Machine Learning for Network Management: Techniques for Traffic Management and Anomaly Detection

Arusyak Manasyan
IIAP
Yerevan, Armenia
e-mail: armanasyan@iiap.sci.am

Robert Tadevosyan
IIAP
Yerevan, Armenia
e-mail: robert@sci.am

Hrachya Astsatryan
IIAP
Yerevan, Armenia
e-mail: hrach@sci.am

Vladimir Sahakyan
IIAP
Yerevan, Armenia
e-mail: vladimir.sahakyan@sci.am

Arthur Petrosyan
IIAP
Yerevan, Armenia
e-mail: arthur@sci.am

***Abstract*—This paper proposes a real-time anomaly detection method that integrates machine learning (ML) with software-defined networking (SDN) in the GNS3 environment. It will generate a dataset of benign and malicious traffic (including website, DNS, scanning, and DoS attacks) and extract flow-level features to train supervised and unsupervised models, such as random forest, support vector machine, and isolation forest. The framework will provide a reproducible approach to adaptive network security using publicly available datasets and administrative scripts...**

***Keywords*—Machine learning (ML), Software-defined network (SDN), network management, anomaly detection.**

I. Introduction

Modern digital infrastructures depend on efficient, secure network management because they require supporting dynamic services like cloud computing and multimedia applications, which need both high availability and low latency and adaptive resource utilization. The existing methods fail to handle extensive, complicated networks that have grown beyond their operational limits. The separation of control and data planes in Software-defined networks (SDN) allows the network administrators to control network behavior through centralized programmable systems [1]. The new security problems that arise from this programmability exceed the capabilities of standard rule-based systems to solve them effectively [2].

Machine learning (ML) functions as a solution because it enables SDN systems to operate through adaptive and intelligent methods. The AI models detect unusual network behavior through learning patterns, which enables them to automatically stop DDoS attacks and port scans, and spoofing threats that standard security systems cannot detect [3, 4].

The research examines how machine learning systems operate with software-defined networking to create an intelligent security system that operates in real-time. Our experiment runs on GNS3 simulations to produce different traffic types, including website traffic and DNS requests and DDoS attacks and port scans and spoofing attempts, and password guessing attempts for feature extraction and model training (random forest and isolation forest, and MAB-based decision making). The Ryu controller uses OpenFlow rules to block forward and control malicious network traffic.

Our proposed framework is a closed-loop system that combines custom data collection, machine learning-based detection, and SDN automated response. The evaluation is based on criteria such as detection accuracy, false positive rate, vulnerability remediation latency, and resource efficiency. The results demonstrate how the combination of SDN programmability and machine learning analytics enables the creation of adaptive, resilient, and self-protecting networks for modern digital ecosystems [2].

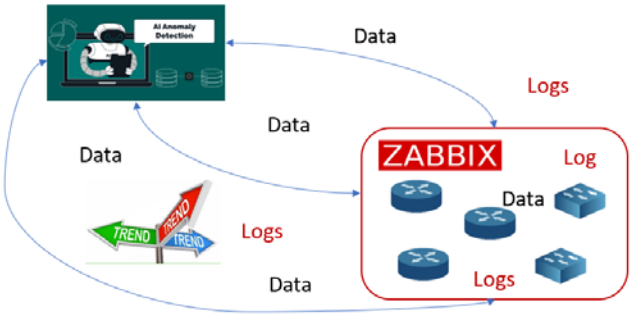

Figure 1. ML in network monitoring and anomaly detection

https://doi.org/10.51408/csit2025_74

II. Challenges in Current Approaches: Limitations in ML Models and Their Integration with SDN/NFV

Machine Learning (ML) combined with Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) offers great potential for better network management. However, several challenges prevent these technologies from achieving their full capabilities. These challenges come from the complexities of ML models and the difficulty of integrating them with SDN and NFV systems [5, 6]. While ML can improve network management, fully integrating these technologies is not straightforward. The inherent complexity of ML models and the challenges of integrating them with SDN and NFV make the process even harder. Addressing these issues is crucial to fully harness the benefits of ML, SDN, and NFV for more intelligent and efficient network management.

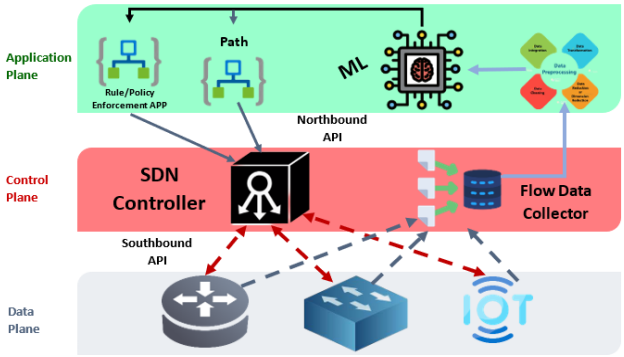

Figure 2. The proposed framework diagram

The proposed framework will be implemented in a controlled emulated environment using GNS3. A closed-loop anomaly detection and mitigation system will be designed and evaluated, which integrates machine learning (ML) algorithms with software-defined networking (SDN) [7]. The planned implementation will proceed through four main phases: construction of the virtual topology, dataset generation and preprocessing, ML-based anomaly detection, and SDN-enabled dynamic mitigation. The workflow is structured in phases to maintain reproducibility and adaptability across various network conditions.

GNS3 will host the experimental testbed to duplicate the conditions of a real network environment. The topology will include:

Various end hosts generate both normal network traffic through web browsing and DNS queries and file transfers, and email communication, and perform malicious activities, including port scanning and DoS flooding, as well as ARP spoofing and brute-force attempts.

- The programmable data plane operates through Open vSwitch (OVS) switches [8].
- The Ryu SDN Controller operates as a centralized control plane to provide dynamic policy enforcement functions [9].
- Network infrastructure devices, including routers and switches, enable authentic multi-hop routing of network traffic.
- The Zabbix monitoring system connects with SDN components to observe their CPU utilization, together with memory consumption and network latency during experimental procedures.

This topology will deliver a versatile test platform to prove the effectiveness of anomaly detection systems and response strategies.

A dataset specific to this project will be established inside the GNS3 platform. The system will create benign network traffic by reproducing typical network functions such as HTTP browsing, DNS resolution, FTP file transfers, and email communication. The system will produce malicious network traffic through these methods: use tools to generate DoS/DDoS traffic, run port scans using nmap, send custom ARP spoofed packets, and perform brute-force login attempts using Hydra. All traffic data will be recorded into PCAP files before transforming into flow-level features. The system will extract packet counts together with inter-arrival times and byte distributions from the network traffic. The flows will receive labels indicating either benign or malicious status to support the supervised machine learning model training.

III. Evaluation

The performance of machine learning models will be evaluated through precision, alongside accuracy and recall. The time delay between detecting anomalies and implementing rules represents the response time metric. The Ryu controller and OVS switches consume CPU and memory resources, which Zabbix monitors for system evaluation. The MAB framework demonstrates its effectiveness by adapting to various attack scenarios through dynamic selection of mitigation strategies.

Potential challenges may include:

The system experiences synchronization delays during rule deployment after detection, which can be solved by controller optimizations.

The training process encounters difficulties because of dataset imbalance, which needs balancing techniques to handle benign and malicious traffic.

The unsupervised detection system generates false positives, which can be reduced by adjusting thresholds and implementing ensemble methods.

The system faces scalability problems when handling heavy loads, which need stress testing for future optimization guidance.

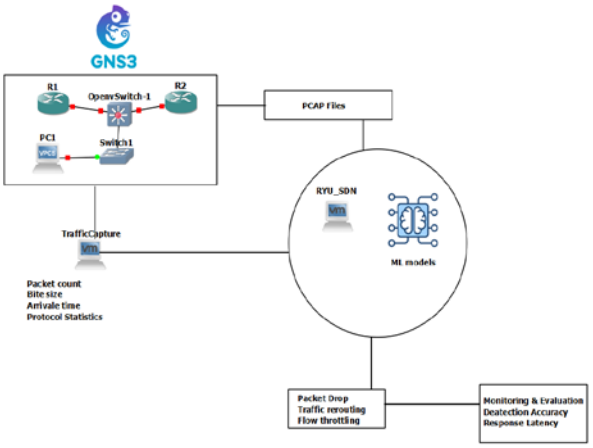

Figure 3. Implementation process

The project will develop an automated closed-loop security system through GNS3, together with Ryu controller and machine learning models. The system will use supervised learning and unsupervised learning together with adaptive learning to detect anomalies precisely and react automatically in real time. The experimental framework will provide a robust starting point to investigate large-scale deployments and advanced ML integration within SDN environments.

REFERENCES

[1] N. McKeown et al., "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008.

[2] Scott-Hayward, S., O'Callaghan, G., & Sakir Sezer, SDN security: A survey. *IEEE SDN for Future Networks and Services (SDN4FNS)*, pp. 1–7, 2013.

[3] Q. Yan, F. R. Yu, Q. Gong, and J. Li, "Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 602–622, 2015.

[4] L. Wang, L. Xu, and W. Guo, "Machine learning-based network anomaly detection in SDN: A survey," *IEEE Access*, vol. 7, pp. 132104–132119, 2019.

[5] J. Qadir, M. Naeem, and M. Bilal, "Integration challenges of ML with SDN and NFV: A review," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1869–1893, 2020.

[6] K. Zhang, Y. Mao, S. Leng, and T. Chang, "Machine learning and SDN integration: challenges and solutions," *IEEE Network*, vol. 33, no. 3, pp. 128–135, May/June 2019.

[7] M. Zekri, S. El Kafhali, N. Aboutabit, and Y. Saadi, "DDoS attacks detection using machine learning techniques in SDN environment," *Proc. IEEE ICC*, pp. 1–6, 2017.

[8] Open vSwitch, “Open vSwitch,” [Online]. Available: https://www.openvswitch.org/

[9] Ryu SDN Framework Community, “Ryu SDN Framework,” [Online]. Available: https://osrg.github.io/ryu/