IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 29 January 2026, accepted 5 March 2026, date of publication 12 March 2026, date of current version 18 March 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3673172

SURVEY

SoK: Software-Defined Networks Security With Machine Learning

ALPEREN ÖRSDEMİR[1], UTKU TEFEK[2], ERTEM ESİNER[2], AND ALPTEKİN KÜPÇÜ[1,3], (Senior Member, IEEE)

[1]Koç University, 34450 İstanbul, Türkiye
[2]Illinois Advanced Research Center at Singapore, Singapore 138602
[3]KUIS AI Center, 34450 İstanbul, Türkiye

Corresponding author: Alperen Örsdemir (aorsdemir22@ku.edu.tr)

This work was supported in part by TÜBİTAK, the Scientific and Technological Research Council of Türkiye under Grant 123E462 and Grant 124N941; and in part by the National Research Foundation, Prime Minister's Office, Singapore, through the Campus for Research Excellence and Technological Enterprise (CREATE) Program.

ABSTRACT Software-Defined Networks (SDN) have revolutionized modern networking by enabling flexible, programmable management of network resources. This flexibility facilitates the effective design and deployment of Machine Learning (ML)-based defense mechanisms, including Intrusion Detection Systems (IDS) and anomaly detection. However, the validity of existing SDN-based threat detection solutions for systems that use SDN utilities remains unresolved. This work presents a Systematization of Knowledge (SoK) that synthesizes the literature on ML-based SDN security. The study aims to: (i) analyze and strengthen the validity of reported success in SDN security with ML by reviewing 50 recent high-ranking papers, using a taxonomy-driven analysis that categorizes evaluation metrics and the use of ML models, datasets, controllers, and SDN frameworks; (ii) critically assess the state of the literature by comparing these findings with primary surveys and questioning reported accuracy rates; and (iii) identify future perspectives and key takeaways for security framework deployment, to propose solutions to address validation challenges, and to outline a hybrid model. The outlined hybrid model combines passive DL-based traffic monitoring with triggered active mitigation, mapping datasets, ML model families, and programmable enforcement mechanisms into a layered SDN defense to improve validity, efficiency, and real-world deployability.

INDEX TERMS Anomaly detection, intrusion detection system, machine learning, network functions virtualization, software-defined networking.

I. INTRODUCTION

Sdn [1], [2] offers a flexible and customisable paradigm for modern networking. Countries such as China and the UK have adopted SDN to improve infrastructure scalability, particularly in SD-IoT (Software Defined Internet of Things), SD-IIoT (Software Defined Industrial IoT), SDV (Software Defined Vehicles), and SD-LEO (Software Defined Low Earth Orbit Satellite Networks) [3], [4].

SDN's separation of Control and Data planes supports the effectiveness of ML-based security frameworks in virtualized environments. Growing reliance on computing resources for security frameworks further accelerates SDN adoption for secure, virtualized infrastructures [5]. This accelerated growth demands an equivalent increase in security measures, making the development of effective ML-based security frameworks crucial. Wider GPU adoption has provided the necessary computational power to process large volumes of data. In addition, this makes AI-driven security more feasible against sophisticated cyber threats in real time [6].

The effectiveness of ML in SDN-based security frameworks depends on the strategic selection of ML models, which balance detection accuracy with real-world resilience against adversarial threats. Tree-based models such as Decision Trees (DT), Random Forests (RF) offer high interpretability for threat response, while sequential models

The associate editor coordinating the review of this manuscript and approving it for publication was Mehul S. Raval.

such as Recurrent Neural Networks and Long Short-Term Memory (LSTM) excel at capturing temporal patterns in network traffic, identifying evolving attack signatures [7], [8], [9]. However, these models face significant vulnerabilities: attack detection evasion, for example, adversarial perturbations that bypass detection thresholds and dataset poisoning in examples; malicious data injection leading to false positives/negatives directly undermines their reliability [10], [11]. Crucially, such vulnerabilities are exacerbated by the interplay between model architecture and dataset integrity [12]. For instance, poisoned datasets can cause tree-based models to misclassify benign traffic as threats, while sequential models may exhibit catastrophic failure during evasion attacks targeting temporal sequences. Thus, optimizing ML efficacy in SDN security requires not only model selection but also robust defenses against poisoning and evasion challenges that directly impact the attack detection phase of the SDN defense pipeline.

This study surveys 50 recent high-quality publications in the literature to dissect the knowledge from ML studies on SDN security. This research generates a taxonomy out of *security frameworks*, *datasets*, *ML models*, examining successes, and *controllers* used in SDN simulations.

A. CONTRIBUTION

To address the best use of SDN studies, research gaps are identified with 10 high-quality related works as mentioned in Section I-B, open issues, and future perspectives for each result are examined. In particular, two major unresolved problems emerge: the validity of ML-based SDN security frameworks and the lack of rigorous validation of ML model effectiveness and reported success rates. To systematically analyze these aspects, a research taxonomy has been defined to capture security frameworks with an analysis in Figure 1 consisting of framework elements such as datasets, ML models, frameworks, and controllers. Considering the taxonomy created, a research gap has been addressed with the 5 research questions as follows.

1) Which components of SDN security ML frameworks must be fine-tuned for effective security for a real-world validation?
2) Which data handling methods can be improved for better validity of success on SDN security ML frameworks?
3) What perspectives for ML models can improve validity in SDN security frameworks? Which ML models are most successful, and what are the overfitting issues? What can be considered for further validation of ML?
4) How is the research balance across SDN security ML frameworks?
5) What factors define an optimal ML framework for SDN security?

After formulating guiding research questions to address the literature gap, we undertook evaluations to inform the following resolutions.

1) Demonstrated a discussion on the trade-off between security and efficiency, outlining an optimal framework architecture to address open problems and challenges.
2) Examined dataset fields for security-specific results and suggested data-handling methods to improve the validity of SDN security ML frameworks.
3) Analyzed ML families for overage and success validity, examined ML architectures for optimal validity.
4) Various emerging frameworks are identified, and a balance in the literature is examined for potential further study direction.
5) Following the research gap examination and discussions, an **optimal security framework** is identified as a proposal combining the structural elements of taxonomy.

B. METHODOLOGY

To ensure a comprehensive and reproducible evaluation, we conducted a structured literature search and screening as part of an SoK study spanning from January 2024 to June 2025 (both inclusive). A total of 633 papers were initially identified from Google Scholar using 11 distinct search phrases generated with ChatGPT 4o [13] covering the domains of SDN and ML security. After filtering using strict inclusion and exclusion criteria, we selected 50 primary research papers and 10 reviews.

In addition, the taxonomy comparison in Table 4 presents a side-by-side evaluation across eight dimensions ranging from architectural depth to dataset coverage, demonstrating that this work uniquely examines SDN virtualization, ML model taxonomy, controller usage, and datasets into frameworks.

1) USED SEARCH PHRASES

- SDN Machine Learning Security
- ML-based threat detection in SDN
- DDoS attack detection on SDN with ML
- Enhancing SDN security through ML-based network behavior analysis
- Real-Time Anomaly Detection in SDN using supervised learning techniques
- Advanced Security Framework for SDN
- ML-Based Threat Detection on SDN
- Deep Learning Approaches for IDS's in SDN
- A Review of ML models for Securing SDN Infrastructures
- Implementing AI and ML models for SDN Security: Challenges and Solutions
- Hybrid ML Models for Efficient SDN Security

2) INCLUSION CRITERIA

For a fair comparison of performance claims across heterogeneous SDN security studies, we analyze the literature through the taxonomy in Figure 1. The taxonomy captures five dimensions that influence reported success rates: SDN foundations, controller environment, security objective, ML model family

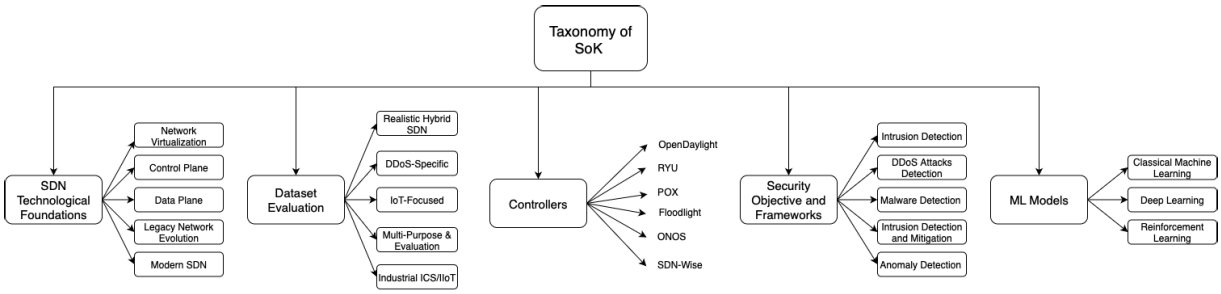

FIGURE 1. Taxonomy of SoK.

used within the framework, and dataset. Throughout this paper, we "digest" reported performance claims by mapping each framework to these dimensions and then interpreting metrics within the corresponding context (e.g., objective, controller, and dataset realism). This enables (a) grouping results under comparable settings, (b) explaining why similar models yield different outcomes, and (c) identifying gaps where high reported accuracy is not transferable due to mismatched datasets, controllers, or objectives. To assess academic quality, we referenced the CORE Conference Rankings [14] for conferences and the SCImago Journal Rank (SJR) [15] for journal quartile classification.

- **A-tier Conferences / Q1 Journals:** 131 papers identified, 40 retained after screening.
- **B-tier Conferences / Q2 Journals:** 25 papers identified, 10 retained.
- **Review Papers:** 10 state-of-the-art SoK or survey/review papers included for comparison and benchmarking.
- Inclusion of a synthetic or benchmark dataset suitable for SDN security tasks.
- Application of at least one success metric provided ML model for a framework on SDN-specific security (e.g., IDS, IDPS, DDoS, anomaly detection).

3) EXCLUSION CRITERIA

- Absence of a dataset or ML application.
- No security relevance or not SDN specific.

This methodology supports a layered analysis of models, datasets, and architectural components, forming the foundation for our taxonomy-driven review in the next section.

II. BACKGROUND: SDN COMPONENTS

SDN separates the control and data planes, enabling centralized management of network traffic and flexible deployment of security mechanisms [4]. This separation is foundational to integrating ML models for tasks like anomaly detection and IDS. Below, the key components of SDN and their role in enabling ML-based security solutions are explored.

A. VIRTUALIZATION TECHNOLOGIES

Virtualization is central to SDNs' adaptability, allowing the deployment of ML models and frameworks such as IDS on shared infrastructure.

Network Virtualization is a key milestone in SDN development [4], offering expanded functionalities and utilities which result in further development of networking.

Network virtualization abstracts physical hardware, enabling synthetic datasets to simulate diverse traffic scenarios [16], [17], [18], [19], [20].

Network Functions Virtualization (NFV) enables a flexible deployment of network functions across the infrastructure [21]. This enhances the flexibility, scalability, and cost-effectiveness, making NFV an essential tool for managing modern networks. When combined with SDN, these capabilities allow for the seamless deployment of ML models to strengthen network security.

Mininet [22] is a containerization technology that facilitates communication between containers through virtual tunnels within a virtual network [23]. These virtualized environments allow for the ML models and computational power to bridge SDN's control and data planes to optimize network performance.

By combining efficiency and flexibility, containerization enhances SDN's ability to deploy and manage sophisticated security solutions.

B. CONTROL PLANE

Before the advent of the Control Plane [4], network elements like in-band signaling and forwarding systems managed both data and control processes, leading to inefficiencies [18], [19], [20].

The introduction of the Network Control Point (NCP) helped separate data and control operations, improving network management [24]. The Control Plane began handling logic and management tasks, while the Data Plane focused on forwarding packets, making large networks [25] easier to manage and more scalable.

However, deploying multiple Control Planes can create scalability and routing challenges. OpenFlow protocol moves routing intelligence to software, improving network performance and efficiency [26].

C. DATA PLANE

The Data Plane's functionality can be summarized in three main tasks: receiving packets, processing them, and taking action. Actions include forwarding, mapping, deep packet inspection, or dropping packets [4].

On the hardware level, advancements are improving Data Planes by focusing on increasing packet and processing speeds [27]. Hardware Data Planes are fast but less flexible compared to software Data Planes, which offer greater flexibility [28], [29].

At the software level, the Data Plane operates on a protocol-independent layer, where data is processed before being configured by a programming language. Specialized languages, like P4 (Programming Protocol-Independent Packet Processors), are designed for SDN switching and play a crucial role in enhancing Data Plane functionality in SDN [4], [30].

P4 allows flexible control of the Data Plane compared to the limited control over the hardware-based Data Plane. It is possible to alter the stage number on the packet processing pipeline in the Data Plane for functionality [30]. Data Plane should have at least some programmability to quickly deploy protocols [4]. Another technology, NetASM, demonstrates intermediate representation between the programmable Data Plane and the lower part of the networks, so that the devices can communicate in a more efficient environment. The intermediate representation NetASM can demonstrate target-specific configurations with front-end and back-end demonstration capabilities [31].

D. EBPF AND BPF

Berkeley Package Filtering (BPF), extended Berkeley Package Filtering (eBPF), and SDN are completely different technologies created by unrelated study groups but serving in very close domains, which have the potential to be complementary for information security.

BPF has been popularized as a programmable network before SDN was generalized. BPF provides observation utility with a packet-filtering mechanism as a Unix tool [32]

eBPF is developed as an extension of classic BPF for kernel subsystem observability of networking and security. BPF extension as programmable utilities for network and system level enforcements [32], [33].

eBPF can implement and enforce rules on the system and network, whereas SDN observes the system at a higher level to manage network decisions on the controller [34]. SDN emerged to decouple the communication control and flow; eBPF was later developed as a kernel-native way to program and interact without custom kernel modules.

SDN's foundational components, virtualization technologies, the Control Plane, and the Data Plane offer high adaptability and efficiency. Understanding these components establishes a strong foundation for analyzing the role of ML in SDN security, which we explore in the next section. The next section will dig into frameworks and datasets, illustrating how they leverage these components to address security challenges.

To evaluate the state of the literature on threat detection, statistics of various frameworks are digested with observations in the Section III we have used. To summarize the performance measures reported in the reviewed studies, we use the standard classification metrics: accuracy, precision, recall, and F1-score. Let *TP*, *TN*, *FP*, and *FN* denote the number of true positives, true negatives, false positives, and false negatives, respectively. The metrics are defined as follows:

$$\begin{cases} \text{Acc.} = \dfrac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}}, \\ \text{Prec.} = \dfrac{\text{TP}}{\text{TP}+\text{FP}}, \\ \text{Rec.} = \dfrac{\text{TP}}{\text{TP}+\text{FN}}, \\ \text{F1} = \dfrac{2\text{TP}}{2\text{TP}+\text{FP}+\text{FN}}, \qquad \text{where TP, TN, FP, FN} \in \mathbb{N}_0. \end{cases} \tag{1}$$

Accuracy can be misleading under class imbalance, which is common in attack detection settings; therefore, precision, recall, and F1-score are frequently reported to better reflect false-alarm and miss-detection behavior.

III. REVIEW: SDN SECURITY WITH ML

This section summarizes prior work and the literature, highlighting advancements in SDN security with ML, which is increasingly adopted as an alternative to traditional networks, motivating extensive investigation of ML-based defense mechanisms within SDN environments. ML models have become increasingly vital with high success rates over attack detections, addressing security risks by enabling dynamic and adaptive defenses rather than traditional static rule sets with white listing and black listing [35].

SDN has various applications such as SD-IoT (Software Defined Internet of Things), SD-IIoT (Software Defined Industrial IoT), SDV (Software Defined Vehicles), SD-WAN, SDN-enabled 5G, and SD-LEO (Software Defined Low Earth Orbit Satellite Networks), as mentioned on I, are small to large, addressing computational and systematic needs of the host infrastructures, in other words, IBN Intent-based Networks. The architecture of SDN changes, but the base capabilities remain similar across various SDN [3], [36].

While applications of SDN vary, the foundational security needs and security frameworks are not clearly differentiated [36]. Each SDN application has the application plane, the control plane, and the data plane.

A. SECURITY OF SDN

Each component of the SDN itself has distinct security risks as they inherently mimic various parts of an information system, application plane mimics the applications on the host controller, control plane mimics the controller, and data plane mimics switches. The risk of each plane is similar to the system that the SDN component uses for the role.

As network elements are exposed to DDoS attacks, exhaustion on the communication surface is the main risk for the data planes, which can result in system unavailability [36].

As the controller is in charge of the most decisive operations within the network, the authorization and the authentication could be the single point of failure for the system where the used API's, such as the southbound northbound API, are vulnerable [36].

Applications on the host that support networking operations rely primarily on system memory; therefore, any malware execution may lead to exploitation at the application plane [36], [37].

Although SDN expands the attack surface across planes and interfaces, this SoK focuses on whether ML-based SDN defenses are evaluated in a valid and comparable way.

B. ML MODELS EFFECTIVENESS

Tables 6 and 7 summarize the primary taxonomy targets of this SoK by listing ML-based SDN defense studies together with their associated ML models, datasets, framework types, and reported success metrics. The successor ML model and its corresponding performance results are highlighted to emphasize validation patterns across studies.

ML models, including DL, have shown potential in enhancing the effectiveness of SDN security, particularly for DDoS detection and IDS. Detailed Table 6 proves that on various datasets, ML models can achieve success with an average of 99%.

Effective feature selection is a critical step, a catalyzer for improving the efficiency and accuracy of these models, with correct choices with the help of techniques like chi-square and feature importance scoring being employed [38]. However, achieving real-time detection with low computational overhead remains a challenge, especially for deployment in resource-constrained SDN environments like the data plane.

Over the observations on the reviewed literature, there are several ML models improving the success rates, such as Sequential models and tree-based models shown in the ML taxonomy Figure 2. %24 of studies used RF, following with 22% XGBoost a DT variant and 20% Sequential Models such as LSTM/GRU. We can observe that tree-based models are dominating over attack detection for SDN datasets. Sequential DL models follow tree-based models on attack detection for SDN.

1) OVERAGE & ALGORITHMS

ML models with distinct logic guessing over various patterns offer solutions to various network behaviors, such as source and destination guessing or time patterns. For instance, RF and SVM can detect timestamp-based patterns in time-dependent attacks. ML models address SDN security challenges at different architectural layers. Figure 2 shows their hierarchy; Tables 6 link models to specific use cases, and Tables 3 outline their operational logic for targeted deployment.

While various ML models like RF [10], [11], [39], [40], [41], [42], [43], [44], [45], [46], XGBoost [47], [48], [49], [50], [51], [52], [53], [54], and LSTM [11], [47], [55], [56], [57], [58], [59] have demonstrated success in detecting DDoS attacks, their effectiveness in broader, real-world applications are still being explored, requiring real-time optimization.

Over the studies reviewed in the Table 6 and 7, 34% of models have reached above 99% precision, where CNN dominated with 8%, RF with 12%. As RF has shown %24 of accuracy over the literature with a similar number of precision results with CNN, we can examine whether CNN proves proportionally better performance with FP metrics.

On the Fig. 2, we can examine ML models and the success of the sequential models, such as LSTM, and GRU, recorded on studies on Table 6 and 7 are excelling IDS metrics all over 98% on 18% of the studies. However, simply performing with high metrics might not be accepted as a valid outcome; it can also be explained as overfitting.

In parallel, high detection accuracy has also been reported for Deep Neural Networks (DNN) [60] in studies. On the **ML Models Table** 1, the algorithmic behaviour of each model provides insights for those investigating SDN security for matching datasets to succeed. It highlights successful algorithms used in various research environments. Efficiency and effectivity trade-off can also be an open discussion for ML models, as ML models' resource usage can be excessive while success rates can outperform the DL models [59].

2) ARCHITECTURE EFFECT

Hybrid and ensemble methods are also being investigated to further improve detection capabilities [57]. Furthermore, ensuring the adaptability of IDS to evolving threats and addressing scalability issues in large networks are ongoing research areas [51], [53], [61].

Techniques like Federated Learning (FL) [57], [62], [63], [64], [65] offer promising privacy-preserving approaches for collaborative intrusion detection in distributed SDN environments.

The overall effectiveness of ML in SDN and networking security depends on overcoming data and validation limitations and continuously refining detection techniques for dynamic and complex network environments.

The **ML Taxonomy** in Figure 2 provides ML parenting relationship with insights for researchers focusing on SDN and networking security, offering a structured view of how different ML models are utilized, helping those who wish to understand the relationship and application of various ML models in SDN security, making it easier to grasp complex connections and identify effective models for specific security challenges such as DDoS attacks, application attacks or MiTM attacks. As each attack is different, defending ML algorithms have the potential to address various types of attacks. For example, a DDoS attack could bind with a timestamp, and an application-layer attack could bind the packet sender to a different address than the DDoS attack, which could be considered for feature selection of models.

For instance, GANs and VAEs under "Generative Models" have been utilized in anomaly detection frameworks to

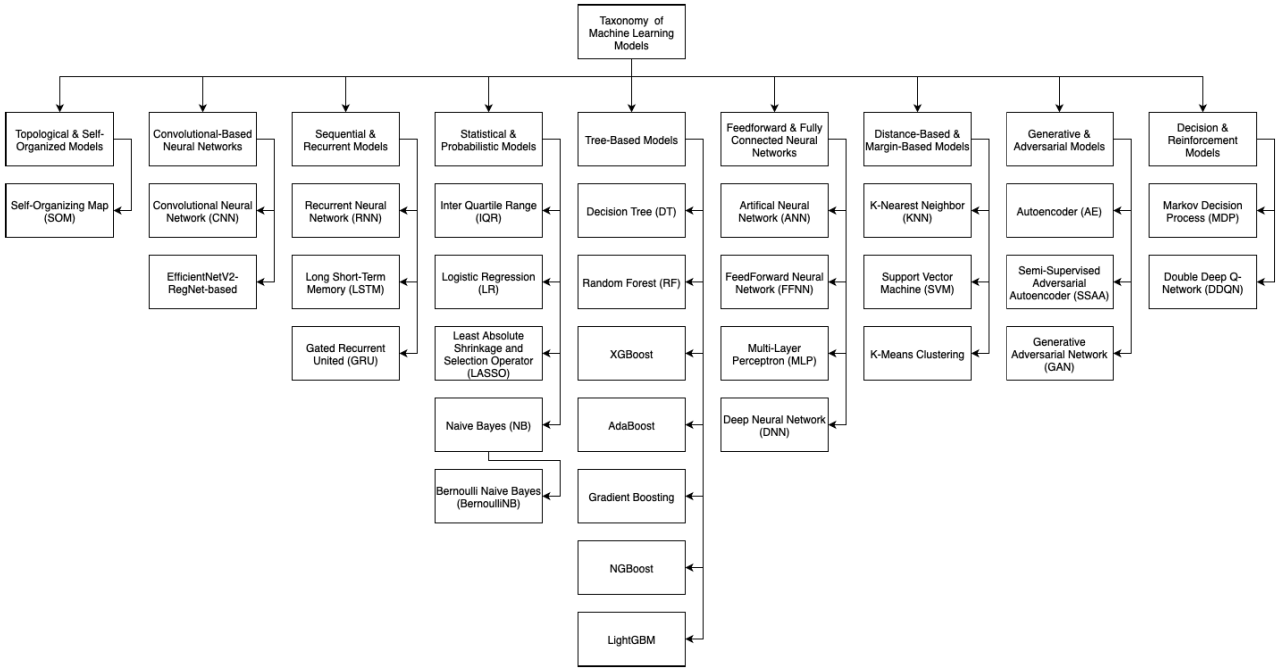

FIGURE 2. Taxonomy of ML models.

generate synthetic datasets that balance imbalanced data distributions. A notable example is the work of [66], which leveraged GANs to improve the class balance problem on datasets.

Similarly, RF, listed under "Tree-Based Models," has been applied in DDoS detection studies like [67], where it demonstrated high accuracy rates in identifying and mitigating DDoS attacks within SDN environments. This highlights the versatility and robustness of RF in handling structured datasets typical of SDN scenarios.

Hybrid approaches such as CNN-LSTM [68] combinations enable both sequential and convolutional pattern recognition, improving accuracy for DDoS attacks. These models increasingly merge DL with classical techniques like RF and SVM to enhance scalability and real-time detection.

While many individual ML models exist, as shown in Figure 2, integrated frameworks remain limited. Future work should develop unified, adaptable ML systems capable of handling diverse attack types and dynamic SDN environments through multi-model fusion. In order to develop an adaptable ML system, efficiency and security may need to be balanced for optimal trade-offs.

C. DATASETS

Benchmark datasets, which have been examined in Table 2, often lack the diversity of traffic needed for effective SDN security research. Many studies still use outdated sets like KDD'99 [98] and NSL-KDD [122], limiting model applicability to current threats.

Tables 2 highlight the need for comprehensive and attack-specific datasets, whereas Industrial, Healthcare, and IoT environments have various datasets as shown in Table 2.

Over the results on the literature shown in Table 6, we can point out the success dominance over CiC-based datasets, while for InSDN and NSL-KDD, the results are distributed. In studies with CiC-based datasets, the results are completely over 98%.

It could also be discussed about specified fields, such as protocol, so it can provide further details with detection quality and potentially enable researchers to keep protocol-based security metrics [116]. Synthetic dataset results are also open for discussion, as results may seem diverse; however, real-world scenarios will be the expected threshold.

Diverse Datasets must span varied industries and traffic types to support generalizable ML defenses. While datasets exist for industrial subdomains, broader environments such as Software Defined Unmanned Aerial Vehicles (SD-UAV), which are not found in the literature, are essential for training adaptable models.

Remaining limited, but they are vital for testing customized defenses. Network and application-specific datasets would support more precise mitigation strategies. Comparing and building upon older datasets can also improve future dataset design.

It has been presented on the **SDN datasets Table** 2, notably diverse datasets, each focused on different network environments. Researchers can select environment-specific datasets and the results shown on the Table 6 and 7 to ensure their security-focused studies are more applicable and effective. Key challenges include imbalanced data, lack of public availability, and limited reproducibility. Ethical standards must guide dataset generation and use to ensure reliability and fairness.

TABLE 1. ML models used in the reviewed SDN security studies and their brief operational descriptions.

ML Model	Explanation	ML Model	Explanation
Isolated Forest [69]	Tree-based algorithm, isolates anomalies by recursively partitioning the data, with outliers being easier to isolate.	Naive Bayes (NB) [70]	Probabilistic classifier based on Bayes' theorem, assuming independence between features, used for classification tasks with a simple and fast implementation.
Autoencoder (AE) [71]	Neural network for unsupervised learning, compresses data into a lower-dimensional representation, reconstructs it, learning to identify important features.	Self-Organizing Map (SOM) [72]	Unsupervised neural network that clusters data by mapping high-dimensional data into lower-dimensional grids while preserving the topological structure.
RF [8]	Ensemble of DTs, improves classification or regression accuracy by averaging multiple models to reduce overfitting.	Bernoulli Naive Bayes (BNB) [73]	Variant of NB for binary/boolean features, assuming each feature follows a Bernoulli distribution.
Support Vector Machine (SVM) [74]	Supervised learning algorithm, finds the optimal hyperplane to separate different classes by maximizing the margin between them.	Passive-Aggressive [75]	Linear model used for classification and regression that adjusts its parameters based on the data's change while being minimally aggressive in learning.
K-Nearest Neighbor (KNN) [76]	Non-parametric algorithm, classifies a data point based on the majority class of its nearest neighbors in the feature space.	Stochastic Gradient Descent (SGD) [77]	Optimization technique used in ML where the model's parameters are updated incrementally using a subset of data to minimize a loss function.
XGBoost [78]	Gradient boosting method, improves predictive performance by sequentially correcting the errors of weak learners (DT).	Logistic Regression (LR) [79]	Linear model used for binary classification, predicts the probability of outcome using the logistic function.
Artificial Neural Network (ANN) [80]	Computational model inspired by biological neural networks, consisting of layers of nodes (neurons), processes, and learn from input data.	Convolutional Neural Network (CNN) [81]	Type of DNN mainly used for image recognition, utilizing convolutional layers to detect spatial hierarchies of features.
DNN [82]	Type of neural network with multiple layers between input and output, enabling learning of complex representations and features from large datasets.	EfficientNetV2 [83]	State-of-the-art CNN architecture that combines EfficientNetV2 for efficient training.
AdaBoost [84]	Ensemble learning method that adjusts the weights of misclassified instances and trains subsequent models to focus on correcting those errors.	Generative Adversarial Network (GAN) [85]	Framework of two neural networks, a generator and a discriminator, compete to create and evaluate realistic data, such as images or text.
Multi-Layer Perceptron (MLP) [86]	Type of artificial neural network with one or more hidden layers, used for supervised learning tasks such as classification and regression.	LSTM [7]	Type of RNN designed to remember long-range dependencies and mitigate vanishing gradient problems in sequential data.
Feedforward Neural Network (FFNN) [87]	A simple type of artificial neural network where connections between nodes do not form cycles, often used for supervised learning tasks.	LightGBM [88]	Type of gradient boosting framework for tree-based models, optimized for speed and scalability in classification and regression tasks.
NGBoost [89]	Gradient boosting method that generates probabilistic forecasts by combining the strengths of traditional gradient boosting with a probabilistic model.	Markov Decision Process (MDP) [90]	Mathematical framework for modeling decision-making where outcomes are partly random and partly under the control of the decision maker.
Double Deep Q-Network (DDQN) [91]	Reinforcement learning algorithm that addresses the overestimation bias in Q-learning by using two neural networks to estimate the Q-values.	K-Means Clustering [92]	Unsupervised learning algorithm that partitions data into K clusters by minimizing the variance within each cluster.
DT [9]	Tree-like model of decisions, where each node represents a feature, and branches represent decisions or outcomes, used for both classification and regression.	Gradient Boosting [93]	Boosting technique that builds strong predictive models by sequentially training weak models, each correcting the errors of the previous one.
Extra Tree [94]	Type of DT ensemble method where trees are built using random splits at each node, aiming to reduce variance.	Variational Autoencoder (VAE) [95]	A generative model that encodes input data into a probabilistic latent space and decodes it back, enabling controlled generation and smooth interpolation of new data.

ML models listed are those reported in the reviewed SDN security frameworks.

D. EMERGING FRAMEWORKS

Our research revealed a wide variety of ML frameworks designed to directly detect attacks in SDN. However, we noticed a lack of emerging techniques as they have been examined in the Table 3, such as FL, network slicing, and malware detection, which are crucial for enhancing SDN security. As shown in Table 3, specifically, the number of up-to-date malware detection frameworks is smaller compared to the number of IDS or FIDS. Future studies should explore these advanced methods further. Incorporating techniques like slicing, tunneling, and BPF [33] could pave the way for more robust and specialized intrusion response frameworks, strengthening SDN defense mechanisms.

The Frameworks Used in Research **Table** 3 provides critical insights into the factors influencing the success rates and coverage of each study. For example, anomaly detection frameworks can identify a wider range of attack types compared to DDoS-specific detection models, though they may flag benign anomalies, leading to false positives. When compared with corresponding studies in Table 6, understanding these frameworks helps researchers assess their strengths, limitations, and suitability for various SDN scenarios.

E. CONTROLLERS & PROGRAMMABLE DATA PLANES

While SDN architecture provides significant benefits with central control for network management and innovation, it also introduces unique security challenges, which are addressed with multi-controller solutions [50]. Without explicitly mentioning the direct relationship between the controllers and the models' successes, controllers affect the efficiency of the models on the system differently [148]. Even if there are various efficiency metrics used and it is hard to standardize and compare directly by the studies with controller on the Table 5 and 6, the ONOS [149] controller has preferable results compared with POX [148], [150].

Programmable data planes remain underexplored in SDN security. Leveraging technologies like P4, they embed security logic directly into switches, enabling faster and more responsive detection (e.g., LANTERN framework [151],

TABLE 2. Benchmark datasets.

Dataset Name	Dataset Notes	Dataset Name	Dataset Notes
KYOTO2016 [96]	Real university network including attack flow, used for academic network traffic research.	DARPA1998 [97]	Synthetic network flow with multiple attack types, used for botnet research.
DARPA1999 [98]	Similar to DARPA1998 but includes different attacks and network configurations.	SD-IoT Dataset [99]	Specific dataset focused on IoT security within SDN environments.
BotIoT [100]–[105]	IoT dataset focusing on botnet and attack traffic, designed for IoT anomaly detection.	CIC-Ton-IoT [106]	Industrial IoT dataset with focus on communication patterns and cyber-attacks.
IoT-23 [107]	Industrial IoT dataset with focus on communication patterns and cyber-attacks.	MAWI [108]	Real-world traffic dataset, focusing on internet backbone and large-scale network analysis.
N-BaIoT [109]	Focuses on IoT-based networks for anomaly detection and traffic analysis.	ITU Challenge Dataset [110]	IoT dataset used for evaluating IDS solutions in smart cities and large-scale environments.
SWaT [111]	Water treatment system dataset for Industrial Control systems (ICS) security.	AWID [112]	Wireless intrusion detection dataset focused on WiFi networks.
MalImgDataset [113]	Malware image dataset for image-based anomaly detection and malware classification.	SDN-Based-Iot [114]	High-traffic IoT dataset.
CICIOT2023 [115]	IoT traffic dataset for intrusion/anomaly detection.	CICDDoS2019 [116]	DDoS-focused, realistic traffic; widely used for IDSs.
CICIDS2018 [117]	Labeled attack types with a realistic setup for anomaly detection.	CICDoS2017 [118]	Labeled attack types with a realistic setup for anomaly detection.
InSecLab-IDS2021 [119]	IDS-focused dataset for industrial environments.	CICIDS2017 [117]	Simulated enterprise network traffic dataset.
Ember [120]	Open dataset for training static PE malware models using ML.	Orion [121]	Improved successor to KDD 1999.
NSL-KDD [122]	Enhanced KDD 1999; suitable for anomaly detection.	InSDN [123]	Realistic SDN-security dataset.
Bennett Univ. [124]	Hybrid testbed (real + virtual) for flows and attacks.	Edge IIoT [125]	Industrial IoT simulation for industrial scenarios.
TON-IoT [126]	Realistic industrial IoT testbed.	X-IIoTID [127], [128]	Advanced IoT dataset tailored for IDSs.
TOTEM 2006 [129], [130]	Traditional enterprise traffic; common in security research.	UNSW-NB15 [131], [132]	Comprehensive multi-attack dataset for classification.
ISCX 2012 [133], [134]	Various malicious flows for attack detection.	CTU-13 Botnet [135]	Botnet-focused traffic for detection/analysis.
ICS-CSR 2013 [136]	ICS attacks on real-like industrial systems.	UNR-IDD [137]	IoT/industrial traffic including attack scenarios.
Hogzilla [138]	High-volume enterprise-like traffic.	CAIDA [139]	Diverse backbone traffic traces.
SDN-NF-TJ [140]	NFV-focused scenarios.	KDD Cup 1999 [141]	Classic intrusion-detection benchmark.
CERNET [142]	Real traffic from the Chinese education network.	DNP3 IDS [143]	Synthetic + captured DNP3 traffic for anomaly detection.
Slow-Rate DDoS [144]	Slow-rate DDoS attack traces.	SDN-IoT [145]	Generated for healthcare systems.
BODMAS [146]	Temporal analysis for PE malware behavior patterns.	Portable Executable (PE) Malware ML Dataset [147]	Practical security analytics for malware classification.

Table 6). As SDN demands greater control and speed, it can be discussed whether the current P4 studies have reached maturity or if future research should focus on harnessing programmable data planes for real-time, in-network ML-based defense.

These findings highlight key advancements for handling traffic on the data plane and challenges in SDN security, providing a roadmap for future innovations.

TABLE 3. Frameworks used in corresponding papers.

Framework Category	Corresponding Papers
Intrusion Detection	[43], [47]–[49], [54], [56], [58]–[60], [64], [65], [68], [152]–[157]
DDoS Attack Detection	[10], [11], [38], [39], [42], [45], [46], [50]–[52], [57], [158]–[162]
Intrusion Detection and Prevention	[12], [40], [41], [44], [53], [61], [163]–[166]
Anomaly Detection	[55]
Network Slicing with NFV	[167]
Malware Detection	[168]
Federated Intrusion Detection	[62], [63]
Botnet Detection	[169], [170]

To inspect and understand the efficiency of studies, within the controller Table 5, it is possible to check each literature and shed light on how different studies have optimized resource usage, such as CPU and RAM efficiency [61], [153], [163]. It makes it possible to inspect and cross-check notes with Table 6 that certain controllers are more cost-effective, and their performance can be consistently replicated in similar settings.

Covering the ML models employed, strategies adopted, and key result metrics, the Summary of Frameworks Tables 6 offer nuanced technical details about each study, such as accuracy, precision, recall, and F1-Score. These tables serve as a guide for understanding the methodologies and practical applications of ML models in SDN security, providing in-depth information for strategic research planning.

IV. DISCUSSIONS: STATE OF LITERATURE

In this section, clear outcomes of the reviewed data are discussed. Review abstractions and the synthesis of the literature in Section III made the path to discussions before addressing takeaways, bridging the way into the state of the literature.

The results of this research indicate that while significant progress has been made in securing SDN using ML models, several challenges remain.

A. GENERAL OBSERVATION

Dominant success results over %98, with attack detection results from the literature showing the studies are promising. However, there is also a potential for questioning the success rates from the several discussions. In this section, observations from the literature are questioned with the research questions.

1) RQ1

The efficiency of the attack detection frameworks varies, making it difficult to understand. However, under-standardized efficiency metrics with various overhead measurements keep the reviews over the literature and synthesis impossible and incomparable [11], [55], [59], [60], [158]. For a serious consideration of a framework, the resource usage should not be a burden for the main operations of the system.

2) RQ2

The effectiveness of models like SVM, RF, and LSTM in detecting DDoS attacks is dominating, but the limitations of current datasets and the need for real-time optimization are lacking. The success rate of the specified ML models is also open for discussion, where network fundamentals and several fields can be open for statistical exploitation, which leaves the success rates in question if they are valid [49].

3) RQ3

Synthetic and benchmark datasets in Table 2 are used for studies on the literature mapped on Table 6. Mixed results are observable in the Table 6 for synthetic datasets and several benchmark datasets such as InSDN. The results of CiC-based datasets can be observed with overwhelming success; accuracy is typically over %98.

4) RQ4

There are several synthetic dataset creation tools, including D-ITG, HTTProxy, and The CiCFlowmeter [44], [57], [60], [157], [161]. The CiCFlowmeter is used for benchmark datasets, too. Each has a varying number of fields for ML models to train on.

5) RQ5

Overall, the rate of Malware Detection frameworks is only %2, and the usage of P4 within the literature is %8. Literature shows that the studies are imbalanced.

B. LIMITATIONS

The following limitations are highlighted to demonstrate why the high success rates reported in current literature often struggle with real-world applicability and technical validity. By identifying specific constraints in simulation environments and dataset relevance, a necessary critical perspective for transitioning these theoretical frameworks into practical SDN environments is provided [39], [56], [171].

1) LIMITED VALIDATION POSSIBILITIES

Several proposed systems require further validation in real-world scenarios to assess their practicality and effectiveness [40], [56]. Simulation environments may not fully replicate the complexities of live networks [39], [49].

2) SCALABILITY CONCERNS

Scalability and performance issues in large-scale SDN implementations are potential issues requiring additional research [171], [172].

3) DATASET FIELDS

Even if several datasets have attack protocol mentioned [115], [118], existing datasets with specified protocol/layer details about datasets are lacking.

4) DATASET RELEVANCE

Many studies rely on datasets that may not fully represent current DDoS attack patterns [42], [55]. Outdated datasets like KDD99 may not capture the complexity and diversity of modern attacks [10], [11]. Synthetic datasets [39], [40], [41], [44] can result in invalid success rates where the training ML model may not represent a detection rationale.

5) LACK OF BENCHMARKING

Some studies lack benchmarking against existing public datasets, limiting the evaluation and validation of proposed methods [39], [40]. The absence of benchmarking limits meaningful comparisons and advancements in the field [49], [56]. Lack of benchmarking can also lead to ML imbalanced training and overfitting, where several ML models can prove unreliable success [68].

6) FRAMEWORK IMBALANCE

Attacks detection for system-based studies and Malware detection are less frequently studied. Further research in the field can promise new findings [171].

C. CHALLENGES

Fundamental challenges to underscore the significant computational and operational hurdles that hinder the seamless integration of ML-based IDS with existing network infrastructures are identified. Addressing these issues, particularly regarding resource constraints and evolving attack patterns, is vital for developing a defense system that remains resilient without becoming a performance burden [11], [173].

1) INTEGRATION WITH EXISTING SECURITY MECHANISMS

Integrating ML-based IDSs with existing security mechanisms, such as firewalls and traditional IDS's, can enhance network security but presents integration challenges due to efficiency problems [173].

2) EVOLVING ATTACK STRATEGIES

Continuous refinement and adaptation to evolving attack specific strategies are necessary [63].

3) COMPUTATIONAL RESOURCES

ML-based methods often require significant computational resources, potentially impacting real-time performance [11], [41], [49], [173], [174].

4) DATASET GAPS

Current benchmark datasets shown in Table 2 focus on conventional attack detection, leaving Federated IDS and other advanced approaches with limited validation resources.

5) OVERFITTING & ML MODELS VALIDITY

Several ML models on the "Distance-Based & Margin-Based Models" or "Sequential & Recurrent Models" shown on ML Taxonomy Figure 2 are excelling on most of the studies on Table 6. However, this success can also be explained by the exploitation of the quantitative logic of the models over imbalanced datasets, which can result in invalidated studies.

6) FALSE POSITIVES

Anomaly-based detection methods can suffer from high false alarm rates [44], [46], [55], [155].

7) DATA PRIVACY

Ensuring data privacy while implementing collaborative intrusion detection approaches like FL is a key challenge [48], [62], [63], [64].

D. FUTURE OUTLOOK

Emerging approaches like network slicing, blockchain, and access control [175] have shown promise, but further work is needed to ensure their validity in live environments. The development of hybrid ML models, along with more comprehensive and balanced datasets and validated ML strategies, is crucial for advancing the current situation of SDN ML security frameworks. Future research should focus on these areas to create more efficient, scalable, and responsive security solutions.

Having analyzed the results of current SDN security research in Section III, following the taxonomy, it is observed that while progress has been made, significant challenges and opportunities remain with datasets and ML models for successful validation of the security frameworks. Section V digs into these perspectives, offering actionable insights and proposing strategies to advance the field further, alongside the comparisons to recent quality-related works.

V. RESULTS AND RELATED WORK

This section discusses the key points of review for future work in SDN security, with a particular focus on ML applications. It highlights frameworks, effective ML models, and emerging approaches aimed at enhancing security in SDN environments. The discussion identifies gaps in current literature, emphasizing the need for an optimum efficiency/effectivity trade-off for ML models and comprehensive datasets to address evolving security challenges.

After careful examination of the literature and discussion of the main results in Section IV, in this section, takeaways are considered for security frameworks.

A. TAKE AWAYS

Core takeaways to synthesize the taxonomy-driven insights gathered from our analysis into a cohesive roadmap for future security implementations are presented. These points serve to bridge the gap between fragmented research findings and the development of unified, hybrid frameworks that balance detection accuracy with scalable efficiency [48], [64].

1) FL POTENTIAL

FL offers a collaborative training approach without direct data sharing [63], addressing privacy concerns [64] and data unavailability in traditional, centralized IDS systems [48], [64]. FL can also be a potential approach against inefficiency with a distributed structure and resource allocation. FL lets multiple networks train a shared intrusion detection model without sending their raw traffic logs to a central place. Each network trains locally on its own traffic features and sends only model updates to be combined into a global model. This supports privacy and data-sharing limits, but training can be harder because traffic patterns differ across networks and because a malicious participant could try to corrupt the training updates [64].

2) STANDARDIZATION OF EFFICIENCY METRICS

While various efficiency metrics are reported in the literature, standardized System-efficiency reporting remains an unresolved challenge, hindering comparability across ML-based SDN security studies. Many works either evaluate different definitions of efficiency or omit efficiency analysis altogether. In addition to the controllers that directly affect efficiency in Table 5, we propose standardized metrics such as *overhead*, *processing time*, and *memory usage*, which quantify how much the defense mechanism burdens the SDN infrastructure and how swiftly it can respond. Without standardized efficiency metrics, high-accuracy models may remain theoretical exercises rather than validated security solutions suitable for real-world deployment. Therefore, we propose future SDN security studies to report at least the following:

- **Controller Computational Overhead:** The CPU and memory load imposed by the ML inference engine on the SDN controller [153], [163], [172], [173].
- **Inference Latency:** The time difference between traffic arrival, such as packet/flow observation and detection decision/policy execution in milliseconds [57], [62].
- **Throughput Impact:** Reduction in network throughput or flow handling performance under varying traffic loads when the security module is active [11], [148].
- **Control-plane Reaction Overhead:** The number and frequency of rule installations/updates triggered by mitigation actions, reflecting controller workload and scalability constraints [61], [163].

TABLE 4. Comparison with related work over taxonomy.

Paper	Year	ML Coverage	Controllers	Examined Frameworks	Datasets	Taxonomy
SoK: SDN Security with ML	2025	Fundamentals & Detailed	9 Controllers	IDS & IDPS & Malware Detection	40+ Datasets	Defense Frameworks Taxonomy
Resilience in Internet of Medical Things (IOMT) [177]	2024	Architectural perspective	Architectural perspective	IOMT Resilience Focus frameworks	✗	Microservice Architecture
DDoS Detection review [178]	2025	DL/ML segmentation & in literature	5 controllers	DDoS detection & mitigation	7 datasets	DDoS attack taxonomy
DDoS Detection & Mitigation Survey [37]	2024	DL/FL segmentation & in literature	Touches within the literature	DDoS detection & mitigation.	8 benchmark datasets	DDoS attack taxonomy
Distributed Firewalls for Mobile Cloud [173]	2025	✗	Architectural perspective	NFV Solutions for Mobile Cloud Firewalls	✗	✗
Controller Vulnerabilities [148]	2025	Mentioned briefly	In depth analysis	Controller focused aspects	7 datasets	Detection methods Taxonomy
Vehicular Ad Hoc Networks Review (VANET) Review [172]	2025	VANET Emerging Trends in AI including Security	Touches within the literature	VANET AI	✗	✗
East-West Interface Security [179]	2025	Touches within the literature	✗	ML/AI, Hybrid, Cryptology, Auditing for East-West Interface	Touches within the literature	Vulnerability Taxonomy
Wireless Sensor Networks (WSN) Review [174]	2025	Touches within the literature	Touches within the literature	WSN Resource management Frameworks including Security	✗	Organization Chart/Taxonomy
Survey on Data Plane Security [36]	2025	Touches within the literature	Only in general	General defense frameworks of Data Planes	Only NSL-KDD	No Taxonomy
Malware Analysis [171]	2025	Touches within the literature	4 controllers	Malware detection frameworks	Touches within the literature	Frameworks Taxonomy

a: ML DOMINANCE

ML models are crucial for enhancing the capabilities of IDSs. Yet it is an open discussion with the fundamentals of ML models and success rates on the SDN datasets to correlate with each other for a validated framework scheme or a potential layered defense system.

b: IMPORTANCE OF FEATURE SELECTION

Selecting significant features within datasets is crucial for the effectiveness, efficiency, and also in various datasets, validity of ML models [62] used for DDoS detection and mitigation [57]. However, selected features are addressing specified network fields, which can be selected with an aim rather than automated algorithms against potential overfitting issues or senseless addressing.

c: DDOS ATTACK FOCUS

A primary application of ML in SDN security is the detection and mitigation of DDoS attacks, as shown in Table 3. Various ML models, including SVM [44], [48], [57], RF, and DL models, have demonstrated effectiveness in classifying network traffic and identifying malicious patterns [10], [45]. After considering the efficiency, the mentioned models can be applied for a layered defense against specified threats.

d: SDN CONTROLLER CAPABILITIES

Leading open-source controllers like ONOS and OpenDaylight (ODL) address critical network requirements; where ONOS stands out with its scalability for large-scale deployments, ODL excels in high-precision data plane control, enabling flexible and reliable traffic management, as cited in the Table 5, [149], [176].

e: P4-PROGRAMMABLE SWITCHES

offer significant advantages in network security by allowing full control over packet processing actions via software. This enables customizing forwarding logic and data plane behavior to suit distributed processing [11], [42], [43], [61]. Leveraging P4 allows the ability to customize forwarding logic and data plane behavior to suit distributed processing, potentially boosting efficiency and reducing the burden on the SDN controller [42], [61].

f: SDN FLEXIBILITY AND SCALABILITY

SDN's programmable centralized controller gives network administration employees more authority, allowing for more seamless supervision [171], [172]. The network's scalability, flexibility, and programmability support its widespread usage to create a dynamic environment [171], [172].

B. COMPARISON WITH EXISTING RESEARCH

Table 4 compares existing surveys and review papers on ML-Based SDN defenses, highlighting differences in scope, taxonomy structure, and coverage of datasets, models, controllers, and frameworks. This comparison positions the present SoK relative to prior review efforts. Some focus on a single security problem, such as DDoS [37], malware [171], or data-plane vulnerabilities [36], while others take a broader approach and look at multiple parts of the SDN architecture and several types of attacks [178]. The way these works are organized also differs. Some provide taxonomies that group threats, defenses, and testing setups [36], [178], while others evaluate algorithms or focus on defending a specific protocol rather than presenting a general structure [148].

ML is given different levels of attention. In some works, ML is the main topic, and there are direct comparisons of models, datasets, and performance results [178]. In others, ML is mentioned as part of the solution, not as the main focus [148]. The treatment of controllers also changes between studies. Some works show which controllers are used most often and how they relate to certain attacks or defenses [148], while many emphasize traffic features [177], programmable data planes [36], or dataset collection [179]

The handling of datasets is another point where these works differ. The most thorough studies link datasets to specific security problems and ML models, which makes it easier to repeat results and compare them [179]. Others prefer to list datasets with limited explanation of how they are used [148], and some do not explicitly consider controller datasets [177]. Overall, the works that combine a clear taxonomy, evaluation of ML models, mapping of controller use, and a direct link between datasets and tasks provide the most complete and reusable insights for SDN security research.

TABLE 5. Controllers used in research papers.

Controller Name	Corresponding Papers
RYU	[38], [40]–[42], [44], [46], [53], [55]–[57], [61], [163]
ODL	[61], [153]
SDN-Wise	✗
POX	[61], [156], [161], [170]
ONOS	[42], [61], [63], [165]
Floodlight	[61], [164]
SD-WAN	✗
ns-3 Library	✗
TensorFlow SDN	✗

VI. FUTURE PERSPECTIVES

SDN security studies in the literature could address the identified gaps through the seven future perspectives and the outlined framework.

- **Enhanced Anomaly Detection through Explainable AI (XAI):** While current models detect anomalies effectively, they often lack transparency. Integrating Explainable AI into SDN anomaly detection can prove *why* certain patterns are flagged, increase operator trust, and provide deeper threat insights.
  Could XAI-driven IDS with specified ML techniques over rationally selected Dataset fields on frameworks uncover not just anomalies but also their contextual causes in SDN traffic?
  Could reinforcement learning allow SDN controllers to autonomously reconfigure policies in real time to combat novel attack vectors?
- **Development of Adaptive Datasets via GANs:** GANs can generate evolving, realistic datasets that mimic modern threats, training models to generalize better across unseen attacks. This approach helps counter dataset stagnation in SDN research.
  Can GAN-based datasets drive continuous model evolution and study validity against zero-day attacks in SDN?
- **Advancing Malware Detection via Anomaly-based Methods:** Anomaly-based detection approaches can identify zero-day malware attacks by recognizing patterns that deviate from normal network behavior. This strategy complements traditional signature-based methods and enhances overall SDN security. **Can anomaly-based detection algorithms be integrated with ML models to create a more robust and adaptive malware detection system in SDNs?**
- **Exploring Emerging Technologies like eBPF:** With the capability to run sandboxed code in the kernel, eBPF [33] provides high-speed, programmable packet filtering and telemetry at the data plane level. This may optimize SDN performance through non-delayed hardware actions and observability.
  Could eBPF become a core mechanism for lightweight, real-time packet analysis and policy enforcement in future SDN systems?
- **Security Zoning via Network Virtualization and Slicing:** Future SDN architectures may adopt network virtualization and slicing to establish isolated security zones tailored for each application. Using containerization and NFV technologies, lightweight and specialized ML models can be deployed within each slice. This isolation prevents threat propagation across zones.
  Could containerized ML-based intrusion detectors deployed per network slice replace monolithic systems with a more resilient and distributed SDN defense?
- **Dynamic Security Policies via Reinforcement Learning:** Unlike static rules, reinforcement learning can continuously adapt to threats by learning optimal security policies through environmental feedback. This enables responsive and intelligent SDN defense.

- **Integration of Blockchain for Secure Logging and Auditing:** Blockchain offers a tamper-evident record of security logs and configuration changes, strengthening

TABLE 6. Summary of ML approaches in SDN security frameworks with success metrics.

Paper	ML Algorithms	Notes	Acc., Rec.	Prec., F1	Paper	ML Algorithms	Notes	Acc., Rec.	Prec., F1
[39]	KNN, DT, RF, ANN, DNN, **CNN**	IDS; Synthetic dataset	99%	✗	[63]	GAN, Autoencoder, **SSA**	IDS; SIEM architecture; **Datasets**:CIC-ToN-IoT, CIC-IDS2018, NF-UNSW-NB15, InSDN, InSecLab-IDS2018, DNP3-CICFlowMeter, DNP3-CustomParser	98.67%	✗
			✗	99.26%				✗	99%
[40]	NB, KNN, RF, **CNN**	IDPS; MitM detection; Blocking port clearing ARP cache as defense; Syntetic dataset	99.96%	✗	[55]	LSTM, CNN, **NetSeqDL**	Anomaly Detection; Zero-trust framework for communication, monitoring; **Datasets:**Bennett University dataset	99.65%	✗
			✗	✗				✗	99.35%
[47]	Bi-GRU-CNN, Bi-GRU-LSTM, **Bi-GRU-LSTM-CNN**	IDS for Industrial systems; **Datasets:**NSL-KDD, CIC-IDS-2018, and N-BaIoT	99.87%	✗	[41]	LR, SVM, DT, RF, GBM, **KNN**, XGBoost, AdaBoost, NB	IDPS, SOM for feature extraction; Performance metrics; Synthetic dataset.	98.8%	98%
			✗	98%				✗	✗
[42]	RF, **NB**, LR, KNN	DDOS Attack Detection; Details about network protocols, P4 ; **Dataset:**CICDDoS2019	98.04%	98.15%	[56]	**Deep Bi-LSTM**	IDS; Feature Selection; **Dataset:**InSDN	98.77%	97.66%
			99.82%	98.98%				97.32%	97.48%
[48]	KNN, SVM, CNN, CTGAN, **CoAtNet XGBoost**	IDS; Feature Selection; Minimal overhead; **Dataset:**InSDN,BoT-IoT,IoT-23	99.64%	99.60%	[64]	**OCC-VAE**	IDS; FL; **Dataset:**InSDN	99.37%	✗
			99.58%	99.59%				99.99%	✗
[49]	RF, DT, KNN, **XGBoost**, CNN, GRU, LSTM	IDS; Resource consumption metrics; **Datasets:**Bennett, IEC 60870-5-104	✗	✗	[43]	**RF**	IDS; P4; Syntetic dataset	97.33%	✗
			99.97%	99.97%				✗	✗
[57]	CNN, BiGRU, **LSTM-SVM**	DDoS attack detection with 16 features; FL ; Syntetic dataset	99.75%	99.8%	[44]	KNN, NB, LR, RF, **SVM**	Anomaly Detection; Syntetic dataset	99.43%	99.41%
			99.43%	96.2%				98.2%	97.2%
[10]	NB, **RF**, KNN, SVM, LDA	DDOS Attack detection; **Datasets:**aikenkazin-kaggle	99.99%	99.99%	[58]	**Bi-GRU-LSTM**	IDS; **Dataset:**E-IioT, ToN-IoT	99.15%	99.31%
			99.99%	99.99%				98.97%	99.14%
[11]	**RF**, KNN, DT, LSTM, CNN, GRU, MLP	DDoS attack detection; P4; **Datasets:**CICIoT2023	98.28%	✗	[60]	**DNN**	Ecrypted IDS; **Datasets:**InSDN	87%	79%
			97.57%	98.76%				81%	79%
[152]	**QTS-SGRU**	IDS with blockchain; **Datasets:**NSL-KDD	98.04%	98.80%	[158]	**CNN-Bi-GRU-AM**	DDOS attack detection; **Datasets:**aikenkazin-kaggle	99.4%	99.4%
			98.56%	98.68%				99.0%	99.4%
[59]	CNN, LSTM, KNN, DT, **DNN**	IDS;Performance analysis; **Datasets:**NSL-KDD	99.85%	99.85%	[65]	**DNN**	IDS; FL; **Datasets:**NSL-KDD	96%	92%
			✗	94%				✗	✗
[45]	KNN, **RF**, SVM, GMM	DDOS attack detection; Syntetic dataset	100%	100%	[159]	**DNN-LSTM**	DDOS attack detection; **Datasets:**CICDDOS2019, CICIDS2017, CICIDS2018, CTU-BOTNET, DARPA98, UNSW-NB15	98.84%	98.54%
			100%	100%				98.65%	98.33%
[46]	LR, SVM, DT, **RF**, KNN	DDOS attack detection; **Datasets:**Bennett	99%	✗	[163]	GNB, LR, **AdaBoost Classifier**	DDOS attack detection and mitigation; Blocks attacker ports to mitigate; Syntetic dataset(D-ITg, HTTPref)	95.29%	95.41%
			✗	✗				95.29%	95.31%
[168]	LGBM, **XGBoost**, DNN, CNN	Malware detection; Feature extraction, grey scale image 64x64; **Datasets:**EMBER, PEMachineLearning, BODMAS	98%	✗	[169]	LDA, GNB, LR, NN, DT, **RF**, LSVC	Botnet detection; **Datasets:**Neris, Rbot, Virut, QakBot, Trickbot	96.83%	93.28%
			✗	98%				98.2%	95.66%

accountability and forensic readiness across SDN infrastructures.

Might blockchain-backed auditing become the backbone of verifiable and immutable SDN security monitoring? Quantum-Resistant Security Mechanisms: Quantum advancements may render traditional cryptography obsolete. Forward-looking SDN designs must proactively integrate quantum-safe algorithms to future-proof communications.

TABLE 7. Summary of ML approaches in SDN security studies with metrics.

Paper	ML Algorithms	Notes	Acc., Rec.	Prec., F1	Paper	ML Algorithms	Notes	Acc., Rec.	Prec., F1
[160]	**GRU**	IDS; Data balancing wtih GAN; **Datasets:**InSDN, EDGE-IIoT, BoT-IoT	97.9%	98.1%	[161]	**RF**, DT, KNN, SVM, NB, LR	DDOS attack detection, IDS; **Datasets:**CICDDoS2019	99%	99%
			98.5%	98.2%				99%	99%
[50]	**XGBoost**	IDS; Multi-controller; **Datasets:**Bennett	98.5%	97.0%	[51]	XGBoost, **RF**, DT, DNN	IDPS; Feature extraction; **Datasets:**InSDN	100%	100%
			97.0%	✗				100%	100%
[52]	RF, **XGBoost**, LGB	IDS; Execution time metrics **Datasets:**InSDN	99.96%	99%	[162]	**Osprey Optimized Versatile RF (IOO-VRF)**	IDS; **Datasets:**IW-IB-5GNET	98%	97.4%
			95%	96%				94%	93%
[38]	LR, SGB, **DT**, NB, MLP	IDPS; Recursive Feature Elimination (RFE), Chi-Square , RF; **Datasets:**InSDN, NITSDN	99.18%	98%	[164]	**RF**,DT, SVM, KNN, LSTM	DDOS attack detection and mitigation without interrupting traffic; Syntetic dataset	99.71%	99.62%
			100%	99%				99.87%	99.74%
[170]	**SSRNN**	Botnet detection; **Datasets:**N-BaIoT	96.6%	95.71%	[61]	RF, KNN, DT, SVM, GNB, BLR (Binary LR), MLP, **CO-STOP**	IDS; Multi-controller and P4; Efficiency metrics; **Datasets:**CICIoT2023	99.25%	98.41%
			95.6%	95.58%				✗	98.83%
[153]	SVM, KNN, **SVM-KNN**	IDS; Feature extraction with min-max scaling and z-score; **Datasets:**NSL-KDD	95%	93%	[53]	K-NN, RF , **XGBoost**, FFNN, MLP	DDOS attack detection; Bandwidth metrics; **Datasets:**CICIDS2017, Edge-IIoTset	99.9%	100%
			91%	92%				99.99%	99.99%
[12]	**CNN-GNN**	DDOS attack detection; Syntetic dataset	98%	✗	[165]	**RF**, LSTM, SVM, CNN, MLP	IDPS, Moving Target Defense; Syntetic dataset	99.97%	✗
			✗	90%				✗	✗
[166]	**ANN**	DDOS Attack detection; **Datasets:**KDD Cup 99	98.89%	✗	[154]	**LR**, KNN	DDOS attack detection; Syntetic dataset	99.96%	99.75%
			✗	✗				99.85%	99.80%
[62]	**DNN-LSTM**	IDS; FL; Syntetic	98%	✗	[54]	**XGBoost**, DT	IDS; **Datasets:**MOORE-SET, ISCX VPN-nonVPN	98%	✗
			✗	✗				✗	✗
[68]	**CNN-LSTM**	IDS; **Datasets:**InSDN	99.19%	99.03%	[155]	**KNN**	IDS; houndreds of datasets	✗	95%
			99.05%	99.02%				100%	94%
[156]	**KNN**	IDS; Dataset benchmarking **Datasets:**NSL-KDD, CICIDS2017, CIC-DDoS2019	99%	99%	[157]	**LightGBM**	IDS; Syntetic dataset	99.02%	99%
			✗	99%				99%	99%

Should SDN frameworks begin adopting quantum-resistant ciphers to preempt vulnerabilities in the post-quantum era?

A. A HYBRID MODEL

As answers for validity, efficiency, and security problems on future SDN deployments, examining the current results from the literature, it is possible to envision a hybrid IDPS architecture that balances cost efficiency with detection accuracy and technical validity. The framework leverages DL for continuous passive monitoring, providing low-latency, high-accuracy baseline detection without the operational overhead of paradigms such as Moving Target Defense or Zero-Trust enforcement.

Under normal network conditions, a **DL-based passive detection layer** continuously analyzes aggregated traffic flows. When anomalous behavior or threshold-triggered alarms occur, such as during suspected DDoS campaigns, Man-in-the-Middle (MitM) attempts, or web application exploitation, an **active mitigation layer** is engaged. An outlined framework, addressing the articulated materials by the taxonomy details provided in the following:

- **InSDN** could be applied as *attack-type-specific*, leveraging realistic dataset for instance-based learning and robust evaluation.
- **GANs** can potentially balance the dataset by generating realistic samples that are representative of the actual data distribution.
- **Multi-controller architecture** with ONOS, or ODL, for cost-effective and scalable deployment could be framed.
- **Sequential Models (e.g., LSTM)** for DDoS attack detection, exploiting the temporal correlation inherent in volumetric and protocol-level flooding patterns.
- **CNN** for web application attack labeled fields detection, learning from repetitive functional patterns in front-end/back-end interactions to uncover obfuscated enumeration or injection attempts.
- **P4** could be used for early detection and response to DDoS attacks on the Data Plane level, using tree-based ML models without an efficiency cost.

- **eBPF** for cost-effective prevention/mitigation, enabling robust field-level matching to react against malicious behavior on the action in.

For feature extraction and selection, a neutral approach is suggested, taking into account the risk of overfitting. In this context, the outlined framework will potentially use the most varied resulting datasets, using GAN for specific attack scenarios, the best effective/efficient trade-off detection with sequential models, a layered defense addressing from physical to application, a cost-effective mitigation strategy for DDoS with P4, and a general solution with eBPF. Architecture could potentially achieve scalable threat coverage without sacrificing real-time responsiveness to address the gap in the literature. This design aligns with the resource-sensitive nature of large-scale SDN environments, positioning it as an adaptable model for next-generation programmable networks.

VII. CONCLUSION

This SoK addresses critical validity gaps in SDN security literature by synthesizing evidence across five research questions. The analysis confirms that: (1) real-world applicability requires fine-tuning ML efficiency/effectiveness trade-offs via field-matching within layered defenses; (2) dataset validation necessitates strategies for field alignment and class balance using datasets like InSDN augmented by GANs; (3) ML success validation requires explicit matching of model families to dataset fields within computational constraints; (4) research balance demands multi-controller frameworks (e.g., ODL/ONOS) to address scalability and heterogeneity; and (5) optimal SDN security frameworks must integrate hardware-aware mitigation (P4/eBPF) with field-specific model selection. Crucially, this synthesis identifies a conceptual framework for future implementation where dataset fields are dynamically aligned with ML models within a layered defense system, computational efficiency is monitored, and models are selected against burden, as a validated pathway to bridge literature gaps. This approach directly addresses the absence of production-ready validation mechanisms for real-world SDN security, providing an open-ended direction for future work without committing to implementation.

ACKNOWLEDGMENT

The authors sincerely thank Prof. Öznur Özkasap for invaluable guidance and thoughtful advice. They also acknowledge the use of AI tools, such as ChatGPT-4o, Gemini 2, and DeepSeek, in refining the expression of ideas in the Future Perspectives section.

REFERENCES

[1] N. McKeown, "Software-defined networking," *INFOCOM Keynote Talk*, vol. 17, no. 2, pp. 30–32, 2009.

[2] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008.

[3] S. Forge and K. Vu, "Forming a 5G strategy for developing countries: A note for policy makers," *Telecommun. Policy*, vol. 44, no. 7, Aug. 2020, Art. no. 101975.

[4] N. Feamster, J. Rexford, and E. Zegura, "The road to SDN: An intellectual history of programmable networks," *Queue*, vol. 11, no. 12, pp. 20–40, Dec. 2013.

[5] A. Vahdat, D. Clark, and J. Rexford, "A purpose-built global network: Google's move to SDN," *Commun. ACM*, vol. 59, pp. 46–59, Mar. 2016.

[6] J. D. Owens, M. Houston, D. Luebke, S. Green, J. E. Stone, and J. C. Phillips, "GPU computing," *IEEE*, vol. 96, no. 5, pp. 879–899, May 2008.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[8] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[9] J. R. Quinlan, "Induction of decision trees," in *Proc. 1st Nat. Conf. Artif. Intell. (AAAI)*, vol. 1, 1986, pp. 81–106.

[10] M. S. Sawah, H. Elmannai, A. A. El-Bary, K. Lotfy, and O. E. Sheta, "Distributed denial of service (DDoS) classification based on random forest model with backward elimination algorithm and grid search algorithm," *Sci. Rep.*, vol. 15, no. 1, May 2025, Art. no. e8332.

[11] E. D. Ramirez-Martinez, J. A. Pérez-Díaz, and N. M. Yungaicela-Naula, "P4-assisted slowloris DDoS attack detection in IoT environments by using ML and DL," *Comput. Netw.*, vol. 267, Jul. 2025, Art. no. 111364.

[12] Y. Li, R. Fang, Q. Song, and X. Yang, "STGCN-based link flooding attack detection and mitigation in software-defined network," in *Proc. IEEE 23rd Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom)*, Dec. 2024, pp. 346–353.

[13] OpenAI. (2023). *Prompt-Generated Search Phrases for Surveying ML-based SDN Security Frameworks. Used To Generate Search Phrases for Systematic Literature Retrieval*. [Online]. Available: https://chat.openai.com/

[14] Computing Research and Education Association of Australasia (CORE). (2023). *CORE Conference Rankings*. [Online]. Available: https://portal.core.edu.au/conf-ranks/

[15] SCImago Lab. (2025). *SCImago Journal & Country Rank*. [Online]. Available: https://www.scimagojr.com/journalsearch.php

[16] S. Bhattacharjee, K. L. Calvert, and E. Zegura, "An architecture for active networking," in *Proc. IFIP HPN*. Melbourne, VIC, Australia: Georgia Institute of Technology, 1997, pp. 265–279.

[17] A. Greenberg, G. Hjalmtysson, D. A. Maltz, A. Myers, J. Rexford, G. Xie, H. Yan, J. Zhan, and H. Zhang, "A clean slate 4D approach to network control and management," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 5, pp. 41–54, Oct. 2005.

[18] J. Carr, S. Winder, and S. Bigelow, *Understanding Telephone Electronics*. Amsterdam, The Netherlands: Elsevier, 2001.

[19] D. Sheinbein and R. P. Weber, "Stored program controlled network: 800 service using SPC network capability," *Bell Syst. Tech. J.*, vol. 61, no. 7, pp. 1737–1744, Sep. 1982.

[20] D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015.

[21] M. Chiosi, D. Clarke, P. Willis, A. Reid, J. Feger, M. Bugenhagen, W. Khan, M. Fargano, C. Cui, and H. Deng, "Network functions virtualisation: An introduction, benefits, enablers, challenges and call for action," in *Proc. SDN OpenFlow WC*, vol. 48, 2012, pp. 1–16.

[22] ON Foundation. (2022). *Mininet Address*. Accessed: Sep. 1, 2024. [Online]. Available: https://mininet.org

[23] B. Lantz and B. O'Connor, "A mininet-based virtual testbed for distributed SDN development," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 4, pp. 365–366, Sep. 2015.

[24] A. Bianco, R. Birke, L. Giraudo, and M. Palacin, "OpenFlow switching: Data plane performance," in *Proc. IEEE Int. Conf. Commun.*, May 2010, pp. 1–5.

[25] A. Tootoonchian and Y. Ganjali, "HyperFlow: A distributed control plane for OpenFlow," in *Proc. IEEE REN*, Apr. 2010, p. 3.

[26] B. Pfaff. (2011). *Openflow Specification Version 1.1.0*. Accessed: Sep. 1, 2024. [Online]. Available: http://www.openflow.org/documents/openflow-spec-v1

[27] R. Niranjan Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Radhakrishnan, V. Subramanya, and A. Vahdat, "PortLand: A scalable fault-tolerant layer 2 data center network fabric," in *Proc. ACM SIGCOMM Conf. Data Commun.*, Aug. 2009, pp. 39–50.

[28] M. B. Anwer and N. Feamster, "Building a fast, virtualized data plane with programmable hardware," in *Proc. 1st ACM workshop Virtualized infrastructure Syst. architectures*, Aug. 2009, pp. 1–8.
[29] S. Han, K. Jang, K. Park, and S. Moon, "PacketShader: A GPU-accelerated software router," *ACM SIGCOMM CCR*, vol. 40, no. 4, pp. 195–206, 2010.
[30] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, "P4: Programming protocol-independent packet processors," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 3, pp. 87–95, Jul. 2014.
[31] M. Shahbaz and N. Feamster, "The case for an intermediate representation for programmable data planes," in *Proc. 1st ACM SIGCOMM Symp. Softw. Defined Netw. Res.*, Jun. 2015, pp. 1–6.
[32] H. Sharaf, I. Ahmad, and T. Dimitriou, "Extended Berkeley packet filter: An application perspective," *IEEE Access*, vol. 10, pp. 126370–126393, 2022.
[33] A. Starovoitov, "LKML: Alexei starovoitov [patch net-next] extended BPF," 2013.
[34] M. A. M. Vieira, M. S. Castanho, R. D. G. Pacífico, E. R. S. Santos, E. P. M. C. Júnior, and L. F. M. Vieira, "Fast packet processing with eBPF and XDP: Concepts, code, challenges, and applications," *ACM Comput. Surveys*, vol. 53, no. 1, pp. 1–36, Jan. 2021.
[35] M. Medhat, S. G. Sayed, S. M. Abd-Alhalem, and A. E. Takieldeen, "Whitelisting requirements for effective cyber defense solutions," in *Proc. Int. Telecommun. Conf. (ITC-Egypt)*, Jul. 2023, pp. 484–489.
[36] A. Tankovic, E. Dervisevic, M. Mehic, and E. Kaljic, "A survey on data plane security in software-defined networks: Toward adaptive security of data planes," *IEEE Access*, vol. 13, pp. 97058–97093, 2025.
[37] D. Kalambe, D. Sharma, P. Kadam, and S. Surati, "A comprehensive plane-wise review of DDoS attacks in SDN: Leveraging detection and mitigation through machine learning and deep learning," *J. Netw. Comput. Appl.*, vol. 235, Mar. 2025, Art. no. 104081.
[38] B. Khanal, C. Kumar, and M. S. A. Ansari, "Real-time anomaly detection framework to mitigate emerging threats in software defined networks," *J. Netw. Syst. Manage.*, vol. 33, no. 2, Apr. 2025, Art. no. 29159.
[39] S. Majumder, M. K. Deb Barma, and A. Saha, "ARP spoofing detection using machine learning classifiers: An experimental study," *Knowl. Inf. Syst.*, vol. 67, no. 1, pp. 727–766, Jan. 2025.
[40] N. Karmous, Y. Ben Dhiab, M. Ould-Elhassen Aoueileyine, N. Youssef, R. Bouallegue, and A. Yazidi, "Deep learning approaches for protecting IoT devices in smart homes from MitM attacks," *Frontiers Comput. Sci.*, vol. 6, pp. 3146–3154, Oct. 2024.
[41] D. Sendil Vadivu and N. Rajagopalan, "RyuGuard—Combining ryu and machine learning for proactive DDoS defense in software-defined networks," *Concurrency Computation: Pract. Exper.*, vol. 36, no. 28, pp. 756–784, Dec. 2024.
[42] D. P. Andrade, K. Akkaya, A. Perez-Pons, S. Uluagac, and A. Sahin, "DDoS attack detection and mitigation in 5G networks using P4 and SDN," in *Proc. IEEE 49th Conf. Local Comput. Netw. (LCN)*, Oct. 2024, pp. 1–9.
[43] R. Dai, D. Tang, Z. Qin, K. Chen, K. Li, and J. Zhang, "Detecting congestion-related attacks via fine-grained queue diagnosis," *IEEE Trans. Cognit. Commun. Netw.*, vol. 12, no. 1, pp. 1255–1268, Feb. 2026.
[44] L. Priya and N. Rajagopalan, "A hybrid intrusion detection system for mitigating flow table buffer saturation attacks in software-defined networking," *Social Netw. Comput. Sci.*, vol. 6, no. 5, Jun. 2025.
[45] S. Thapliyal, M. Wazid, and D. Singh, "Design of distributed denial-of-service attack mechanism for IoT-driven data fusion system," *Cyber Secur. Appl.*, vol. 3, Apr. 2025, Art. no. 100092.
[46] L. Barolli, "Lecture notes on data engineering and communications technologies 249 advanced information networking and applications," LNDECT, Cham, Switzerland, Tech. Rep., 2025.
[47] S. Krishnaveni, S. Sivamohan, B. Jothi, T. M. Chen, and M. Sathiyanarayanan, "TwinSec-IDS: An enhanced intrusion detection system in SDN-digital-twin-based industrial cyber-physical systems," *Concurrency Computation: Pract. Exper.*, vol. 37, no. 3, Feb. 2025.
[48] D. S. Rao and A. J. Emerson, "An effective IDS using CondenseNet and CoAtNet based approach for SDN-IoT environment," *Comput. Electr. Eng.*, vol. 123, Apr. 2025, Art. no. 110305.
[49] A. Alharthi, M. Alaryani, and S. Kaddoura, "A comparative study of machine learning and deep learning models in binary and multiclass classification for intrusion detection systems," *Array*, vol. 26, Jul. 2025, Art. no. 100406.
[50] B. Sapkota, A. Ray, M. K. Yadav, B. R. Dawadi, and S. R. Joshi, "Machine learning-based attack detection and mitigation with multi-controller placement optimization over SDN environment," *J. Cybersecurity Privacy*, vol. 5, no. 1, p. 10, Mar. 2025.
[51] A. Bajenaid, M. Khemakhem, F. E. Eassa, F. Bourennani, J. M. Qurashi, A. A. Alsulami, and B. Alturki, "Towards robust SDN security: A comparative analysis of oversampling techniques with ML and DL classifiers," *Electronics*, vol. 14, no. 5, p. 995, Feb. 2025.
[52] M. P. Singh, Haimashreelakshmi, V. P. Singh, and M. Gupta, "Enhancing the security of SDN in 5G: A hybrid feature selection based ensemble machine learning framework for classification of cyber-attacks," *Social Netw. Comput. Sci.*, vol. 6, no. 3, Feb. 2025, Art. no. 100080.
[53] H. M. Belachew, M. Y. Beyene, A. B. Desta, B. T. Alemu, S. S. Musa, and A. J. Muhammed, "Design a robust DDoS attack detection and mitigation scheme in SDN-edge-IoT by leveraging machine learning," *IEEE Access*, vol. 13, pp. 10194–10214, 2025.
[54] C. Jisi, B.-H. Roh, and J. Ali, "An effective scheme for classifying imbalanced traffic in SD-IoT, leveraging XGBoost and active learning," *Comput. Netw.*, vol. 257, Feb. 2025, Art. no. 110939.
[55] J. Barach, "Towards zero trust security in SDN: A multi-layered defense strategy," in *Proc. 26th Int. Conf. Distrib. Comput. Netw.*, Jan. 2025, pp. 331–339.
[56] S. R. Alotaibi, H. Alfraihi, N. Alruwais, M. Maray, A. B. Miled, A. M. Al-Sharafi, M. Alotaibi, and S. H. Alajmani, "Two-tiered privacy preserving framework for software-defined networking driven defence mechanism for consumer platforms," *IEEE Access*, vol. 13, pp. 26684–26694, 2025.
[57] M. Fan, J. Lan, Y. Zhou, M. Pan, J. Li, and D. Zhang, "DDoS attack detection in SDN-assisted federated learning environment based on contrastive learning," *IEEE Access*, vol. 13, pp. 108798–108814, 2025.
[58] M. Kokila, "DeepSDN: Deep learning based software defined network model for cyberthreat detection in IoT network," *ACM Trans. Internet Technol.*, vol. 26, no. 1, pp. 1–29, Feb. 2026.
[59] S. Jamshidi, K. W. Nafi, A. Nikanjam, and F. Khomh, "Evaluating machine learning-driven intrusion detection systems in IoT: Performance and energy consumption," *Comput. Ind. Eng.*, vol. 204, Jun. 2025, Art. no. 111103.
[60] V. S. Naresh and D. Ayyappa, "Enhancing security in software defined networks: Privacy-preserving intrusion detection with homomorphic encryption," *J. Inf. Secur. Appl.*, vol. 92, Jul. 2025, Art. no. 104084.
[61] A. El-Sayed, A. A. Toony, F. Alqahtani, Y. Alginahi, and W. Said, "CO-STOP: A robust P4-powered adaptive framework for comprehensive detection and mitigation of coordinated and multi-faceted attacks in SD-IoT networks," *Comput. Secur.*, vol. 151, Apr. 2025, Art. no. 104349.
[62] A. Chetouane and K. Karoui, "New continual federated learning system for intrusion detection in SDN-Based edge computing," *Concurrency Computation: Pract. Exper.*, vol. 37, no. 2, pp. 49187–49213, Jan. 2025.
[63] P. T. Duy, D. T. T. Hien, T. D. Luong, N. H. Quyen, and V.-H. Pham, "Fed-evolver: An automated evolving approach for federated intrusion detection system using adversarial autoencoder in SDN-enabled networks," *Internet Things*, vol. 28, Dec. 2024, Art. no. 101397.
[64] S. H. A. Kazmi, F. Qamar, R. Hassan, and K. Nisar, "FOCC: A synthetically balanced federated one-class-classification for cyber threat intelligence in software defined networking," *IEEE Open J. Comput. Soc.*, vol. 6, pp. 701–713, 2025.
[65] M. Ali, Y.-F. Hu, and J.-P. Li, "Federated learning augmented cybersecurity for SDN-based aeronautical communication network," *Electronics*, vol. 14, no. 8, p. 1535, Apr. 2025.
[66] S. Bendale and B. Gupta, "Performance evaluation and validation of intelligent security mechanism in software defined network," *Int. J. Recent Innov. Trends Comput. Commun.*, vol. 11, no. 7s, pp. 359–367, Jul. 2023.
[67] R. A. M. Rudro, M. F. A. A. Sohan, S. K. Chaity, and R. I. Reya, "Enhancing DDoS attack detection using machine learning: A framework with feature selection and comparative analysis of algorithms," *TURCOMAT*, vol. 14, no. 3, pp. 1185–1192, 2023.
[68] M. S. Ataa, E. E. Sanad, and R. A. El-Khoribi, "Intrusion detection in software defined network using deep learning approaches," *Sci. Rep.*, vol. 14, no. 1, pp. 465–478, Nov. 2024.
[69] F. T. Liu, K. M. Ting, and Z. Zhou, "Isolation forest," in *Proc. 20th ICML*, 2008, pp. 413–422.
[70] R. O. Duda and P. E. Hart, "Pattern classification using neural networks," *IEEE Trans. Syst., Man, Cybern.*, vol. 3, no. 1, pp. 1–10, 1973.
[71] D. H. Ballard, "Modular learning in neural networks," in *Proc. 6th AAAI*, 1987, pp. 279–284.

[72] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biol. Cybern.*, vol. 43, no. 1, pp. 59–69, 1982.
[73] A. McCallum and K. Nigam, "A comparison of event models for naive Bayes text classification," in *Proc. AAAI-98*, Madison, WI, USA, 1998, pp. 41–48.
[74] M. A. Hearst, S. Dumais, E. Osuna, J. Platt, and B. Schölkopf, "Support vector machines," *IEEE Intell. Syst. their Appl.*, vol. 13, no. 4, pp. 18–28, Jul. 1998.
[75] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive-aggressive algorithms," *J. Mach. Learn. Res.*, vol. 7, no. 19, pp. 551–585, Mar. 2006.
[76] P. J. Clark and F. C. Evans, "Distance to nearest neighbor as a measure of spatial relationships in populations," *Ecology*, vol. 35, no. 4, pp. 445–453, Oct. 1954.
[77] S.-I. Amari, "Backpropagation and stochastic gradient descent method," *Neurocomputing*, vol. 5, nos. 4–5, pp. 185–196, Jun. 1993.
[78] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 785–794.
[79] D. W. Hosmer Jr., S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*. Hoboken, NJ, USA: Wiley, 2013.
[80] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas of a neuron," *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, 1943.
[81] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[82] J. A. Anderson, *An Introduction To Neural Networks*. Cambridge, MA, USA: MIT Press, 1995.
[83] M. Tan and Q. V. Le, "EfficientNetV2: Smaller models and faster training," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 10096–10106.
[84] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proc. icml*, Italy, 1996, pp. 148–156.
[85] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," in *Proc. 27th ICML)*, vol. 63, 2020, pp. 139–144.
[86] F. Rosenblatt et al., *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, vol. 55. Washington, DC, USA: Spartan books, 1962.
[87] G. Bebis and M. Georgiopoulos, "Feed-forward neural networks," *IEEE Potentials*, vol. 13, no. 4, pp. 27–31, Nov. 1994.
[88] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.
[89] T. Duan, A. Avati, D. Y. Ding, K. K. Thai, S. Basu, A. Y. Ng, and A. Schuler, "NGBoost: Natural gradient boosting for probabilistic prediction," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 2690–2700.
[90] M. L. Puterman, "Markov decision processes," *Handbooks operations Res. Manage. Sci.*, vol. 2, pp. 331–434, Apr. 1990.
[91] H. v. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Proc. AAAI*, 2016, vol. 30, no. 1, pp. 2094–2100.
[92] G. A. F. Seber, "Multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probab.*, Jun. 1984, pp. 281–297.
[93] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001.
[94] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn.*, vol. 63, no. 1, pp. 3–42, Apr. 2006.
[95] D. P Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.
[96] Kyoto University. (2015). *Benchmark Data–Description V5*. Accessed: Sep. 1, 2024. [Online]. Available: https://www.takakura.com/Kyoto
[97] MIT Lincoln Laboratory. (1998). *1998 Darpa Intrusion Detection Evaluation Dataset*. Accessed: 2024. [Online]. Available: https://www.ll.mit.edu/r-d/datasets/1998-darpa-intrusion-detection-evaluation-dataset
[98] *1999 Darpa Intrusion Detection Evaluation Dataset*. Accessed: Sep. 1, 2024. [Online]. Available: https://www.ll.mit.edu/r-d/datasets/1999-darpa-intrusion-detection-evaluation-dataset
[99] A. K. Sarica and P. Angin, "A novel SDN dataset for intrusion detection in IoT networks," in *Proc. 16th Int. Conf. Netw. Service Manage. (CNSM)*, Nov. 2020, pp. 1–5.
[100] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: BoT-IoT dataset," *Future Gener. Comput. Syst.*, vol. 100, pp. 779–796, Nov. 2019.
[101] N. Koroniotis, N. Moustafa, E. Sitnikova, and J. Slay, "Towards developing network forensic mechanism for botnet activities in the IoT based on machine learning techniques," in *Proc. 9th MONAMI*. Melbourne, VIC, Australia: Georgia Institute of Technology, Dec. 2018, pp. 30–44.
[102] N. Koroniotis, N. Moustafa, and E. Sitnikova, "A new network forensic framework based on deep learning for Internet of Things networks: A particle deep framework," *Future Gener. Comput. Syst.*, vol. 110, pp. 91–106, Sep. 2020.
[103] N. Koroniotis and N. Moustafa, "Enhancing network forensics with particle swarm and deep learning: The particle deep framework," 2020, *arXiv:2005.00722*.
[104] N. Koroniotis, N. Moustafa, F. Schiliro, P. Gauravaram, and H. Janicke, "A holistic review of cybersecurity and reliability perspectives in smart airports," *IEEE Access*, vol. 8, pp. 209802–209834, 2020.
[105] N. Koroniotis, "Designing an effective network forensic framework for the investigation of botnets in the Internet of Things," Ph.D. dissertation, School Comput. Sci. Eng., UNSW, Sydney, NSW, Australian, 2020.
[106] M. Sarhan, S. Layeghy, and M. Portmann. (2023). *CIC-ton-IoT: Machine Learning-based Network Intrusion Detection System Datasets in Netflow and Cicflowmeter Representations*. [Online]. Available: https://espace.library.uq.edu.au/view/UQ:f6884ce
[107] N. Abdalgawad, A. Sajun, Y. Kaddoura, I. A. Zualkernan, and F. Aloul, "Generative deep learning to detect cyberattacks for the IoT-23 dataset," *IEEE Access*, vol. 10, pp. 6430–6441, 2022.
[108] P. Borgnat, G. Dewaele, K. Fukuda, P. Abry, and K. Cho, "Seven years and one day: Sketching the evolution of Internet traffic," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 711–719.
[109] F. Abbasi, M. Naderan, and S. E. Alavi, "Intrusion detection in IoT with logistic regression and artificial neural network: Further investigations on n-baiot dataset devices," *J. Comput. Secur.*, vol. 8, no. 2, pp. 27–42, 2021.
[110] U Communication. (May 2023). *Intrusion and Vulnerability Detection in Software-defined Networks*. ITU AI for Good, challenge description for SDN security solutions. [Online]. Available: https://challenge.aiforgood.itu.int/match/matchitem/81
[111] A. P. Mathur and N. O. Tippenhauer, "SWaT: A water treatment testbed for research and training on ICS security," in *Proc. Int. Workshop Cyber-physical Syst. Smart Water Netw. (CySWater)*, Apr. 2016, pp. 31–36.
[112] T. Morris et al. (2014). *Awid (auto-ID) Dataset for RFID/IoT Anomaly Detection*. Dataset contains normal and attack traffic (impersonation, injection, flooding) for RFID/IoT systems. [Online]. Available: https://sites.google.com/a/uah.edu/tommy-morris-uah/awid-datasets
[113] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware images: Visualization and automatic classification," in *Proc. 8th Int. Symp. Visualizat. Cyber Secur.*, Jul. 2011, pp. 1–7.
[114] A. K. Sarica and P. Angin, "Explainable security in SDN-based IoT networks," *Sensors*, vol. 20, no. 24, p. 7326, Dec. 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/24/7326
[115] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol. 23, no. 13, p. 5941, Jun. 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/13/5941
[116] University of New Brunswick. (2019). *CIC DDOS 2019 Dataset*. Accessed: Sep. 1, 2024. [Online]. Available: https://www.unb.ca/cic/datasets/ddos-2019.html
[117] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," *ICISSp*, vol. 1, pp. 108–116, Jan. 2018.
[118] H. H. Jazi, H. Gonzalez, N. Stakhanova, and A. A. Ghorbani, "Detecting HTTP-based application layer DoS attacks on Web servers in the presence of sampling," *Comput. Netw.*, vol. 121, pp. 25–36, Jul. 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128617301172
[119] V. H. Nguyen and I Team. (2021). *Inseclab-IDS-2021 Dataset: A Comprehensive Dataset for Intrusion Detection Systems*. Accessed: Oct. 2023. [Online]. Available: https://link.uit.edu.vn/InSecLab-IDS-2021-dataset
[120] H. S. Anderson and P. Roth, "EMBER: An open dataset for training static PE malware machine learning models," 2018, *arXiv:1804.04637*.
[121] ORION Research Group. (2023). *Orion Research Group Datasets*. Londrina, Brazil. [Online]. Available: http://www.uel.br/grupos/orion/datasets.html
[122] University of New Brunswick. *NSL-KDD Dataset*. Accessed: Sep. 1, 2024. [Online]. Available: http://nsl.cs.unb.ca/NSL-KDD/

[123] M. S. Elsayed, Nhien-An Le-Khac, and A. D. Jurcut, "InSDN: A novel SDN intrusion dataset," *IEEE Access*, vol. 8, pp. 165263–165284, Sep. 2020, doi: 10.1109/access.2020.3022633.
[124] D. N. Ahuja and G. Singal. (2020). *Ddos Attack SDN Dataset*. [Online]. Available: https://data.mendeley.com/datasets/jxpfjc64kr/1
[125] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," *IEEE Access*, vol. 10, pp. 40281–40306, 2022.
[126] UNSW. *Toniot Datasets*. Accessed: Sep. 1, 2024. [Online]. Available: https://research.unsw.edu.au/projects/toniot-datasets
[127] M. Al-Hawawreh, E. Sitnikova, and N. Aboutorab, "X-iiotid: A connectivity- and device-agnostic intrusion dataset for industrial Internet of Things," 2021, doi: 10.21227/mpb6-py55.
[128] M. Al-Hawawreh, E. Sitnikova, and N. Aboutorab, "X-IIoTID: A connectivity-agnostic and device-agnostic intrusion data set for industrial Internet of Things," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3962–3977, Mar. 2022.
[129] (2020). *DDos Attack SDN Dataset*. [Online]. Available: https://totem.info.ucl.ac.be/
[130] S. Uhlig, B. Quoitin, J. Lepropre, and S. Balon, "Providing public intradomain traffic matrices to the research community," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 1, pp. 83–86, Jan. 2006, doi: 10.1145/1111322.1111341.
[131] U.-U of New Brunswick. (2015). *UNSW-NB15 Dataset*. [Online]. Available: http://www.unsw.adfa.edu.au/australian-centre-for-cyber-security/
[132] UNSW. *UNSW-NB15 Dataset*. Accessed: Sep. 1, 2024. [Online]. Available: https://research.unsw.edu.au/projects/unsw-nb15-dataset
[133] U.-U of new brunswick. (2017). *ISCX Dataset*. Accessed: Dec. 18, 2023. [Online]. Available: http://www.unb.ca/research/iscx/dataset/iscx-IDS-dataset.html
[134] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, "Toward developing a systematic approach to generate benchmark datasets for intrusion detection," *Comput. Secur.*, vol. 31, no. 3, pp. 357–374, May 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404811001672
[135] S. Garcia, M. Grill, J. Stiborek, and A. Zunino, "An empirical comparison of botnet detection methods," *Comput. Secur.*, vol. 45, pp. 100–123, Sep. 2014, doi: 10.1016/j.cose.2014.05.011. Accessed: Sep. 1, 2024.
[136] ScienceOpen. *ICS-CSR 2013–index*. Accessed: Sep. 1, 2024. [Online]. Available: https://www.scienceopen.com/hosted-document?doi=10.14236/ewic/ICSCSR2013.0
[137] T. Das. *UNR IDD Dataset*. Accessed: Sep. 1, 2024. [Online]. Available: https://www.tapadhirdas.com/unr-idd-dataset
[138] IDS-Hogzilla. *IDS Hogzilla Dataset*. Accessed: Sep. 1, 2024. [Online]. Available: https://ids-hogzilla.org/dataset/
[139] CAIDA, "Ddos attack dataset (2007-08-04)," https://www.caida.org/catalog/datasets/ddos-20070804 dataset/, last accessed 2024/09/01.
[140] T. Jafarian, "SDN-NF-TJ," *IEEE Dataport*, 2019. [Online]. Available: https://ieee-dataport.org/documents/sdn-nf-tj
[141] UCI KDD. (1999). *KDD Cup 1999 Dataset*. Accessed: Sep. 1, 2024. [Online]. Available: https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
[142] China Education and Research Network (CERNET). *Cernet Information*. Accessed: Sep. 1, 2024. [Online]. Available: https://www.edu.cn/english/cernet/
[143] P. Radoglou-Grammatikis, V. Kelli, T. Lagkas, V. Argyriou, and P. Sarigiannidis. (2022). *Dnp3 Intrusion Detection Dataset*. [Online]. Available: https://dx.doi.org/10.21227/s7h0-b081
[144] N. M. Yungaicela-Naula, C. Vargas-Rosales, J. A. Pérez-Díaz, E. Jacob, and C. Martinez-Cagnazzo. (2023). *SDN-slowrate-DDoS Dataset*. [Online]. Available: https://dx.doi.org/10.21227/amrt-8y98
[145] A. Kaan Sarica and P. Angin, "A novel SDN dataset for intrusion detection in IoT networks," in *Proc. 16th Int. Conf. Netw. Service Manage. (CNSM)*, Nov. 2020, pp. 1–5.
[146] L. Yang, A. Ciptadi, I. Laziuk, A. Ahmadzadeh, and G. Wang, "BODMAS: An open dataset for learning based temporal analysis of PE malware," in *Proc. IEEE Secur. Privacy Workshops (SPW)*, May 2021, pp. 78–84.
[147] M. Lester. (2021). *Pe Malware Machine Learning Dataset*. [Online]. Available: https://practicalsecurityanalytics.com/pe-malware-machine-learning-dataset/
[148] J. Arevalo-Herrera, J. Camargo Mendoza, J. I. Martínez Torre, T. Zona-Ortiz, and J. M. Ramirez, "Assessing SDN controller vulnerabilities: A survey on attack typologies, detection mechanisms, controller selection, and dataset application in machine learning," *Wireless Pers. Commun.*, vol. 140, nos. 1–2, pp. 739–775, Jan. 2025.
[149] P. Berde, M. Gerola, J. Hart, Y. Higuchi, M. Kobayashi, T. Koide, B. Lantz, B. O'Connor, P. Radoslavov, and W. Snow, "Onos: Towards an open, distributed SDN OS," in *Proc. 3rd ACM HotSDN*, 2014, pp. 1–6.
[150] N Github Repository. (2014). *Pox Github Address*. Accessed: Oct. 22, 2023. [Online]. Available: https://github.com/noxrepo/pox
[151] K. Hou, D. Saharia, V. Yegneswaran, and P. Porras, "LANTERN: Layered adaptive network telemetry collection for programmable data-planes," in *Proc. 6th Eur. P4 Workshop*, Dec. 2023, pp. 1–7.
[152] S. Rao Pappu and K. Chakravarthy Chilukuri, "SDN controller selection and secure resource allocation," *IEEE Access*, vol. 13, pp. 77278–77290, 2025.
[153] M. W. Asif, A. Aqdus, R. Amin, S. A. Chaudhry, F. S. Alsubaei, and S. Iqbal, "An efficient intrusion detection system using advanced machine learning techniques in software-defined networks (SDN) for healthcare system," *IEEE J. Biomed. Health Informat.*, pp. 1–14, 2025.
[154] A. O. M. Salih, "Exploring LDoS attack detection in SDNs using machine learning techniques," *Eng., Technol. Appl. Sci. Res.*, vol. 15, no. 1, pp. 19568–19574, Feb. 2025.
[155] J. Malik and F. Pastore, "Field-based security testing of SDN configuration updates," *IEEE Trans. Rel.*, vol. 74, no. 3, pp. 3469–3483, Sep. 2025.
[156] S. A. AlSharman, O. Al-Khaleel, and M. Al-Ayyoub, "A detailed inspection of machine learning based intrusion detection systems for software defined networks," *IoT*, vol. 5, no. 4, pp. 756–784, Nov. 2024.
[157] F. Rustam, R. Shafique, S. K. Posa, and A. D. Jurcut, "Malicious traffic detection in multi-environment network using dual-data trained LightGBM approach," in *Proc. IEEE 21st Int. Conf. Mobile Ad-Hoc Smart Syst. (MASS)*, Mali, Sep. 2024, pp. 598–603.
[158] C. L. Kumar, S. Betam, D. Pustokhin, E. Laxmi Lydia, K. Bala, R. Aluvalu, and B. S. Panigrahi, "Metaparameter optimized hybrid deep learning model for next generation cybersecurity in software defined networking environment," *Sci. Rep.*, vol. 15, no. 1, Apr. 2025.
[159] A. S. Zaidoun and Z. Lachiri, "A hybrid deep learning model for multi-class DDoS detection in SDN networks," *Ann. Telecommun.*, vol. 80, nos. 5–6, pp. 459–472, Jun. 2025.
[160] R. Shameli and S. Rajkumar, "High-speed threat detection in 5G SDN with particle swarm optimizer integrated GRU-driven generative adversarial network," *Sci. Rep.*, vol. 15, no. 1, Mar. 2025.
[161] F. Ashfaq, M. Wasim, M. A. Shah, A. Ahad, and I. M. Pires, "Enhancing security in 5G edge networks: Predicting real-time zero trust attacks using machine learning in SDN environments," *Sensors*, vol. 25, no. 6, p. 1905, Mar. 2025.
[162] Y. Zhou, "Network security threats and defense mechanisms for 6G multi-virtual network scenarios," *Int. J. Netw. Manage.*, vol. 35, no. 2, Mar. 2025.
[163] S. Kaur, K. Kumar, and N. Aggarwal, "Enhancing DDoS defense in SDN using hierarchical machine learning models," *J. Netw. Comput. Appl.*, vol. 239, Jul. 2025, Art. no. 104168.
[164] M. Sinha, P. Bera, M. Satpathy, K. S. Sahoo, and J. J. P. C. Rodrigues, "DDoSBlocker: Enhancing SDN security with time-based address mapping and AI-driven approach," *Comput. Netw.*, vol. 259, Mar. 2025, Art. no. 111078.
[165] F. Shi, Z. Zhou, J. Guo, R. Li, Z. Zhang, S. Li, Q. Liu, and X. Bao, "LightRL-AD: A lightweight online reinforcement learning approach for autonomous defense against network attacks," in *Proc. IEEE 23rd Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom)*, Dec. 2024, pp. 1614–1621.
[166] S. N, N. Parthiban, S. Vijay, and S. N. Sheela, "Comparison of mitigating DDoS attacks in software defined networking and IoT platforms," *Cyber Secur. Appl.*, vol. 3, Dec. 2025, Art. no. 100080.
[167] S. Venkatapathy, T. Srinivasan, H.-G. Jo, and I.-H. Ra, "An E2E network slicing framework for slice creation and deployment using machine learning," *Sensors*, vol. 23, no. 23, p. 9608, Dec. 2023.
[168] S. H. Almotiri, "AI driven IOMT security framework for advanced malware and ransomware detection in SDN," *J. Cloud Comput.*, vol. 14, no. 1, Apr. 2025.

[169] D. Jyothi, M. A. H. Farquad, and G. Narsimha, "A network security framework for hybrid botnet detection in critical infrastructure by using machine learning algorithms," *J. Tianjin Univ. Sci. Technol.*, 2025. [Online]. Available: https://www.researchgate.net/publication/390144870
[170] N. V. M. Bindu, V. K. Nassa, P. Vasuki, G. Manikandan, R. Jeena, and R. Mahaveerakannan, "IoT botnet detection from software defined network using American zebra optimization algorithm with SSRNN-ELM," *Int. J. Inf. Technol.*, vol. 17, no. 2, pp. 959–967, Mar. 2025.
[171] C. H. M. Souza, T. Pascoal, E. P. Neto, G. B. Sousa, F. S. L. Filho, D. M. Batista, and F. S. Dantas Silva, "SDN-based solutions for malware analysis and detection: State-of-the-art, open issues and research challenges," *J. Inf. Secur. Appl.*, vol. 93, Sep. 2025, Art. no. 104145.
[172] O. José Salcedo Parra, L. Correa Sánchez, and J. Gómez, "The evolution of VANET: A review of emerging trends in artificial intelligence and software-defined networks," *IEEE Access*, vol. 13, pp. 49187–49213, 2025.
[173] C. G. Suetor, D. Scrimieri, A. Qureshi, and I.-U. Awan, "An overview of distributed firewalls and controllers intended for mobile cloud computing," *Appl. Sci.*, vol. 15, no. 4, p. 1931, Feb. 2025.
[174] S. Hudda and K. Haribabu, "A review on WSN based resource constrained smart IoT systems," *Discover Internet Things*, vol. 5, no. 1, May 2025.
[175] P. V. Rajkumar and R. Sandhu, "Safety decidability for pre-authorization usage control with identifier attribute domains," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 3, pp. 465–478, May 2020.
[176] J. Medved, R. Varga, A. Tkacik, and K. Gray, "OpenDaylight: Towards a model-driven SDN controller architecture," in *Proc. IEEE Int. Symp. World Wireless, Mobile Multimedia Netw.*, Jun. 2014, pp. 1–6.
[177] V. Tomer, S. Sharma, and M. Davis, "Resilience in the Internet of Medical Things: A review and case study," in *Proc. Future Internet*, Nov. 2024.
[178] A. Kaur, C. R. Krishna, and N. V. Patil, "A comprehensive review on software-defined networking (SDN) and DDoS attacks: Ecosystem, taxonomy, traffic engineering, challenges and research directions," *Comput. Sci. Rev.*, vol. 55, Feb. 2025, Art. no. 100692.
[179] H. Alrashede, F. Eassa, and A. Marish, "Security of east-west interface of SDN: A review of challenges, solutions, and future directions," *Eng., Technol. Appl. Sci. Res.*, vol. 15, no. 3, pp. 23376–23385, Jun. 2025.

ALPEREN ÖRSDEMİR was born in Türkiye, in 1997. He received the B.Sc. degree in computer engineering from Sakarya University, İstanbul, Türkiye, in 2022, and the M.Sc. degree in cyber security from Koç University, Türkiye, in 2024. He is currently pursuing the thesis-based M.Sc. degree in computer science and engineering. He is also an Application Security Engineer with Ford Otosan, Sancaktepe, İstanbul. His current research interests include application security, intrusion detection in SDN environments, DevSecOps architecture, and machine learning for cyber defense. He is the President of the Koç University Cyber Security Society.

UTKU TEFEK received the B.S. degree in electrical and electronics engineering from Bilkent University, Türkiye, in 2013, and the Ph.D. degree in electrical and computer engineering from the National University of Singapore (NUS), in 2017. He is currently a Senior Research Scientist with Illinois Advanced Research Center at Singapore (IARCS), a research center of the University of Illinois Urbana–Champaign. He also holds an appointment as an Adjunct Senior Lecturer with NUS. He received the Singapore International Graduate Award (SINGA) to support his doctoral studies. His research interests include network security, cyber-physical system security, and applied cryptography.

ERTEM ESİNER received the French Baccalaureat degree from Galatasaray Lycee, and the B.S. degree in computer engineering and the M.Sc. degree in computer science and engineering from Koç University, Türkiye, in 2011 and 2013, respectively. He is currently pursuing the Ph.D. degree in computer science and engineering with Nanyang Technological University (NTU), supported by the Singapore International Graduate Award (SINGA). His doctoral research focused on cryptography, security, and privacy. He received the prestigious Vehbi Koç Scholarship to support his master's studies and the UPE/ACM Scholarship Award for outstanding academic performance and professional commitment in computing. He is also a Senior Research Scientist and a Coordinator with Illinois Advanced Research Center at Singapore (IARCS), a research center of the University of Illinois Urbana–Champaign. He leads the center's research activities and serves as a principal investigator on multiple government-funded projects in Singapore.

ALPTEKİN KÜPÇÜ (Senior Member, IEEE) received the Ph.D. degree from the Computer Science Department, Brown University, in 2010. Since then, he has been a Faculty Member with Koç University, leading the Cryptography, Cyber Security and Privacy Research Group that he founded. He is also a Co-Founder of Xtinge Technology Inc. He has various accomplishments, including eight international patents granted, 16 funded research projects (for 14 of which he was the principal investigator), two European Union COST Action Management Committee memberships, four outstanding teaching awards, four outstanding young scientist awards, an ACM Senior Member Awards, and the Royal Society of UK Newton Advanced Fellowship. His research mainly focuses on provably-secure applied cryptography, with applications to cloud security, privacy-preserving and adversarial machine learning, peer-to-peer networks, blockchains, and game theory and mechanism design.