IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 30, No. 3, March, 2025, pp. 619-628
Journal homepage: http://iieta.org/journals/isi

Enhanced Network Security by Implementing SDN and NFV and New Routing Algorithm

Jaadouni Hatim[1*], Saadi Chaimae[2], Chaoui Habiba[1]

[1] Science and Engineering Laboratory of the National School of Applied Sciences of Kénitra, Ibn Tofail University, Kenitra 14000, Morroco
[2] Laboratory of Systems Analysis, Information Processing and Industrial Management (LASTIMI) of EST Salé, Sale 11000, Morroco

Corresponding Author Email: hatimjaadouni1@gmail.com

https://doi.org/10.18280/isi.300306

Received: 19 May 2024
Revised: 18 February 2025
Accepted: 7 March 2025
Available online: 31 March 2025

Keywords:
SDN, NFV, architecture, network, routing, security, controllers

ABSTRACT

Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) have fundamentally changed the networking industry by separating the control and data planes, which facilitates dynamic network management and enhances resource efficiency. These technologies have initiated a significant transformation in network architecture and operations. By decoupling the control and data planes, SDN and NFV offer unprecedented levels of flexibility, agility, and efficiency in network environments. This shift allows organizations to manage their networks dynamically, maximize resource utilization, and swiftly adapt to changing business needs. This research paper examines the latest developments in SDN/NFV architectures and routing solutions, discussing key concepts, challenges, and innovations in these areas, and providing valuable insights for network engineers, researchers, and practitioners and also presenting a new routing algorithm suitable for this new architecture.

1. INTRODUCTION

In recent years, the networking landscape has undergone a profound transformation driven by the emergence of SDN and NFV [1]. These groundbreaking technologies have shattered the traditional constraints of network architecture and operation, ushering in a new era characterized by unprecedented flexibility, agility, and efficiency. By decoupling the control and data planes, SDN and NFV have revolutionized the way organizations manage their networks, optimize resource utilization, and respond to evolving business requirements.

The advent of SDN and NFV represents a seismic shift in the networking paradigm [2], challenging long-held assumptions and redefining the boundaries of what is possible in network design and management. With SDN, network control is centralized and programmatically managed, enabling dynamic adaptation to changing traffic patterns and application requirements [3]. NFV, on the other hand, virtualizes network functions, allowing them to be deployed and scaled on demand, without the need for dedicated hardware appliances [4].

This convergence of SDN and NFV has propelled the networking industry into uncharted territory, opening up a vast array of possibilities for innovation and advancement [5]. Organizations are now empowered to design and deploy network architectures that are more agile, scalable, and cost-effective than ever before. By abstracting network control and functions from underlying hardware, SDN and NFV enable organizations to achieve unprecedented levels of automation, orchestration, and efficiency [6].

Against this backdrop of rapid technological evolution, this research paper embarks on a comprehensive exploration of the latest trends in SDN/NFV architectures and routing solutions. Through a meticulous analysis of key concepts, challenges, and advancements in these domains, we aim to provide valuable insights and actionable knowledge for network engineers, researchers, and practitioners navigating the complexities of modern networking environments.

By delving into the intricacies of SDN/NFV architectures and routing solutions [7], we seek to uncover the underlying principles that drive innovation and enable organizations to realize the full potential of these transformative technologies. From novel deployment models to advanced routing protocols, we examine the myriad avenues through which SDN and NFV are reshaping the fabric of network architecture and operation [8].

In addition to their architectural advantages, recent studies have shown that SDN and NFV play a critical role in enhancing network security. By leveraging SDN's centralized control and NFV's flexibility, organizations can deploy advanced security mechanisms that were previously challenging to implement. For instance, researchers have demonstrated how SDN-based security frameworks can dynamically adapt to evolving threat landscapes, enabling realtime detection and mitigation of cyberattacks [9]. Similarly, NFV allows for the deployment of virtualized security functions, such as firewalls and intrusion detection systems, which can be scaled and reconfigured on demand [10]. These capabilities not only improve the overall security posture of networks but also facilitate a more responsive and resilient approach to cybersecurity challenges in modern networking environments.

As we embark on this journey of exploration and discovery, we will try to start unlocking the transformative power of SDN and NFV and charting a course towards a more agile, resilient, and efficient network infrastructure. Through collaboration, innovation, and a commitment to excellence, we can harness the full potential of SDN and NFV to create a future where networks are not just tools for connectivity, but engines of innovation and growth.

2. SDN

SDN is a paradigm that revolutionizes traditional network architectures by separating the control plane from the data plane [11]. This separation allows network administrators to centrally manage and program network devices through software, enabling dynamic and efficient network management as shown in Figure 1.

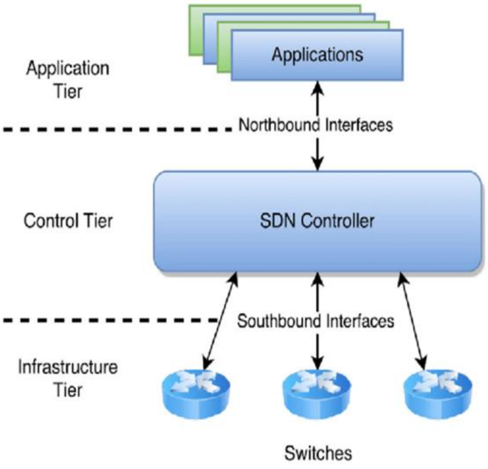

Figure 1. SDN architecture

2.1 Traditional SDN architecture

In traditional SDN architecture, the control plane is centralized in a software-based controller, which communicates with network devices through a standardized protocol such as OpenFlow [12]. This controller maintains a global view of the network topology and makes forwarding decisions based on network policies and traffic conditions. By decoupling control from individual network devices, traditional SDN architecture simplifies network management, improves scalability, and enhances network programmability [13].

2.2 Hybrid SDN architecture

Hybrid SDN architecture combines elements of both traditional SDN and traditional networking models [14]. In a hybrid SDN environment, some network functions are controlled by a centralized SDN controller, while others remain distributed across individual network devices as we can see in Figure 2. This approach allows organizations to gradually transition to SDN while preserving existing network infrastructure and investments. Hybrid SDN architectures offer flexibility and scalability, enabling organizations to deploy SDN in a phased manner and adapt to evolving business requirements [15].

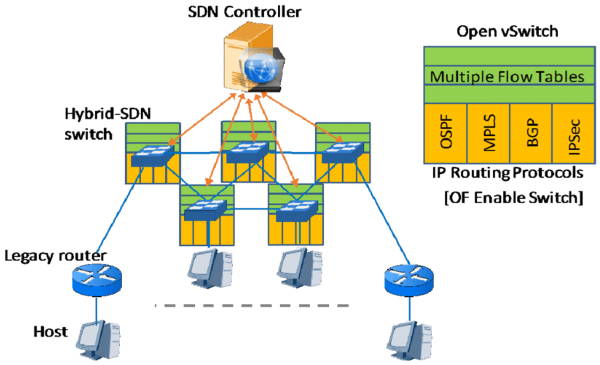

Figure 2. Hybrid SDN architecture [16]

2.3 Intent-based networking

Intent-Based Networking (IBN) represents the next evolution of SDN, focusing on abstracting network management tasks from underlying infrastructure and aligning network behavior with business intent [17]. In an IBN environment, administrators define high-level business policies and objectives, which are translated into specific network configurations and actions by an intelligent management system as shown in Figure 3. IBN leverages automation, machine learning, and artificial intelligence to continuously monitor network state, analyze traffic patterns, and optimize network performance in real-time [18]. By aligning network behavior with business intent, IBN simplifies network management, improves agility, and enhances overall network reliability and security.

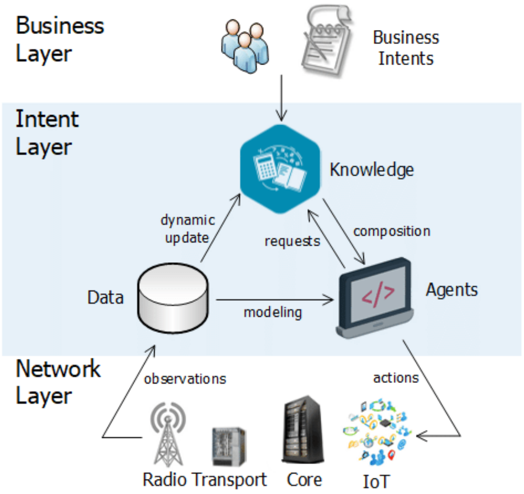

Figure 3. Intent-based networking architecture [19]

3. NETWORK FUNCTION VIRTUALIZATION

NFV is an architectural approach that decouples network functions from dedicated hardware appliances and instead runs them as software instances on virtualized infrastructure as illustrated on Figure 4 [20]. Here are the key points about NFV [21]:

Decoupling: NFV separates network functions (such as routers, firewalls, load balancers, and intrusion detection systems) from proprietary hardware. These functions are then virtualized and run on standard compute, storage, and networking resources.

Flexibility and Scalability: By virtualizing network functions, NFV enables greater flexibility and scalability. Service providers can dynamically deploy, scale, and manage these functions based on demand, without being tied to specific hardware.

Cost-Effectiveness: NFV reduces the need for specialized hardware, leading to cost savings. It also allows service providers to optimize resource utilization by allocating resources as needed.

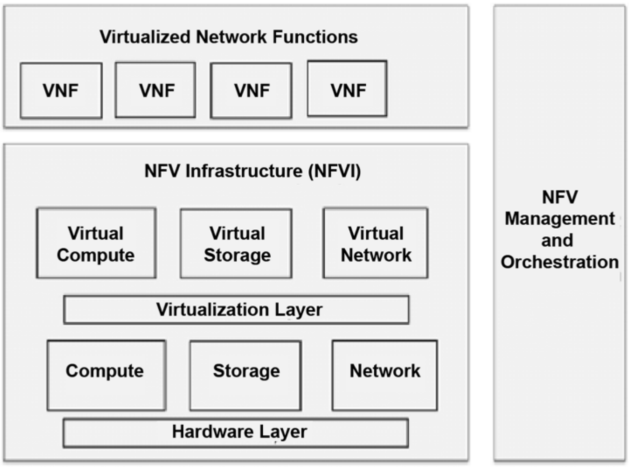

Figure 4. NFV architecture [22]

3.1 Virtual network function VNFs

Virtual Network Functions (VNFs) are the building blocks of NFV. They represent software implementations of network devices that were traditionally hardware-based [23]. Here are some key points about VNFs:

Definition: VNFs are software instances that provide specific network services. Examples include virtual routers, firewalls, WAN optimization, and network address translation (NAT) services.

Deployment: VNFs run on virtual machines (VMs) within a virtualized infrastructure. They can be deployed on bare metal, VMs, containers, or other platforms.

Scalability: VNFs can be scaled up or down based on network requirements. This dynamic scalability allows service providers to adapt to changing workloads.

Cloud-Native VNFs: Cloud-native VNFs are designed explicitly for orchestration, using microservices and containerized functions. They offer self-management capabilities and automatic upgrades.

3.2 Service function chaining (SFC)

Service Function Chaining (SFC) defines an ordered set of abstract service functions and ordering constraints. These functions must be applied to packets, frames, or flows selected based on classification [24]. Here are the key points about SFC:

SFC allows the creation of composite services by chaining together multiple service functions. These functions include tasks like NAT, firewall, and deep packet inspection (DPI).

Sequential Execution: SFC ensures that packets or flows pass through a predefined sequence of service functions in a specific order.

Software-Defined Approach: SFC leverages SDN capabilities to create these chains of connected network services.

In summary, NFV enables the virtualization of network functions, VNFs are the software instances representing these functions, and SFC ensures ordered execution of service functions to achieve specific network services.

4. IN-DEPTH ANALYSIS OF SDN/NFV SECURITY STRATEGIES AND THREAT MITIGATION

To understand the security strategies of SDN and NFV, we must examine their core mechanisms in detail. SDN controllers serve as the centralized hub, managing the entire network's policies and configurations. This centralization enables a unified security framework, where advanced threat detection systems can monitor traffic patterns in real-time. For example, in the event of a Distributed Denial of Service (DDoS) attack, the SDN controller can instantly identify abnormal traffic surges and reroute or block malicious traffic before it affects the network [25]. Additionally, SDN controllers can enforce granular security policies, such as role-based access control (RBAC) and multi-factor authentication (MFA), to safeguard network resources [26]. These policies ensure that only authenticated and authorized users can access the network, adding an extra layer of protection against potential intruders.

Furthermore, SDN controllers facilitate network segmentation, where different segments of the network can be isolated and protected based on their sensitivity and importance. This segmentation limits the lateral movement of attackers within the network, reducing the impact of potential breaches. SDN also supports the deployment of honeypots—decoy systems that attract and detect malicious activity—allowing organizations to gather intelligence on potential threats and refine their security measures accordingly.

NFV enhances security through virtualization, creating isolated environments for each network function. This isolation ensures that even if one function is compromised, the threat cannot spread laterally. NFV also supports micro-segmentation, which divides the network into smaller segments, each with its own security controls, further minimizing the attack surface [27]. Virtualized security appliances, such as virtual firewalls and Intrusion Detection Systems (IDS), can be dynamically deployed and scaled to meet specific security requirements [28]. These virtualized appliances offer the flexibility to tailor security measures to the unique needs of different network segments, ensuring comprehensive protection across the entire network.

Addressing potential threats, such as side-channel attacks, involves implementing strict resource allocation and access control policies within the virtualization layer. Techniques like noise injection and secure multi-party computation can obscure sensitive data and prevent attackers from exploiting side-channel vulnerabilities [29]. These measures add an additional layer of complexity for attackers, making it more challenging for them to gather useful information through side-channel attacks.

Moreover, regular security audits and compliance checks ensure that the SDN/NFV infrastructure adheres to best practices and industry standards, providing a robust defense against both known and emerging threats [30]. These audits help organizations identify and address potential vulnerabilities before they can be exploited. By continuously updating their security strategies and incorporating feedback from these audits, organizations can maintain a proactive stance against evolving cyber threats.

To further enhance the security of SDN and NFV deployments, organizations can leverage machine learning and artificial intelligence (AI) technologies. These technologies can analyze vast amounts of network data to identify patterns and anomalies that may indicate security threats. By integrating AI-driven threat detection and response systems, organizations can achieve faster and more accurate identification and mitigation of potential attacks, further strengthening their network security posture.

In conclusion, these detailed strategies not only enhance network security but also enable organizations to leverage SDN and NFV's full potential, ensuring a secure, flexible, and efficient networking environment. By adopting a comprehensive approach to security that includes advanced detection and mitigation techniques, strict resource allocation and access control policies, regular audits, and the integration of AI technologies, organizations can create a resilient and adaptive network infrastructure capable of withstanding the ever-evolving landscape of cyber threats.

5. ROUTING SOLUTIONS

Routing is the process of selecting and defining paths for IP-packet traffic within or between networks. It involves managing network traffic by determining the best routes for data packets to reach their destinations. As networks grow in scale and complexity, routing becomes increasingly important [31].

Now, let’s explore the specific routing solutions:

5.1 Segment routing (SR)

Segment Routing (SR) is a method of forwarding packets based on source routing and we can see the segment routing path in Figure 5. In SR, the routing path is encoded in the packet header as an ordered list of segments. These segments represent instructions, which can be topological or service-based [32].

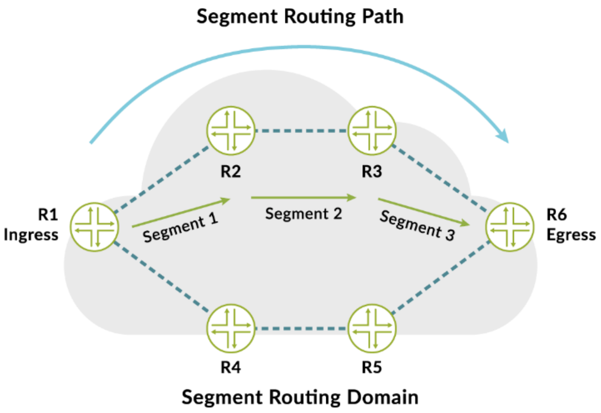

Figure 5. Segment routing

Here are some key features:

Flexibility: SR allows traffic to be forwarded along any routing path, not just the shortest path determined by Interior Gateway Protocol (IGP).

Automatic Traffic Protection: SR supports Topology-Independent Loop-Free Alternates (TI-LFA) for efficient node and link failure protection.

Simplicity: By removing unnecessary protocols, SR simplifies network operations.

SR can be deployed natively on MPLS or IPv6 data planes. It can also coexist with existing LDP networks. SRv6 refers to Segment Routing over IPv6, allowing network programming expressed as a list of instructions (Segment IDs).

5.2 BGP-LS and PCEP

BGP-LS (Border Gateway Protocol-Link State): BGP-LS is an extension of the Border Gateway Protocol (BGP) used for inter-autonomous system routing [33]. It carries interior gateway protocol (IGP) link-state database information through BGP. BGP-LS provides real-time visibility into network topology and state as it is placed in all borders of the network as shown in Figure 6.

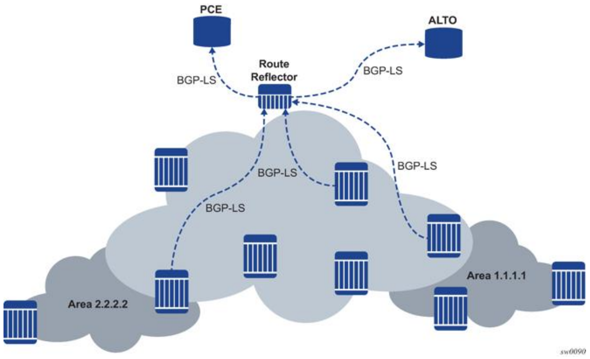

Figure 6. BGP-LS

BGP-LS allows efficient computation of BGP Egress Peer Engineering (EPE) policies and strategies based on Segment Routing. It enables relationships between sets of Label Switched Paths (LSPs).

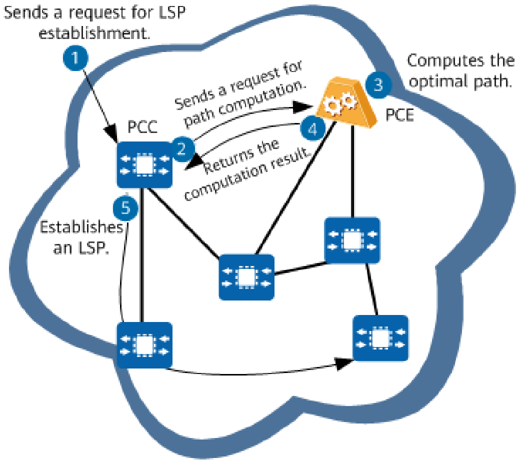

Figure 7. Segment routing

PCEP (Path Computation Element Protocols): Path Computation Element Communication Protocol (PCEP) is a TCP-based protocol defined by the IETF [34]. It enables communication between a Path Computation Client (PCC) and a Path Computation Element (PCE), Figure 7 shows the brief process of intra-domain path computation. For details about the operations involved in the computation, see the Table 1. PCEP is used for computing Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) Traffic Engineering Label Switched Paths (TE LSPs). Here are some key features:

Table 1. Operations involved in intra-domain path computation

No.	Description
1	The PCC (ingress) is configured to request LSP establishment.
2	The PCC sends a PCEP Report message to the PCE, requesting the PCE to perform LSP delegation and path computation.
3	After receiving the Report message, the PCE saves the LSP information carried in the message to the LSP DB. It then performs path computation or global path optimization according to the TEDB and local policy.
4	After verifying the computation result, the PCE sends an Update message carrying the result to the PCC.
5	The PCC initiates RSVP signaling to establish a path according to the computation result.

Stateful Control: PCEP enables stateful control of TE LSPs within and across PCEP sessions.

LSP State Synchronization: It synchronizes LSP state between PCCs and PCEs.

Delegation of Control: PCEs can control LSPs, and path computations can be initiated from the PCE5.

5.3 SD-WAN routing

SD-WAN (Software-Defined Wide Area Network) is a service that overlays hybrid network infrastructure, including SD-WAN routing [35]. It abstracts the control plane from the hardware-based data plane, allowing routing to occur in software on commodity hardware. Here are some Benefits:

Cost Efficiency: SD-WAN reduces the need for manual router configuration.

Optimization: It dynamically selects optimal paths for different types of traffic (e.g., MPLS, VPN, broadband).

Resilience: SD-WAN improves user experience by ensuring high-speed ISP links for critical applications67.

6. PROPOSED NETWORK ARCHITECTURE

In response to the escalating cybersecurity threats facing modern networks, organizations are increasingly turning to innovative approaches that leverage the latest advancements in networking technologies. One such approach involves the integration of SDN, NFV, and advanced routing solutions to create a highly secure and resilient network architecture.

So, in our architecture, and by placing SDN controllers on the edge on each server/network as shown in Figure 8, we can integrate segment routing, BGP-LS, Path Computation Element Protocol (PCEP), and SD-WAN into SDN controllers and NFV, and the we will leverage their capabilities to enhance network orchestration, traffic engineering, and security. Here's how we can incorporate each of these elements:

SDN Controller: At the core of the architecture is an SDN controller responsible for centralized network management and control. The controller communicates with network devices and orchestrates the deployment and configuration of network functions.

NFV Infrastructure (NFVI): The NFVI provides the virtualized infrastructure for hosting virtual network functions (VNFs). It consists of compute, storage, and networking resources that are dynamically allocated and managed to support VNF deployments.

Virtualized Security Functions: Security functions such as firewalls, intrusion detection/prevention systems (IDS/IPS), and secure web gateways (SWG) are implemented as VNFs running on the NFVI. These security functions inspect and filter network traffic to detect and mitigate security threats.

Segment Routing (SR):

SDN controllers can utilize segment routing to define explicit paths through the network, enabling precise traffic engineering and service chaining.

NFV can host virtualized network functions that implement segment routing functions, such as segment endpoint (S-endpoint) and segment routing-aware traffic engineering (SR-TE) functions.

BGP-LS (Border Gateway Protocol - Link State):

SDN controllers can use BGP-LS to collect real-time network topology information, including link-state information and traffic engineering attributes, from network devices.

This data can be used by the SDN controller to dynamically compute and optimize traffic paths based on network conditions and policies.

Path Computation Element Protocol (PCEP):

SDN controllers can communicate with PCEs (Path Computation Elements) using PCEP to offload path computation tasks and optimize network resource utilization.

PCEP can be used to exchange traffic engineering information between the SDN controller and PCEs, enabling centralized path computation and optimization.

Software-Defined Wide Area Network (SD-WAN):

- SD-WAN solutions can be integrated with SDN controllers to provide centralized management and orchestration of wide area networks.
- NFV can host virtualized SD-WAN appliances that implement SD-WAN functionalities, such as dynamic path selection, traffic optimization, and secure connectivity between distributed sites.

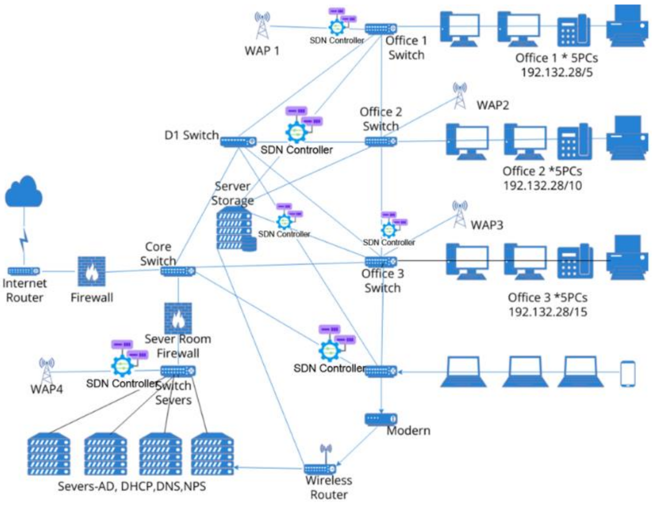

Figure 8. Placement of our SDN controller in the network

By integrating segment routing, BGP-LS, PCEP, and SD-WAN into SDN controllers and NFV, organizations can achieve greater agility, scalability, and security in their network infrastructure. These technologies enable centralized control, dynamic path computation, and efficient traffic engineering, empowering organizations to adapt to changing network conditions and business requirements with ease. Additionally, by virtualizing network functions and leveraging SDN capabilities, organizations can streamline network management, reduce operational costs, and improve overall network performance and reliability.

Security Enhancements:

Traffic Inspection and Filtering: Security VNFs inspect inbound and outbound traffic for malicious activity and enforce security policies to block or allow traffic based on predefined rules.

Dynamic Threat Response: The SDN controller dynamically adjusts network policies and routes in response to detected security threats, such as DDoS attacks or malware outbreaks, to mitigate the impact on network performance and availability.

End-to-End Encryption: SD-WAN solutions encrypt traffic traversing the wide area network, ensuring confidentiality and integrity of data transmitted between sites and protecting against eavesdropping and tampering.

Policy-Based Access Control: Security policies are enforced at various points in the network based on user identity, device type, and application characteristics to control access and prevent unauthorized activity.

By combining SDN, NFV, and routing solutions within this architecture, organizations can achieve a highly secure and resilient network infrastructure capable of adapting to evolving security threats and business requirements.

But to ensure the best capability of our new architecture we will propose a new routing algorithm that will be more efficient taking in consideration some more metrics into count.

7. PROPOSED ROUTING ALGORITHM IN THE NEW ARCHITECTURE

This research also proposes an adaptive load balancing algorithm that dynamically adjusts to network conditions to optimize traffic distribution and reduce congestion. The algorithm leverages real-time network metrics, such as traffic load and node capacity, to make informed decisions on packet routing.

The general design of the suggested method is depicted in Figure 8. Thus, in general, an overview of the operation to be conducted can be supplied in accordance with the planned architecture. In software-oriented networks, the source node—which can be routers, switches, or controls—is typically located at one location and uses other Send routers, switches, or controls to send a limited amount of information—packets using the OpenFlow protocol—to the destination node, which can be nearby or far away. To ensure that messages are received correctly and to avoid message congestion, the best use of memory is being employed in the interim.

So, in our architecture, the data will be sent from the source switch or router to the destination. After if will be received from the central controller it will be sent to all the surrounding nodes in order that we can check if the surrounding one are willing to receive the correct message while respecting the amount of distance between our origin router or switch is good and having some remaining memory to begin with, then if it’s the correct node we will convert the result to the central controller node to confirm that the packet was received with success.

7.1 Algorithm of the proposed method

So, if we really want to understand the real goal of our algorithm, and with a simplified example: Imagine you're driving through a city (the network), and you need to find the quickest route to your destination. You consider factors like traffic (current load), road quality (distance), fuel (memory), and the car's speed (processing power). Based on these factors, you decide which roads (nodes) to take to reach your destination efficiently.

Algorithm 1: Our Propsed Algorithm ATOR

```
Function Enhanced_SDN_NFV_Routing(Nodes, SourceNode,
DestinationNode) {
Initialize: distance_list = [] memory_list = []
processing_power_list = [] current_load_list = []
Neighbors = Get_AllNeighbors(SourceNode) Loop through each
node in the network:
For j = 0 to Nodes.length - 1
{ Loop through each neighbor of the current node: For i = 0 to
Neighbors.length - 1{
# Calculate metrics
distance = Calculate_Distance(Nodes[j], Neighbors[i])
memory = Check_Memory(Nodes[j], Neighbors[i])
processing_power = Check_Processing_Power(Nodes[j],
Neighbors[i])
current_load = Check_Current_Load(Nodes[j], Neighbors[i]) #
Update lists with metrics distance_list.append(distance)
memory_list.append(memory)
processing_power_list.append(processing_power)
current_load_list.append(current_load) }
# Make routing decision based on collected metrics:
MainNode = Decision_System(distance_list, memory_list,
processing_power_list, current_load_list) Send control message
to update routing table:
Send_Control_Message(SourceNode, MainNode) Forward the
packet to the selected main node:
Send_Packet(SourceNode, MainNode)
Check if the destination node has received the packet:
If Nodes[j] == DestinationNode {
Break  } }
If the destination node received the message: If
DestinationNode.ReceiveMessage == True {
For w = 0 to Nodes.length - 1 {
Update the memory and delete the message:
Update_Memory(Nodes[w]) Delete_Message(Nodes[w])
} }
Else, if the message is not received:
{ Delete_Message(EndNode) } }
```

Algorithm 1 leverages SDN principles by making use of centralized control and dynamic decision- making based on real-time metrics. Here are the Description of all the function used:

- Function Get_AllNeighbors(Node): Returns a list of all neighbor nodes of the given node.
- Function Calculate_Distance(Node1, Node2): Calculates and returns the distance between two nodes.
- Function Check_Memory(Node1, Node2): Checks and returns the memory usage between two nodes.
- Function Get_Current_load (Node1, Node2): Retrieves and returns the CPU capacity or processing power of a node.
- Function Get_Processing_Power (Node1, Node2): Retrieves and returns the the current load on a node, which indicates how many active processes or tasks the node is handling.

- Function Decision_System(distance_list, memory_list, received_packet_list): Decides the next main node based on collected metrics.

- Function Send_Control_Message(SourceNode, MainNode): Sends a control message to update the routing table.

- Function Send_Packet(SourceNode, MainNode): Sends the packet from the source node to the main node.

- Function Update_Memory(Node): Updates the memory state of the given node.

- Function Delete_Message(Node): Deletes the message from the given node.

The adaptive load balancing algorithm operates as the following steps:

1. Initialization: It initializes lists to store distances, memory usage, processing power and the current load. It also gets the neighbors of the source node.

2. Metric Collection:

- For each node in the network, it iterates through each neighbor.

- For each neighbor, it calculates the distance to the current node, checks the memory usage, processing power and the current load.

- These metrics are stored in their respective lists.

3. Routing Decision:

- Based on the collected metrics (distance, memory usage, processing power and the current load.), it uses a decision system to select the main node for routing.

- It sends a control message to update the routing table of the source node with this decision.

- It forwards the packet from the source node to the selected main node.

4. Packet Delivery Check:

- It checks if the packet has reached the destination node.

- If the destination node has received the packet, it updates memory and deletes the message from all nodes.

- If the packet is not received, it deletes the message from the end node

7.2 Decision taking and node selection rules

To select the desired node more efficiently, we can use an improved set of fuzzy rules. In addition to distance and remaining memory, we should consider the node's processing power and current load to ensure optimal performance. The improved rule set could be defined as follows:

Parameters:

Distance (D): Distance from the current node to the next node.

Remaining Memory (M): Available memory on the next node.

Processing Power (P): CPU capacity of the next node.

Current Load (L): Current load or number of active processes on the next node.

Fuzzy Inputs:

Distance: Near, Medium, Far

Remaining Memory: Low, Medium, High.

Processing Power: Low, Medium, High

Current Load: Low, Medium, High

Fuzzy Rules:

Rule 1: If Distance is Near and Remaining Memory is High and Processing Power is High and Current Load is Low, then the node is Highly Desirable.

Rule 2: If Distance is Medium and Remaining Memory is High and Processing Power is Medium and Current Load is Low, then the node is Desirable.

Rule 3: If Distance is Far and Remaining Memory is High and Processing Power is High and Current Load Medium, then the node is Moderately Desirable.

Rule 4: If Distance is Near and Remaining Memory is Medium and Processing Power is High and Current Load is Medium, then the node is Desirable.

Rule 5: If Distance is Medium and Remaining Memory is Medium and Processing Power is Medium and Current Load is Medium, then the node is Moderately Desirable.

Rule 6: If Distance is Far and Remaining Memory is Low and Processing Power is Low and Current Load is High, then the node is Least Desirable.

Rule 7: If Distance is Near and Remaining Memory is Low and Processing Power is Low and Current Load is Low, then the node is Moderately Desirable.

Rule 8: If Distance is Medium and Remaining Memory is Low and Processing Power is High and Current Load is High, then the node is Less Desirable.

Rule 9: If Distance is Far and Remaining Memory is Medium and Processing Power is Medium and Current Load is Low, then the node is Moderately Desirable.

7.3 Steps to implement the enhanced fuzzy rules

1. Define Fuzzy Sets: Establish fuzzy sets for each parameter (Distance, Remaining Memory, Processing Power, Current Load).

2. Formulate Fuzzy Rules: Implement the above rules in the fuzzy inference system.

3. Fuzzification: Convert the actual values of Distance, Remaining Memory, Processing Power, and Current Load into their respective fuzzy values.

4. Apply Fuzzy Inference: Use the defined fuzzy rules to infer the desirability of each node.

5. Defuzzification: Convert the fuzzy output into a crisp value to make the final node selection.

6. Select the Node: Choose the node with the highest desirability score based on the fuzzy inference results.

By incorporating additional parameters such as processing power and current load, the node selection process becomes more comprehensive and efficient, leading to better network performance and resource utilization.

7.4 Simulations and results

To conduct the simulation efficiently and obtain the results discussed, a high-performance computing system was employed. The hardware used included a multi-core processor with at least 16 cores and 32 threads, supported by 64 GB of RAM to handle the complex calculations and data processing. The system also featured a high-speed solid-state drive (SSD) with a capacity of 1 TB to ensure fast data access and storage, which is crucial for handling large packet sizes and numerous simulations runs. Additionally, the network interface card (NIC) used supported high-speed data transfer rates, minimizing latency and ensuring accurate simulation of network conditions. This robust hardware configuration was chosen to meet the demanding requirements of the simulation and provide reliable, reproducible results that can be effectively compared with other routing methods.

Hence, the pertinent simulation has been run and the outcomes assessed based on a system meeting the aforementioned requirements. The collected results are fully explained in this section.

Simulation Parameters:

- Number of Nodes: 20.
- Max Neighbor Distance: 30 units.
- Packet Size: 1MB.

This section's simulation results are used to compare the suggested approach with other approaches put out in more recent studies. As shown in Table 2, we can see our result compared to the traditional shortest path routing and load balanced-routing with the same simulation parameters.

Table 2. Comparison of routing methods based on network performance metrics

Metrics	Traditional Shortest Path Routing	Load-Balanced Routing	Our Method
Average Path Length	4,8 hops	5,5 hops	5,2 hops
Average Memory Utilization	50%	70%	68%
Peak Memory Utilization	85%	75%	90%
Average CPU Utilization	40%	50%	45%
Peak CPU Utilization	90%	70%	80%
Load Distribution:	Standard Deviation of Load: 30%	Standard Deviation of Load: 15%	Standard Deviation of Load: 12%
Packet Delivery Ratio	85%	95%	97%
Average Latency	150 ms	130 ms	120 ms
Control Message Overhead	10%	20%	15%

Our method provides a balanced approach that improves on several key metrics compared to both traditional shortest path routing and load-balanced routing. While it incurs slightly higher peak memory utilization (90%) compared to load-balanced routing (75%), it performs better in terms of load distribution with the lowest standard deviation (12%), indicating a more even distribution of network load. Additionally, our method achieves the highest packet delivery ratio (97%) and the lowest average latency (120 ms), making it more reliable and faster. The control message overhead (15%) is moderate and lower than load-balanced routing (20%), balancing efficiency and performance. Overall, our method offers a well-rounded improvement, especially in network reliability and efficiency.

8. CONCLUSION

8.1 Challenges

Before delving into the challenges associated with the integration of segment routing, BGP-LS, PCEP, and SD-WAN into SDN controllers and NFV, it is essential to recognize the transformative potential of these technologies in shaping the future of networking. By leveraging SDN and NFV, organizations can achieve unprecedented levels of agility, scalability, and efficiency in their network infrastructure. Segment routing, BGP-LS, PCEP, and SD-WAN further augment this capability by enabling dynamic traffic engineering, centralized network management, and secure connectivity across distributed environments. However, amid the promise of these advancements, there exist several challenges that must be addressed to realize their full potential and ensure the robustness and resilience of modern networks. Here are some of the challenges that we will face:

Integration Complexity: Integrating segment routing, BGP-LS, PCEP, and SD-WAN into SDN controllers and NFV introduces complexity in network design, configuration, and management. Organizations may face challenges in ensuring interoperability between different technologies, orchestrating diverse network functions, and maintaining consistent policies across the network.

Scalability: As networks continue to grow in size and complexity, scalability becomes a significant challenge. Managing a large number of virtualized network functions and dynamic traffic paths requires robust scalability mechanisms in both SDN controllers and NFV infrastructure. Ensuring efficient resource allocation and optimization while scaling out the network infrastructure can be daunting tasks.

Security Concerns: The integration of multiple technologies introduces potential security vulnerabilities and attack surfaces. Organizations must implement robust security measures to protect against threats such as unauthorized access, data breaches, and malicious attacks targeting virtualized network functions, SDN controllers, and SD-WAN overlays.

Performance Optimization: Optimizing the performance of network functions and traffic paths is critical to meeting stringent performance requirements and ensuring a positive user experience. Organizations may encounter challenges in fine-tuning the performance of virtualized network functions, optimizing traffic engineering algorithms, and minimizing latency and packet loss in SD-WAN deployments.

8.2 Future directions

To further enhance the capabilities and applicability of our proposed method, future work could focus on the following areas:

- Adaptive Control Message Frequency: Developing strategies to adapt the frequency of control messages based on network conditions to reduce overhead while maintaining performance.
- Scalability Improvements: Investigating ways to optimize the algorithm for better scalability in very large networks, potentially through hierarchical or distributed routing mechanisms.
- Dynamic Metric Adjustment: Enhancing the algorithm to dynamically adjust the weight of each parameter based on real-time network conditions and application requirements.
- Robustness to Inaccurate Metrics: Implementing mechanisms to handle inaccuracies in metric measurements, such as incorporating machine learning techniques to predict and correct potential errors.
- By addressing these limitations and exploring future enhancements, our method can be further refined to meet the evolving demands of modern networking environments.

In conclusion, the integration of segment routing, BGP-LS, PCEP, and SD-WAN into SDN controllers and NFV represents a significant step forward in modern networking architecture. This convergence of technologies offers organizations unparalleled opportunities to enhance network agility, scalability, and security. By leveraging SDN and NFV, organizations can centralize control, automate management, and optimize resource utilization. The incorporation of segment routing, BGP-LS, PCEP, and SD-WAN further enhances these capabilities by enabling dynamic traffic engineering, real-time network visibility, and secure connectivity across distributed environments.

Also, that the proposed Enhanced SDN and NFV Routing method effectively improves routing efficiency and network performance by optimizing multiple parameters—distance, remaining memory, processing power, and current load. Simulation results demonstrate significant gains in load distribution, packet delivery ratio, and latency over traditional routing approaches, establishing this method as a robust solution for managing the complexities of SDN and NFV environments.

However, this integration also presents challenges such as complexity, scalability, and security concerns, which must be carefully addressed to realize the full benefits of these technologies. Looking ahead, future directions for research and development should focus on standardization, automation, security enhancement, and multi-domain orchestration to drive continued innovation and advancement in network architecture. By embracing these future directions, organizations can navigate the evolving landscape of networking technology and unlock new opportunities for network optimization and transformation.

And despite its advantages, our Enhanced SDN and NFV Routing method has some limitations:

- Control Message Overhead: The need for frequent control messages to update routing decisions can introduce additional overhead, which might impact performance in very large or highly dynamic networks.
- Complexity: The algorithm's complexity due to multiple parameter considerations may result in higher computational requirements, potentially affecting the scalability of the solution.
- Dependency on Accurate Metrics: The method relies heavily on the accuracy of distance, memory, processing power, and load metrics. Any inaccuracies in these measurements can affect the overall performance of the routing decisions.

ACKNOWLEDGMENT

I would like to express my sincere gratitude to SAADI Chaimae & CHAOUI Habiba for their guidance and support throughout this research endeavor. Special thanks to University Ibn Tofail for providing resources. Lastly, I appreciate the encouragement from my friends and family.

REFERENCES

[1] Bonfim, M.S., Dias, K.L., Fernandes, S.F. (2019). Integrated NFV/SDN architectures: A systematic literature review. ACM Computing Surveys (CSUR), 51(6): 1-39. https://doi.org/10.1145/3172866

[2] Matias, J., Garay, J., Toledo, N., Unzilla, J., Jacob, E. (2015). Toward an SDN-enabled NFV architecture. IEEE Communications Magazine, 53(4): 187-193. https://doi.org/10.1109/MCOM.2015.7081093

[3] Chica, J.C.C., Imbachi, J.C., Vega, J.F.B. (2020). Security in SDN: A comprehensive survey. Journal of Network and Computer Applications, 159: 102595. https://doi.org/10.1016/j.jnca.2020.102595

[4] Adamuz-Hinojosa, O., Ordonez-Lucena, J., Ameigeiras, P., Ramos-Munoz, J.J., Lopez, D., & Folgueira, J. (2018). Automated network service scaling in NFV: Concepts, mechanisms and scaling workflow. IEEE Communications Magazine, 56(7): 162-169.

[5] Jaadouni, H., Saadi, C., Chaoui, H. (2022). Performance of OpenFlow-based SDN on IoT network. IET Conference Proceedings, 2022: 146-153. https://doi.org/10.1049/icp.2022.2431

[6] El Rajab, M., Yang, L., Shami, A. (2024). Zero-touch networks: Towards next-generation network automation. Computer Networks, 243: 110294. https://doi.org/10.1016/j.comnet.2024.110294

[7] Almutairi, H., Zhang, N. (2024). A survey on routing solutions for low-power and lossy networks: Toward a reliable path-finding approach. Network, 4(1): 1-32. https://doi.org/10.3390/network4010001

[8] Hatim, J., Chaimae, S., Habiba, C. (2022). Improved IOT/SDN architecture with the concept of NFV. In International Conference on Digital Technologies and Applications, pp. 294-301. https://doi.org/10.1007/978-3-031-01942-5_29

[9] Ebadinezhad, S., Bayemi, P.F.N. (2025). SDN and NFV security challenges and solutions for minimizing failures in IoT networks: Literature review. Research Advances in Network Technologies, 22-40.

[10] Shayegan, M.J., Damghanian, A. (2024). A method for DDoS attacks prevention using SDN and NFV. IEEE Access, 12: 108176-108184. https://doi.org/10.1109/ACCESS.2024.3438538

[11] Thirupathi, V., Sandeep, C.H., Kumar, N., Kumar, P.P. (2019). A comprehensive review on SDN architecture, applications and major benifits of SDN. International Journal of Advanced Science and Technology, 28(20): 607-614.

[12] Bholebawa, I.Z., Jha, R.K., Dalal, U.D. (2016). Performance analysis of proposed network architecture: OpenFlow vs. traditional network. International Journal of Computer Science and Information Security, 14(3): 30.

[13] Haji, S.H., Zeebaree, S.R., Saeed, R.H., Ameen, S.Y., Shukur, H.M., Omar, N., et al. (2021). Comparison of software defined networking with traditional networking. Asian Journal of Research in Computer Science, 9(2): 1-18.

[14] Khorsandroo, S., Sánchez, A.G., Tosun, A.S., Arco, J.M., Doriguzzi-Corin, R. (2021). Hybrid SDN evolution: A comprehensive survey of the state-of-the-art. Computer Networks, 192: 107981. https://doi.org/10.1016/j.comnet.2021.107981

[15] Sinha, Y., Haribabu, K. (2017). A survey: Hybrid SDN. Journal of Network and Computer Applications, 100: 35-55. https:/doi.org/10.1016/j.jnca.2017.10.003

[16] Lee, A., Wang, X., Nguyen, H., Ra, I. (2018). A hybrid software defined networking architecture for next-generation IoTs. KSII Transactions on Internet & Information Systems, 12(2): 932-945. https://doi.org/10.3837/tiis.2018.02.024

[17] Leivadeas, A., Falkner, M. (2022). A survey on intent-based networking. IEEE Communications Surveys & Tutorials, 25(1): 625-655. https://doi.org/10.1109/COMST.2022.3215919

[18] Velasco, L., Signorelli, M., De Dios, O.G., Papagianni, C., Bifulco, R., Olmos, J.J.V., Pryor, S., Carrozzo, G., Schulz-Zander, J., Bennis, M., Martinez, R., Cugini, F., Salvadori, C., Lefebvre, V., Valcarenghi, L., Ruiz, M. (2021). End-to-end intent-based networking. IEEE Communications Magazine, 59(10): 106-112. https://doi.org/10.1109/MCOM.101.2100141

[19] Zeydan, E., Turk, Y. (2020). Recent advances in intent-based networking: A survey. In 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), pp. 1-5. https://doi.org/10.1109/VTC2020-Spring48590.2020.9128422

[20] Herrera, J.G., Botero, J.F. (2016). Resource allocation in NFV: A comprehensive survey. IEEE Transactions on Network and Service Management, 13(3): 518-532. https://doi.org/10.1109/TNSM.2016.2598420

[21] Hatim, J., Chaimae, S., Habiba, C. (2023). SDN/NFV security challenges and proposed architecture. In 2023 7th IEEE Congress on Information Science and Technology (CiSt), pp. 145-149. https://doi.org/10.1109/CiSt56084.2023.10409955

[22] Bhamare, D., Samaka, M., Erbad, A., Jain, R., Gupta, L., Chan, H.A. (2017). Optimal virtual network function placement in multi-cloud service function chaining architecture. Computer Communications, 102: 1-16. https://doi.org/10.1016/j.comcom.2017.02.011

[23] Schardong, F., Nunes, I., Schaeffer-Filho, A. (2021). NFV resource allocation: A systematic review and taxonomy of VNF forwarding graph embedding. Computer Networks, 185: 107726. https://doi.org/10.1016/j.comnet.2020.107726

[24] Wang, B., Li, J., Cao, S., Guler, E., Zheng, D. (2024). Security-aware service function chaining and embedding with asymmetric dedicated protection. IEEE Access, 12: 53944-53957. https://doi.org/10.1109/ACCESS.2024.3387083

[25] Doriguzzi-Corin, R., Siracusano, G., Bertino, E. (2018). Characterizing DDoS attacks and their impact on the SDN control plane. In 2018 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 898-899.

[26] Scott-Hayward, S., Natarajan, S., Sezer, S. (2015). A survey of security in software defined networks. IEEE Communications Surveys & Tutorials, 18(1): 623-654. https://doi.org/10.1109/COMST.2015.2453114

[27] Xia, W., Wen, Y., Foh, C.H., Niyato, D., Xie, H. (2015). A survey on software-defined networking. IEEE Communications Surveys & Tutorials, 17(1): 27-51. https://doi.org/10.1109/COMST.2014.2330903

[28] Benzaid, C., Taleb, T. (2020). AI-driven zero touch network and service management in 5G and beyond: Challenges and research directions. Ieee Network, 34(2): 186-194. https://doi.org/10.1109/MNET.001.1900252

[29] Anwer, M.B., Benson, T., Feamster, N., Levin, D. (2015). A slice of control: Managing the interaction between SDN and cloud providers. Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies, pp. 1-13.

[30] Sezer, S., Scott-Hayward, S., Chouhan, P.K., Fraser, B., Lake, D., Finnegan, J., Viljoen, N., Miller, N., Rao, S. (2013). Are we ready for SDN? Implementation challenges for software-defined networks. IEEE Communications Magazine, 51(7): 36-43. https://doi.org/10.1109/MCOM.2013.6553676

[31] Priyadarshi, R. (2024). Energy-efficient routing in wireless sensor networks: A meta-heuristic and artificial intelligence-based approach: A comprehensive review. Archives of Computational Methods in Engineering, 31(4): 2109-2137. https://doi.org/10.1007/s11831-023-10039-6

[32] Voyer, D., Filsfils, C., Parekh, R., Bidgoli, H., Zhang, Z. (2024). RFC 9524 segment routing replication for multipoint service delivery. https://doi.org/10.17487/RFC9524

[33] Ginsberg, L., Previdi, S., Wu, Q., Tantsura, J., Filsfils, C. (2019). BGP-link state (BGP-LS) advertisement of IGP traffic engineering performance metric extensions (No. rfc8571).

[34] Boldrini, L., Bachiddu, M., Koning, R., Grosso, P. (2024). User controlled routing exploiting pceps and inter-domain label switched paths. In Proceedings of the Second International Conference on Advances in Computing Research, pp. 465-478. https://doi.org/10.1007/978-3-031-56950-0_39

[35] Mine, G., Hai, J., Jin, L., Huiying, Z. (2020). A design of SD-WAN-oriented wide area network access. In 2020 International Conference on Computer Communication and Network Security (CCNS), pp. 174-177. https://doi.org/10.1109/CCNS50731.2020.00046