An Advanced Security Model for Virtualized Network Functions in the Cloud

Abstract. Cloud computing and virtualization have revolutionized the IT industry, providing scalability and flexibility. But the dynamic and dispersed character of NFV-based cloud systems presents serious security issues including vulnerabilities in virtualized network functions (VNFs), inter-VNF communication, and hypervisor attacks. This paper focuses on the security implications of network virtualization and proposes a novel security model based on artificial intelligence and deep learning algorithms. The model combines a Deep Autoencoder (DAE) and machine learning techniques for network intrusion detection in NFV environments. The results demonstrate the model's effectiveness in detecting network intrusions in virtual networks. Our model achieved a high detection rate of 96%, demonstrating its effectiveness in detecting network intrusions within virtual networks.

Keywords: Network Function Virtualization, Cloud Computing, Network Intrusion Detection, Deep Autoencoder.

1 Introduction

Cloud computing has reshaped the internet architecture being one of the most significant advancements in the field of information technology in recent years [1]. This technology provides on-demand access to a variety of computing resources, including servers, storage, applications, and services. Virtualization technology is one of the key enablers of cloud computing [2]. Through virtualization, multiple operating systems can operate on a single physical machine, maximizing resource efficiency and reducing hardware expenses. Additionally, virtualization enables the creation of virtual machines (VMs) that may be easily deployed, moved, or scaled up or down in alignment with demand. The technology has been extended to networks, enabling multiple virtual networks to be created on a single physical network [3]. This technology has been the key idea behind what is called "Network Virtualization" which allows the creation of several isolated virtual networks on top of a shared physical network infrastructure [3]. Network Virtualization enables better network management, security, and compliance by allowing organizations to construct logical network segments for various applications or user groups. This call for two paradigms that are tandemly used: Network Functions Virtualization (NFV) and Software-Defined Networking (SDN) [4].

In order to deliver particular network operations, network operators use a variety of hardware running proprietary software. As a result, they must purchase and set up new hardware if they want to offer new network services [1]. This brings up several challenges, including rising equipment costs; rising power consumption along every new device; increased complexity that increases operational costs and misconfiguration risk; low dynamism and scalability.

Therefore, NFV has come as a potential solution to these problems. Network Functions Virtualization is a network architecture idea that the European Telecommunications Standards Institute (ETSI) standardized in October 2012 [4]. It entails using standard hardware to host diverse, vendor-independent network software components. NFV enables network functions (such as packet forwarding and dropping) to be carried out in virtual machines (VMs) in a cloud architecture as opposed to in dedicated devices. The Network Functions (NFs) are virtualized and consolidated onto standard hardware instead of being provided by a large number of devices with vendor-specific hardware and software. As a result, the requirement to purchase a single device for each NF is circumvented, leading to significant cost savings for network operators. The NFV paradigm takes advantage of the decoupling of software from hardware to increase flexibility and agility, allowing Telecommunication Service Providers to provide better service agility by opening up their network capabilities and network controls to users as well as other functionalities like the ability to deploy or support new network services more quickly and affordably.

According to the framework introduced by the (ETSI), NFV is built on three main domains [1]: (1) Virtual Network Functions (VNFs); (2) NFV infrastructure, and (3) NFV management and orchestration (MANO). VNFs are considered as containers of network services provisioned by software. The physical resources (such as CPU, memory, and I/O) needed for storage, processing, and networking in order to set up the execution of VNFs are included in the infrastructure component of NFV. The NFV management and orchestration domain manages all virtualization-specific tasks within the NFV framework. The MANO domain represents a main component dealing with heterogeneity in physical resources in order to ensure the desired level of interoperability in case they are developed by different vendors.

SDN, when coupled with NFV, provides the network programmability and centralized control needed to efficiently manage and orchestrate the virtualized network functions. SDN's decoupling of the control plane from the forwarding plane allows for dynamic control and management of the network infrastructure, while NFV virtualizes and manages the network functions themselves.

This mapped architecture has been a matter of interest due to its importance and the challenges it presents. Hence, several academic researchers have conducted studies and experiments related to NFV, which has found application in conjunction with cloud computing, edge computing, fog computing, the Internet of Things (IoT) [5], as well as 5G, for NFV is a fundamental technology of 5G networks [6], and potential upcoming technologies like 6G. These research projects explore various aspects of NFV such as service function placement, traffic load balancing, dynamic service function chain [7][8], resource orchestration [9], security architecture [6], service function routing [10], and recovery of device failure [11][12].

While NFV-based cloud environments have brought numerous benefits to organizations, they also come with a set of challenges [4]. One of the most significant challenges is security. The cloud environment presents a unique set of security risks and threats that must be addressed to ensure the confidentiality, integrity, and availability of data and applications. Virtual networks within the cloud are particularly vulnerable to security threats as they are highly dynamic and often span multiple physical locations and cloud providers [13]. (NFV) is a dynamic and distributed technology where resources are constantly changing, and the network perimeter is not well defined. This presents new security challenges in such environments. Classic and regular security solutions, such as firewalls, antivirus, and anti-malware systems, are often unable to detect and respond to emerging threats in real-time, making them inadequate within cloud environments and leaving them vulnerable to cyber-attacks [14]. Therefore, securing virtual networks in the cloud requires a comprehensive approach that leverages a range of novel and advanced security mechanisms, including application-aware security and threat intelligence. Emerging technologies such as machine learning and artificial intelligence are being increasingly used in the field of security to enhance threat detection and response capabilities [15]. These technologies have the ability to analyze large amounts of data and identify patterns that may be indicative of a security threat. Therefore, this research proposes a security model that incorporates deep learning techniques to secure VNF-based cloud environments, in which we use a deep autoencoder in order to achieve anomaly detection and then classify the anomalies detected using a machine learning multi-class algorithm.

Our contributions in this paper are:

- Developed SDN-based NIDS architecture for real-time intrusion detection in VNF-based clouds.

- Implemented hybrid security framework with deep learning and ML techniques for precise attack recognition and reduced false positives.
- Enhanced cloud security and incident response capabilities with granular details on detected attacks.
- Employed deep autoencoders as part of the anomaly detection process.

The remainder of this paper is structured as follows: the second section presents the related work about the security axis in the NFV-based clouds. We present, in section 3, NFV-NIDS, the proposed security model. Section 4 provides the results of our experiments and we finish this paper, by a conclusion and perspectives.

2 Related Works

To solve the security issue in NFV and SDN environments in cloud-based systems, several research studies have recently been carried out, as well as service function chains by referring to the Security Service Function Chain (SSFC). Most of the related works focus on security orchestration. Different previous works address the problem of security vulnerabilities in multi-tenant and multi-cloud NFV environments. According to the works, it is risky to trust cloud service providers, and end-to-end services are put in peril if a single VNF at the network core is compromised.

In [16] [28], the authors categorize various NFV security threats, their causes, and countermeasures, emphasizing the need for architectural redesign to mitigate vulnerabilities and enhance future NFV security. The proposed solutions aim to address these vulnerabilities by implementing advanced security protocols and promoting a more resilient architecture that can adapt to emerging threats in the evolving landscape of industrial IoT networks. This proactive approach not only strengthens the overall security posture of NFV systems but also ensures that organizations can confidently leverage the benefits of virtualization while minimizing risks associated with cyber threats. The authors, in [1] [27], address NFV security by proposing a lightweight certificateless secure communication scheme that mitigates security threats like replay, man-in-the-middle, and impersonation attacks, while significantly reducing computation and communication overheads in industrial IoT networks. By integrating this certificateless scheme, organizations can achieve a seamless balance between robust security measures and the operational efficiency required for the growing demands of industrial IoT environments and in [18], an anomaly detection framework for SFC integrity in NFV environments was proposed. It is based on the addition of a new module called SIM in the NFV MANO architecture. The SIM communicates with the Network Function Orchestrator (NFVO) using standard APIs to request information on NFV elements and report anomaly detection results. SIM has been designed detached from NFVO to make it independent. Network operators have direct access to configure SIM. However, using the management interfaces that NFVO has previously established, controlling and setting SIM through NFVO is the most appropriate method. [19] aims to outline a security architecture that addresses the attack vector in existing orchestrators; however, it can be expanded to use the most recent advancements in trusted computing, lightweight virtualization, and microservices. [20] proposes a blockchain based system BSec-NFVO that secures orchestration operations in virtualized networks ensuring the auditability of all operations that manipulate a service function chain. In BSec-NFVO, the manipulation of a service chain is done through signed transactions. Both tenants and orchestrators sign transactions, which are then validated and agreed upon by consensus. This approach provides irrefutable proof of the operations performed and makes the content of the transactions visible to all participants in the network, including tenants, orchestrators, and consensus participants. The authors aim to enhance the security and audibility of NFVO through the use of blockchain technology. Differently to aforementioned works focusing on MANO related issues, [21] proposed an optimal placement of security VNF for Service function chains based on the security level. This method formulates an optimization problem to construct service function chains based on security level and uses a genetic algorithm to find a near-optimal solution. The solution is used to place security VNFs to meet the security requirements for each service function chain, leading to the construction of multiple chains with high revenue and low cost while considering security level. Similar works have utilized SFCs to provide security services, and in a more recent model [22], multiple SFCs were combined into a security service function tree (SecSFT) to reduce the resource requirements for allocating virtual security functions. The SecSFT uses the concept of decision tree for classification; to classify the network attack traffic. Decision and detection rules are assigned to its nodes to distinguish between suspicious and normal network flows and detect or prevent any potential intrusions. The nodes of SecSFT carry out various security-focused virtual functions, such as load balancing, traffic shaping, intrusion detection, firewall, and virtualized network security hardware. An experimental cloud is utilized to construct the SecSFT, and its security services are evaluated and validated through testing against network attacks. [23] where the particularity optimization algorithm of network topology feature extraction using graph neural network is addressed in this research as a potential solution. This paper suggests a SSFC construction model using graph neural network to access different VNFs in a specific order. It predicts QoS indicators like delay and throughput based on network topology, routing policy, and traffic matrix for efficient security. The algorithm uses the representation of nodes in a graph neural network to construct a flexible and efficient security service function chain more comprehensively under the influence of its surrounding neighbor nodes. The model has been implemented on the control plane of the software-defined network. The problem of constructing a security service function chain for the network topology is transformed into a real-time prediction problem of link nodes based on graph neural networks.

NFV enhances 6G security by enabling flexible and scalable security architectures. It allows for the dynamic deployment of security functions, improving resilience and adaptability against emerging threats in next-generation wireless networks, as discussed in [24]. This adaptability is crucial as the complexity of cyber threats continues to evolve, necessitating a robust framework that can effectively mitigate risks while maintaining operational efficiency. As organizations transition to these advanced frameworks, they must also prioritize continuous monitoring and assessment of their security measures to stay ahead of potential vulnerabilities. Implementing proactive strategies such as threat intelligence sharing and automated response mechanisms will further strengthen their defenses, ensuring a comprehensive approach to cybersecurity in the age of 6G.This holistic strategy not only enhances the overall security posture but also fosters a culture of resilience, enabling organizations to quickly adapt and respond to unforeseen challenges in an increasingly interconnected digital landscape.

3 Proposed Model

(VNFs) are dynamic and distributed technologies that present new security obstructions in cloud environments. Classic and regular security solutions such as firewalls, antivirus, and anti-malware systems are often unable to detect and respond to emerging threats in real-time, leaving VNF-based cloud environments vulnerable to cyberattacks. To address these hurdles, recent research has shown that machine learning and deep learning techniques can be effective in identifying potential security risks in real-time and automatically responding to them [25]. Therefore, this paper proposes a security model that incorporates deep learning techniques to secure VNF-based cloud environments in which we propose an SDN-based NIDS architecture. The architecture can take advantage of the global network perspective offered by the SDN to identify and prevent intrusions in real-time by integrating the NIDS module in the controller.

Our intended model will serve as a network intrusion detection system employing anomaly-based detection technique. Subjecting the architecture to closer scrutiny, network statistics gathered by the SDN controller are transmitted to the NFV-NIDS security module for real-time intrusion detection analysis. We apply a hybrid security framework that leverages deep learning and machine learning techniques for detecting and classifying potential security risks in real-time. (See Figure 1).

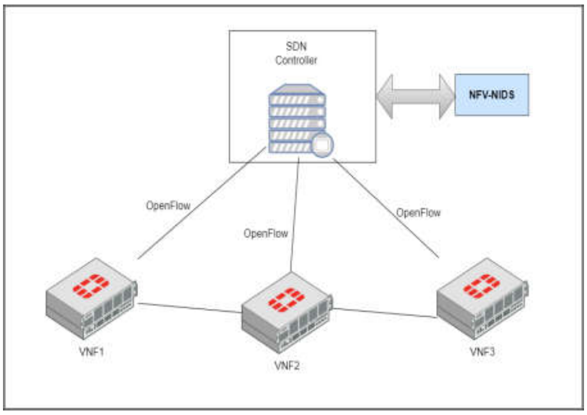

Figure 1: Proposed NFV Security Architecture

The proposed model consists of two phases: the first module (M1) uses a deep autoencoder (DAE) for anomaly detection to distinguish between normal and malicious traffic based on a threshold, and the second module (M2) performs multi-label classification to predict the type of attacks. We name both models output, R1 and R2 respectively (See Figure 2).
The detection process is threshold-based and relies on comparing the reconstruction error between the input and output of the autoencoder. Once a network anomaly is detected and identified, a multiclass classification is utilized to recognize the type of attack. With the type of attack identified, the OpenFlow (OF) [26] protocol can effectively mitigate it by modifying the flow table. Additionally, new security policies can be distributed throughout the network to prevent attacks.

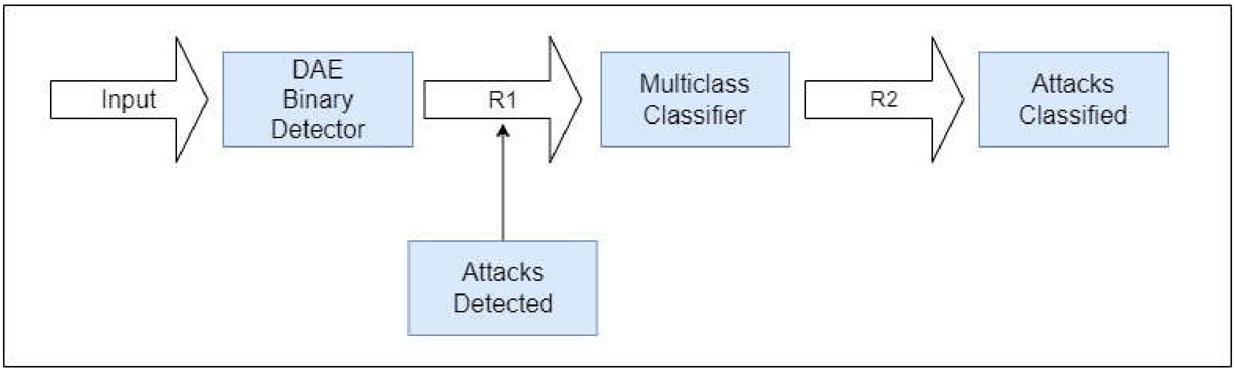

Figure 2: A High-Level of the Proposed NFV-NIDS Architecture

In the first phase of the proposed security framework, a deep autoencoder is utilized to perform anomaly detection on the network traffic. The autoencoder is trained on a dataset of normal traffic flows to learn the patterns and characteristics of benign traffic. During the testing phase, if the input traffic deviates significantly from the learned patterns, the autoencoder will flag it as anomalous traffic and output a binary classification result indicating whether the traffic is benign or malicious. The deep autoencoder consists of symmetric encoder-decoder architecture with multiple layers of fully connected neurons. The input to the model is a vector of shape (input shape), which is first compressed into a lower-dimensional representation by the encoder layers and then reconstructed back to the original shape by the decoder layers. The architecture of this model (M1) consists of three hidden layers on both the encoder and decoder, having the structure (Depth, Size) equivalent to (3, 32) with a latent layer of size 3 (i.e., the bottleneck layer) each with a combination of batch normalization and activation functions, such as ReLU. The encoder part of the model consists of the first four hidden layers, which progressively decrease the dimensionality of the input vector to a three dimensional representation. The decoder part of the model then mirrors the encoder layers, progressively increasing the dimensionality of the representation back to the original shape. The number of neurons in each layer decreases towards the center of the network and increases towards the output layer.

While in the second phase, a multi-label classification algorithm is employed to predict the type of attack. The input to this phase is the output of the first phase (i.e., the anomalies extracted from the binary classification result). If the input is labeled as malicious, the multi-label classification algorithm will predict the type of attack based on the patterns of the malicious traffic. The envisioned security model is designed to support multiple types of attacks and can accurately classify each attack based on its characteristics. During the second phase of the model, the multi-label classification algorithm analyzes the patterns of the malicious traffic and predicts the type of attack. Each attack is classified into one or more categories, depending on the specific characteristics of the attack. This phase is promising to enhance the effectiveness of the proposed security model in protecting VNFs by providing a more granular level of detail about the nature of the attack and allows for more precise and targeted response strategies.

In order to achieve the multi-class classification phase, three algorithms were experimented with, namely Deep Autoencoder, Random Forest (RF), and XGBoost (XGB). For the DAE of the second phase algorithm, we employed an architecture that closely resembles the original (M1), with minor modifications aimed at adapting it to a multi-detection task. For the remaining algorithms; RF and XGB, we performed a hyperparameter tuning technique using Grid search [27] to find the best combination of parameters to maximize the model's performance, avoid overfitting, and improve generalization.

4 Experiments and Results

For the purpose of experimenting with our NFV-NIDS model, we utilize the NF-CSE-CIC-IDS2018-v2 dataset [28]. It is generated from the original PCAP files of the CSE-CIC-IDS2018 dataset [29]. It represents an updated version of the existing CSE-CIC-IDS2018 that has been standardized into a NetFlow format. With a whopping 18,893,708 flows in, NF-CSE-CIC-IDS2018-v2 contains a wealth of insights into the workings of computer networks. Of these flows, 2,258,141 (11.95%) are attack samples, and 16,635,567 (88.05%) are benign ones. To ensure the reliability and generalizability of our model, we have carefully evaluated various publicly available datasets and have selected the NF-CSE-CIC-IDS2018-v2 for its high quality and significant number of labeled samples. Furthermore, the dataset's distribution of 11.95% attack samples and 88.05% benign ones closely resembles real-world network traffic, making it a suitable choice for training and testing our model. In addition, recent studies have demonstrated that the model performance when using the NF-CSE-CIC-IDS2018-v2 dataset is notably more efficient than other commonly used datasets such as CSE-CICIDS2018 and NF-CSE-CIC-IDS 2018 [25].

In order to evaluate our proposed framework's performance, we conduct an experiment of the model settings. We use for the binary detector DAE seven hidden layers each with a Batch Normalization layer. The number of neurons is "32-16-8-3-8-16-32". The activation function in the hidden layers is the RELU function. The batch size for training the DAE is 128, the number of epochs is 10, and the learning rate is 0.001. MSE is used as a loss function and Adam is the optimizer. These hyperparameters were meticulously selected based on a comprehensive process of experimentation and analysis which determined that this particular configuration consistently yields good results on the chosen dataset.

The model's performance is evaluated using various metrics such as accuracy, recall, precision, F1-score, Matthew's correlation coefficient (MCC), false negative rate (FNR), false positive rate (FPR) and the ROC curve (ROC-AUC). Table 2 reveals (M1) results.

Table 2: DAE Binary Detector Performance

Accuracy	Recall	Precision	F1-score	MCC	FNR	FPR	ROC-AUC
96%	99%	94%	96%	93%	0.4%	5%	96%

As we chose along this study to experiment with three different algorithms for the multi-class classification phase, we obtained overall three different hybrid approaches, namely the first one DAE-DAE, the second DAE-RF, and the latter DAE-XGB. We reveal the model's performance results in Table 3.

Table 3: Multi-class Classifier Performance

Model	Accuracy	Recall	Precision	F1-score	Cohen's kappa	FNR	FPR
DAE-DAE	99.65%	99.65%	99.65%	99.65%	99.44%	0.0%	0.3%
DAE-RF	99.82%	99.82%	99.82%	99.82%	99.72%	0.0%	0.0%
DAE-XGB	99.76%	99.76%	99.76%	99.76%	99.62%	0.0%	0.2%

The performance of the three hybrid models were analyzed and compared in this study, with a focus on their ability to accurately classify attacks detected by the (M1) phase. The results indicate that all three models demonstrated high accuracy and strong performance metrics. Among them, the RF hybrid approach stood out as the top-performing model, achieving the highest accuracy. These findings highlight the effectiveness of the RF algorithm in multi-class classification tasks for network security applications. The other two hybrid models, DAE-DAE and DAE-XGB, also performed well, but did not reach the level of accuracy and performance achieved by the RF hybrid model.

In order to clearly emphasize the performance of the hybrid proposed approach and the advantages it presents, we conduct a comparison of our entire model with its two components and the standalone multi-class classifiers, the DAE, RF, and XGB. To concretely demonstrate this, we build the aforementioned standalone classifiers to be able to further compare them. The findings are shown in Table 4, 5, 6:

Table 4: DAE-DAE vs Standalone DAE

Model	Accuracy	Recall	Precision	F1-score	Cohen's kappa	FNR	FPR
DAE (standalone)	97.77%	97.77%	98.17%	96.84%	89.67%	0.0%	0.00005%
DAE-DAE	99.65%	99.65%	99.65%	99.65%	99.44%	0.0%	0.3%

Table 5: DAE-RF vs Standalone RF

Model	Accuracy	Recall	Precision	F1-score	Cohen's kappa	FNR	FPR
RF (standalone)	97.77%	97.77%	97.82%	97.02%	89.14%	4.3%	0.0%
DAE-RF	99.82%	99.82%	99.82%	99.82%	99.72%	0.0%	0.0%

Table 6: DAE-XGB vs Standalone XGB

Model	Accuracy	Recall	Precision	F1-score	Cohen's kappa	FNR	FPR

XGB (standalone)	99.45%	99.45%	99.46%	99.42%	97.54%	0.0%	0.0%
DAE-XGB	99.76%	99.76%	99.76%	99.76%	99.62%	0.0%	0.2%

The tables' findings show that the proposed hybrid technique clearly outperforms the DAE, RF, and XGB standalone multi-class classifiers in terms of accuracy, recall, precision, F1-score, Cohen's Kappa, the false negative rate, and fallout. In their standalone versions, the DAE, RF, and XGB models all received high accuracy ratings, with the XGB model receiving the greatest accuracy of 99.45%. The accuracy scores were significantly raised when the standalone models were coupled with the DAE model in the proposed hybrid framework, and the DAE-RF hybrid technique ended up with a nearly ideal accuracy score of 99.82%.

The other evaluation metrics, in addition to accuracy, significantly increased when the separate models were integrated in the hybrid approach. The hybrid strategy outperformed the standalone models in terms of the Cohen's Kappa values, which assess agreement between predicted and actual classifications. As a result, there appears to be a higher level of agreement between the predicted and actual values, indicating that the hybrid method is more robust and reliable. Furthermore, the lower false negative rate and fallout observed in the proposed hybrid approach are particularly relevant in the context of intrusion detection. In such scenarios, misclassifying a benign network activity as a malicious intrusion can lead to false positives and unnecessary alarm triggering, which can be costly and time-consuming for network administrators. On the other hand, failing to detect a true intrusion can lead to a security breach and potential damage to the system. The proposed hybrid approach's ability to reduce the false negative rate and fallout indicates that it can better differentiate between normal network activities and malicious intrusions, leading to a more accurate and reliable intrusion detection system. This has significant implications for enhancing the security and resilience of computer networks against potential cyber-attacks. In the context of intrusion detection, the use of a hybrid approach that incorporates multiple models can be particularly advantageous. Network data is complex and dynamic, and it can be challenging to capture all the relevant features and patterns using a single model. The combination of multiple models, each trained to capture different aspects of the data, can lead to a more comprehensive and accurate representation of the network activity. This can result in a more effective intrusion detection system that is better able to detect and classify anomalous behavior.

5 Conclusion and Perspectives

Virtual networks within the cloud are particularly vulnerable to security threats. Traditional and conventional security measures are often inadequate. Through this study, we have proposed a hybrid approach of binary and multi-class classification for the Network Intrusion Detection System within NFV environments (NFV-NIDS) based on artificial intelligence and deep learning algorithms that uses a Deep Autoencoder to act as a first-step filter for network traffic classification. Future work can involve implementing our proposed model on real platforms to evaluate its effectiveness in a practical setting. Additionally, we suggest exploring the use of Graph Neural Networks (GNNs) to enhance the performance of the model.

References

[1] L. Haji, O. Ahmed, S. Zeebaree, H. Dino, R. Zebari, and H. Shukur, "Impact of Cloud Computing and Internet of Things on the Future Internet," Technology Reports of Kansai University, June 2020, vol. 62, pp. 2179-2190.

[2] A. Rista, J. Ajdari, and X. Zenuni, "Cloud Computing Virtualization: A Comprehensive Survey" , September 2020, pp. 462-472.

[3] N.M. Mosharaf Kabir Chowdhury and Raouf Boutaba, "A survey of network virtualization," in Computer Networks ,2010, vol. 54, no. 5, pp. 862-876.

[4] R. Mijumbi, J. Serrat, J.L. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network Function Virtualization: State-of-the-Art and Research Challenges," in IEEE Communications Surveys & Tutorials, 2016, vol. 18, pp. 236-262.

[5] S. Iftikhar et al., "AI-based fog and edge computing: A systematic review, taxonomy and future directions," Internet of Things, 2023, vol. 21, p. 100674.

[6] A. K. Alnaim, A. M. Alwakeel, and E. B. Fernandez, "Towards a Security Reference Architecture for NFV," Sensors (Basel, Switzerland), 2022, vol. 22, no. 10, p. 3750.

[7] M. A. Khoshkholghi and T. Mahmoodi, "Edge intelligence for service function chain deployment in NFV-enabled networks," Computer Networks, 2022, vol. 219, p. 109451,

[8] M. A. Abdelaal, G. A. Ebrahim, and W. R. Anis, "Efficient Placement of Service Function Chains in Cloud Computing Environments," Electronics, 2021, vol. 10, no. 3, p. 323.

[9] A. Sarah, G. Nencioni, and M. M. I. Khan, "Resource Allocation in Multi-access Edge Computing for 5G-and-beyond networks," Computer Networks, 2023, vol. 227, p. 109720

[10] J. Pei, P. Hong, K. Xue, and D. Li, "Resource Aware Routing for Service Function Chains in SDN and NFV-Enabled Network," in IEEE Transactions on Services Computing, 2021, vol. 14, no. 4, pp. 985-997.

[11] N. Siasi, A. Jaesim, A. Aldalbahi, and N. Ghani, "Link Failure Recovery in NFV for 5G and Beyond," in 2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Barcelona, Spain, 2019, pp. 144-148.

[12] Z. Huang and H. Huang, "Proactive Failure Recovery for Stateful NFV," in IEEE 26th International Conference on Parallel and Distributed Systems (ICPADS), Hong Kong, 2020, pp. 536-543.

[13] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network Function Virtualization: Challenges and Opportunities for Innovations," in IEEE Communications Magazine, February 2015, vol. 53, no. 2, pp. 90-97.

[14] S. Al-Bakri, B. Shanmugam, G. Narayana Samy, N. Idris, and A. Ahmad, "Traditional Security Risk Assessment Methods in Cloud Computing Environment: Usability Analysis," in Jurnal Teknologi, March 2014, vol. 73, no. 3, pp. TBD.

[15] Y. Song, S. Hyun, and Y.-G. Cheong, ”Analysis of Autoencoders for Network Intrusion Detection,” Sensors, Jun. 2021, vol. 21, no. 12, p. 4294.

[16] B., Zahran, N. Ahmed, A. R. Alzoubaidi and M. A., Ngadi, Security and Privacy Issues in Network Function Virtualization: A Review from Architectural Perspective. International Journal of Advanced Computer Science & Applications, 2024, vol. 15, no 6.

[17] Z., Ashraf, A., Sohail, and M. Iqbal, Design and Implementation of Lightweight Certificateless Secure Communication Scheme on Industrial NFV-Based IPv6 Virtual Networks. Electronics, 2024, vol. 13, no 13, p. 2649.

[18] L. Bondan, T. Wauters, B. Volckaert, F. De Turck, and L. Z. Granville, ”Anomaly detection framework for SFC integrity in NFV environments,” IEEE Conference on Network Softwarization (NetSoft), Bologna, Italy, 2017, pp. 1-5.

[19] N. Paladi, A. Michalas, and H. V. Dang, ”Towards Secure Cloud Orchestration for Multi-Cloud Deployments,” in Proceedings of the 5th Workshop on CrossCloud Infrastructures & Platforms, CrossCloud’18, Porto, Portugal, 2018, pp. 4:1-4:6

[20] G. A. F. Rebello, I. D. Alvarenga, I. J. Sanz, and O. C. M. B. Duarte, ”BSec-NFVO: A Blockchain-Based Security for Network Function Virtualization Orchestration,” in 2019 IEEE International Conference on Communications (ICC), Shanghai, China, 2019, pp. 1-6.

[21] D. Dwiardhika and T. Tachibana, ”Optimal Construction of Service Function Chains Based on Security Level for Improving Network Security,” in IEEE Access, 2019, vol. 7, pp. 145807-145815.

[22] J. -L. Luo, S. -Z. Yu and S. -J. Peng, ”SDN/NFV-Based Security Service Function Tree for Cloud,” in IEEE Access, 2020, vol. 8, pp. 38538-38545.

[23] Li, Wei, Haomin Wang, Xiaoliang Zhang, Dingding Li, Lijing Yan, Qi Fan, Yuan Jiang, Ruoyu Yao. ”Security Service Function Chain Based on Graph Neural Network” Information 13, 2022, no. 2: 78

[24] J. Hatim, C Habiba, S Chaimae. Evolving Security for 6G: Integrating Software-Defined Networking and Network Function Virtualization into Next-Generation. Architectures. International Journal of Advanced Computer Science & Applications, 2024, vol. 15, no 6.

[25] Y. Song, S. Hyun, and Y.-G. Cheong, ”Analysis of Autoencoders for Network Intrusion Detection,” Sensors, Jun. 2021, vol. 21, no. 12, p. 4294

[26] Open Networking Foundation [Online]. Available: https: //opennetworking.org/

[27] ” GridSearchCV - scikit-learn documentation,” scikit-learn website. [Online]. Available: https://scikit-learn.org/stable/modules/ generated/sklearn.model_selection.GridSearchCV.html.

[28] M. Sarhan, S. Layeghy, and M. Portmann, ”Towards a standard feature set for network intrusion detection system datasets,” Mobile Networks and Applications, 2021, vol. 27, no. 1, pp. 357-370.

[29] A Realistic Cyber Defense Dataset (CSE-CIC-IDS2018). Available: https://registry.opendata.aws/cse-cic-ids2018.