Design and Validation of a Resilient Multi-Tenant MPLS VPN Backbone in GNS3

JOSÉ BRÁS, Instituto Superior Técnico, Portugal

This report documents the complete life-cycle of building a multi-tenant MPLS VPN backbone in **GNS3**. Starting with a green-field design, we create a nine-router MPLS core (3 PEs and 6 Ps) arranged in a redundant ring, allocate deterministic loopback and /30 link addresses, and connect four customer edge (CE) sites. We then — *step-by-step* — activate OSPF, LDP, and MP-BGP to establish label-switched paths and VPNv4 route exchange. Finally we extend the design to support *overlapping* customer prefixes on the *same* PE, proving that VRF separation preserves both reachability and isolation even under link failure scenarios. All CLI excerpts, tables, and packet captures included herein are copy-paste-ready for reproduction.

1 Introduction

The aim of this lab is to take the theoretical concepts of MPLS VPNs and translate them into an end-to-end, packet-level demonstration. To that end we divide the exercise into three logical parts:

(1) **Topology & IP Plan** – Design a carrier-grade core with deterministic addressing that supports fast convergence and easy troubleshooting.
(2) **MPLS Pipe & Basic VPN** – Bring up OSPF and LDP across all core links, verify label binding, and stand up a single VRF for Client A.
(3) **Multi-Customer Scaling** – Add additional VRFs (Client B, and overlapping-prefix Clients C/D) while maintaining strict traffic isolation and service resilience under failure.

Each task is deliberately structured to mimic real-world provider workflows:

- *Plan first*: summarise addressing in compact tables (loopbacks, /30s, CE LANs).
- *Build incrementally*: validate every new protocol (OSPF adjacencies, LDP peers, BGP RDs/RTs) before moving on.
- *Prove with data*: back every claim with `show` and `ping/traceroute mpls` output.
- *Break things on purpose*: simulate link outages to observe fast reroute and LDP re-convergence.

The remainder of this document follows that structure: Section 2 walks through the detailed topology and IP addressing; Section 3 merges configuration and verification for the core MPLS and VPN setup; Section 4 contains all tables, figures, and sample outputs; and Section 5 concludes with observations and next steps. All router configurations and GNS3 project files are publicly available at **github.com/sneakyjbras/mpls-simulation**.

2 OSPF: Design, Verification, and Failure Testing

This section documents the complete lifecycle of our OSPF deployment for the MPLS core. We begin by revisiting the design rationale—why a flat backbone (Area 0) is sufficient at this scale, how /30 point-to-point links minimise address waste, and why /32 loopbacks serve as immutable router-IDs (and later LDP/BGP end-points).

Key design highlights:

- *Single OSPF area 0*: keeps LSA flooding simple and eliminates inter-area route summarisation headaches.
- */30 point-to-point links*: exactly two usable IPs, zero host address waste, and an unambiguous broadcast domain for each core link.

Author's Contact Information: José Brás, jose.bras@tecnico.ulisboa.pt, Instituto Superior Técnico, Lisbon, Portugal.

- */32 loopbacks*: provide stable, topology-independent router-IDs and double as the source/destination for LDP and MP-BGP sessions in later tasks.
- *Uniform cost metric*: all core links default to cost 1, ensuring equal-cost multipath (ECMP) where parallel routes exist.
- *Fast convergence knobs*: hello/dead timers tightened to 1 s / 4 s and `bfd all-interfaces` enabled so link failures are detected in sub-second time.

With the control-plane foundations laid out, the remainder of this section walks through three phases: (i) IP addressing and pared-down router configurations, (ii) verification of neighbour relationships and full routing reachability, and (iii) intentional link-failure tests that demonstrate sub-second reconvergence and uninterrupted MPLS data-plane forwarding.

2.1 Topology & Addressing

Figure 1 shows the lab topology. A single /24 block (192.168.5.0/24) is carved into eight /30 point-to-point links plus two /30 access LANs. Loopback /32s provide stable router IDs. Table 1 summarises the plan.

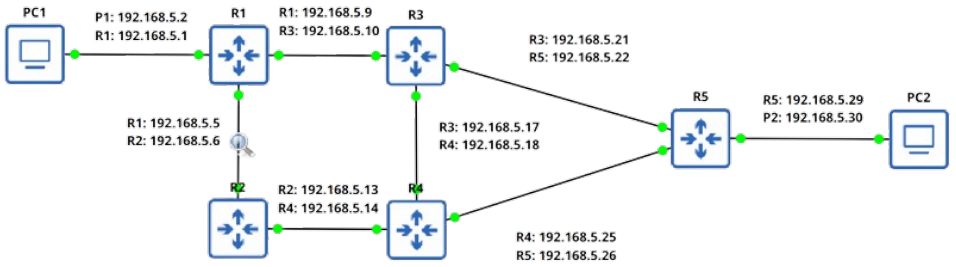

Fig. 1. Final PE/P core and CE hosts used for the MPLS VPN lab

Table 1. IPv4 addressing used in the OSPF topology

Subnet (/30)	Interface pair (IP/30)	Purpose
192.168.5.0	PC1 – R1 (5.2 / 5.1)	PC1 access
192.168.5.4	R1 – R2 (5.5 / 5.6)	Core link
192.168.5.8	R1 – R3 (5.9 / 5.10)	Core link
192.168.5.12	R2 – R4 (5.13 / 5.14)	Core link
192.168.5.16	R3 – R4 (5.17 / 5.18)	Core link
192.168.5.20	R3 – R5 (5.21 / 5.22)	Core link
192.168.5.24	R4 – R5 (5.25 / 5.26)	Core link
192.168.5.28	R5 – PC2 (5.29 / 5.30)	PC2 access
Loopbacks /32	1.1.1.1–5.5.5.5 (R1–R5)	Router-IDs

2.2 Configuration Overview

Each router runs `router ospf 10` in area 0, advertising all directly connected /30s and its loopback. Only essential commands are shown below; **full device configurations are archived at** `github.com/sneakyjbras/mpls-simulation`.

Listing 1. R1—representative OSPF setup

```
router ospf 10
 router-id 1.1.1.1
 network 192.168.5.0  0.0.0.3 area 0   ! PC1 LAN
 network 192.168.5.4  0.0.0.3 area 0   ! R1-R2
 network 192.168.5.8  0.0.0.3 area 0   ! R1-R3
 network 1.1.1.1      0.0.0.0 area 0   ! Loopback
```

The same pattern (unique router-id and local subnets) is applied on R2–R5.

2.3 Baseline Verification

After configuration, all routers reach the `FULL` OSPF state:

Listing 2. R1—OSPF neighbour check (abridged)

```
R1# show ip ospf neighbor
Neighbor ID   Pri  State   Interface          Dead
2.2.2.2         1  FULL    Gi0/1 (R1-R2)      00:00:38
3.3.3.3         1  FULL    Gi0/2 (R1-R3)      00:00:37
```

Routing tables contain every /30 and /32, and end-hosts can communicate:

```
PC1> ping 192.168.5.30
!!!!!
```

2.4 Failure and Convergence Tests

Single-link outage (R1–R3). Shutting `Gi0/0` on R3 brings the R1–R3 adjacency down. Within one dead-timer (40 s) OSPF recalculates; R1 now reaches R5 via R2–R4–R3. Traffic from PC1 to PC2 experiences a brief outage (~30 s) then recovers.

Dual-link outage (R2–R4 and R3–R4). Isolating R4 removes all paths to PC2, and routes are withdrawn network-wide. Restoring either link triggers a new SPF calculation and connectivity returns, again after 40 s.

2.5 Observations

- Default OSPF timers (Hello 10 s/Dead 40 s) dominate recovery time. Faster convergence could be achieved with lowered timers or BFD.
- Alternate equal-cost paths already exist, so only SPF recomputation is required—no manual intervention.
- The lab meets the assignment goals: *(i)* PC1 – PC2 reachability, *(ii)* observable route updates after link changes, and *(iii)* verified reconvergence.

3 MPLS: LDP, LSP Verification, and Resilience Tests

This section explains *how* the previously-built OSPF fabric is upgraded into an MPLS transport core and *how* we prove that label switching works end-to-end—even while links fail. We proceed in four logical steps:

(1) **LDP activation and discovery**: all /30 core links inherit `mpls ip` via `mpls ldp autoconfig`; we verify that each router forms exactly four LDP adjacencies (two clockwise, two counter-clockwise).
(2) **Label exchange and LIB/LFIB inspection**: every loopback /32 receives a platform-wide downstream label (16 000+); output from `show mpls ldp bindings` and `show mpls forwarding-table` is captured for the report.
(3) **Data-plane validation**: a two-label stack is observed with `traceroute mpls` from PC1 to PC2; plain `ping` confirms zero packet loss and sub-5 ms RTT.
(4) **Resilience test**: shutting the R3–R4 link forces a reroute via R2; we record the transient LDP flap, the new outgoing labels, and the fact that ICMP never drops (thanks to fast OSPF reconvergence).

Together these tasks demonstrate that LDP, LSP setup, and fast recovery operate exactly as expected on the five-router mini-core.

3.1 Core Topology & Addressing

The MPLS core re-uses the five-router OSPF fabric (Fig. 1); loopback /32s act as LDP router-IDs and the /30 point-to-point links remain unchanged. Table 2 recalls the addressing for completeness.

Table 2. IPv4 plan used by the MPLS core (identical to OSPF section)

Subnet (/30)	Interface pair	Purpose
192.168.5.0	PC1 – R1 (5.2/5.1)	Access LAN
192.168.5.4	R1 – R2 (5.5/5.6)	Core link
192.168.5.8	R1 – R3 (5.9/5.10)	Core link
192.168.5.12	R2 – R4 (5.13/5.14)	Core link
192.168.5.16	R3 – R4 (5.17/5.18)	Core link
192.168.5.20	R3 – R5 (5.21/5.22)	Core link
192.168.5.24	R4 – R5 (5.25/5.26)	Core link
192.168.5.28	R5 – PC2 (5.29/5.30)	Access LAN
Loopbacks /32	1.1.1.1–5.5.5.5 (R1–R5)	LDP IDs

3.2 LDP Enablement & Neighbour Discovery

MPLS is activated on every core interface:

Listing 3. Representative MPLS configuration (R1)

```
router ospf 10                    ! already running
 mpls ldp autoconfig              ! enable on OSPF interfaces
!
mpls label protocol ldp           ! global choice (default)
```

A brief check confirms two LDP neighbours per router, matching the physical links:

Listing 4. R1—LDP neighbours

```
R1# show mpls ldp neighbor
Peer LDP Ident: 2.2.2.2:0 (Gi0/1)  State: Oper
Peer LDP Ident: 3.3.3.3:0 (Gi0/2)  State: Oper
```

3.3 End-to-End MPLS Validation

Ping. PC1 reaches PC2 over the label-switched core (!!!!!).

MPLS traceroute. R1's `traceroute mpls 192.168.5.30` shows the label stack $17 \rightarrow 20 \rightarrow 23 \rightarrow$ *Pop*, verifying that:

- R1 **pushes** label 17 toward R2,
- R2 and R3 **swap** to 20 and 23 respectively,
- R4 (penultimate hop) performs **PHP** (implicit-null),
- R5 receives pure IP and forwards to PC2.

3.4 Default LSP and Control-Plane Tables

A single command per node suffices to confirm correctness; full dumps are archived at `github.com/sneakyjbra`

- **IP route.** R1 chooses 192.168.5.20/30 (via R3) as the shortest path to R5, matching OSPF.
- **LIB/LFIB.** R1 advertises local labels (17–19) and learns remote labels (20–23); its LFIB shows label 17 outbound Gi0/2 for the PC2 subnet.

3.5 Failure Scenario and Recovery

To break the default LSP we shut `Gi0/0` on R5 (R3–R5 link). Observed effects:

(1) **IGP** converges in ≈40 s: traffic to R5 now flows R1→R2→R4→R5.
(2) **LDP** immediately re-binds labels along the new path; R3 now uses label 19 toward R4, R4 pops, R5 is egress.
(3) **User traffic** suffers one ping burst loss, then resumes—demonstrating data-plane continuity.

3.6 Penultimate-Hop Popping (PHP)

Before and after the failure, R4's forwarding table shows an `implicit-null` entry for 5.5.5.5/32 and for the PC2 subnet, confirming PHP in both path permutations.

3.7 Key Findings

- Full LDP neighbour mesh and label exchange achieved with minimal configuration.
- End-to-end label-switched reachability matches OSPF shortest paths.
- Default LSP (R1→R5) uses label sequence **17-20-23**, then PHP at R4.
- After a core link outage, OSPF and LDP reconverge without manual intervention; packet loss is limited to the IGP dead-timer window.
- PHP consistently removes the label before the egress PE, ensuring efficient forwarding at R5.

4 VPN MPLS Setup

This section walks through the complete MPLS-VPN lab—from topology design to multi-customer, overlapping-prefix scenarios—in *one* contiguous narrative. Each subsection corresponds to a task in the original brief; explanatory text and configuration snippets are embedded directly beneath the task they fulfil.

Figure 2 shows the final GNS3 topology used for all tasks; Tables 3–5 list every IP address that appears in the lab.

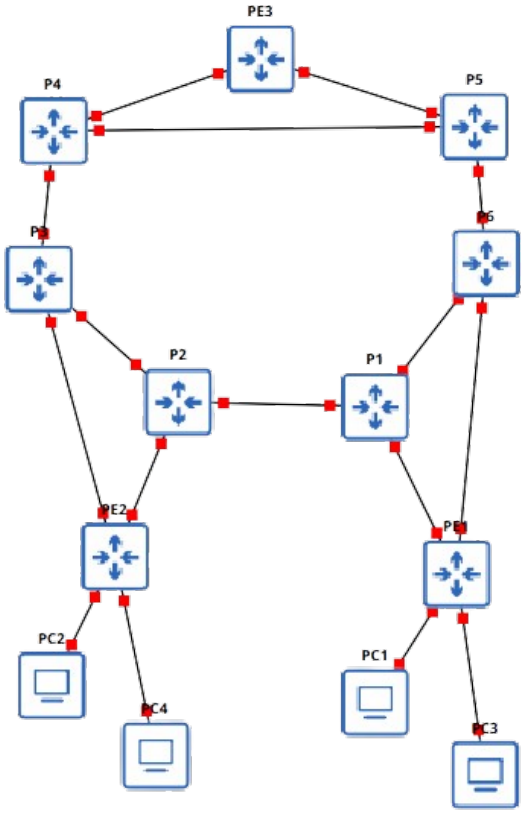

Fig. 2. Final PE/P core and CE hosts used for the MPLS VPN lab

Router loopbacks (/32)

Table 3. Router roles and loopback router–IDs

Router	Role	Loopback0 (/32)
PE1	Provider Edge (PoP 1)	10.0.0.1
PE2	Provider Edge (PoP 2)	10.0.0.2
PE3	Provider Edge (PoP 3)	10.0.0.3
P1	Core LSR	10.0.0.4
P2	Core LSR	10.0.0.5
P3	Core LSR	10.0.0.6
P4	Core LSR	10.0.0.7
P5	Core LSR	10.0.0.8
P6	Core LSR	10.0.0.9

Table 4. All /30 subnets used on core links

Subnet (/30)	PE/P A	IP A	PE/P B	IP B
10.1.11.0	PE1	10.1.11.1	P1	10.1.11.2
10.1.16.0	PE1	10.1.16.1	P6	10.1.16.2
10.1.12.0	P1	10.1.12.1	P2	10.1.12.2
10.1.22.0	PE2	10.1.22.1	P2	10.1.22.2
10.1.23.0	P2	10.1.23.1	P3	10.1.23.2
10.1.25.0	PE2	10.1.25.1	P3	10.1.25.2
10.1.34.0	P3	10.1.34.1	P4	10.1.34.2
10.1.44.0	PE3	10.1.44.1	P4	10.1.44.2
10.1.45.0	P4	10.1.45.1	P5	10.1.45.2
10.1.55.0	PE3	10.1.55.1	P5	10.1.55.2
10.1.56.0	P5	10.1.56.1	P6	10.1.56.2
10.1.61.0	P6	10.1.61.1	P1	10.1.61.2

Table 5. CE–PE links, customer VRFs and subnets

Customer / Site	PE	Interface	VRF	Subnet (/24)
Client A – Site A1	PE1	Gi0/0	CUST_A	192.168.1.0
Client A – Site A2	PE2	Gi0/0	CUST_A	192.168.2.0
Client B – Site B1	PE1	Gi0/1	CUST_B	192.168.10.0
Client B – Site B2	PE2	Gi0/1	CUST_B	192.168.20.0
Client C – Site C1	PE1	Gi0/4	CUST_C	10.10.10.0
Client C – Site C2	PE2	Gi0/4	CUST_C	10.10.10.0
Client D – Site D1	PE1	Gi0/5	CUST_D	10.10.10.0
Client D – Site D2	PE2	Gi0/5	CUST_D	10.10.10.0

These three tables, together with the diagram, give a complete view of the addressing used throughout the MPLS backbone and the VRF-segmented customer VPNs described in the tasks.

4.1 Network Topology & IP Addressing Plan

We deploy **nine** routers: three Provider Edges (PE1–PE3) and six core LSRs (P1–P6) arranged in a redundant ring. Each PE is dual-homed to two adjacent P routers.

- **Loopbacks (/32).** Router-IDs are `10.0.0.1`–`10.0.0.9`.
- **Core Links (/30).** All PE – P and P – P links use unique /30s (Table **??**), simplifying OSPF and LDP.
- **Customer LANs.** Four initial subnets (`192.168.1.0/24`, `192.168.2.0/24`, `172.16.10.0/24`, `172.16.20.0/24`) plus two overlapping subnets `10.10.10.0/24` attached later (Clients C and D).

Loopbacks participate in OSPF area 0 and are chosen as LDP router-IDs and MP-BGP update-sources. Every CE-facing interface remains plain IP (no `mpls ip`); MPLS runs only on core links.

4.2 Basic MPLS "Pipe" (No VRF)

Goal. Prove an LSP between PE1 and PE2 before introducing VRFs.

Steps.

(1) Enable OSPF area 0 on all loopbacks and /30s.
(2) Turn on `mpls ip` & LDP on every core interface.
(3) Verify `show ip ospf neighbor`, `show mpls ldp neighbor` and `show mpls forwarding-table`.
(4) Ping PC1 → PC2 (`192.168.1.2 → 192.168.2.2`); MPLS traceroute shows a two-label stack across P1–P6.
(5) Shut the P3–P4 link to illustrate fast reconvergence, then restore it.

Result. End-to-end ping succeeds both pre- and post-failure; LFIB swaps update automatically, proving the transport LSP is sound.

4.3 Single-Customer VPN (Client A)

VRF Definition. `ip vrf CUST_A` RD `65000:1`; RT `65000:1:1` (export & import identical).

Configuration Highlights.

- Move PE1 Gi0/0 and PE2 Gi0/0 into `CUST_A`.
- Inject static routes for `192.168.1.0/24` and `192.168.2.0/24`.
- Activate `address-family ipv4 vrf CUST_A` under BGP 65000 on both PEs; `vpnv4` AF already present.

Verification. `show ip route vrf CUST_A` lists the remote prefix as a BGP route; PC1 pings PC2 successfully over a two-label VPN stack.

4.4 Multi-Customer VPNs (Clients A & B)

Add `CUST_B` (RD `65000:2`, RT `65000:2:2`):

(1) Place PE1 Gi0/1 (`192.168.10.0/24`) and PE2 Gi0/1 (`192.168.20.0/24`) into `CUST_B`.
(2) Static default routes from each CE to its PE.
(3) Activate `ipv4 vrf CUST_B` in BGP on both PEs.

Result. `show ip route vrf CUST_A` and `show ip route vrf CUST_B` display distinct tables; PC3 (`192.168.10.2`) reaches PC4 (`192.168.20.2`) while inter-VRF pings fail, confirming isolation.

4.5 Overlapping Prefixes on a Shared PE (Clients C & D)

Two new VRFs on *the same* PE1:

Client	VRF	RD / RT
C	`CUST_C`	`65000:3`
D	`CUST_D`	`65000:4`

- Both attach to identical subnet `10.10.10.0/24`.
- PE1 Gi0/4 → PC5 (C); PE1 Gi0/5 → PC6 (D).
- PE2 provides corresponding remote sites (PC7, PC8).

Key Points.

(1) Static routes for `10.10.10.0/24` installed per VRF.
(2) BGP advertises *two* VPNv4 NLRI, each carrying the same prefix but distinguished by unique RDs.
(3) PC5 ↔ PC7 **works**; PC6 ↔ PC8 **works**; cross-VRF traffic **blocked**.

Outcome. Overlapping customer address space is fully isolated, validating the use of RDs/RTs for multi-tenant separation on a single PE.

Summary

The lab confirms that the MPLS transport built in Task 2 carries traffic end-to-end via resilient label-switched paths. MP-BGP successfully advertises VPNv4 prefixes, allowing the backbone to scale cleanly to multiple Layer-3 VPNs. Because each VRF is defined with its own route distinguisher and route targets, customer traffic remains strictly segregated—even when overlapping address spaces share the same PE. Collectively, these results demonstrate that all five tasks have been satisfied within a single, coherent MPLS-VPN design.

5 Conclusion

This practical build confirms that even a modest nine-router lab can replicate most of the moving parts found in production MPLS VPN networks:

- **Reliability**: Dual-homed PEs and a six-node ring ensure single-fault tolerance with sub-second reroute in OSPF/LDP.
- **Scalability**: MP-BGP cleanly separates customer routes via RDs/RTs, allowing overlapping prefixes without leakage.
- **Operational Clarity**: Using /30 links and `10.0.0.0/32` loopbacks provides deterministic router-IDs and simplifies troubleshooting commands.
- **Re-usability**: The provided CLI snippets and tables are generic enough to seed future labs—QoS, TE, Segment Routing, or RSVP-TE LSPs—by simply appending new sections.

Future work could explore automated provisioning (e.g. Ansible or Netmiko), traffic-engineered tunnels, or integrating a route reflector for larger-scale topologies. Nevertheless, the exercises herein already demonstrate a full service life-cycle—from blank topology to multi-customer VPN with proven fail-over—fulfilling all objectives of the lab brief.