllers can face bottlenecks in processing large-scale traffic. For instance, a typical citywide IoT deployment may generate millions of flows, overwhelming single SDN controllers [122]. Dynamic load-balancing strategies or multi-controller architectures are needed to handle bursts of malicious activity, ensuring minimal latency and uninterrupted service. Most current proposals overlook scalability in ultra-dense deployments such as smart cities or industrial 5G systems. As SDN controllers scale with increasing flows, response latency and control overhead rise significantly. We recommend architectural shifts toward multi-controller planes, federation schemes, or edge-augmented detection nodes. Load balancing via consistent hashing or service-mesh overlays may further optimize response time under peak load. Emulation platforms like FABRIC[19] should be employed for stress-testing.

5.2. Operational challenges

Network impacts and resource overhead. Many works fail to consider the proposed solution's impact on the protected network. For example, the execution of mechanisms for traffic inspection can cause a slight delay in response to the client due to the processing time of the tool [123]. Some questions must be considered:

1. How can the applied defense processes degrade the functioning of the network?
2. How can the delay imposed by the processing performed by the tool directly influence the quality of service (QoS)?
3. What are the potential damages to the quality of experience (QoE) of users?

Complex ML-driven detection can cause CPU or memory spikes on the SDN controller, risking a single point of failure. Proposed solutions like Souza and Arima [59] mitigate this by offloading classification to the data plane. Further research might evaluate multi-controller or distributed control planes to alleviate performance bottlenecks.

Prevalence of prototyping controllers. A significant number of proposals presented by the scientific community employ controllers used for proof of concept, such as POX or Ryu. Although they streamline the development of new solutions, authors could also evaluate their tools on SDN controllers used by the industry, such as ONOS or OpenDayLight. Another possibility is to perform benchmarks to compare the performance of algorithms on different controllers, facilitating the choice of the best controller and proposal by network operators.

Transitioning from lab setups to production environments may reveal unforeseen integration issues (e.g., plugin compatibility, versioning conflicts). Researchers should consider synergy with large-scale frameworks or real-world 5G testbeds to ensure wide applicability.

Practical deployment barriers. Deploying advanced SDN-based malware detection can face barriers such as the high cost of SDN-capable hardware, the complexity of updating firmware on devices, and the requirement of specialized skill sets. Some organizations may also be reluctant to adopt new architectures without standardized guidelines or robust vendor support. Future solutions should aim for incremental deployment strategies, possibly leveraging containerized or virtualization-based overlays that integrate with legacy networks.

As a real-world example, in certain critical infrastructure sectors (e.g., electrical grids), updating legacy switches to SDN-compatible devices may require significant downtime and retraining staff. The cost and risk implications can slow adoption, despite the heightened benefits of robust SDN-based security.

[14] https://sdn.ieee.org/about
[15] https://www.tpc.org/information/benchmarks5.asp
[16] https://www.spec.org/products/
[17] https://www.docker.com
[18] https://kubernetes.io
[19] https://www.keysight.com/us/en/products/network-test/protocol-load-test/fabric-emulator.html

Granular performance metrics and deployment requirements. Current literature frequently underreports key runtime metrics such as detection latency, throughput, and scalability under realistic traffic volumes. We advocate for benchmarking environments that measure end-to-end latency and accuracy under variable flow rates, packet sizes, and encrypted traffic scenarios. For example, detection latencies reported for models in controlled simulations may not translate effectively to real-time 5G infrastructures operating at 10 Gbps or higher. Moreover, deployment feasibility depends heavily on specific hardware accelerators such as P4-programmable devices, or GPU clusters, which influence latency and energy consumption. Detailed profiling of computational requirements, memory footprint, and hardware dependencies is essential for informed deployment decisions in heterogeneous SDN ecosystems.

Real-world hybrid security validation. While several studies propose hybrid SDN-traditional security integrations, few validate their approaches through comprehensive simulations or live deployments. For instance, combining SDN-based anomaly detection with legacy firewall rule bases necessitates compatibility and consistency testing, particularly under asynchronous policy updates or failover conditions. Pilot case studies leveraging simulated traffic across NGFW (Next-Generation Firewall) plus SDN topologies or experimental deployments in university labs and smart grid testbeds can demonstrate feasibility, latency impact, and policy conflict resolution effectiveness. Such empirical validation is vital to understand operational constraints, interoperability challenges, and performance trade-offs before industrial adoption.

5.3. Ethical and privacy challenges

Deep packet inspection (DPI) and user privacy. While DPI can be highly effective for malware detection, it raises privacy concerns when packets contain sensitive information (e.g., personal data, medical records). Balancing robust security with data privacy obligations (GDPR, HIPAA, etc.) remains an ongoing challenge. Researchers must explore selective or partial inspection, privacy-preserving ML, or on-device detection to reduce potential privacy infringements.

Privacy-preserving detection techniques. Emerging privacy-preserving approaches such as federated learning and homomorphic encryption offer promising alternatives. Federated learning enables decentralized model training on-device (e.g., smartphones, IoT sensors), preserving data locality and reducing central aggregation risks Liu et al. [124]. However, it suffers from challenges including non-Independent and Identically Distributed (IID) data distribution, communication overhead, and degraded model convergence. Homomorphic encryption techniques allow inference directly on encrypted data, ensuring confidentiality but often incur high computational costs and latency. Hybrid schemes combining federated learning with secure multiparty computation may offer balanced trade-offs but require further research to optimize scalability and accuracy in SDN-based detection scenarios.

Accountability in automated defense. Automated quarantine or blocking decisions by an AI-driven SDN controller can inadvertently affect legitimate users. Ensuring accountability and transparency in these automated processes is essential, especially in critical infrastructures (e.g., healthcare). Formal policy verification and explainable AI techniques [125] can provide logs and rationales for each blocking decision, limiting legal or ethical liability.

Section summary: By examining technical, operational, and ethical challenges, we address **RQ3**, underscoring where SDN-based malware defenses require future research. Technical constraints (e.g., realistic testbeds, standardized datasets) must be resolved alongside operational considerations (e.g., overhead, production-grade controller integration) and ethical imperatives (e.g., privacy-preserving inspection). The final section concludes our findings, highlights contributions, and identifies specific areas for future research.

6 Conclusion

This paper provides a comprehensive review of the significant advances and persistent challenges in leveraging the SDN paradigm to enhance malware detection and analysis. Through an in-depth survey of literature spanning from 2013 to 2025, we classify over twenty SDN-based proposals, covering a wide range of approaches, from static analysis to deep learning-driven techniques. We also evaluate their effectiveness based on detection accuracy, false positives, latency, and computational overhead.

Our findings highlight detection rates frequently exceeding 95%, and in some cases reaching 99%, with overhead often contained below 2%. These results emphasize the potential of SDN to meet the demands of modern threat landscapes, particularly in dynamic 5G and IoT environments where flexibility, programmability, and real-time reactivity are essential. We demonstrate how integrating machine learning models into SDN controllers enables agile and adaptive responses that can detect and quarantine threats almost instantaneously.

Despite these advances, several key barriers remain. Our analysis identifies ongoing challenges such as the absence of standardized datasets and benchmarking frameworks, limited testing in real-world scenarios, reliance on prototype SDN controllers, and privacy concerns related to deep packet inspection. Hybrid architectures that combine SDN with traditional security mechanisms offer promising synergies, but they also introduce complexity and require careful attention to integration and performance.

To address these gaps, we propose actionable directions for future research and development:

- **Standardized frameworks:** Develop open-access testbeds and benchmarking methodologies that reflect realistic large-scale IoT environments and traffic patterns.
- **Adversarial-resilient machine learning:** Advance defenses against obfuscation, adversarial examples, and dataset poisoning to ensure robust performance under evolving threats.
- **Privacy-preserving inspection:** Design detection techniques that are compatible with encryption and selective inspection while maintaining user privacy.
- **Scalable and distributed SDN architectures:** Support multi-controller and edge-based SDN designs capable of handling high-volume and mission-critical deployments.
- **Hybrid integration:** Encourage collaboration between academia and industry to align SDN programmability with traditional security tools such as firewalls and endpoint protection.
- **Ethical and regulatory compliance:** Create frameworks that handle personal data responsibly in accordance with regulatory standards and ethical guidelines.

We propose a research roadmap that prioritizes achievable and impactful goals over time. Initially, the focus should be on creating shared resources and standardized evaluation methods to support reproducible research. Mid-term efforts should develop advanced, distributed detection systems with transparent AI, strict security policies, and adaptive threat intelligence exchange. In the long term, emphasis should be placed on global coordination, establishing ethical and legal guidelines, and implementing scalable privacy-focused defenses for critical infrastructures. The following roadmap summarizes these objectives by timeline:

- **Short-term:** Establish open, shared benchmarking datasets, federated simulation testbeds, and standardized adversarial evaluation suites specifically designed for SDN malware detection. These foundational tools will catalyze reproducible research and robust baseline comparisons.
- **Medium-term:** Advance distributed SDN-based detection frameworks that seamlessly integrate explainable AI for transparency, deploy DPI-enabled zero-trust policy enforcement, and enable dynamic, adaptive threat intelligence sharing across network domains.

- **Long-term:** Foster global coordination mechanisms such as dedicated SDN-CERTs, formalize ethical and legal standards for AI-empowered SDN operations, and realize scalable, privacy-preserving defense architectures deployed across critical infrastructure sectors.

In conclusion, SDN presents a powerful foundation for the development of adaptive and intelligent malware defense systems. By addressing the outlined challenges and pursuing the proposed research directions, both academic and industrial communities can fully leverage SDN to strengthen the cybersecurity posture of increasingly connected digital ecosystems.

CRediT authorship contribution statement

Cristian H.M. Souza: Writing – review & editing, Writing – original draft, Conceptualization. **Túlio Pascoal:** Writing – review & editing, Writing – original draft. **Emidio P. Neto:** Writing – review & editing. **Galileu B. Sousa:** Writing – review & editing, Writing – original draft. **Francisco S.L. Filho:** Writing – review & editing. **Daniel M. Batista:** Writing – review & editing, Validation. **Felipe S. Dantas Silva:** Writing – review & editing, Writing – original draft, Conceptualization.

Declaration of competing interest

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

Data availability

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

References

[1] Urooj U, Al-rimy BAS, Zainal A, Ghaleb FA, Rassam MA. Ransomware detection using the dynamic analysis and machine learning: A survey and research directions. Appl Sci 2022;12(1):172.

[2] Davidson R. The fight against malware as a service. Netw Secur 2021;2021(8):7–11.

[3] Meland PH, Bayoumy YFF, Sindre G. The ransomware-as-a-service economy within the darknet. Comput Secur 2020;92:101762.

[4] (GERT) GERT. Incident response analyst report 2024. Technical report, Kaspersky Lab; 2025, URL: https://content.kaspersky-labs.com/fm/site-editor/33/3318ec849851138088d24f26d236f469/source/irreport.pdf.

[5] Morgan S. Cybercrime to cost the world 8 trillion annually in 2023. 2023, URL: https://cybersecurityventures.com/cybercrime-to-cost-the-world-8-trillion-annually-in-2023/.

[6] Networks PA. 2022 unit 42 ransomware threat report. 2022, URL: https://start.paloaltonetworks.com/unit-42-ransomware-threat-report.html.

[7] Dantas Silva FS, Lima MPS, Corujo D, Venâncio Neto AJ, Esposito F. A comprehensive step-wise survey of multiple attribute decision-making mobility approaches. IEEE Access 2024;12:108616–56.

[8] Hassan WH, et al. Current research on internet of things (IoT) security: A survey. Comput Netw 2019;148:283–94.

[9] Ji X, Huang K, Jin L, Tang H, Liu C, Zhong Z, et al. Overview of 5G security technology. Sci China Inf Sci 2018;61(8):1–25.

[10] Ahmad I, Kumar T, Liyanage M, Okwuibe J, Ylianttila M, Gurtov A. Overview of 5G security challenges and solutions. IEEE Commun Stand Mag 2018;2(1):36–43.

[11] Schneider P, Horn G. Towards 5G security. In: 2015 IEEE trustcom/bigDataSE/ISPA, vol. 1, IEEE; 2015, p. 1165–70.

[12] Liyanage M, Ahmad I, Abro AB, Gurtov A, Ylianttila M. A comprehensive guide to 5G security. Wiley Online Library; 2018.

[13] Dutta A, Hammad E. 5G security challenges and opportunities: a system approach. In: 2020 IEEE 3rd 5G world forum. IEEE; 2020, p. 109–14.

[14] Erunkulu OO, Zungeru AM, Lebekwe CK, Mosalaosi M, Chuma JM. 5G mobile communication applications: A survey and comparison of use cases. IEEE Access 2021;9:97251–95.

[15] Schiller E, Aidoo A, Fuhrer J, Stahl J, Ziörjen M, Stiller B. Landscape of IoT security. Comput Sci Rev 2022;44:100467.

[16] Gangolli A, Mahmoud QH, Azim A. A systematic review of fault injection attacks on IOT systems. Electronics 2022;11(13):2023.

[17] Dantas Silva FS, Silva E, Neto EP, Lemos M, Venancio Neto AJ, Esposito F. A taxonomy of DDoS attack mitigation approaches featured by SDN technologies in IoT scenarios. Sensors 2020;20(11):3078.

[18] De Donno M, Dragoni N, Giaretta A, Spognardi A. DDoS-capable IoT malwares: Comparative analysis and mirai investigation. Secur Commun Netw 2018;2018:1–30.

[19] De Donno M, Dragoni N, Giaretta A, Spognardi A. Analysis of DDoS-capable IoT malwares. In: 2017 federated conference on computer science and information systems. IEEE; 2017, p. 807–16.

[20] Vasques AT, Gondim JJ. Amplified reflection ddos attacks over iot mirrors: A saturation analysis. In: 2019 workshop on communication networks and power systems. IEEE; 2019, p. 1–6.

[21] Zaman U, Mehmood F, Iqbal N, Kim J, Ibrahim M. Towards secure and intelligent internet of health things: A survey of enabling technologies and applications. Electronics 2022;11(12):1893.

[22] Shahid J, Ahmad R, Kiani AK, Ahmad T, Saeed S, Almuhaideb AM. Data protection and privacy of the internet of healthcare things (IoHTs). Appl Sci 2022;12(4):1927.

[23] Sadhu PK, Yanambaka VP, Abdelgawad A. Internet of things: Security and solutions survey. Sensors 2022;22(19):7433.

[24] Martins de Souza CH, Lemos MO, Dantas Silva FS, Souza Alves RL. On detecting and mitigating phishing attacks through featureless machine learning techniques. Internet Technol Lett 2020;3(1):e135.

[25] Sadiq A, Anwar M, Butt RA, Masud F, Shahzad MK, Naseem S, et al. A review of phishing attacks and countermeasures for internet of things-based smart business applications in industry 4.0. Hum Behav Emerg Technol 2021;3(5):854–64.

[26] Abbas SG, Vaccari I, Hussain F, Zahid S, Fayyaz UU, Shah GA, Bakhshi T, et al. Identifying and mitigating phishing attack threats in IoT use cases using a threat modelling approach. Sensors 2021;21(14):4816.

[27] Boyapati M, Gutta BC, Bhuiyan MZA, Son J. Anti-phishing approaches in the era of the internet of things. In: Towards a wireless connected world: achievements and new technologies. Springer; 2022, p. 35–63.

[28] Farhin F, Kaiser MS, Mahmud M. Towards secured service provisioning for the internet of healthcare things. In: 2020 IEEE 14th international conference on application of information and communication technologies. IEEE; 2020, p. 1–6.

[29] Raghuvanshi A, Singh UK, Joshi C. A review of various security and privacy innovations for IoT applications in healthcare. Adv Heal Syst: Empower Physicians IoT Enabled Technol 2022;43–58.

[30] Sharma R, Arya R. Security threats and measures in the internet of things for smart city infrastructure: A state of art. Trans Emerg Telecommun Technol 2022;e4571.

[31] Silva FSD, Silva SN, da Silva LM, Bessa A, Ferino S, Paiva P, et al. ML-based inter-slice load balancing control for proactive offloading of virtual services. Comput Netw 2024;246:110422.

[32] Mahdi SS, Abdullah AA. Survey on enabling network slicing based on SDN/NFV. In: International conference on information systems and intelligent applications. Springer; 2022, p. 733–58.

[33] Babiker Mohamed M, Matthew Alofe O, Ajmal Azad M, Singh Lallie H, Fatema K, Sharif T. A comprehensive survey on secure software-defined network for the internet of things. Trans Emerg Telecommun Technol 2022;33(1):e4391.

[34] Kreutz D, Ramos FM, Verissimo PE, Rothenberg CE, Azodolmolky S, Uhlig S. Software-defined networking: A comprehensive survey. Proc IEEE 2014;103(1):14–76.

[35] Maleh Y, Qasmaoui Y, El Gholami K, Sadqi Y, Mounir S. A comprehensive survey on SDN security: threats, mitigations, and future directions. J Reliab Intell Environ 2022;1–39.

[36] Liatifis A, Sarigiannidis P, Argyriou V, Lagkas T. Advancing SDN from OpenFlow to P4: A survey. ACM Comput Surv 2023;55(9):1–37.

[37] Dantas Silva FS, Neto EP, Nunes RS, Souza CH, Neto AJ, Pascoal T. Securing software-defined networks through adaptive moving target defense capabilities. J Netw Syst Manage 2023;31(3):61.

[38] Dantas Silva FS, Neto EP, Nunes RS, Souza CH, Neto AJ, Pascoal T. Securing software-defined networks through adaptive moving target defense capabilities. J Netw Syst Manage 2023;31(3):61.

[39] Hasneen J, Sadique KM. A survey on 5G architecture and security scopes in SDN and NFV. In: Applied information processing systems: proceedings of ICCET 2021. Springer; 2022, p. 447–60.

[40] Abdul Ghaffar AA, Mahmoud A, Sheltami T, Abu-Amara M. A survey on software-defined networking-based 5G mobile core architectures. Arab J Sci Eng 2022;1–18.

[41] Vishwakarma R, Jain AK. A survey of ddos attacking techniques and defence mechanisms in the IoT network. Telecommun Syst 2020;73(1):3–25.

[42] Borys A, Kamruzzaman A, Thakur HN, Brickley JC, Ali ML, Thakur K. An evaluation of IoT ddos cryptojacking malware and mirai botnet. In: 2022 IEEE world AI ioT congress. IEEE; 2022, p. 725–9.

[43] Kumari P, Jain AK. A comprehensive study of ddos attacks over IoT network and their countermeasures. Comput Secur 2023;103096.

[44] Savithri G, Mohanta BK, Dehury MK. A brief overview on security challenges and protocols in internet of things application. In: 2022 IEEE international IOT, electronics and mechatronics conference. IEEE; 2022, p. 1–7.

[45] Kenaza R, Khemane A, Bendjenna H, Meraoumia A, Laimeche L. Internet of things (IoT): Architecture, applications, and security challenges. In: 2022 4th international conference on pattern analysis and intelligent systems. IEEE; 2022, p. 1–5.

[46] Alhijawi B, Almajali S, Elgala H, Salameh HB, Ayyash M. A survey on DoS/DDoS mitigation techniques in SDNs: Classification, comparison, solutions, testing tools and datasets. Comput Electr Eng 2022;99:107706.

[47] de Assis MV, Carvalho LF, Rodrigues JJ, Lloret J, Proença Jr. ML. Near real-time security system applied to SDN environments in IoT networks using convolutional neural network. Comput Electr Eng 2020;86:106738.

[48] Yoon C, Park T, Lee S, Kang H, Shin S, Zhang Z. Enabling security functions with SDN: A feasibility study. Comput Netw 2015;85:19–35.

[49] Bhardwaj S, Harit S. SDN-Enabled Secure IoT Architecture Development: A Review. Inventive communication and computational technologies: Proceedings of ICICCT, Springer; 2022, p. 599–619.

[50] Ceron JM, Margi CB, Granville LZ. MARS: An SDN-based malware analysis solution. In: 2016 IEEE symposium on computers and communication. IEEE; 2016, p. 525–30.

[51] Rahouti M, Xiong K, Xin Y, Jagatheesaperumal SK, Ayyash M, Shaheed M. SDN security review: Threat taxonomy, implications, and open challenges. IEEE Access 2022;10:45820–54.

[52] Sahbi A, Jaidi F, Bouhoula A. Machine learning algorithms for enhancing intrusion detection within SDN/nfv. In: 2023 international wireless communications and mobile computing. IEEE; 2023, p. 602–7.

[53] Paladi N. Towards secure SDN policy management. In: 2015 IEEE/ACM 8th international conference on utility and cloud computing. IEEE; 2015, p. 607–11.

[54] Alsaeedi M, Mohamad MM, Al-Roubaiey AA. Toward adaptive and scalable OpenFlow-SDN flow control: A survey. IEEE Access 2019;7:107346–79.

[55] Taylor CR, MacFarland DC, Smestad DR, Shue CA. Contextual, flow-based access control with scalable host-based sdn techniques. In: IEEE INFOCOM 2016-the 35th annual IEEE international conference on computer communications. IEEE; 2016, p. 1–9.

[56] Chang H-F, Wang MI-C, Hung C-H, Wen CH-P. Enabling malware detection with machine learning on programmable switch. In: NOMS 2022-2022 IEEE/iFIP network operations and management symposium. IEEE; 2022, p. 1–5.

[57] Chaganti R, Suliman W, Ravi V, Dua A. Deep learning approach for SDN-enabled intrusion detection system in IoT networks. Information 2023;14(1):41.

[58] Kumar S, Kumar A. Image-based malware detection based on convolution neural network with autoencoder in industrial internet of things using software defined networking honeypot. Eng Appl Artif Intell 2024;133:108374.

[59] Souza CHM, Arima CH. A hybrid approach for malware detection in SDN-enabled IoT scenarios. Internet Technol Lett 2024;n/a(n/a):e534. http://dx.doi.org/10.1002/itl2.534.

[60] Or-Meir O, Nissim N, Elovici Y, Rokach L. Dynamic malware analysis in the modern era—A state of the art survey. ACM Comput Surv 2019;52(5):1–48.

[61] Ucci D, Aniello L, Baldoni R. Survey of machine learning techniques for malware analysis. Comput Secur 2019;81:123–47.

[62] Aslan ÖA, Samet R. A comprehensive review on malware detection approaches. IEEE Access 2020;8:6249–71.

[63] Alsmadi T, Alqudah N. A survey on malware detection techniques. In: 2021 international conference on information technology. IEEE; 2021, p. 371–6.

[64] Singh J, Singh J. A survey on machine learning-based malware detection in executable files. J Syst Archit 2021;112:101861.

[65] Muthanna MSA, Alkanhel R, Muthanna A, Rafiq A, Abdullah WAM. Towards SDN-enabled, intelligent intrusion detection system for internet of things (IoT). IEEE Access 2022;10:22756–68.

[66] Bazrafshan Z, Hashemi H, Fard SMH, Hamzeh A. A survey on heuristic malware detection techniques. In: The 5th conference on information and knowledge technology. IEEE; 2013, p. 113–20.

[67] Odusami M, Abayomi-Alli O, Misra S, Shobayo O, Damasevicius R, Maskeliunas R. Android malware detection: A survey. In: International conference on applied informatics. Springer; 2018, p. 255–66.

[68] Qiu J, Zhang J, Luo W, Pan L, Nepal S, Xiang Y. A survey of android malware detection with deep neural models. ACM Comput Surv 2020;53(6):1–36.

[69] Kouliaridis V, Kambourakis G. A comprehensive survey on machine learning techniques for android malware detection. Information 2021;12(5):185.

[70] Ye Y, Li T, Adjeroh D, Iyengar SS. A survey on malware detection using data mining techniques. ACM Comput Surv 2017;50(3):1–40.

[71] Souri A, Hosseini R. A state-of-the-art survey of malware detection approaches using data mining techniques. Human Centric Comput Inf Sci 2018;8(1):1–22.

[72] Tripathy SN, Kapat SK, Kumar DA, Nayak M, Das SK. A survey on malware detection approaches using EULA analysis with text mining. In: 2018 2nd international conference on data science and business analytics. IEEE; 2018, p. 517–22.

[73] Mahdavifar S, Ghorbani AA. Application of deep learning to cybersecurity: A survey. Neurocomputing 2019;347:149–76.

[74] Negera WG, Schwenker F, Debelee TG, Melaku HM, Ayano YM. Review of botnet attack detection in SDN-enabled IoT using machine learning. Sensors 2022;22(24):9837.

[75] Madan S, Sofat S, Bansal D. Tools and techniques for collection and analysis of internet-of-things malware: A systematic state-of-art review. J King Saud Univ Comput Inf Sci 2022.

[76] Tayyab U-e-H, Khan FB, Durad MH, Khan A, Lee YS. A survey of the recent trends in deep learning based malware detection. J Cybersecur Priv 2022;2(4):800–29.

[77] Gopinath M, Sethuraman SC. A comprehensive survey on deep learning based malware detection techniques. Comput Sci Rev 2023;47:100529.

[78] Janabi AH, Kanakis T, Johnson M. Survey: Intrusion detection system in software-defined networking. IEEE Access 2024;12:164097–120.

[79] Alzahrani S, Xiao Y, Asiri S, Zheng J, Li T. A survey of ransomware detection methods. IEEE Access 2025;13:57943–82.

[80] Yang Z, Liu X, Li T, Wu D, Wang J, Zhao Y, Han H. A systematic literature review of methods and datasets for anomaly-based network intrusion detection. Comput Secur 2022;102675.

[81] Thakkar A, Lohiya R. A survey on intrusion detection system: feature selection, model, performance measures, application perspective, challenges, and future research directions. Artif Intell Rev 2022;55(1):453–563.

[82] Campos EM, Saura PF, González-Vidal A, Hernández-Ramos JL, Bernabé JB, Baldini G, et al. Evaluating federated learning for intrusion detection in internet of things: Review and challenges. Comput Netw 2022;203:108661.

[83] Kumar R, Kela R, Singh S, Trujillo-Rasua R. APT attacks on industrial control systems: A tale of three incidents. Int J Crit Infrastruct Prot 2022;37:100521.

[84] Do Xuan C, Huong D. A new approach for APT malware detection based on deep graph network for endpoint systems. Appl Intell 2022;52(12):14005–24.

[85] Chen Z, Liu J, Shen Y, Simsek M, Kantarci B, Mouftah HT, et al. Machine learning-enabled iot security: Open issues and challenges under advanced persistent threats. ACM Comput Surv 2022;55(5):1–37.

[86] Oz H, Aris A, Levi A, Uluagac AS. A survey on ransomware: Evolution, taxonomy, and defense solutions. ACM Comput Surv 2022;54(11s):1–37.

[87] Muralidharan T, Cohen A, Gerson N, Nissim N. File packing from the malware perspective: Techniques, analysis approaches, and directions for enhancements. ACM Comput Surv 2022;55(5):1–45.

[88] Galloro N, Polino M, Carminati M, Continella A, Zanero S. A systematical and longitudinal study of evasive behaviors in windows malware. Comput Secur 2022;113:102550.

[89] Sharma A, Gupta BB, Singh AK, Saraswat V. Orchestration of APT malware evasive manoeuvers employed for eluding anti-virus and sandbox defense. Comput Secur 2022;115:102627.

[90] Jin R, Wang B. Malware detection for mobile devices using software-defined networking. In: 2013 second GENI research and educational experiment workshop. IEEE; 2013, p. 81–8.

[91] Abaid Z, Rezvani M, Jha S. MalwareMonitor: an SDN-based framework for securing large networks. In: Proceedings of the 2014 coNEXT on student workshop. 2014, p. 40–2.

[92] Lee C, Shin S. Shield: an automated framework for static analysis of sdn applications. In: Proceedings of the 2016 ACM international workshop on security in software defined networks & network function virtualization. 2016, p. 29–34.

[93] Cabaj K, Mazurczyk W. Using software-defined networking for ransomware mitigation: the case of cryptowall. IEEE Netw 2016;30(6):14–20.

[94] Nguyen T-H, Yoo M. A behavior-based mobile malware detection model in software-defined networking. In: 2017 international conference on information science and communications technologies. IEEE; 2017, p. 1–3.

[95] Tatang D, Quinkert F, Frank J, Röpke C, Holz T. SDN-guard: Protecting SDN controllers against SDN rootkits. In: 2017 IEEE conference on network function virtualization and software defined networks. IEEE; 2017, p. 297–302.

[96] Cabaj K, Gregorczyk M, Mazurczyk W. Software-defined networking-based crypto ransomware detection using HTTP traffic characteristics. Comput Electr Eng 2018;66:353–68.

[97] Lee C, Yoon C, Shin S, Cha SK. INDAGO: A new framework for detecting malicious SDN applications. In: 2018 IEEE 26th international conference on network protocols. IEEE; 2018, p. 220–30.

[98] Cusack G, Michel O, Keller E. Machine learning-based detection of ransomware using SDN. In: Proceedings of the 2018 ACM international workshop on security in software defined networks & network function virtualization. 2018, p. 1–6.

[99] Letteri I, Della Penna G, De Gasperis G. Botnet detection in software defined networks by deep learning techniques. In: Cyberspace safety and security: 10th international symposium, CSS 2018, amalfi, Italy, October 29–31, 2018, proceedings 10. Springer; 2018, p. 49–62.

[100] Maeda S, Kanai A, Tanimoto S, Hatashima T, Ohkubo K. A botnet detection method on SDN using deep learning. In: 2019 IEEE international conference on consumer electronics. IEEE; 2019, p. 1–6.

[101] Akbanov M, Vassilakis VG, Logothetis MD. Ransomware detection and mitigation using software-defined networking: The case of WannaCry. Comput Electr Eng 2019;76:111–21.

[102] Rouka E, Birkinshaw C, Vassilakis VG. SDN-based malware detection and mitigation: The case of expetr ransomware. In: 2020 IEEE international conference on informatics, ioT, and enabling technologies. IEEE; 2020, p. 150–5.

[103] Alotaibi FM, Vassilakis VG. Sdn-based detection of self-propagating ransomware: The case of badrabbit. IEEE Access 2021;9:28039–58.

[104] Khan S, Akhunzada A. A hybrid DL-driven intelligent SDN-enabled malware detection framework for internet of medical things (IoMT). Comput Commun 2021;170:209–16.

[105] Ahmed J, Gharakheili HH, Russell C, Sivaraman V. Automatic detection of DGA-enabled malware using SDN and traffic behavioral modeling. IEEE Trans Netw Sci Eng 2022;9(4):2922–39.
[106] Almotiri SH. AI driven IOMT security framework for advanced malware and ransomware detection in SDN. J Cloud Comput 2025;14(1):19.
[107] Lam P, Bodden E, Lhoták O, Hendren L. The soot framework for java program analysis: a retrospective. In: Cetus users and compiler infastructure workshop (CETUS 2011), vol. 15, (35):2011, p. 1–8.
[108] Garcia S, Grill M, Stiborek J, Zunino A. An empirical comparison of botnet detection methods. Comput Secur 2014;45:100–23.
[109] Sharafaldin I, Lashkari AH, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: ICISSp, vol. 1, 2018, p. 108–16.
[110] Garcia S, Parmisano A, Erquiaga MJ. IoT-23: A labeled dataset with malicious and benign IoT network traffic. Zenodo; 2020, http://dx.doi.org/10.5281/zenodo.4743746, More details here https://www.stratosphereips.org/datasets-iot23.
[111] Sarica AK, Angin P. A novel sdn dataset for intrusion detection in iot networks. In: 2020 16th international conference on network and service management. IEEE; 2020, p. 1–5.
[112] jafarian T. SDN-NF-TJ. IEEE Dataport; 2019, http://dx.doi.org/10.21227/0s6v-p761.
[113] Wang Y, Hsin W-J, Lamsal M. EdGENI: Making GENI user-friendly for general computer education. In: Proceedings of the 53rd ACM technical symposium on computer science education-volume 1. 2022, p. 801–7.
[114] Baldin I, Nikolich A, Griffioen J, Monga IIS, Wang K-C, Lehman T, et al. Fabric: A national-scale programmable experimental network infrastructure. IEEE Internet Comput 2020;23(6):38–47.
[115] Bacanin N, Stoean C, Zivkovic M, Rakic M, Strulak-Wójcikiewicz R, Stoean R. On the benefits of using metaheuristics in the hyperparameter tuning of deep learning models for energy load forecasting. Energies 2023;16(3):1434.
[116] Wang S, Ko RK, Bai G, Dong N, Choi T, Zhang Y. Evasion attack and defense on machine learning models in cyber-physical systems: A survey. IEEE Commun Surv Tutor 2023;26(2):930–66.
[117] Apruzzese G, Conti M, Yuan Y. SpacePhish: The evasion-space of adversarial attacks against phishing website detectors using machine learning. In: Proceedings of the 38th annual computer security applications conference. 2022, p. 171–85.
[118] Rimon SI, Haque MM. Malware detection and classification using hybrid machine learning algorithm. In: International conference on intelligent computing & optimization. Springer; 2022, p. 419–28.
[119] Marpaung JA, Sain M, Lee H-J. Survey on malware evasion techniques: State of the art and challenges. In: 2012 14th international conference on advanced communication technology. IEEE; 2012, p. 744–9.
[120] Olaimat MN, Maarof MA, Al-rimy BAS. Ransomware anti-analysis and evasion techniques: A survey and research directions. In: 2021 3rd international cyber resilience conference. IEEE; 2021, p. 1–6.
[121] Bulazel A, Yener B. A survey on automated dynamic malware analysis evasion and counter-evasion: Pc, mobile, and web. In: Proceedings of the 1st reversing and offensive-oriented trends symposium. 2017, p. 1–21.
[122] Houhamdi Z, Athamena MR, Athamena B, Eletter S. An optimized SDN framework for the internet of things. In: 2024 11th international conference on software defined systems. IEEE; 2024, p. 164–9.
[123] Silva FSD, Pascoal T, Neto EP, Nunes RSS, Souza CHM, Neto A. An adaptive moving target defense approach for software-defined networking protection. In: NOMS 2023 IEEE/iFIP network operations and management symposium. NOMS 2023 IEEE/IFIP Network Operations and Management Symposium; 2023, p. 1–6.
[124] Liu B, Lv N, Guo Y, Li Y. Recent advances on federated learning: A systematic survey. Neurocomputing 2024;128019.
[125] Minh D, Wang HX, Li YF, Nguyen TN. Explainable artificial intelligence: a comprehensive review. Artif Intell Rev 2022;1–66.

Cristian Souza is an incident response specialist at Kaspersky Lab and a researcher at the LaTARC Research Lab. He holds a B.Tech. in Computer Networks from the Federal Institute of Education, Science, and Technology of Rio Grande do Norte (IFRN), an MSc in Management and Technology from Centro Paula Souza (CPS), and is currently pursuing a Ph.D. in Computer Science at the University of São Paulo (USP). He also holds several international cybersecurity certifications, including CISSP, GCFA, GREM, GXPN, GCTI, GRID and GCIH. His research interests include Malware Analysis, Reverse Engineering, Software-Defined Networking, and Artificial Intelligence.

Túlio Pascoal is a Cybersecurity & Privacy Expert at PricewaterhouseCoopers Luxembourg. He holds a Ph.D. in Computer Science specialized in privacy-enhancing technologies for secure and end-to-end private multiparty environments from the Université du Luxembourg. He has been a reviewer board member in several peer-reviewed journals and conferences, such as IEEE Transactions on Network and Service Management and Communications Letters. His research interests include network and system security, privacy-preserving and privacy-enhancing protocols, and software-defined networking.

Emidio P. Neto is a researcher at the LaTARC Research Lab. In addition, he holds a Master's degree in Computer Science from the Federal University of Rio Grande do Norte (UFRN) and is a member of the Research Group in Future Internet Service and Applications (REGINA). His research interests include Software-Defined Networking, Cloud Computing, Network Slicing MANO, Programmable Data Planes, and Network Security.

Galileu Batista is a Computer Forensics Expert at the Brazilian Federal Police and is also a Professor at the Federal Institute of Education, Science, and Technology of Rio Grande do Norte (IFRN). In addition, he holds a Master's degree in Computer Science from the State University of Campinas (UNICAMP) and has more than 20 years of experience in Computer Security. His research interests include malware analysis, reverse engineering, operating systems, and compiler construction.

Francisco S.L. Filho is a Professor at the Federal Institute of Education, Science, and Technology of Rio Grande do Norte (IFRN). He received the Ph.D. degree in Computer Engineering from the Federal University of Rio Grande do Norte (UFRN). His research interests include IT Service Management, Software-Defined Networking (SDN), Cloud Computing, Machine Learning, and Security.

Daniel M. Batista received his Computer Science (2003) degree from the Federal University of Bahia, Brazil, MSc in Computer Science (2006), and PhD in Computer Science (2010) degrees from the State University of Campinas (Unicamp), Brazil. Currently, he is an Associate Professor at the University of São Paulo (USP) and the Executive Coordinator of Information Technology at USP. His main research interests are IoT Security, B5G, and Data Analytics applied to Computer Networks.

Felipe S. Dantas Silva received the Ph.D. degree in Computer Science from the Federal University of Rio Grande do Norte (UFRN). He is currently an Associate Professor at the Federal Institute of Education, Science, and Technology of Rio Grande do Norte (IFRN), Brazil. He is also the Research Team Lead of the LaTARC Research Laboratory, IFRN. His research interests include network softwarization/virtualization, mobility management, cloud/edge computing, network/cloud slicing, QoS/QoE, machine learning, and security.