Ministry of Higher Education and scientific Research.

SAAD DAHLAB University -1-

Faculty of sciences

Department of Computer Science

To obtain the diploma:

Master's degree in computer science

Option: **Computer Systems and Networks**

Theme:

Optimizing Network performance in a multi-transport environment using SD-WAN

Host Organization: The general directorate of OOREDOO Algeria

Prepared by: SALHI Manal

Supervised by: MR. BENYAHIA Mohamed

MR. ZEGGARI Mohamed El Amine

Academic Year : **2024/2025**

Abstract :

With the constant evolution of connectivity demands and the increasing intricacy of network infrastructures, many organizations are re-evaluating the effectiveness of conventional network models. In this context, SD-WAN has emerged as a promising alternative, offering smarter traffic handling, improved scalability across multiple sites, and better overall network efficiency.

This project, conducted in collaboration with the company Ooredoo, explores and contrasts two different networking approaches: the conventional network setup relying on standard routing protocols, and the SD-WAN model, which is designed for more adaptable and centralized traffic control. The evaluation takes place in a complex network setting, involving various connection types and hardware components.

Our goal is to assess how these two network designs perform in terms of speed, security, adaptability, and ease of administration. Based on this analysis, we aim to offer practical insights that may assist in deciding whether a shift towards SD-WAN can better support modern enterprise connectivity needs.

Keywords: SD-WAN, traditional architecture, network optimization, Ooredoo, WAN, heterogeneous infrastructure.

Résumé:

Face à l'évolution constante des besoins en connectivité et à la complexité croissante des infrastructures réseau, de nombreuses organisations réévaluent l'efficacité des modèles de réseau conventionnels. Dans ce contexte, le SD-WAN s'est imposé comme une alternative prometteuse, offrant une gestion plus intelligente du trafic, une évolutivité accrue sur plusieurs sites et une meilleure efficacité globale du réseau.

Ce projet, mené en collaboration avec la société Ooredoo, explore et compare deux approches réseau : la configuration réseau conventionnelle, basée sur des protocoles de routage standard, et le modèle SD-WAN, conçu pour un contrôle du trafic plus adaptable et centralisé. L'évaluation se déroule dans un environnement réseau complexe, impliquant différents types de connexions et composants matériels.

Notre objectif est d'évaluer les performances de ces deux conceptions réseau en termes de vitesse, de sécurité, d'adaptabilité et de simplicité d'administration. Sur la base de cette analyse, nous souhaitons proposer des éclairages pratiques pour déterminer si une transition vers le SD-WAN peut mieux répondre aux besoins de connectivité des entreprises modernes.

Mots-clés : SD-WAN, architecture traditionnelle, optimisation réseau, Ooredoo, WAN, infrastructure hétérogène.

ملخص :

في ظلّ احتياجات الاتصال المتطورة باستمرار والتعقيد المتزايد للبنى التحتية للشبكات، تُعيد العديد من المؤسسات تقييم فعالية تصاميم الشبكات التقليدية. في هذا السياق، برزت شبكات SD-WAN كبديل واعد، إذ تُوفّر إدارةً أذكى لحركة البيانات، وقابليةً أكبر للتوسّع عبر مواقع متعددة، وكفاءةً إجماليةً مُحسّنةً للشبكة.

يستكشف هذا المشروع، الذي أُجري بالتعاون مع Ooredoo ويُقارن بين منهجين للشبكات: تكوين الشبكة التقليدي، القائم على بروتوكولات التوجيه القياسية، ونموذجSD-WAN ، المُصمّم للتحكم في حركة البيانات بشكل أكثر مرونةً ومركزيةً. يُجرى التقييم في بيئة شبكة مُعقّدة، تتضمن أنواعًا مُختلفة من الاتصالات ومكونات الأجهزة.

هدفنا هو تقييم أداء هذين التصميمين للشبكات من حيث السرعة والأمان والقدرة على التكيف وسهولة الإدارة. بناءً على هذا التحليل، نهدف إلى تقديم رؤى عملية لتحديد ما إذا كان الانتقال إلى SD-WAN يُمكن أن يُلبّي احتياجات الاتصال للشركات الحديثة بشكل أفضل.

الكلمات المفتاحية: SD-WAN، البنية التحتية التقليدية، تحسين الشبكة، Ooredoo، WAN، البنية التحتية غير المتجانسة.

Acknowledgement

First and foremost, I am profoundly grateful to* ***ALLAH* *for His countless blessings and for granting me the strength, determination, and perseverance to bring this work to fruition.*

I extend my heartfelt thanks to my supervisors,* ***Mr. Mohamed Benyahia* *and* ***Mr. Zeggari Mohamed El Amine****, for their invaluable guidance and support throughout the course of this project. Their expertise, encouragement, and insightful advice were fundamental to the successful completion of this work.*

I would also like to sincerely thank the* ***members of the jury* *for the honor of evaluating my work. Their constructive feedback and professional insight will undoubtedly contribute to the enrichment and refinement of this thesis.*

Finally, I wish to express my deep appreciation to* ***all those who supported me**, whether directly or indirectly, during the preparation and realization of this work.*

Dedications

I dedicate this modest work to:

****To my dear mother**, your love, your silent strength, and your unwavering support have carried me throughout this journey. No words can express the infinite gratitude I feel towards you. You are, and will remain, my pillar of strength.*

****To my sister and my dear besties**, who were there in times of doubt as well as in times of joy, with reassuring words, sincere smiles, and a precious presence. Thank you for your energy, your attentiveness, and your kindness.*

****To those two mates**, with whom I shared endless discussions, support, and laughter. Thank you for always being there in your own way — your presence made a real difference.*

****To the CDEC club and team**, which played a meaningful role in my personal growth. Being part of such a dynamic and inspiring community helped me discover myself, build confidence, and develop valuable soft skills that shaped me far beyond academics.*

****To Ooredoo team**, for welcoming me and offering me a concrete insight into the professional world. This internship was a significant step in my career, short but enriching.*

****To all my friends,** without exception, who have always encouraged me with kindness, patience, and humor. Your presence, even discreet, has meant a lot to me.*

****And finally, to the little girl I used to be** — curious, anxious, full of dreams, and sometimes full of doubt. I hope you're proud of how far we've come. This is for you.*

Table of contents :

List of figures

List of tables

General Introduction

In today’s rapidly evolving business landscape, organizations face increasing pressure to optimize their networks and improve connectivity between their geographically distributed branches. Traditional network architectures often struggle to meet these demands, pushing companies to explore innovative solutions like SD-WAN, which offer enhanced performance, flexibility, and simplified management.

This project, carried out within Ooredoo, focuses on comparing two network approaches: the conventional network model and the emerging SD-WAN technology. Instead of directly simulating Ooredoo’s network, we designed a random heterogeneous network environment to analyze how each solution performs in various real-world scenarios, utilizing diverse connections and devices.

The primary goal of this research is to evaluate essential factors such as security, performance, centralization, and flexibility, with a focus on optimizing inter-site communication. Through this study, we aim to offer actionable insights for businesses considering a shift to SD-WAN, helping them make informed decisions to improve their network efficiency and adapt to modern connectivity challenges

Chapter 1: State of art

1-Presentation of the host organization : Ooredoo Algeria

Ooredoo, A key player in telecommunications and socioeconomic development. Founded in August 2004, Ooredoo Algeria quickly established itself as a key leader in telecommunications, particularly among young Algerians, thanks to a bold vision and a solid national network.

An innovator, Ooredoo was a pioneer in the launch of 3G in 2013 and 4G in 2016, now covering 90% of the Algerian population.

With a dense sales network (more than 250 points of sale across the country), ambitious projects such as the deployment of 5G, and actions promoting sustainability, Ooredoo continues to play a central role in Algeria's digital and social transformation, combining innovation, excellence, and social responsibility. [6]

1.1- Company's organization :

Ooredoo Algeria adopts a geographical structure spread throughout the country to ensure the availability and quality of its services everywhere .

The company is structured around its headquarters in Ouled Fayet, Algiers, which serves as the planning and decision-making center, as well as a sub-directorate in Bab Ezzouar. These two levels house general management, technology and engineering departments, and the network monitoring and infrastructure development teams.. In addition to the 2 others operational directorates, one in the east at Constantine, and the other at Oran in the west.

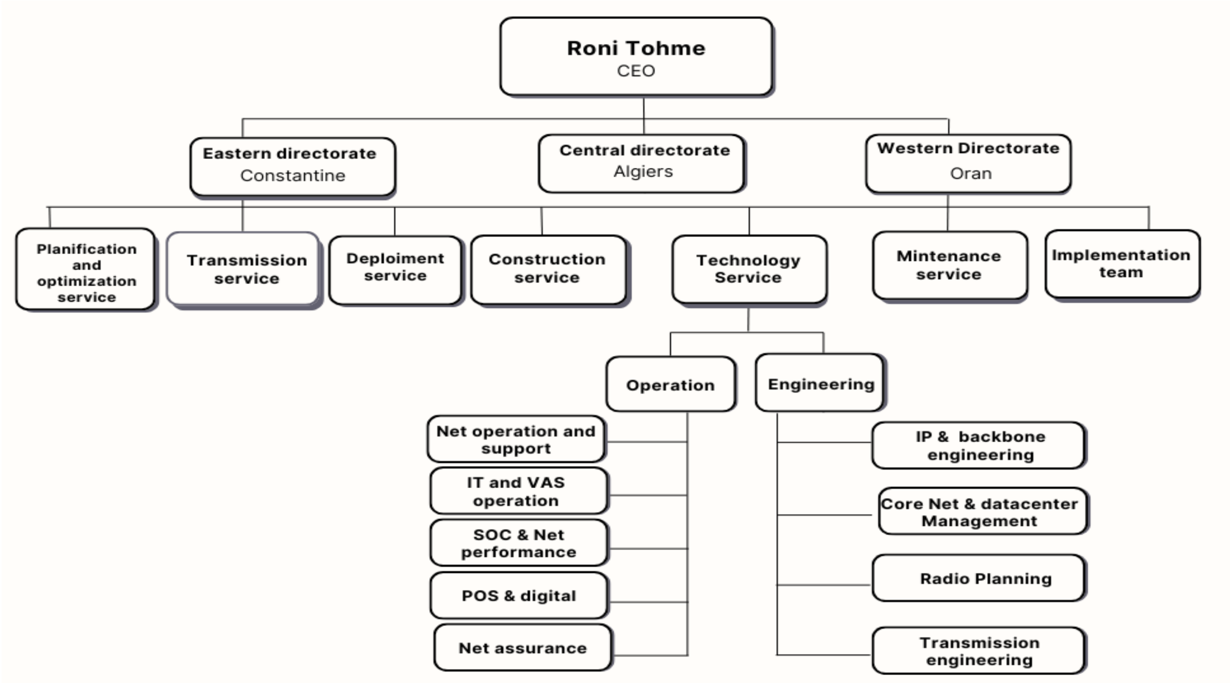


Figure 1.1- Structure diagram of Ooredoo 1

Ooredoo currently uses a traditional WAN architecture to interconnect its three main directorates and additional remote sites. This setup is based on two transport types**:** MPLS**,** used as the primary link, and 4G LTE as a backup**.**

MPLS provides reliable, high-performance connectivity for critical services such as voice, video, and business applications. In the event of an MPLS failure, traffic is rerouted through the 4G link, which ensures basic continuity but is not optimized for high-demand traffic.

1.2- The company's main mission :

Ooredoo Algeria's main missions are :

-To provide high-quality telecommunications services for individuals and businesses, covering voice, data, internet connectivity, and value-added services.

-To support the digital transformation in Algeria by developing accessible, efficient, and scalable digital services.

-To support businesses in their digital transition by offering customized solutions in connectivity, cybersecurity, cloud computing, and managed services.

-To promote digital inclusion and sustainable development by investing in projects with a strong societal and environmental impact

2- General context of multi-transport Wan :

A multi-transport WAN refers to an infrastructure that leverages multiple types of physical and logical network connections; including but not limited to MPLS , broadband Internet, LTE/4G/5G, and fiber-optic links; to interconnect various organizational sites, data centers, and cloud resources.

This architecture is particularly relevant in an environment where organizations are expanding geographically and increasingly adopting cloud-based services such as SaaS, IaaS, and PaaS. By utilizing diverse transport mechanisms, enterprises can ensure redundancy, load balancing, cost efficiency, and improved fault tolerance. Each transport type offers distinct characteristics in terms of latency, bandwidth, reliability, and cost, making their strategic combination a key enabler of resilient and adaptable networks. [1]

3- Limitations of traditional multi-transport networks :

However, as the number of connection types increases, so does the complexity of managing traffic flows and ensuring consistent QoS across all links. Traditional WAN architectures, often built around MPLS backbones, are being challenged to meet the agility, visibility, and automation requirements of modern business operations. This context underscores the need for more intelligent network management solutions capable of dynamically adapting to evolving demands.

Despite offering diversified connectivity, traditional multi-transport WAN architectures are increasingly viewed as inadequate in addressing the needs of modern enterprises. These limitations stem primarily from their rigid and hardware-centric design, as well as the lack of centralized visibility and control.

One of the most significant drawbacks is the reliance on static routing or conventional dynamic protocols (e.g., OSPF, BGP) that do not account for real-time network conditions or application requirements. Traffic engineering is typically complex and inflexible, making it difficult to optimize performance across multiple links dynamically. Furthermore, traditional WANs lack application-level awareness, meaning they are unable to prioritize critical business applications over less important traffic without extensive manual configuration.

Security is another concern. Traditional architectures often require the backhauling of traffic to centralized data centers for inspection and enforcement of security policies, which introduces additional latency and degrades performance, especially for cloud-based applications.

Operationally, the management of legacy WAN infrastructures is labor-intensive and time-consuming. Any modification—such as adding a new branch or adjusting routing policies—typically requires manual intervention across multiple devices, increasing the risk of human error and configuration inconsistencies. Additionally, the cost of MPLS circuits remains high compared to broadband alternatives, placing financial strain on enterprises seeking to expand or upgrade their networks.

These challenges collectively highlight the structural and functional inefficiencies of traditional multi-transport WANs and serve as a driving force for the adoption of more adaptive and software-driven solutions, such as SD-WAN. [2]

4- Problem statement: why migrate to SD-WAN?

Ooredoo currently operates a traditional WAN architecture to connect its three main directorates, using MPLS as the primary transport and 4G as a backup lin. This model has long ensured network stability and availability. However it is now showing its limitations in the face of the growing demands of modern applications, cloud connectivity, and cost-reduction imperatives.The existing infrastructure lacks dynamic traffic management, centralized control, and the ability to efficiently utilize multiple transport links. Failover mechanisms are basic, and performance optimization is limited, especially for latency-sensitive service like voice and video.

In the light of these constraints, Ooredoo is considering a migration to SD-WAN as a more flexible, scalable, and cost-effective solution.

5- Objectives of the comparative study

This study aims to compare traditional multi-transport WAN and SD-WAN architectures based on four key criteria:

- **Performance:** Evaluate latency, throughput, and application handling.
- **Cost:** Analyze infrastructure and operational expenses.
- **Flexibility:** Assess scalability and ease of network management.

The goal is to determine which approach better supports modern enterprise connectivity requirements.

Chapter 2: Traditional Network Architecture vs. SD-WAN

1- Traditional Multi-transport Network

1.1-Description of the classic architecture (MPLS, 4G)

This section focuses on the diverse transport mechanisms traditionally employed in multi-site WAN environments. It offers a detailed technical analysis of the core transport technologies that make up traditional WAN backbones:

- **Multiprotocol Label Switching (MPLS):** A high-performance protocol that directs data along predetermined paths using labels rather than IP addresses. Widely adopted for its support of Quality of Service (QoS), reliability, and service-level agreements (SLAs), MPLS has been the cornerstone of many enterprise WAN deployments. [3]
- **4G/LTE Networks:** Used primarily for backup or temporary connectivity, 4G/LTE provides wireless access that can supplement or replace wired links in remote or mobile settings. While flexible and quick to deploy, such connections often face performance inconsistencies and data cap limitations. [3]
- **Fiber Optic Lines:** Known for their high bandwidth and low latency, fiber connections are essential for interconnecting data centers or supporting bandwidth-intensive applications. However, deployment can be cost-prohibitive and time-consuming, especially in underdeveloped regions. [3]

1.2- Traditional routing methods (static, dynamic)

Routing is the fundamental mechanism that determines how data traverses a WAN. In traditional architectures, routing intelligence is typically embedded within edge devices—such as branch routers—and configured using either static or dynamic methods. This section offers a thorough examination of both strategies:

- **Static Routing:** Static routes are manually defined by network administrators and do not change unless explicitly modified. While this approach is simple and predictable, it lacks adaptability and fault tolerance. If a link fails, traffic must be rerouted manually or through additional scripts, which can delay recovery and increase operational overhead. [4]
- **Dynamic Routing Protocols:** Protocols like Open Shortest Path First **(OSPF)**, Enhanced Interior Gateway Routing Protocol **(EIGRP)**, and Border Gateway Protocol **(BGP)** introduce automation into routing by allowing devices to exchange information about network topology and choose optimal paths based on metrics such as bandwidth, delay, or hop count. These protocols enable route recalculation during link failures or congestion, providing a level of responsiveness absent in static configurations. [4]

Despite the advantages of dynamic routing, traditional WANs still struggle to meet modern demands. The inability to prioritize traffic based on application type, user identity, or real-time performance metrics limits the effectiveness of traditional routing methods. Moreover, the absence of centralized policy management makes it difficult to coordinate security, quality, and compliance controls across multiple sites.

2- SD-WAN Networks

2.1- SD-WAN Definition:

A Software-Defined Wide Area Network (SD-WAN) is a virtual WAN architecture that allows enterprises to securely connect users to applications using any combination of transport services, including MPLS, LTE, and broadband internet services. [5]

Software-Defined WAN provides the advantages typically associated with Software-Defined Networking (SDN) in data centers but for wide area network solutions for enterprise branch offices. Both SDN and SD-WAN virtualize resources to provide accelerated services delivery, better performance and improved availability by automating network deployment and management while reducing the total cost of ownership. [7]

2.2- Components of the SD-WAN architecture

Modern SD-WAN solutions are architected with a clear separation between the control plane and the data plane. This division enables centralized intelligence and management, while maintaining distributed and hight-performance traffic forwarding. This model improves scalability, agility, and simplifies operations. [8]

2.2.1- Control Plane

The control plane is responsible for centralized orchestration, policy management, and network-wide visibility. It defines how the traffic should flow across the network without directly handling data forwarding. Its key Components [8]:

- **SD-WAN Orchestrator (Central Controller):**
  Acts as the centralized brain of the SD-WAN solution. It manages:
  - Configuration and policy deployment;
  - Network-wide monitoring and automation;
  - Zero-touch provisioning and topology control.
- **Policy and Routing Logic:**
  Hosted within the controller, this defines:
  - Routing decisions based on performance and application type;
  - Security and segmentation policies for consistent enforcement across all sites.

2.2.2- Data Plane

The data plane is in charge of forwarding the actual user traffic between endpoints, based on policies defined in the control plane. Its key Components [9] :

- **Edge Devices (SD-WAN Appliances):**
  Deployed at branches and data centers, these:

- Enforce routing and QoS policies;
    - Establish secure IPsec tunnels;
    - Select optimal paths based on real-time link performance.
- **Overlay Network:**
  A virtual network formed by encrypted tunnels (e.g., IPsec), allowing:
    - Abstracted routing over various underlay transports (MPLS, Internet, LTE);
    - Application-aware routing and traffic prioritization.
- **Performance Monitoring:**
  Integrated into edge devices to measure metrics like latency, jitter, and packet loss, which are fed back into routing decisions.

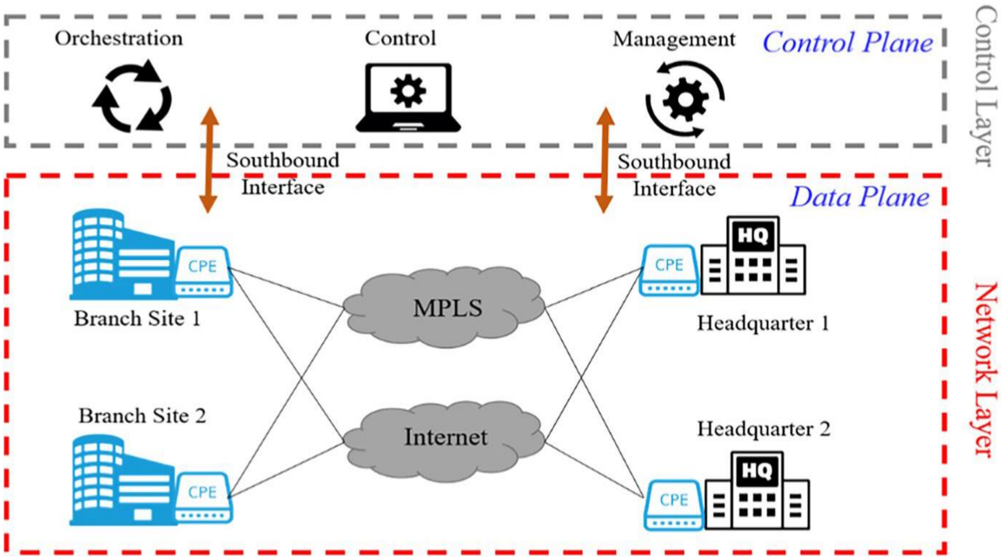


Figure 2.1- SD-WAN architecture 1 [10]

2.2.3- Transport (Underlay) Networks

Though not part of SD-WAN logic, underlay networks—such as MPLS, broadband Internet, and LTE—form the physical infrastructure. SD-WAN overlays dynamically leverage these links based on predefined SLAs and real-time conditions.

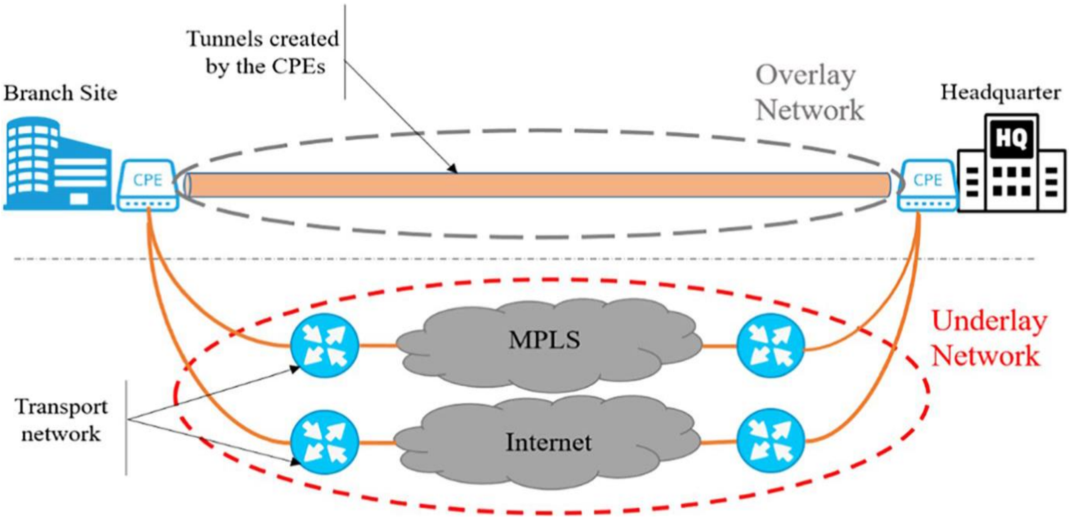


Figure 2.2- Physical and virtual network decomposition in SD-WAN 1 [10]

2.3- Different SD-WAN on the international market:

The Gartner Magic Quadrant for SD-WAN as of September 2023 highlights Fortinet as the clear leader, positioned highest in both "Ability to Execute" and "Completeness of Vision." Cisco and Broadcom (VMware) also appear in the Leaders quadrant, though slightly behind Fortinet. Companies like Palo Alto Networks, Versa Networks, and HPE are positioned as strong contenders with robust visions but slightly lower execution. Juniper Networks is classified as a Visionary, showing innovation yet lacking in execution compared to the leaders. Huawei stands alone in the Challengers quadrant, indicating strong execution but a less complete vision. Niche Players such as Peplink, Ericsson (Cradlepoint), and Barracuda reflect more limited market impact and strategic vision in the SD-WAN space. This chart provides a strategic snapshot of the competitive SD-WAN vendor landscape.

Figure 2.3- Gartner's magic quadrant for SD-WAN vendors – September 2024 1 [11]

2.3.1- Which SD-WAN technology to use? :

Feature	Cisco SD-WAN	Fortinet SD-WAN
Core platform	Viptela (integrated with IOS-XE)	Integrated into FortiGate new generation firewall (NGFW)
Components required	vManage, vSmart, vBond, vEdge/cEdge, optional: Umbrella/Duo	FortiGate , optional: FortiManager, FortiAnalyzer
Security integration	Requires separate integration (Umbrella, Firepower)	Natively built-in with FortiOS
Management interface	vManage	FortiManager/ FortiGate GUI/ FortiOS
Routing support	Advanced, enterprise-grade (BGP, OSPF, EIGRP)	Full routing support with simpler configuration
SD-WAN policies	Template-driven in vManage; flexible but complex	GUI or CLI-based; easier for smaller team
Scalability	Excellent; built for large-	String for both SMBs and

	scale networks	large distributed entreprises
Licensing model	Multi-tier (DNA, vBond, vManage, etc)	More unified license per FortiGate unit
Cost	Higher TCO with multiple components and licenses	Lower TCO with bundled features
Deployment Complexity	High, requires multiple controllers, certificates, config	Low to Medium, simpler with FortiGate's all-in-one design
Ideal use case	Large enterprises with deep Cisco investment & skilled teams	Distributed networks needing security+SD-WAN simplicity

Table 2.1- Comparison table for Fortinet and Cisco SD-WAN 1

After careful evaluation, we chose Fortinet Secure SD-WAN for our project due to its seamless integration of networking and security features within a single platform. Unlike Cisco SD-WAN, which requires multiple components such as vManage, vSmart, vBond, and additional security tools, Fortinet provides a consolidated solution through FortiGate appliances, simplifying both deployment and ongoing management. We were particularly drawn to the ease of configuration, the built-in next-generation firewall and ZTNA capabilities, and the ability to manage everything through a unified interface. This not only reduced complexity and deployment time but also lowered the overall cost of ownership. Fortinet's approach aligned perfectly with our need for a secure, scalable, and easy-to-operate SD-WAN solution, especially for a distributed environment where simplicity and strong security are critical.

2.4- Operation of SD-WAN

In this subsection, we explore how SD-WAN operates from a high-level perspective, focusing on the principles that drive its functionality and decision-making processes. Unlike traditional networks that rely on static configurations and decentralized control, SD-WAN leverages **centralized, software-driven control** for managing and optimizing network traffic. Key operational elements include [12] :

- **Dynamic Path Selection:** SD-WAN uses real-time performance metrics such as bandwidth, latency, jitter, and packet loss to intelligently route traffic across the most suitable path. This decision-making process is automated, allowing the network to adjust dynamically as conditions change, such as during link failure, congestion, or fluctuating traffic demands.
- **Application-Aware Traffic Steering:** One of the standout features of SD-WAN is its ability to prioritize network traffic based on application requirements. It ensures that high-priority applications, such as VoIP or video conferencing, receive preferential treatment over less critical traffic. This is achieved through deep packet inspection (DPI) and integrated traffic management algorithms that allow the SD-WAN to understand the nature of traffic at the application level, enabling more granular control.
- **Centralized Policy Management:** The SD-WAN orchestrator centralizes the creation and enforcement of policies for the entire network. Administrators define policies based on business needs, performance requirements, and security protocols. These policies are

automatically applied across all connected edge devices, ensuring consistency and ease of management across all sites, regardless of geographic location.

- **Security and Encryption:** SD-WAN incorporates robust security measures such as end-to-end encryption, secure tunneling protocols, and firewall protections to safeguard data as it traverses potentially unsecured public or private networks. Unlike traditional WANs, where security is often bolted onto the network as an afterthought, SD-WAN integrates security features into its core architecture, ensuring that secure connectivity is inherent in the network design.

Through these principles, SD-WAN operates in a highly flexible, efficient, and secure manner, drastically reducing manual intervention and offering a level of agility and control that is often unattainable in conventional WAN architectures.

2.5- Advantages compared to classic topologies

This section outlines the advantages of SD-WAN compared to traditional multi-transport WANs, emphasizing its ability to address the inherent limitations of conventional network models. The benefits of SD-WAN are especially relevant in the context of modern enterprises, where cloud computing, mobile workforces, and high-bandwidth applications are increasingly common. [2]

- **Improved Agility and Flexibility:** SD-WAN allows for the rapid deployment of new sites, services, or applications, often without the need for new hardware. Network configurations and policies can be changed on-the-fly via the centralized orchestrator, which provides a level of operational flexibility that traditional networks simply cannot match. This is particularly beneficial in organizations with rapidly changing or expanding network needs.
- **Reduced Operational Costs:** By leveraging low-cost broadband Internet connections, SD-WAN enables businesses to reduce their reliance on expensive MPLS circuits. The ability to mix multiple transport types—such as broadband, LTE, and MPLS—provides cost-effective solutions for geographically distributed sites. Additionally, SD-WAN's centralized management model reduces the administrative burden, further lowering operational costs.
- **Enhanced Performance and Reliability:** SD-WAN improves network performance through dynamic path selection and real-time traffic optimization, ensuring that high-priority traffic consistently receives the necessary resources. In comparison, traditional WAN architectures often struggle with inefficient traffic routing, leading to performance bottlenecks and poor user experiences.
- **Centralized Security:** Unlike traditional WANs, where security often requires separate configurations at each site, SD-WAN integrates security directly into the architecture. Features such as end-to-end encryption, secure tunneling, and built-in firewalls ensure that security is managed centrally and uniformly across all network endpoints.
- **Simplified Network Management:** With SD-WAN's centralized management and policy-based control, administrators can configure, monitor, and manage network performance from a single platform. This significantly reduces the complexity of managing a multi-site WAN compared to traditional networks, where each site may require separate configuration and troubleshooting.

This comparative analysis highlights SD-WAN as a more efficient, cost-effective, and future-proof solution for modern enterprises, offering greater flexibility, improved performance, and stronger security than traditional network architectures.

3-Conclusion:

By contrasting traditional multi-transport WANs with SD-WAN architectures, it becomes evident that while legacy models provide stability through MPLS and static or dynamic routing, they fall short in adaptability and visibility. SD-WAN, through its centralized intelligence, dynamic routing capabilities, and link-aware decision-making, offers a more responsive and efficient alternative. The selection of Fortinet as the SD-WAN solution in this study reflects its strong integration features and market reliability, preparing the ground for practical validation through simulation.

Chapter 3: Methodology and simulation scenarios

1- Simulation environment

1.1- Presentation of the *EVE-NG* platform

EVE-NG (Emulated Virtual Environment – Next Generation) est une plateforme avancée d'émulation de réseaux, utilisée massivement par les ingénieurs, chercheurs et étudiants pour simuler des architectures complexes en environnement entièrement virtualisé, sans matériel réel. [13]

EVE-NG operates as a centralized virtual machine that runs on a hypervisor (such as VMware Workstation, ESXi, or other virtualization solutions). Its architecture allows users to interconnect various virtual network appliances—routers, firewalls, switches, and end devices—within a graphical topology builder that can be accessed via any modern web browser. [14]

One of the key strengths of EVE-NG lies in its multi-vendor compatibility. It supports a wide range of images from network equipment vendors such as Cisco, Fortinet, Juniper, Palo Alto, and others. This flexibility makes it ideal for simulating heterogeneous network environments, such as hybrid WAN architectures involving SD-WAN overlays, IPsec tunnels, BGP routing, and transport-layer diversity (e.g., MPLS, LTE, broadband).

1.2- Software resources

In order to build and run the network simulation environment, two primary software resources were used:

- **EVE-NG Community Edition**
  The EVE-NG (Emulated Virtual Environment - Next Generation) Community Edition is a free and open-source network emulation platform. It provides a robust framework for designing and testing complex network topologies that include multiple vendors and protocols. It allows for interactive CLI-based configurations through a web browser, enabling real-time simulation of routing, switching, tunneling, and SD-WAN functionalities. Its scalability and vendor-neutral architecture make it ideal for both academic and professional experimentation.
- **VMware Workstation Community**
  VMware Workstation is a type 2 hypervisor that enables the creation and management of virtual machines on a single physical computer. In this project, it was used as the host environment for deploying the EVE-NG virtual machine. It provides reliable hardware-level virtualization, resource allocation (CPU, memory, network interfaces), and network adapter bridging, all of which are essential for ensuring the smooth operation of the emulated lab environment.

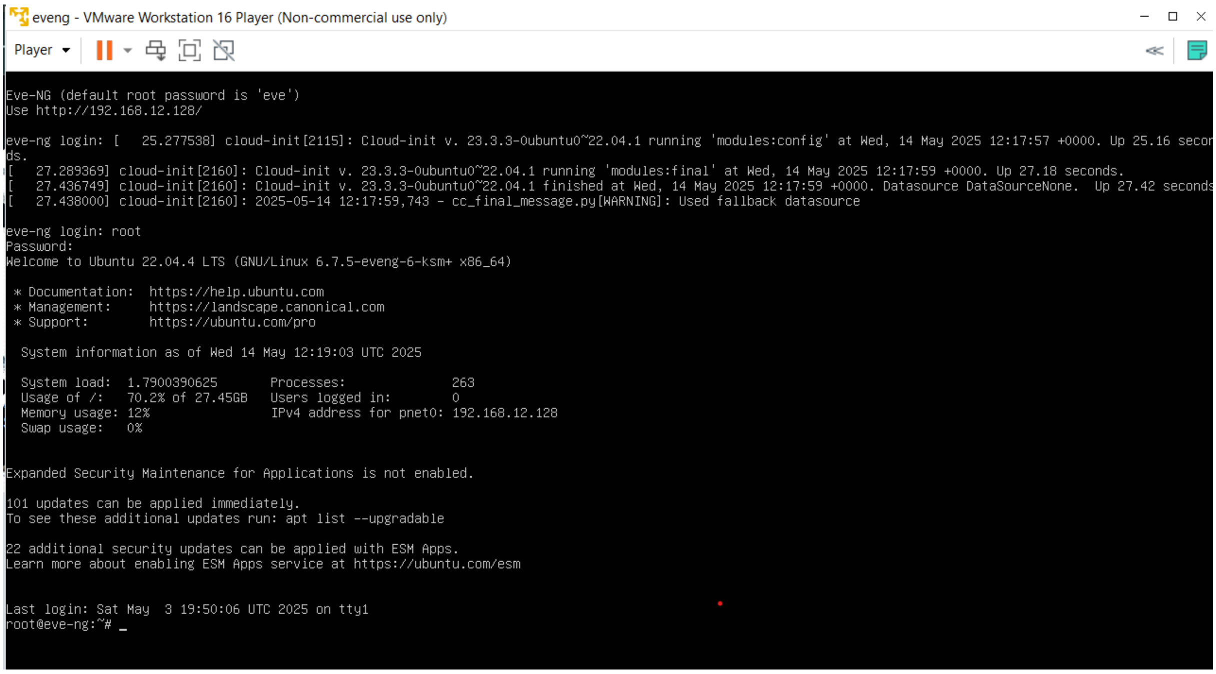


Figure 3.1 –Console of the EVE-NG Virtual machine running on VMware Workstation 1

1.3- Justification for choosing this tool

EVE-NG was chosen for this study due to its ability to integrate multi-vendor images within a single topology. Unlike vendor-specific simulators (e.g., Cisco Packet Tracer or Fortinet's FortiManager lab tools), EVE-NG supports both proprietary and open-source images, making it ideal for simulating hybrid environments such as multi-transport WANs. Furthermore, EVE-NG allows for advanced features such as IPsec tunneling, BGP peering, traffic analysis, and SD-WAN behavior—all within a scalable virtual framework. Its support for CLI-based and GUI-based configurations enhances its applicability for academic experimentation and real-world enterprise scenarios.

1.4- Virtual images used:

To implement our two topologies we are using the virtual images as follows:

Image	Description	File Name
	**Cisco vios router**	**vios-adventerprisek9-m.spa.159-3.m6.qcow2**
	**Cisco vios switch**	**i86bi-linux-l2-adventerprisek9-15.1a.bin**
	**Fortigate firewall VM64**	**FGT_VM64_KVM-v7.2.M-build2726-FORTINET.out.kvm**

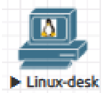
	**Virtual Pc**	**Eve-ng integrated**
	**Linux Netem virtual machine**	**eve-ng-addons-netem_2.0.3-61_amd64.deb**
	**Linux desktop**	**linux-ubuntu-18.04.06-desktop.tar.gz**

Table 3.1- virtual images used 1

2- Topology of the traditional network

2.1- Topology :

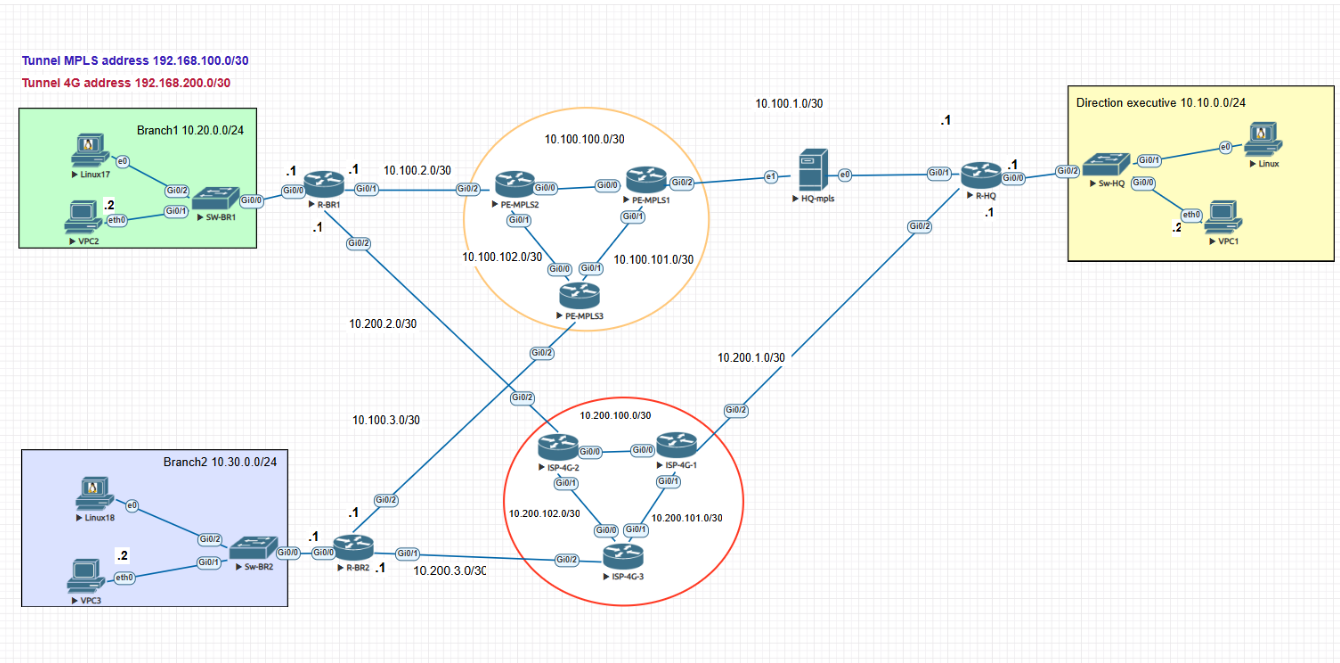


Figure 3.2- Traditional Multi-Transport topology with MPLS and 4G connectivity 1

2.1.1- Explication of the Topology

The displayed figure represents a traditional multi-transport WAN architecture used to interconnect Ooredoo's three main sites: the Head Office (HQ – 10.10.0.0/24), Branch 1 (10.20.0.0/24), and Branch 2 (10.30.0.0/24). This design incorporates two distinct WAN transports: **MPLS** as the primary connection (orange cloud) and **4G LTE** as the secondary backup (red cloud). Each site is equipped with a router (R-HQ, R-BR1, R-BR2), a Layer 2 switch, and a PC. The routers are dual-homed to both the MPLS and 4G infrastructures, using two separate IPsec VPN tunnels—one over MPLS (192.168.100.0/30) and the other over 4G (192.168.200.0/30)—to ensure secure inter-site communication.

The two main steps of implementation for this traditional multi-transport WAN topology are:

1. **Tunnel Configuration and Routing Setup**
   This step involves configuring two IPsec tunnels per site: one over the MPLS transport (primary) and one over the 4G transport (backup). Each tunnel is secured with crypto policies, with appropriate tunnel interfaces assigned IP addresses. Routing protocol is implemented to enable inter-site communication through these tunnels.
2. **Redundancy and Failover Mechanisms**
   IP SLA and tracking are configured to monitor the health of the MPLS tunnel. If the MPLS tunnel fails, the tracking mechanism triggers a failover to the 4G tunnel, ensuring continuous connectivity. This setup provides resilience and automatic backup switching in case of primary link degradation or outage.

2.2- Configuration :

2.2.1- Implementing the routing protocol:

R-HQ

```
router bgp 65000
 bgp log-neighbor-changes
 network 10.10.0.0 mask 255.255.255.0
 network 10.254.254.1 mask 255.255.255.255
 redistribute connected
 neighbor 10.100.1.2 remote-as 65010

 neighbor 10.100.1.2 allowas-in

 neighbor 10.200.1.2 remote-as 65020
 neighbor 10.200.1.2 allowas-in
```

Figure 3.3 - BGP protocol implementation 1

Figure analysis:

We enabled the BGP routing on the three edge routers with the same AS-number which is **65000**

The **"neighbor 10.X.X.2 allowas-in"** command allows this router's own AS-number to appear in the as-path because we're in an iBGP case between the 3 branches, where the BGP policy doesn't allow the bgp router to accept a prefix if it sees its own AS number in it, to avoid loops.

PE-MPLS1

```
!
router bgp 65010
 bgp log-neighbor-changes
 redistribute connected
 neighbor 10.100.1.1 remote-as 65000
 neighbor 10.100.100.2 remote-as 65010
 neighbor 10.100.101.2 remote-as 65010
```

Figure 3.4 - PE-MPLS BGP implementation 1

ISP-4G-1

```
!
router bgp 65020
 bgp log-neighbor-changes
 redistribute connected
 neighbor 10.200.1.1 remote-as 65000
 neighbor 10.200.100.2 remote-as 65020
 neighbor 10.200.101.2 remote-as 65020
!
ip forward-protocol nd
!
```

Figure 3.5 - ISP-4G BGP implementation 1

2.2.2- GRE over IPsec implementation

R-HQ

```
interface Loopback0
 ip address 10.254.254.1 255.255.255.255
!
interface Tunnel10
 description GRE mpls to br1
 ip address 192.168.100.1 255.255.255.252
 tunnel source GigabitEthernet0/1
 tunnel destination 10.100.2.1
!
interface Tunnel11
 description gre mpls to br2
 ip address 192.168.100.5 255.255.255.252
 tunnel source GigabitEthernet0/1
 tunnel destination 10.100.3.1
!
interface Tunnel20
 description gre 4g to br1
 ip address 192.168.200.1 255.255.255.252
 tunnel source GigabitEthernet0/2
 tunnel destination 10.200.2.1
!
interface Tunnel21
 description gre 4g to br2
 ip address 192.168.200.5 255.255.255.252
 tunnel source GigabitEthernet0/2
 tunnel destination 10.200.3.1
```

Figure 3.6 – GRE tunnels creation 1

Figure analysis:

Since we're configuring a full-mesh topology, each branch should have a tunnel over each link to each branch.

R-HQ

```
!
crypto ipsec transform-set ts-gre esp-aes esp-sha-hmac
 mode transport
!
!
!
crypto map gre-map 10 ipsec-isakmp
 set peer 10.100.2.1
 set transform-set ts-gre
 match address 101
crypto map gre-map 20 ipsec-isakmp
 set peer 10.100.3.1
 set transform-set ts-gre
 match address 102
crypto map gre-map 30 ipsec-isakmp
 set peer 10.200.2.1
 set transform-set ts-gre
 match address 103
crypto map gre-map 40 ipsec-isakmp
 set peer 10.200.3.1
 set transform-set ts-gre
 match address 104
```

Figure 3.7 – IPsec crypto map configuration 1

Figure analysis:

- The "**crypto ipsec transform-set ts-gre ..**" command create a transform-set named ts-gre that defines the encryption and integrity algotithms to use, needed to be used in a crypto map
- The "**mode transport**" command to encrypts only the payload, not the ip header because they're used by GRE tunnels
- The **"crypto map gre-map X ipsec-isakmp"** commands are used to apply an IPSec encryption to the GRE tunnels, mentioning the other peer of the tunnel, and the transform-set already created , and the ACL matched to the encryption

```
!
!
crypto isakmp policy 10
 encr aes
 hash sha256
 authentication pre-share
 group 14
crypto isakmp key ooredoo123 address 10.100.2.1
crypto isakmp key ooredoo123 address 10.100.3.1
crypto isakmp key ooredoo123 address 10.200.3.1
crypto isakmp key ooredoo123 address 10.200.2.1
!
```

Figure 3.8 – ISAKMP policy and pre-shared keys 1

Figure analysis:

- The “**crypto isakmp policy 10”** command creates IKE phase1 policy with the priority id 10, defining the encryption and hashing algorithms, the authentication method, and D-H group to use during the key negotiation between the VPN peers
- **“crypto isakmp key ooredoo123 address 10.x.x.x”** sets the shared key used for the authentication between the VPN nodes
-

R-HQ

```
!
access-list 101 permit gre host 10.100.1.1 host 10.100.2.1
access-list 102 permit gre host 10.100.1.1 host 10.100.3.1
access-list 103 permit gre host 10.200.1.1 host 10.200.2.1
access-list 104 permit gre host 10.200.1.1 host 10.200.3.1
```

Figure 3.9 – GRE access-lists implementation 1

- Configuring access-lists matching numbers with the crypto-map, that permit a GRE access between tunnels nodes

2.2.3- Track and failover implementation

Our network is based on the MPLS connection, and using 4G as a backup link, so here we configure the failover and backup

```
route-map prefer-mpls permit 10
 set local-preference 200
!
route-map backup-4g permit 10
 set as-path prepend 65000 65000 65000
!
```

Figure 3.10 – route-map configuration for priority links 1

Figure analysis:

- Configure a route-map named prefer-mpls with a local preference 200 higher than the default value to prefer the mpls link as main link
- Configure a route-map named backup-4g making the 4G less preferred by artificially lengthening the AS path

R-HQ

```
redundancy
!
!
track 1 ip sla 1 reachability
!
ip sla 1
 icmp-echo 10.100.1.2 source-interface GigabitEthernet0/1
 threshold 2
 timeout 1000
 frequency 5
ip sla schedule 1 life forever start-time now
logging host 10.10.0.100
```

Figure 3.11 – Tracking and SLA configuration 1

Figure analysis :

- The command **" track 1 ip sla 1 reachability"** created a track object 1 linked to IP SLA 1, to track the reachability of a remote ip
- The **ip sla 1** creates SLA operation id 1, that pings remote MPLS link ip to check its reachability and availability.

  -The pings are sent every **5 seconds**, with a wait up to **1000ms** for each response, and a minimum of **2 failed replies** to consider the link as unreachable.

```
!
no scheduler allocate
event manager applet enable-mpls-peer
 event track 1 state up
 action 1.0 cli command "enable"
 action 2.0 cli command "configure terminal"
 action 3.0 cli command "router bgp 65000"
 action 4.0 cli command "no neighbor 10.100.1.2 shutdown"
event manager applet disable-mpls-peer
 event track 1 state down
 action 1.0 cli command "enable"
 action 2.0 cli command "configure terminal"
 action 3.0 cli command "router bgp 65000"
 action 4.0 cli command " neighbor 10.100.1.2 shutdown"
!
end
```

Figure 3.12 – Event manager for failover 1

Figure analysis :

- The event manager **"enable-mpls-peer"** script watches track 1, if it goes up, it enables the bgp neighbor
- The event manager **"disable-mpls-peer"** script runs when track 1 is down, it shuts down the bgp neighbor, and the traffic switched to 4g

```
router bgp 65000
 bgp log-neighbor-changes
 network 10.10.0.0 mask 255.255.255.0
 network 10.254.254.1 mask 255.255.255.255
 redistribute connected
 neighbor 10.100.1.2 remote-as 65010
 neighbor 10.100.1.2 fall-over bfd
 neighbor 10.100.1.2 allowas-in
 neighbor 10.100.1.2 route-map prefer-mpls in
 neighbor 10.200.1.2 remote-as 65020
 neighbor 10.200.1.2 allowas-in
 neighbor 10.200.1.2 route-map backup-4g out
 neighbor 10.254.254.2 remote-as 65000
 neighbor 10.254.254.2 update-source Loopback0
!
```

Figure 3.13 – Apply route-map on BGP neighbors 1

Figure analysis:

- The "**neighbor 10.100.1.2 fall-over bfd"** command stands for bidirectional forwarding detection to monitor the neighbor link and provide fast failure detection
- **"neighbor 10.100.1.2 route-map prefer-mpls in"** applies the prefer-mpls route map on incoming routes
- **"neighbor 10.200.1.2 route-map backup-4g out"** applies the backup-4g route-map on outgoing routes

2.2.4- Quality of service implementation

We'll define class maps and policy map and allocate bandwidth/priority to specific traffic classes.

```
class-map match-any voip
 match dscp ef
class-map match-any critical-apps
 match access-group name APPS
class-map match-any video
 match dscp af41
!
policy-map WAN-QOS
 class voip
  priority percent 30
 class video
  bandwidth percent 20
 class critical-apps
  bandwidth percent 20
 class class-default
  fair-queue
!
```

Figure 3.14 – Quality of service configuration 1

Figure analysis :

- The command **“match dscp ef”** matches traffic with DSCP value EF which marks latency-sensitive traffic which is VOIP in our case
- The command **“match dscp af41”** matches traffic with DSCP value AF41 which is meant for high-priority data like video-conf in our case
- Define a policy-map named **WAN-QOS** which specifies how traffic classes are handled
    - Assign **priority percent 30** for VOIP and guarantee 30% of the bandwidth for this class and ensure low latency
    - Assign **bandwidth percent 20** for video traffic without a strict priority, and also for other critical APPS defined in the acl APPS

2.2.5- Access-lists implementation

```
!
ip access-list extended APPS
 permit tcp 10.10.0.0 0.0.0.255 any eq 443
 permit tcp 10.10.0.0 0.0.0.255 any eq 22
 permit udp 10.10.0.0 0.0.0.255 any eq snmp
ip access-list extended HQ-IN
 permit ip 10.20.0.0 0.0.0.255 10.10.0.0 0.0.0.255
 permit ip 10.30.0.0 0.0.0.255 10.10.0.0 0.0.0.255
 deny   ip any any log
!
```

Figure 3.15 – ACLs configuration 1

Figure analysis:

- Access list **APPS** to identify application traffic, matches the application traffic **HTTPS, SSH, and SNMP** coming from the lan
- Implement access-list HQ_IN to allow the traffic from the other 2 LANs

3- Topology of the SD-WAN network

3.1- Topology:

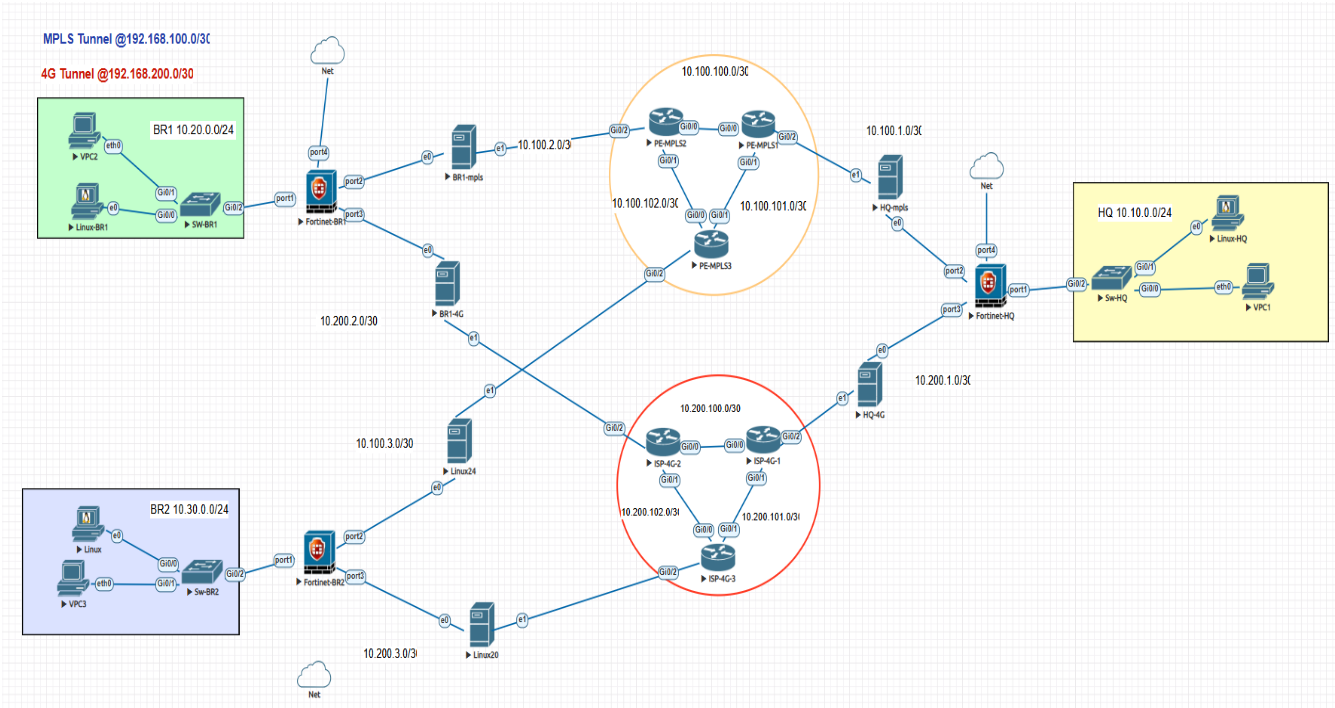


Figure 3.16 – SD-WAN topology 1

Topology Explanation:

The illustrated topology represents a secure and redundant SD-WAN architecture using IPsec tunnels, enabling intelligent traffic steering based on application type and real-time link performance metrics such as latency, jitter, and packet loss, where MPLS link serving as the primary transport for latency-sensitive applications, and a 4G network acting as a secondary failover path and an active transport medium for low-priority or non-latency-sensitive traffic. IPsec tunnels are established over both transport links, providing secure, encrypted communication between sites.

3.2- Configuration

Fortinet offers two methods for the fortigate's configuration, the first one by the console interface using a set of commands, and the second one using the management graphic user interface GUI

3.2.1- IPSec tunnels phase 1 configuration:

Since we are configuring a full-mesh topology, we will need to configure 2 tunnels between each 2 sites, one for each link. Each tunnel should be configured in interface mode to be used later in SD-WAN configuration.

The phase 1 configuration of IPsec tunnels includes configuring peers ip addresses, interfaces, and encryption methods. Starting with central site :

```
FortiGate-VM64-KVM # config vpn ipsec phase1-interface

FortiGate-VM64-KVM (phase1-interface) # show
config vpn ipsec phase1-interface
    edit "ipsec-br1-mpls"
        set interface "port2"
        set peertype any
        set net-device disable
        set proposal des-md5 des-sha1
        set wizard-type simplified-static-fortigate
        set remote-gw 10.100.2.1
        set psksecret ENC CNqSdWBq8w0pv2lwgLORzkAuzCPzkp9KloItynw7BZ0gBCuXAYHUPr
7AFFJPZ9PqgTc0g231lC0oLvnKFjfgwvov9h1MD/PDOmPp8qsOvlzMiy9U2HKOmT6V9YZKPhHjZO5VfT
B9Gw15hBK7UJS9TUegspZOp2Q1xDuFSLmgnFEAtRFWo7QfNw9pRsJocGEltcWCew==
    next
    edit "ipsec-br1-4g"
        set interface "port3"
        set peertype any
        set net-device disable
        set proposal des-md5 des-sha1
        set wizard-type simplified-static-fortigate
        set remote-gw 10.200.2.1
        set psksecret ENC YEDJ1wX+j0hb+ZbTc+u+hF+Lof+3LSQyJIjCV7Y/AItqA3isAjXlSn
kvzv4XOGSYOLZqx6j1XODnRvRLNl7t6JYvK6Qp0KJ3i6VfZjvB3SYY7GJWguaxbpNDZOz98zN0MijKSY
kUt0V7Kp2p5K6KcCTXwJZUtPPKl1I5yZSL+Z4mWP/+YH0bnmlirUNZgah4WQtDnQ==
    next
```

Figure 3.17 – HQ-BR1 IPSec tunnel phase1 configuration 1

Figure analysis:

- Create an IPsec tunnel **ipsec-br1-mpls** for the MPLS link, on the interface port2, and **ipsec-br1-4g** for the 4g link on port3
- The command **"set peertype any"** allows the fortigate to accept a VPN connection from any remote peer, regardless of its identity
- The command **"set wizard-type simplified-static-fortigate"** indicates that the VPN tunnel was configured using fortinet's simplified VPN setup wizard, tailored for static site-to-site Fortigate-to-Fortigate deployments.
- We used an encryption with a pre-shared key which is ooredoo123, the value after the command **"set psksecret ENC…"** represents the key after encryption.
- **"set proposal des-md5 des-sha1"** sets legacy encryption ans hashing algorithms.
- We configure the same parameters for both links IPsec tunnels from BR1 to central site changing the convenient peers ip addresses, as it's shown in the figure below

```
FortiGate-Br1 # config vpn ipsec phase1-interface

FortiGate-Br1 (phase1-interface) # show
config vpn ipsec phase1-interface
    edit "ipsec-hq-mpls"
        set interface "port2"
        set peertype any
        set net-device enable
        set proposal des-md5 des-sha1
        set remote-gw 10.100.1.1
        set psksecret ENC mTqK/awt9JKDJsGoFdIDmQWu2lLftLnC7oZLqiQDI+OFccp6AarGjM
GhD3iDyo3p9ZNIdAmC7SQLeCoGF7bJtEpjxoynnqVMbBQzuUR+Z5b0fbGFV04Vu9BZob67NDHG9Nqhc3
r8McypPKL1nHfrPOb7pRNEpGSsfDh1vOaL0OY1vJn7Ewb1kHJuXD9ASn4eL0uxGA==
    next
    edit "ipsec-hq-4g"
        set interface "port3"
        set peertype any
        set net-device enable
        set proposal des-md5 des-sha1
        set remote-gw 10.200.1.1
        set psksecret ENC km2M7b5bLBILx/X4fZm6VsNz0hwsHF7+u/1WAl0KHGF9OCYNmDNnVy
Y6WpId7mUEcCNCPnErVBgsfOWJzv4saD05hkD8FQzHCvn59Tk+jNi+2CEbDBVFurK5ODwi2MsGZol690
JjJBBQ3Qy7ukD7QNpEviRchAZuQXezB88w4vlX+KBp6Dq2T1XSFVX9SJXZXaoQ0g==
    next
```

Figure 3.18 – BR1-HQ IPSec tunnel phase1 1

With graphic user interface :

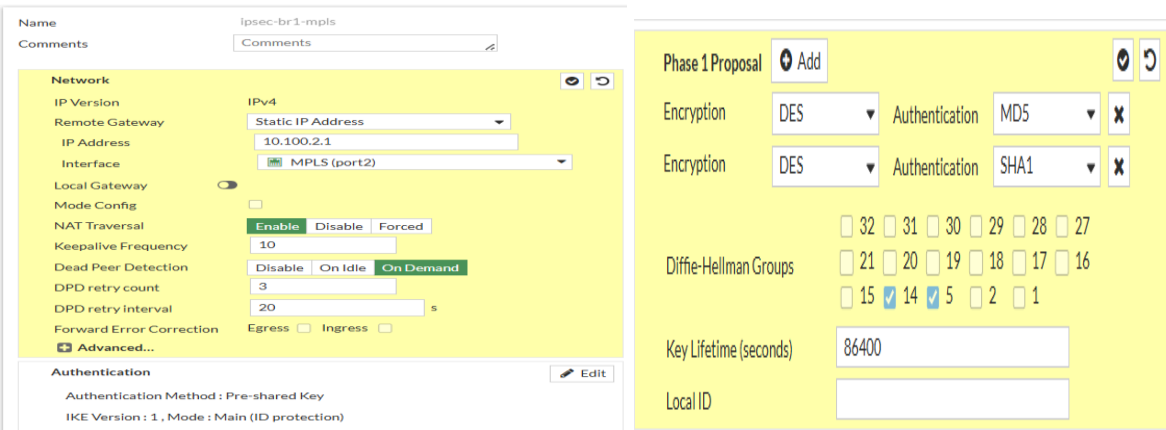


Figure 3.19 - IPsec phase 1 configuration 1

3.2.2- IPSec tunnels phase 2 configuration

```
FortiGate-VM64-KVM # config vpn ipsec phase2-interface

FortiGate-VM64-KVM (phase2-interface) # show
config vpn ipsec phase2-interface
    edit "ipsec-br1-mpls"
        set phase1name "ipsec-br1-mpls"
        set proposal des-md5 des-sha1
        set src-addr-type name
        set dst-addr-type name
        set src-name "all"
        set dst-name "all"
    next
    edit "ipsec-br1-4g"
        set phase1name "ipsec-br1-4g"
        set proposal des-md5 des-sha1
        set src-addr-type name
        set dst-addr-type name
        set src-name "all"
        set dst-name "all"
    next
```

Figure 3.20 – HQ-BR1 ipsec tunnel phase2 interface configuration 1

Figure analysis:

Phase 2 configuration defines the encryption parameters and traffic selectors used to establish the secure tunnel for actual data transfer after phase 1 has successfully completed.

- Link the phase2 configuration with the appropriate phase1 tunnel using **"set phase1name ipsec-br1-mpls/4g".**
- Use the same legacy algorithms used in phase 1 for phase2 encryption
- **" set src-addr-type name"** indicates that the source and destination addresses for the traffic selectors are specified using address object names, not IP ranges or subnets directly.
- **"set src-name all/ set dst-name all"** means this tunnel will match and allow all traffic between the two sites.
- We configure the same parameters for both IPsec tunnels created from BR1 to HQ

```
FortiGate-Br1 (phase2-interface) # show
config vpn ipsec phase2-interface
    edit "ipsec-hq-mpls"
        set phase1name "ipsec-hq-mpls"
        set proposal des-md5 des-sha1
        set src-addr-type name
        set dst-addr-type name
        set src-name "all"
        set dst-name "all"
    next
    edit "ipsec-hq-4g"
        set phase1name "ipsec-hq-4g"
        set proposal des-md5 des-sha1
        set src-addr-type name
        set dst-addr-type name
        set src-name "all"
        set dst-name "all"
    next
```

Figure 3.21 – BR2-HQ ipsec tunnel phase2 1

With graphic user interface :

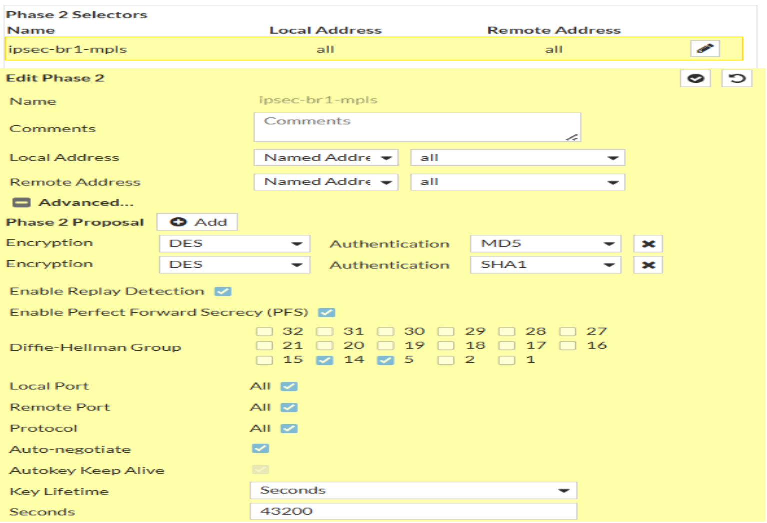


Figure 3.22 - IPsec phase 2 configuration in GUI 1

Ipsec tunnels state's check:

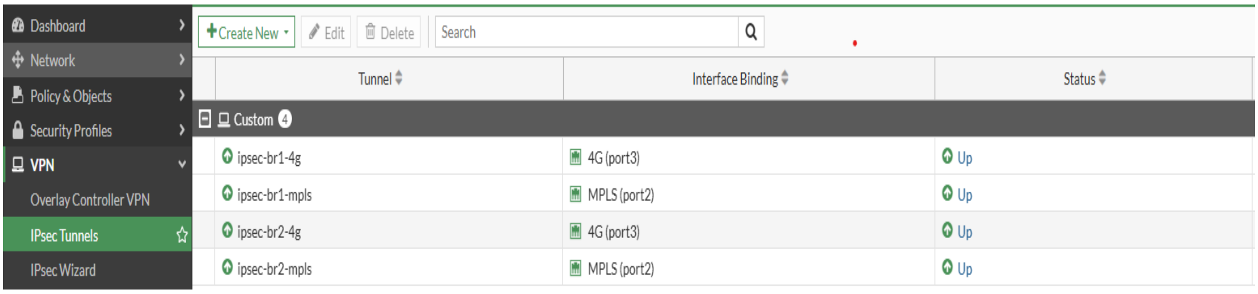


Figure 3.23- IPsec tunnels state on the HQ FortiGate 1

3.2.3- SD-wan configuration:

The configuration of the sd-wan concerns configuring its members first which are going to be the ipsec tunnels already configured, then the health-check SLA for link monitoring, after that the routing rules for each branch, and last thing the static default route.

```
FortiGate-VM64-KVM # config system sdwan

FortiGate-VM64-KVM (sdwan) # show
config system sdwan
    set status enable
    config zone
        edit "virtual-wan-link"
        next
        edit "vpn-zone"
        next
    end
```

Figure 3.24– SD-wan zone configuration 1

Figure analysis:

- The command **"set status enable "** activates the SD-WAN feature on the FortiGate
- We create a new zone "vpn-zone" to group IPsec interfaces into sd-wan logic, beside the default zone "virtual-wan-link"

SD-WAN members configuration:

```
config members
    edit 1
        set interface "ipsec-br1-mpls"
        set zone "vpn-zone"
    next
    edit 2
        set interface "ipsec-br1-4g"
        set zone "vpn-zone"
        set priority 2
    next
    edit 3
        set interface "ipsec-br2-mpls"
        set zone "vpn-zone"
    next
    edit 4
        set interface "ipsec-br2-4g"
        set zone "vpn-zone"
        set priority 2
    next
end
```

Figure 3.25 – SD-wan members configuration 1

Figure analysis:

- We define the actual SD-WAN members and assign them to the appropriate zone
- Each member has an ID to identify it and use it easily later
- All the 4 tunnels are added as members to the vpn-zone with a default priority value 1 for MPLS links tunnels (preferred) and priority 2 for 4G links tunnels (lower preference).

With graphic user interface :

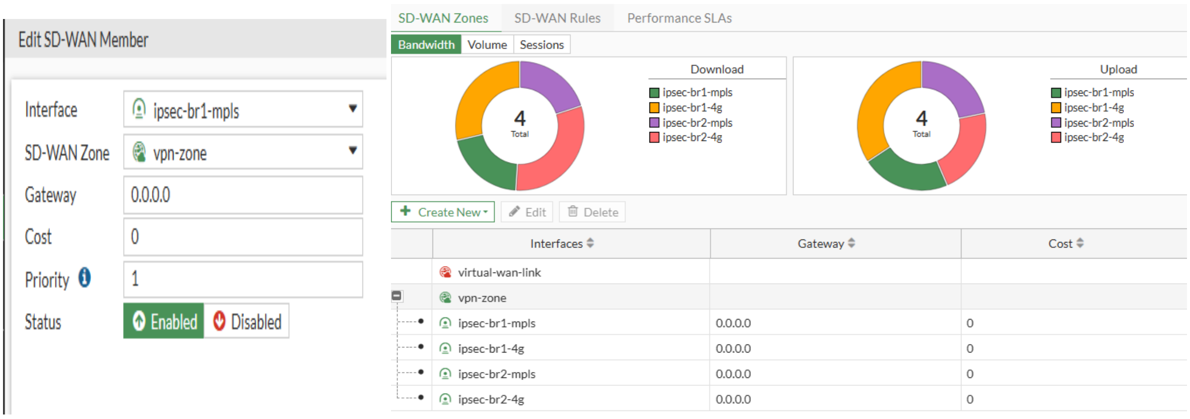


Figure 3.26- SD-WAN zones and members display 1

SD-WAN health-check configuration :

```
CLI Console (2)
Fortigate-HQ # config system sdwan
config system sdwan

Fortigate-HQ (sdwan) # config health-check

Fortigate-HQ (health-check) # edit "br1_monitor"

Fortigate-HQ (br1_monitor) # show
config health-check
    edit "br1_monitor"
        set server "10.20.0.1"
        set members 2 1
        config sla
            edit 1
                set latency-threshold 100
                set jitter-threshold 20
                set packetloss-threshold 5
            next
        end
    next
end

Fortigate-HQ (br1_monitor) #
```

Figure 3.27 – SD-wan SLA configuration 1

Figure analysis:

This figure defines links monitoring criteria to mesure performance, failover, and quality of network links in real time.

- We create 2 link monitors on each fortigate, one for each branch. In the figure we have configured the SLA on the HQ fortigate, one for branch 1 (br1_monitor) and one for branch 2 (br2_monitor)
- For each monitor we set the server as the IP address of a device in the destination lan, and the track members the 2 ipsec tunnels towards the destination to track their availability
- We define the sla performance metrics where : **latency=50, jitter=20, packetloss=5**. If a link exceeds these SLA thresholds it's marked either unhealthy, unstable, or unreliable, and the traffic is routed via the second link.

With graphic user interface :

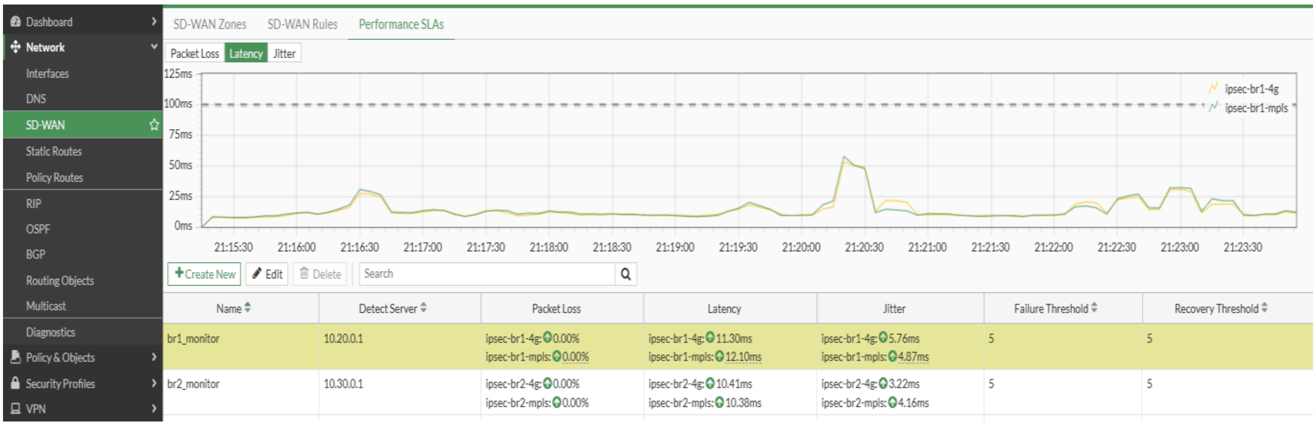


Figure 3.28- SD-WAN SLA performance 1

SD-WAN rules configuration :

```
Fortigate-HQ (service) # show
config service
    edit 1
        set name "voip_traffic_br1"
        set mode priority
        set protocol 17
        set start-port 5060
        set end-port 5061
        set dst "lan2"
        set src "lan1"
        set health-check "br1_monitor"
        set link-cost-factor jitter
        set priority-members 1 2
    next
    edit 2
        set name "http_traffic_br1"
        set protocol 6
        set start-port 80
        set end-port 80
        set dst "lan2"
        set src "lan1"
        set priority-members 2 1
    next
    edit 3
        set name "RTP_traffic_br1"
        set mode priority
        set protocol 17
        set start-port 5004
        set end-port 5006
        set dst "lan2"
        set src "lan1"
        set health-check "br1_monitor"
        set priority-members 1 2
    next
end
```

Figure 3.29 – SD-wan rules configuration 1

Figure analysis:

This is where traffic routing policies are defined based on protocols and link priority.

- Rule 1 for VoIP traffic from central site to branch1 , using UDP protocol with the ports 5060/5061 (default SIP ports), setting the source as the local LAN and destination as the remote LAN, defining priority members as member 1 and member 2 (MPLS and 4G tunnels), and the link-cost-factor as jitter since the VoIP traffic is jitter-sensitive.
- Rule 2 for HTTP traffic from central site to branch1 which is not latency-sensitive or jitter-sensitive. Using protocol TCP with the port number 8080, with same source and destination as first rule, and defining priority members as member 2 then member 1 (4G first then MPLS)
- Rule 3 for RTP traffic (VisioConf traffic) from central site to branch1 , using UDP protocol with the ports 5004 (for audio) and 5006 (for video) , setting the same source and destination as previous rules, defining priority members as member 1 and member 2 (MPLS and 4G tunnels), and the link-cost-factor as latency since the RTP traffic is latency-sensitive.
- We configure the same rules for the connection with branch2, changing the source and destination addresses, and the priority members using 3 and 4.

Best quality link selection:

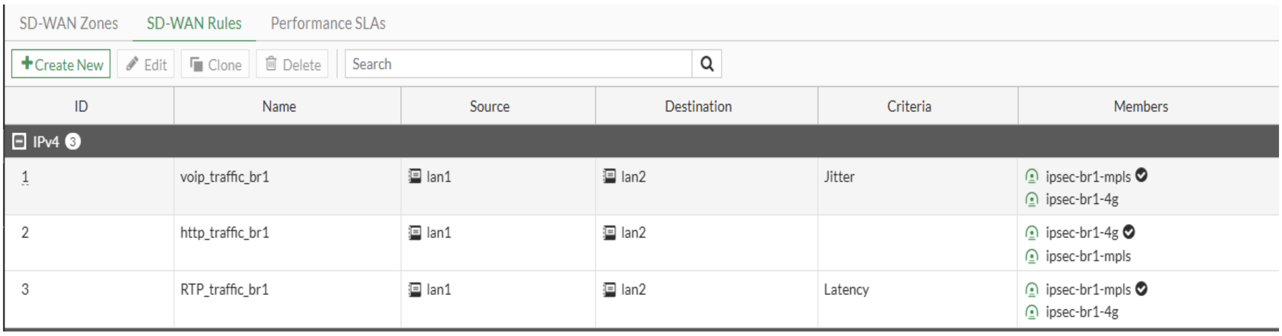


Figure 3.30- SD-WAN rules and link selection 1

3.2.4- SD-wan static route configuration

```
FortiGate-VM64-KVM # config router static

FortiGate-VM64-KVM (static) # show
config router static
    edit 1
        set distance 1
        set sdwan-zone "vpn-zone"
    next
end
```

Figure 3.31 – SD-wan static route configuration 1

Figure analysis:

- We configure a static route to tell the fortigate how to forward traffic using SD-WAN virtual interface rather than a specific physical one,
- It's a default route 0.0.0.0/0 with the exit interface vpn-zone

3.2.5- Routing protocol configuration

Since we are simulating a multiple path topology and we need a scalability and control over how routes are exchanged across multiple sites, BGP is highly recommended for this usecase, and the configuration shown below is a general BGP configuration on a Fortigate. [15]

```
FortiGate-VM64-KVM # config router bgp

FortiGate-VM64-KVM (bgp) # show
config router bgp
    set as 65000
    set router-id 10.10.0.1
    set keepalive-timer 10
    set holdtime-timer 20
    set dampening enable
    set ebgp-multipath enable
    set ibgp-multipath enable
    set additional-path enable
    set scan-time 10
    config neighbor
        edit "192.168.100.2"
            set allowas-in-enable enable
            set soft-reconfiguration enable
            set as-override enable
            set interface "ipsec-br1-mpls"
            set remote-as 65000
            set update-source "ipsec-br1-mpls"
        next
```

```
        edit "192.168.100.6"
            set allowas-in-enable enable
            set soft-reconfiguration enable
            set as-override enable
            set interface "ipsec-br2-mpls"
            set remote-as 65000
            set update-source "ipsec-br2-mpls"
        next
        edit "192.168.200.2"
            set allowas-in-enable enable
            set soft-reconfiguration enable
            set as-override enable
            set interface "ipsec-br1-4g"
            set remote-as 65000
            set update-source "ipsec-br1-4g"
        next
        edit "192.168.200.6"
            set allowas-in-enable enable
            set soft-reconfiguration enable
            set as-override enable
            set interface "ipsec-br2-4g"
            set remote-as 65000
            set update-source "ipsec-br2-4g"
        next
```

Figure 3.32– BGP routing configuration 1

Figure analysis:

- **"set keepalive-timer 10"** defines how often BGP sends a small ' I'm alive ' message to its peer, if no message is received within this interval, fortigate sends one proactively.
- **"set holdtime-timer 20"** defines the maximum time to wait before considering the BGP peer as dead if no keepalive is received.
- **"set ebgp-multipath/ ibgp-multipath enable"** allows load balancing over multiple equal-cost BGP paths in both eBGP and iBGP.
- **"set additional-path enable"** enables advertising multiple paths for the same prefix, it is helpful for the sd-wan performance

- **“set scan-time 10”** defines how often BGP checks the routing table for updates (in seconds), needed for SD-WAN environment where tunnel performance or availability can shift rapidly.
- **“set soft-reconfiguration enable”** enables the fortigate to store inbound BGP updates from peers before applying policies,
- **“set as-override enable”** allows the fortigate to rewrite the ASN in outbound updates to avoid the iBGP split horizon rule.

BGP neighbor check :

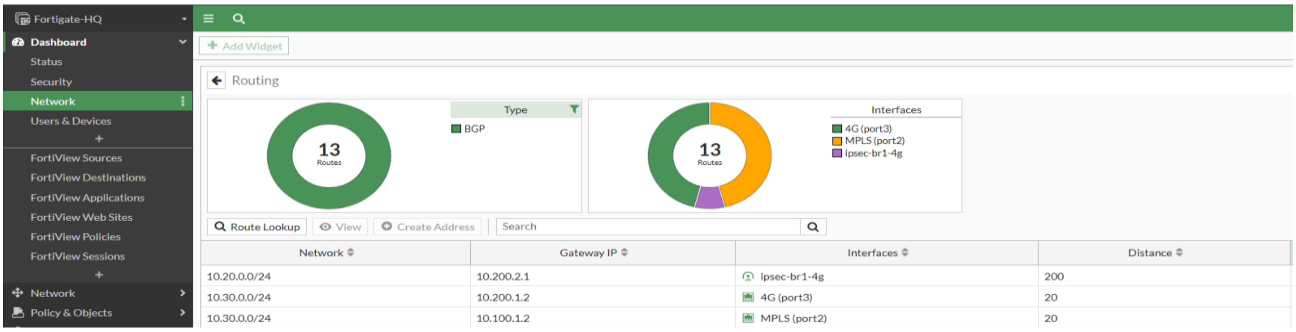


Figure 3.32- BGP routes 1

3.2.6- Firewall policy configuration

```
FortiGate-VM64-KVM # config firewall policy

FortiGate-VM64-KVM (policy) # show
config firewall policy
    edit 1
        set name "wan_to_lan"
        set uuid fa89e274-3c69-51f0-d6ae-07fd8a9d5a37
        set srcintf "vpn-zone"
        set dstintf "port1"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set logtraffic-start enable
    next
    edit 2
        set name "lan_to_wan"
        set uuid 516cf78e-3c6a-51f0-9953-f9a0077f65bd
        set srcintf "port1"
        set dstintf "vpn-zone"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set logtraffic-start enable
    next
```

Figure 3.33 – Firewall policy configuration 1

Figure analysis:

The fortigate is basically a firewall, so it is configured by default with a policy that denies all the traffic, so we have to define 2 firewall policies to allow traffic between LAN and WAN.

- First rule allowing WAN to LAN traffic, defining the vpn-zone as the source interface, and ALL protocols/ports are allowed. Setting the schedule of this policy's activity as 'always'
- Second rule allowing LAN to WAN traffic, configuring the same parameters as the first rule except the source and destination interfaces.
- The commands **"set logtraffic all"** and **"set logtraffic-start enable"** are configured to save the logs of all sessions on this fortigate, when any session starts.

With graphic user interface :

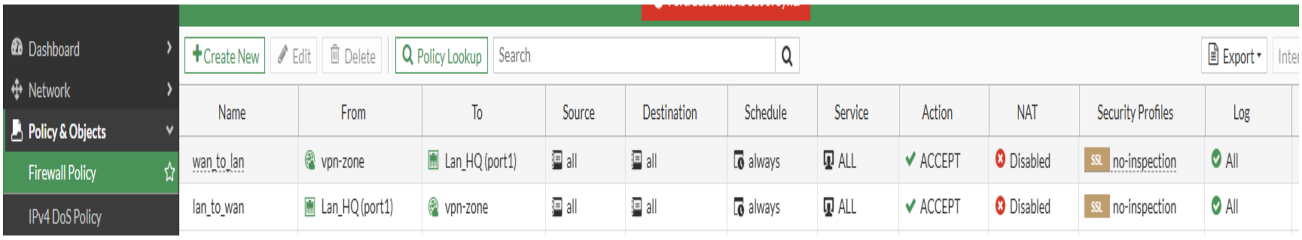


Figure 3.33 – Firewall policy configuration for SD-WAN zone 2

4. Conclusion:

With the simulation environment established on EVE-NG, it was possible to build and test realistic topologies that mirror both traditional and SD-WAN architectures. The use of virtualized network devices, combined with a structured topology design and clear configuration strategies, ensured reliable conditions for performance assessment. This methodological setup provides the foundation for the next phase of analysis, where both models are subjected to real traffic and network impairments.

Chapter 4: Results and discussion

1- Evaluation criteria:

In order to objectively evaluate the performance of the two studied architectures , we defined a set of evaluation criteria focused on QoS and network resilience. These criteria not only help quantify network performance but also reflect each architecture's ability to ensure service continuity under various traffic confitions or link failures.

1.1- **Latency:** measures the transmission delay between two nodes. This is critical for real-time applications such as voice and video conferencing.

1.2- **Packet loss:** The percentage of packets lost during transmission, which directly impacts the quality of sensitive services.

1.3- **Failover capability:** the network's ability to quickly respond to the failure of one of the links in order to ensure service continuity.

- Instead of using a pre-configured tools to simulate the different types of traffic (HTTP, VoIP, RTP), we have used python scripts for each type of traffic. The following scripts represent the VoIP and RTP traffic:

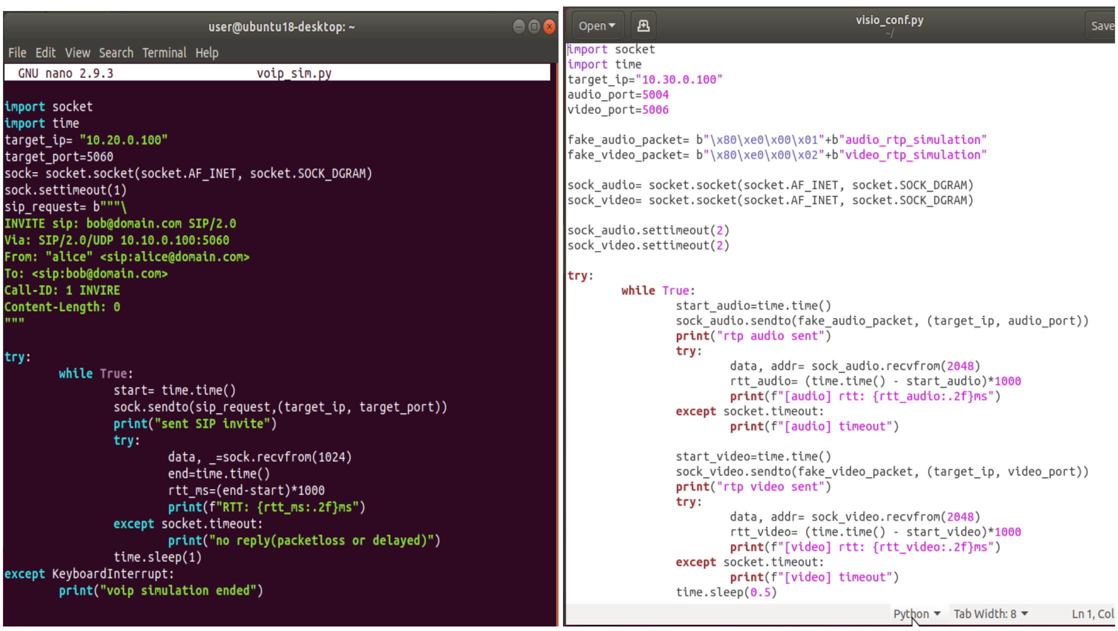


Figure 4.1- Python scripts for traffic simulation 1

2- Traditional network tests :

2.1- Link Redundancy check :

```
r-hq#show ip bgp 10.20.0.1
BGP routing table entry for 10.20.0.0/24, version 17
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1
  Refresh Epoch 2
  65000 65000 65000 65020 65000
    10.200.1.2 from 10.200.1.2 (10.200.101.1)
      Origin IGP, localpref 100, valid, external
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 2
  65010 65000
    10.100.1.2 from 10.100.1.2 (10.100.101.1)
      Origin IGP, localpref 200, valid, external, best
      rx pathid: 0, tx pathid: 0x0
```

Figure 4.2- BGP routing redundancy check 1

Figure's interpretation:

- The command "show ip bgp 10.20.0.1" shows the BGP routing information for the prefix 10.20.0.0/24
- The route 10.100.1.2 (MPLS) is preferred due to higher local preference (200 VS 100)
- The route via 10.200.1.2 (4G) is still valid and available as a backup path

2.2- Failover packet loss:

We initiate a continuous ping command (of 50 packets) to establish a connectivity test to a remote site, sending packets constantly for a period of time. Then, we proceed to disable the interface.

```
vpc1> ping 10.20.0.2 -c 50

84 bytes from 10.20.0.2 icmp_seq=1 ttl=60 time=191.841 ms
84 bytes from 10.20.0.2 icmp_seq=2 ttl=60 time=171.284 ms
84 bytes from 10.20.0.2 icmp_seq=3 ttl=60 time=62.798 ms
84 bytes from 10.20.0.2 icmp_seq=4 ttl=60 time=134.835 ms
84 bytes from 10.20.0.2 icmp_seq=5 ttl=60 time=121.557 ms
84 bytes from 10.20.0.2 icmp_seq=6 ttl=60 time=58.275 ms
84 bytes from 10.20.0.2 icmp_seq=7 ttl=60 time=58.144 ms
84 bytes from 10.20.0.2 icmp_seq=8 ttl=60 time=92.001 ms
84 bytes from 10.20.0.2 icmp_seq=9 ttl=60 time=229.069 ms
84 bytes from 10.20.0.2 icmp_seq=10 ttl=60 time=477.480 ms
```

Figure 4.3- Continuous ping packets send 1

We have disabled the bgp neighbor :

```
pe-mpls1(config)#int g0/2
pe-mpls1(config-if)#shut
pe-mpls1(config-if)#
*Jun 16 23:32:34.109: %BGP-5-NBR_RESET: Neighbor 10.100.1.1 reset (Interface fla
p)
*Jun 16 23:32:34.178: %LDP-5-NBRCHG: LDP Neighbor 10.200.1.1:0 (2) is DOWN (Inte
rface not operational)
```

Figure 4.4- Neighbor interface shutdown 1

As we have initiated a continuous ping, we continue our observation :

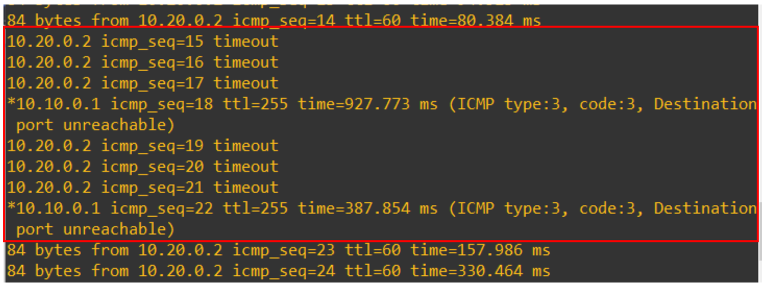


Figure 4.5- Continuous ping behavior 1

Figure analysis:

- We observe that connectivity is restored after a short period of time, as packets continue to be sent normally. When there is an outage on the MPLS BGP neighbor, switching to the 4G neighbor occurs.

- However, this switchover process takes some time, resulting in packet loss during this transition measured at 16%.

- As we continue to make successive failover tests, the link switch result a 5% packet loss from the 5th test onward.

2.3- Traffic overload :

To evaluate the behavior of the traditional network under pressure, we simulated concurrent traffic flows (VoIP, HTTP, and RTP) between the sites (from HQ to BR1). This test aimed to observe how the network handles high-demand scenarios and how it impacts latency.

The results extracted from the RTT values measured during each flow, clearly show increased delay, especially for latency-sensitive applications like VoIP and RTP. The following histogram illustrates the minimum, average, and maximum latency recorded during the test for each traffic type:

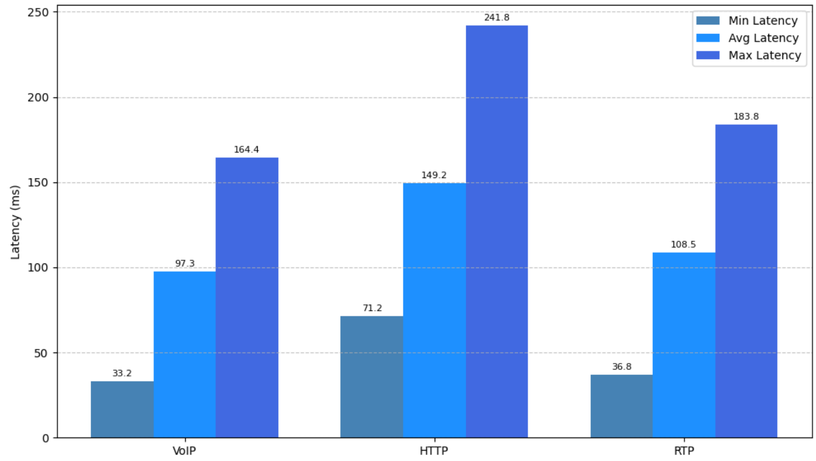


Figure 4.6- Latency per traffic type in traditional network 1

2.4- Behavioral analysis of network under varying impairments:

To assess the resilience of this architecture, we initiated simultaneous VoIP, HTTP, and RTP traffic across the MPLS link and monitored their routing paths and performance. Initially, the traffic followed the MPLS path as expected.

We then artificially increased the jitter on the MPLS link to 40ms.

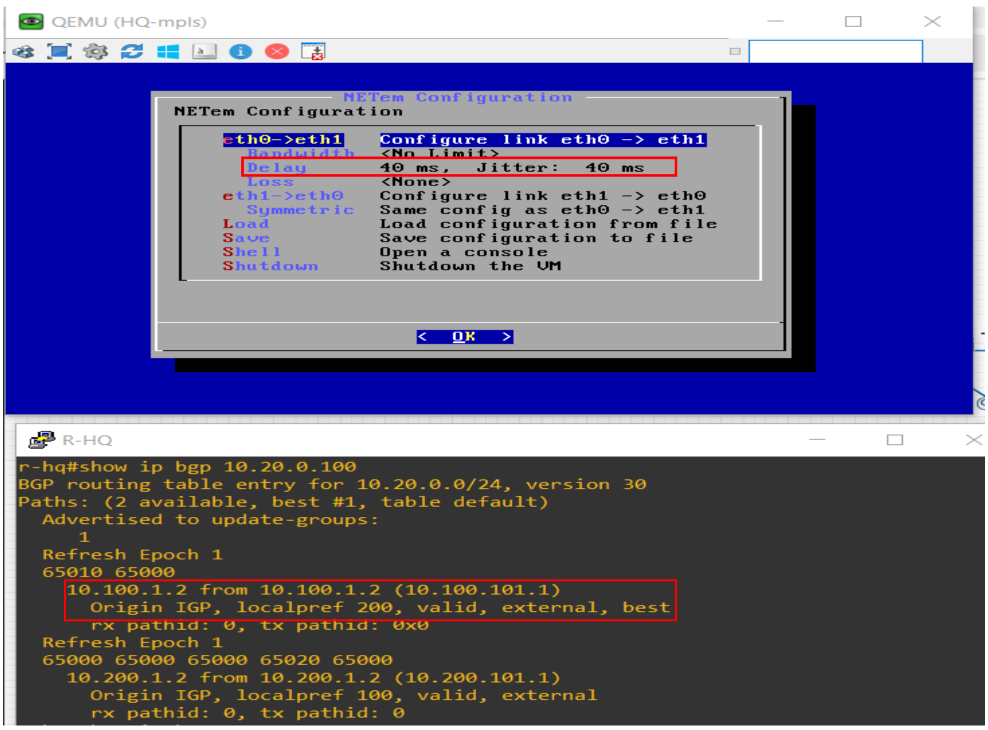


Figure 4.7- Link's Jitter changes and BGP routing check 1

Despite this degradation, the routing remained unchanged and the RTT values began to rise, particularly affecting real-time traffic.

Further we introduced a latency of 100ms on the MPLS link.

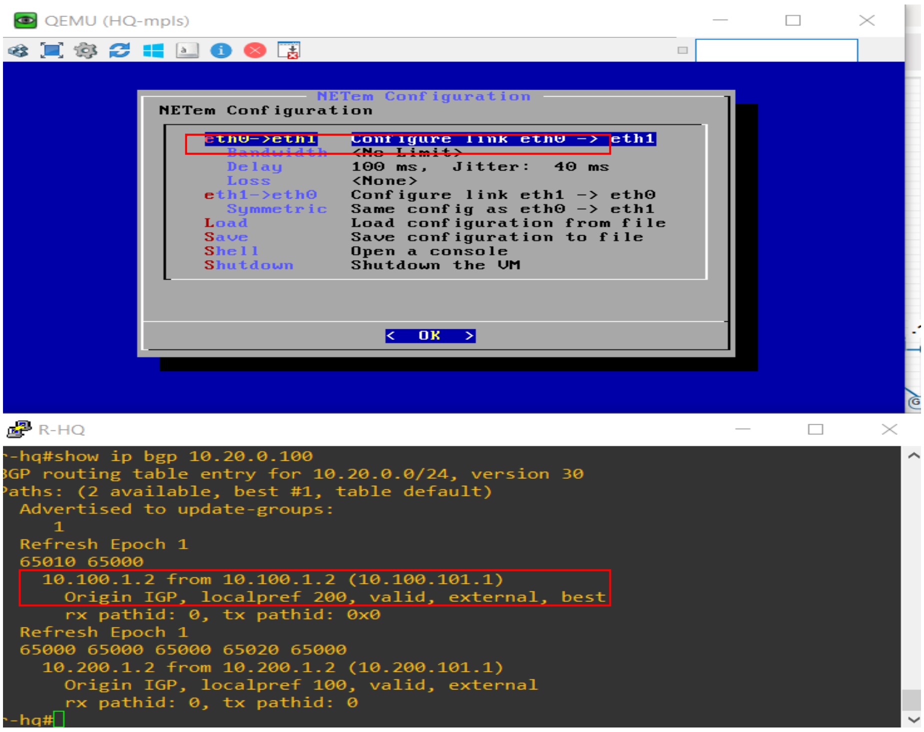


Figure 4.8- Link's Latency changes and BGP routing check 1

Even with this significant impairment, the routing still did not switch to the 4G backup link, due to the absence of dynamic traffic-aware mechanisms in traditional routing.

Consequently, the latency-sensitive traffic experienced further degradation in performance.

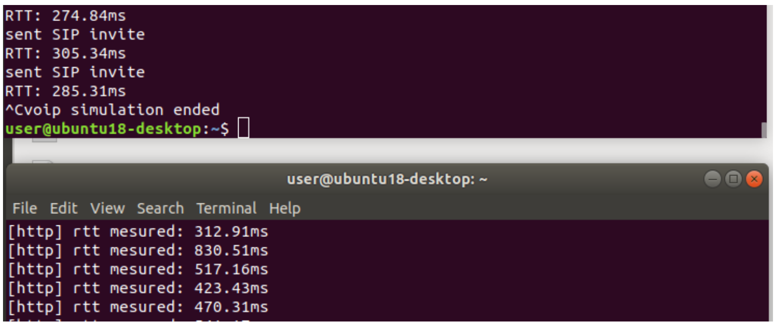


Figure 4.9- RTT values after varying impairments 1

This test highlights the limitations of traditional dynamic routing and the lack of adaptability in traditional WAN setups

3- Tests for SD-WAN network

3.1- Packet path analysis using the Traceroute command:

```
Fortigate-HQ # execute traceroute 10.20.0.2
traceroute to 10.20.0.2 (10.20.0.2), 32 hops max, 3 probe packets per hop, 84 by
te packets
 1  192.168.100.2  8.147 ms  18.866 ms  4.487 ms
 2  10.20.0.2  26.610 ms  54.352 ms  11.659 ms

Fortigate-HQ #
```

Figure 4.10- Traceroute command’s result 1

Figure interpretation:

- The command “traceroute” was executed on device Fortigate-HQ with the destination address 10.20.0.2

- The traffic successfully reaches the destination using the iosec-br1-mpls tunnel, via the address 192.168.100.2.

- This indicates that SD-WAN routing is working properly.

3.2- Failover and link switch :

We use a continuous ping command to establish a connectivity test to a remote site, sending packets constantly for a period of time. Then, we proceed to disable the interface.

```
Fortigate-HQ # execute ping-options repeat-count 50

Fortigate-HQ # execute ping 10.20.0.2
PING 10.20.0.2 (10.20.0.2): 56 data bytes
64 bytes from 10.20.0.2: icmp_seq=0 ttl=63 time=43.0 ms
64 bytes from 10.20.0.2: icmp_seq=1 ttl=63 time=44.2 ms
64 bytes from 10.20.0.2: icmp_seq=2 ttl=63 time=30.7 ms
64 bytes from 10.20.0.2: icmp_seq=3 ttl=63 time=60.4 ms
64 bytes from 10.20.0.2: icmp_seq=4 ttl=63 time=440.7 ms
64 bytes from 10.20.0.2: icmp_seq=5 ttl=63 time=152.8 ms
64 bytes from 10.20.0.2: icmp_seq=6 ttl=63 time=203.7 ms
```

Figure 4.11- Continuous 50 Ping packets 1

We have disabled the ipsec tunnel assigned to the destination:

MPLS (port2)	Physical Interface		10.100.1.1/255.255.255.252
ipsec-br1-mpls	Tunnel Interface		192.168.100.1/255.255.255.255

Figure 4.12- IPsec tunnel shutdown on Frotigate-HQ 1

As we have initiated a continuous ping, we continue our observation :

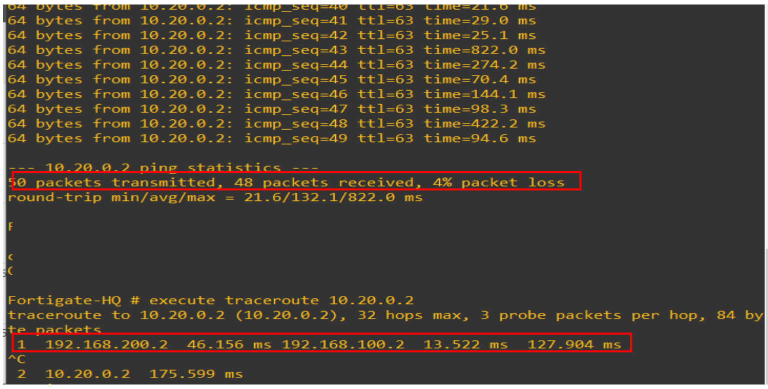


Figure 4.13- Behavior of continuous ping 1

Result interpretation:

- We observe that connectivity is restored after a short period of time, as packets continue to be sent normally. When there is an outage on the IPsec-mpls tunnel, switching to the 4G IPsec tunnel occurs.

- However, this switchover process takes some time, resulting in packet loss during this transition measured at 4%.

- We have enabled the MPLS tunnel again, then we proceed a traceroute command. We observe that the traffic is re-routed to the MPLS tunnel again with no packet loss. As fortinet states: “performance SLA link monitoring measures the health of links.. if a link is broken, the routes on that link are removed and traffic is routed through other links. When the link is working again, the routes are re-enabled.”

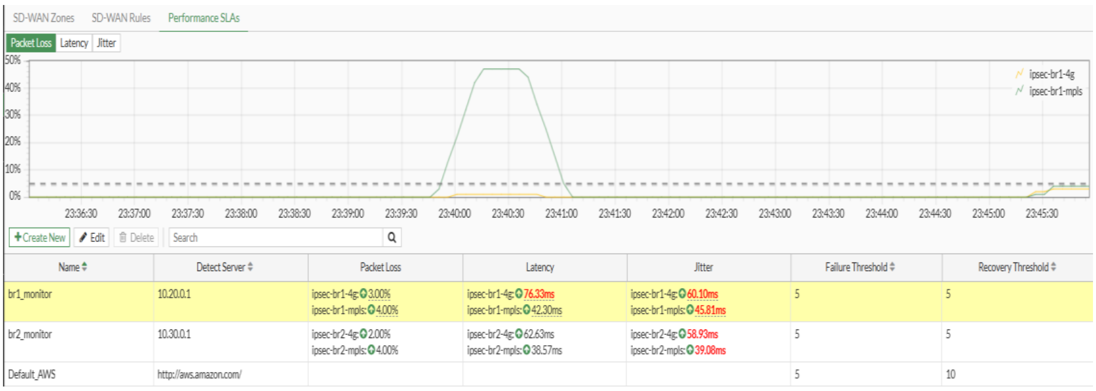


Figure 4.14- Packet loss SLA on GUI 1

Figure analysis :

- The figure represents the packet loss SLA of branch 1 monitor on the fortigate GUI.

- The packet loss has of the ipsec-br1-mpls tunnel has increased during the period when the tunnel was disabled, then it started decreasing once the tunnel is enabled again.

- We have proceed the failover test once more, we observe the packet loss

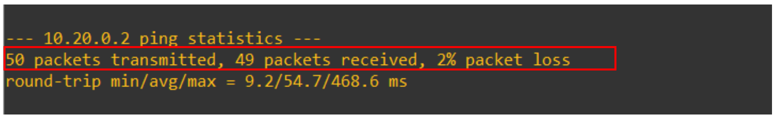

```
--- 10.20.0.2 ping statistics ---
50 packets transmitted, 49 packets received, 2% packet loss
round-trip min/avg/max = 9.2/54.7/468.6 ms
```

Figure 4.15- Packet loss after second failover 1

Result's interpretation :

- We notice that unlike the first test, and since the tunnels and routes already active and established, failover was faster, resulting in a 2% packet loss, because once alternate IPsec tunnels are established after the first failover, subsequent ones are faster and more stable, reducing convergence time and packet loss.
- As we continue to make successive failover tests, the link switch result a 0% packet loss, because no new session state or route recalculation is required.

3.3- Traffic overload:

In this test, we simulated a scenario of high traffic load by generating three types of concurrent flows: VoIP, RTP, and HTTP. The histogram below illustrates the RTT measurements for each traffic class:

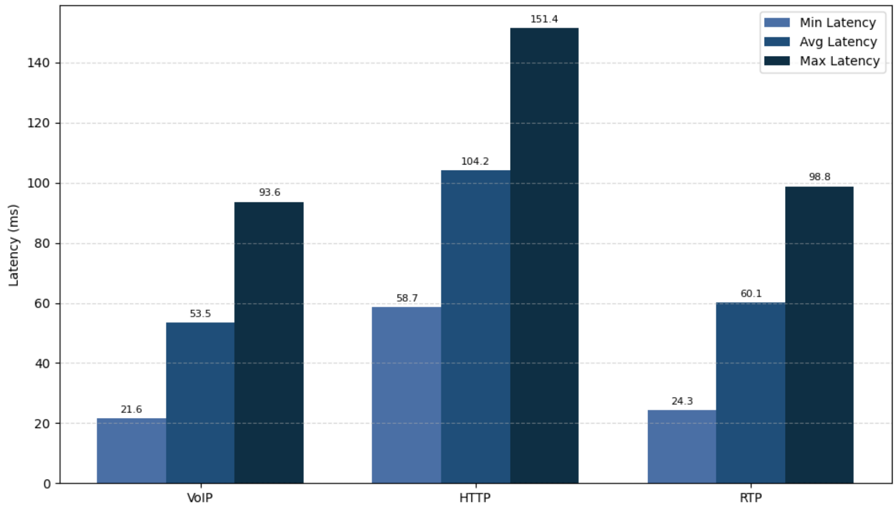


Figure 4.16 – Latency per traffic type in SD-WAN network 1

As shown, SD-WAN maintained lower and more stable latency across all applications, thanks to its application-aware routing and real-time link monitoring, confirming the efficiency of its dynamic traffic steering and SLA-based decision-making.

3.4- Behavioral analysis of SD-WAN under varying network impairments:

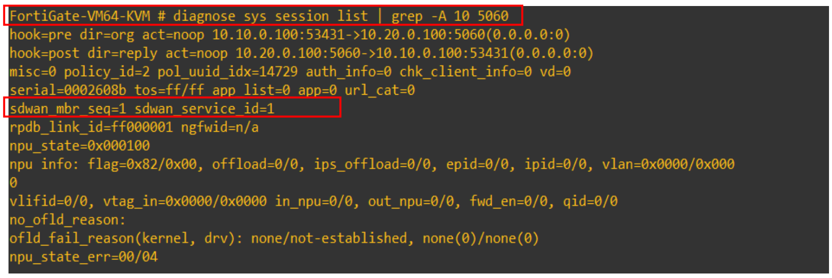
This section focuses on testing SD-WAN's responsiveness to network impairments such as latency, jitter, and packet loss. By intentionally degrading one of the transport links, we evaluate how the SD-WAN solution reacts; specifically, whether it reroutes traffic to the secondary link based on the SLA thresholds defined in the configuration.

These scenarios replicate real-world issues like congestion or unstable connections. The objective is to assess the system's ability to maintain service continuity and enforce optimal path selection under degraded conditions.

3.4.1- MPLS jitter degradation :

To evaluate SD-WAN's ability to ensure VoIP performance, we simulated increasing jitter on the primary link.

First we initiated a continuous VoIP traffic from HQ to BR1 and check on the fortigate the opened sessions on the port 5060



```
FortiGate-VM64-KVM # diagnose sys session list | grep -A 10 5060
hook=pre dir=org act=noop 10.10.0.100:53431->10.20.0.100:5060(0.0.0.0:0)
hook=post dir=reply act=noop 10.20.0.100:5060->10.10.0.100:53431(0.0.0.0:0)
misc=0 policy_id=2 pol_uuid_idx=14729 auth_info=0 chk_client_info=0 vd=0
serial=0002608b tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=1 sdwan_service_id=1
rpdb_link_id=ff000001 ngfwid=n/a
npu_state=0x000100
npu info: flag=0x82/0x00, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0, vlan=0x0000/0x000
0
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason:
ofld_fail_reason(kernel, drv): none/not-established, none(0)/none(0)
npu_state_err=00/04
```

Figure 4.17- SIP session in SD-WAN 1

Figure analysis:

- The figure shows the list and details of opened sessions on port 5060
- The command "diagnose sys session list | grep –A 10 5060"
- The output line "sdwan_mbr_seq= 1 sdwan_service_id=1" describes that the SD-WAN member used in this session is the one with id 1, which is ipsec-br1-mpls, and the SD-WAN rule id followed is the one with id 1 which is the VoIP rule. And this is the normal case of routing without varying the jitter value.

Next, we set the jitter for the MPLS link to a value that's higher than the one configured in the SLA link monitor, in our case we set it to 40ms (> 20ms)

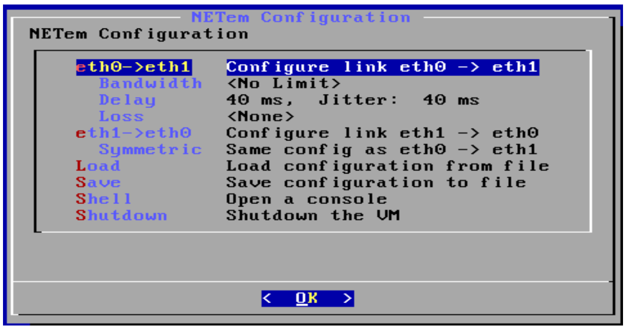


Figure 4.18- MPLS jitter changes 1

Since we have initiated a continuous VoIP traffic, the flow doesn't stop. We check the opened session for the port 5060

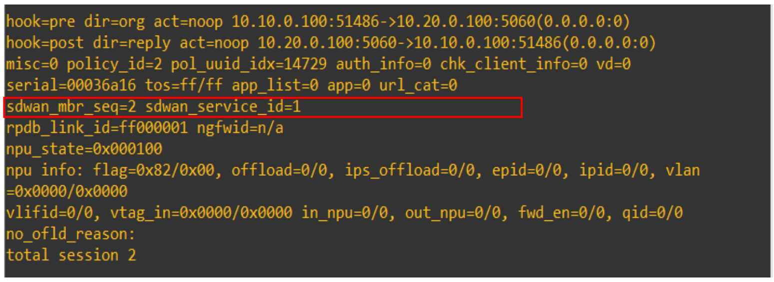


Figure 4.19- SIP session on SD-WAN 1

Result interpretation :

- The opened session this time shows the line " sdwan_mbr_seq=2 sdwan_service_id=1"
- The sdwan VoIP policy (service_id=1) selects the ipsec-br1-4g (mbr_seq=2) as a member for this session, which means that the switch to the better quality link with a lower jitter value is successfully done.

3.4.2- MPLS latency degradation :

This time, we simulated increasing latency on the primary link, to evaluate the ability of SD-WAN to ensure RTP performance.

First we initiated a continuous RTP traffic from HQ to BR1 and check on the Fortigate the opened sessions on the ports 5004 (RTP audio port) and 5006 (RTP video port).

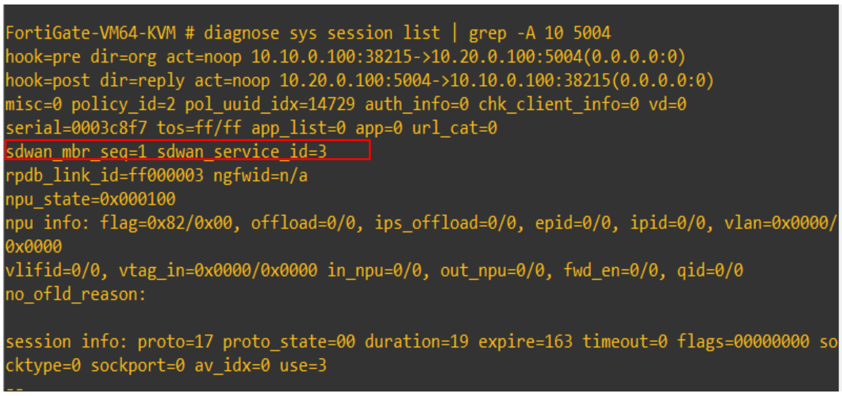


Figure 4.20- RTP session in SD-WAN 1

Figure analysis:

- The figure shows the list and details of opened sessions on port 5004
- The command “diagnose sys session list | grep –A 10 5004”
- The output line “sdwan_mbr_seq= 1 sdwan_service_id=3” describes that the SD-WAN member selected in this session is the one with id 1, which is ipsec-br1-mpls, and the SD-WAN rule followed is the one with id 3 which is the RTP rule. And this is the normal case of routing without varying the Latency value.

Next, we set the latency for the MPLS link to a value that’s higher than the one configured in the SLA link monitor, in our case we set it to 100ms (> 50ms)

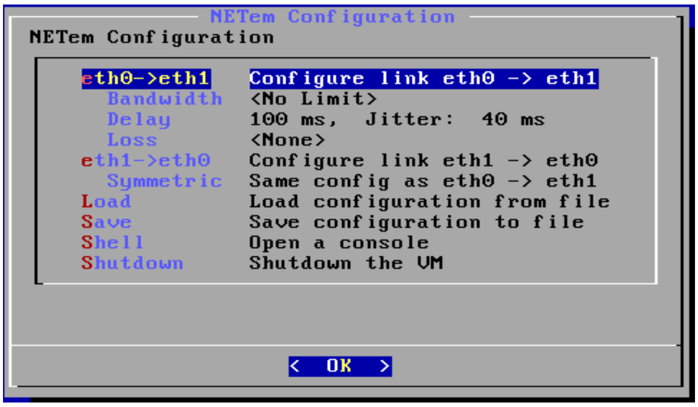


Figure 4.21- MPLS latency changes 1

Since we have initiated a continuous RTP traffic, the flow doesn’t stop. We check the opened session for the port 5006 (same thing for 5004)

```
FortiGate-VM64-KVM # diagnose sys session list | grep -A 10 5006
hook=pre dir=org act=noop 10.10.0.100:54437->10.20.0.100:5006(0.0.0.0:0)
hook=post dir=reply act=noop 10.20.0.100:5006->10.10.0.100:54437(0.0.0.0:0)
misc=0 policy_id=2 pol_uuid_idx=14729 auth_info=0 chk_client_info=0 vd=0
serial=0003e6a6 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=2 sdwan_service_id=3
rpdb_link_id=ff000003 ngfwid=n/a
npu_state=0x000100
npu info: flag=0x82/0x00, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0, vlan
=0x0000/0x0000
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason:
```

Figure 4.22- RTP session in SD-WAN 1

Result interpretation :

- The opened session this time shows the line " sdwan_mbr_seq=2 sdwan_service_id=3"
- The sdwan RTP policy (service_id=3) selects the ipsec-br1-4g (id=2) member for this session, which means that the switch to the better latency link is successfully done.

3.4.3- 4G packet loss degradation:

Now, and since the 3 types of traffics are routed through the 4**G** link, we increase its packet loss to 100% to force the link to become unusable.

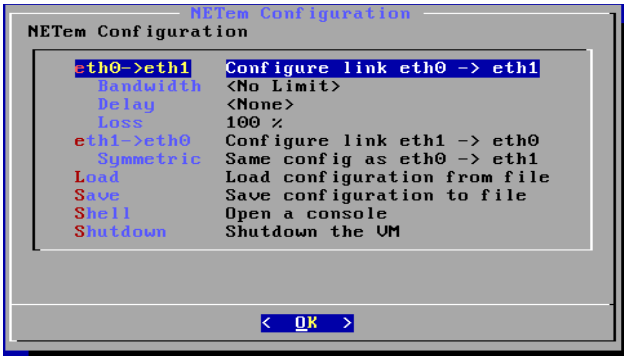


Figure 4.23- 4G packet loss changes 1

We check the opened session for the any of the 3 traffics (ex :checking for port 80 and 5060)

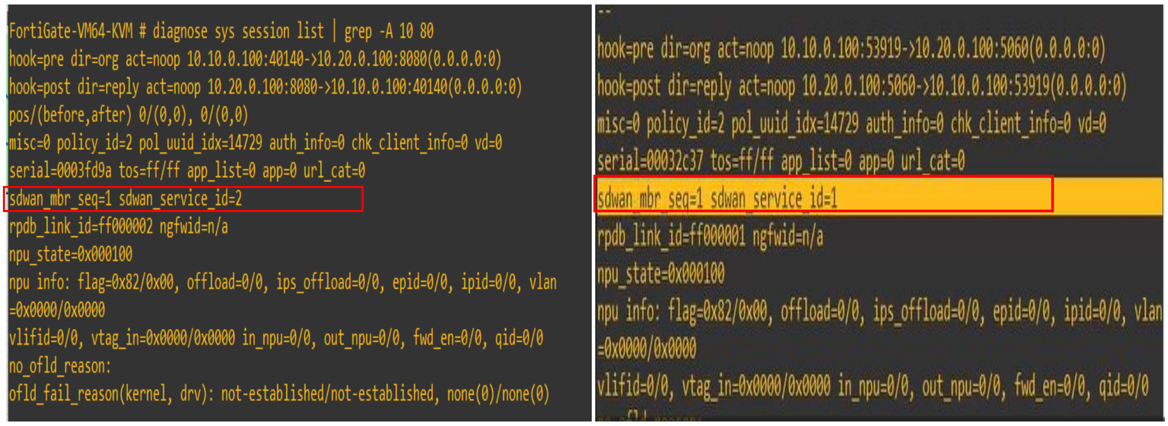


Figure 4.24- HTTP and SIP sessions in SD-WAN 1

Result interpretation:

- Basically, the HTTP traffic is routed through the 4G tunnel, but as the figures shows, this traffic is routed now through the MPLS tunnel (mbr_seq=1) according to the SD-WAN's HTTP service (service_id=2)
- The SIP traffic is also re-routed through the MPLS tunnel again according to the SIP service (service_id=1)
- As the packet loss exceeded the defined SLA thresholds, the SD-WAN mechanism detected the degradation and dynamically rerouted the all the traffic through the MPLS tunnel.

4- Conclusion:

To summarize, the tests and comparative evaluations conducted in this chapter reveal the inherent limitations of traditional architectures, particularly in adaptability and traffic control. Conversely, SD-WAN demonstrates significant strengths, including intelligent routing, application-aware prioritization, and rapid failover capabilities. These insights pave the way for more robust and scalable network design, which will be further reflected in the final analysis and future outlook in the next chapter.

Chapter 5: Conclusion and Future Work

1- Summary of results

This section presents a consolidated view of the key findings obtained through the various tests conducted on both the traditional and SD-WAN topologies. By analyzing metrics such as latency, packet loss, and failover performance under different traffic and impairment scenarios, we highlighted the behavioral differences between the two architectures. The results are summarized through comparative graphs and table to illustrate how SD-WAN improves network efficiency, responsiveness, and resilience.

1.1- Failover efficiency:

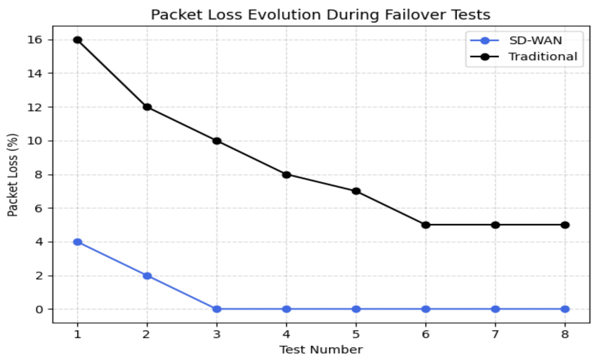


Figure 5.1- Packet loss evolution during 1

- The packet loss curve illustrates that SD-WAN recovers from degraded conditions much faster than the traditional network, starts at 4% in the first test, drops to 0% from the 3rd test onward, indicating a rapid convergence and stabilization thanks to intelligent traffic steering and link monitoring mechanisms.
- In contrast, the traditional setup shows limited improvement and fails to adapt dynamically, resulting in persistent losses.

1.2- Traffic overload resilience:

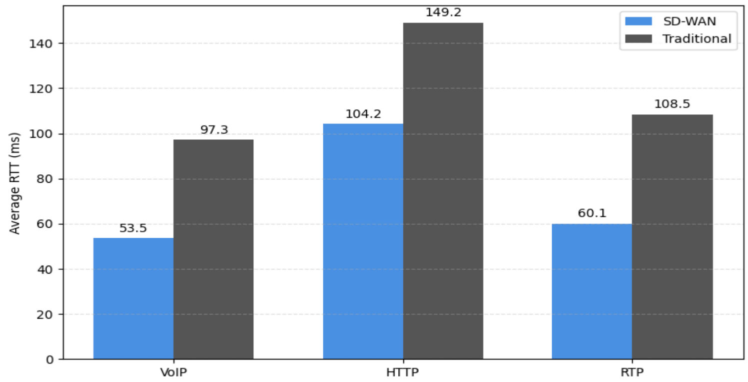


Figure 5.2 – Latency comparison by traffic type 1

- The latency histogram under simultaneous VoIP, RTP, and HTTP flows highlights SD-WAN's ability to maintain lower RTTs for critical traffic by leveraging real-time link metrics and directing less sensitive traffic over secondary link.
- Conversely, traditional routing lacks this granularity, resulting in higher and less predictable latencies.

1.3- Overall behavior under impairments:

The following comparative table summarizes the performance observed under each type of degradation:

Criteria	**Traditional network**	**SD-WAN**
Jitter response	No route adaptation	Dynamic traffic rerouting
Latency increase	No reaction, RTT increases sharply	Rerouting to maintain low RTT
Packet loss recovery	Late, passive failover	Fast failover based on thresholds
Traffic overload	All traffic on main link	Traffic split by SLA rules

Table 5.1-Overall behavior comparison 1

2- Contribution of SD-WAN in a multi-transport environment

- **Dynamic path selection :** SD-WAN continuously monitors link performance (latency, jitter, packet loss) and dynamically selects the optimal for each type of traffic.
- **Application-aware routing:** traffic is classified based on its criticality, allowing real-time services like VoIP and RTP to use MPLS link, while best-effort services (e.g., HTTP) are routed over 4G

- **Centralized policy enforcement:** centralized configuration enables precise control over routing decisions and simplifies management in complex network environments
- **Rapid failover:** in the event of link degradation or failure, SD-WAN quickly reroutes traffic to backup links, minimizing packet loss and downtime.
- **Load balancing & link utilization:** SD-WAN can distribute traffic across multiple links according to defined SLA rules, improving overall link utilization and reducing congestion
- **Enhanced resilience:** through proactive monitoring and intelligent steering, SD-WAN ensures continuity and performance even under traffic overload or link impairments.
- **Transport independence:** the ability to abstract the underlying transport (MPLS, LTE/4G, broadband, etc.) allows greater flexibility and cost optimization

3- Future perspectives and improvements

This study confirmed the advantages of SD-WAN in handling diverse network conditions through dynamic routing and real-time link evaluation. Building on these results, future work could explore:

- Real-world deployment in enterprise environments to validate scalability and reliability.
- Convergence of SD-WAN and security in SASE (Secure Access Service Edge) frameworks [2]
- AI-powered automation for network optimization and troubleshooting [2]
- Enhanced support for remote and hybrid work environments [2]
- Integration with 5G networks for improved mobility and reliability [2]
- Multi-cloud optimization capabilities for distributed applications [2]
- Migration to a hub-and-spoke topology to improve scalability and simplify routing in large enterprise environments.

These perspectives aim to further strengthen SD-WAN's role in building intelligent, resilient, and adaptive network infrastructures.

Conclusion:

The goal of this research was to examine the drawbacks of conventional WAN architectures in a multi-transport setting and assess the advantages of SD-WAN as a cutting-edge substitute. We evaluated both models on important performance measures like latency, packet loss, and failover behavior using theoretical analysis and real-world simulations with EVE-NG.

The conventional architecture showed predictable but constrained adaptation under changing network conditions since it relied on static methods and established failover paths. On the other hand, SD-WAN demonstrated improved responsiveness through intelligent traffic steering and real-time connection monitoring, enabling improved performance and resilience, especially in situations where the system is overloaded or degraded.

The findings demonstrated that SD-WAN offers an application-aware, centralized, and adaptable method of inter-site communication, making it perfect for businesses overseeing a variety of transportation linkages. In dynamic, high-demand situations like Ooredoo's, SD-WAN works better than classic approaches, which are still applicable in simpler settings.

In addition to highlighting SD-WAN's operational advantages, this comparison study set the stage for next innovations that will improve enterprise network efficiency and agility, like cost-performance analysis, cloud service integration, and AI-based traffic management.

References:

[1] J. Qadir and A. Ali, "Exploiting the Power of Multiplicity: A Holistic Survey of Network-Layer Multipath," IEEE, Jul. 2015. Available: https://ieeexplore.ieee.org/document/7152820

[2] Alkira, "What is the Difference Between WAN and SD-WAN?," *Alkira*. https://www.alkira.com/what-is-the-difference-between-wan-and-sd-wan/#:~:text=The%20WAN%20vs%20SD%2DWAN,with%20modern%20digital%20business%20requirements.

[3] "Wan connections types- explaination adn examples," *study ccna*, Dec. 29, 2022. https://study-ccna.com/wan-connection-types/

[4] S. Andrew, Tanenbaum, and J. David, "Network Layer : Routing," in *Computer Networks 5th Edition*, Pearson, 2011.

[5]cisco, "What Is SD-WAN? - Software-Defined WAN (SDWAN)," *Cisco*, 2019. https://www.cisco.com/c/en/us/solutions/enterprise-networks/sd-wan/what-is-sd-wan.html

[6] "À propos de Ooredoo - L'histoire de Ooredoo - Ooredoo Algeria," *Ooredoo Algeria*, 2023. https://www.ooredoo.dz/fr/tout-sur-ooredoo/a-propos

[7] B. Staff, "6 Benefits to a Software-Defined WAN (SD-WAN) Implementation," *Technology Solutions That Drive Business*. https://biztechmagazine.com/article/2016/12/6-advantages-software-defined-wan-implementation

[8] Constantin Mohorea, "Cisco Catalyst SD-WAN Components > Data Plane | Cisco Press," *Ciscopress.com*, 2024. https://www.ciscopress.com/articles/article.asp?p=3203553

[9]R. Bhardwaj, "Cisco SD WAN Data Plane Operation - IP With Ease," *IP With Ease*, Nov. 05, 2024. https://ipwithease.com/cisco-sd-wan-data-plane-operation

[10] Mohamed amine, Ouarbi, et al. *A Comprehensive Survey on Software-defined Wide Area Network (SD-WAN): Principles, Opportunities and Future Challenges*. November 2024, p. 9-10, https://doi.org/10.1007/s11227-024-06718-1

[11] Gartner, "Gartner Magic Quadrant & Critical Capabilities - IT Research," *Gartner*, 2024. https://www.gartner.com/en/research/magic-quadrant

[12] https://www.facebook.com/legarski, "Can you explain Cisco SD-WAN in detail? - SolveForce Communications," *SolveForce*, Sep. 20, 2024. https://solveforce.com/can-you-explain-cisco-sd-wan-in-detail/

[13] netseccloud.com, "Understanding EVE-NG: Essential Tool for Network Emulation," *Netseccloud.com*, 2024. https://netseccloud.com/what-is-eve-ng-an-introduction-to-network-emulation

[14] "Getting Started with EVE-NG | cylab.be," *Cylab.be*, 2024. https://cylab.be/blog/360/getting-started-with-eve-ng

[15] "BGP multiple path support | Administration Guide," *Fortinet.com*, 2025. https://docs.fortinet.com/document/fortigate/7.6.1/administration-guide/773406/bgp-multiple-path-support?