



VyOS
Networks



/ SOLUTION BRIEF

VYOS WITH VPP DATAPLANE

High-Performance Packet Processing for Modern Networks



Introduction

The evolution of network virtualization and cloud-native infrastructures is driving the need for routing platforms that deliver extreme performance, predictable latency, and flexible programmability. VyOS, a fully open-source network operating system, integrates **Vector Packet Processing (VPP)** as an optional high-speed dataplane to dramatically accelerate packet forwarding.

VyOS offers two forwarding architectures: the traditional Linux kernel dataplane and the VPP userspace dataplane. VPP is a high-performance, multi-core-optimized packet processing engine capable of delivering significantly higher throughput and lower latency for demanding network scenarios.

By leveraging VPP, VyOS becomes an even more powerful platform for next-generation routing, advanced firewalling, and service-chaining use cases, where performance, scalability, and deterministic behavior are critical.

Key Characteristics of VPP Dataplane on VyOS

1. High-Speed User-Space Packet Processing

VPP bypasses the Linux kernel network stack, enabling millions of packets per second (Mpps) per core through vectorized packet operations.

2. Multi-Core Scalability

VPP is designed to scale horizontally. VyOS can leverage multiple CPU cores to distribute workloads, improving throughput under heavy traffic.

3. Flexible Plugin Architecture

VPP provides modular plugins for routing, encapsulation, NAT, ACLs, tunneling protocols, and more, easily integrated into VyOS workflows.

4. Hardware Offload and DPDK Integration

VPP supports DPDK NIC drivers, allowing VyOS to take advantage of NIC hardware queues, SR-IOV, and advanced offloading capabilities.

5. Cloud-Ready and Virtualization-Friendly

VPP improves performance on hypervisors such as KVM, ESXi, and cloud environments by reducing overhead and maximizing NIC efficiency.

5. Seamless VyOS Configuration Experience

VyOS maintains its familiar CLI/API while using VPP under the hood, ensuring operational simplicity and consistency.

When to Use VPP

Consider VPP if you have:

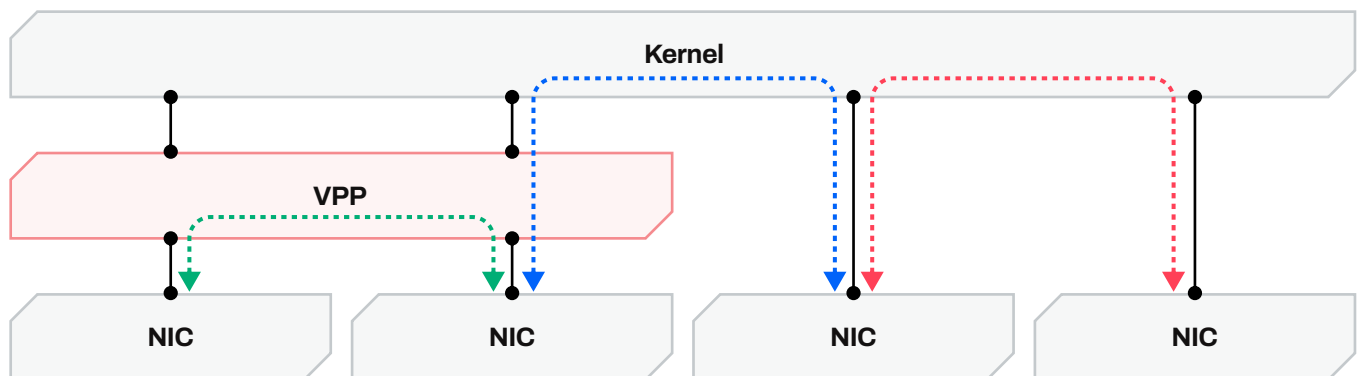
- High-throughput requirements
- Latency-sensitive applications requiring consistent performance

Stay with kernel dataplane if you have:

- Low to moderate traffic volumes
- No latency-sensitive workloads
- Applications requiring specific features not supported by VPP Dataplane

Packets Processing Integration Details

VPP Dataplane integration is done in a way that minimizes configuration changes. Features that exist in kernel dataplane are not removed but continue to operate in kernel dataplane. VPP Dataplane only takes over packet forwarding for interfaces explicitly assigned to it.



Green path

Traffic between two VPP interfaces is processed entirely within VPP for maximum performance. Packets that follow this path can use only features available inside VPP dataplane.

Blue path

Traffic between a VPP interface and a kernel interface is processed by both dataplanes, with VPP handling the VPP side and the kernel handling the kernel side. Packets that follow this path can use features available in both VPP and kernel dataplanes, at the same time. Because packets must follow both dataplanes, performance will be slower than with pure VPP or pure kernel forwarding.



| Red path

Traffic between two kernel interfaces is processed entirely within the kernel dataplane. Packets that follow this path can use only features available inside kernel dataplane, and lack VPP acceleration. This is the traditional VyOS dataplane operation.

Use Cases

■ Virtual Router and NFV Deployments

Ideal for service providers and enterprises requiring high-performance virtual routers, CGNAT gateways, and L3/L4 services.

■ SD-WAN and Edge Appliances

Improves throughput for encrypted tunnels, inter-site connectivity, and edge routing without requiring proprietary hardware.

■ 5G and Telco Cloud

Supports high packet rates, low latency, and service chaining architectures used by telecommunications operators.

■ Cloud and Multi-Cloud Interconnects

Enables scalable VPC/VNet routing, VPN hubs, and inter-cloud gateways with strong performance on virtual machines.

■ High-Performance Firewalling and NAT

VPP's ACL and NAT plugins allow VyOS to deliver predictable performance for stateful packet processing workloads.

Benefits



Massive Performance Gains

Achieve higher forwarding throughput compared to traditional kernel-based dataplanes.



Operational Consistency

Use the same VyOS CLI/API and automation tooling (Ansible, Terraform, PyVyOS) while benefiting from VPP acceleration.



Better Resource Efficiency

Higher throughput with fewer CPU cycles reduces infrastructure costs on both bare metal and virtualized environments.



Cloud-Optimized

Delivers near-line-rate performance in cloud environments—critical for virtual routers, VPN hubs, and NAT gateways.

Flexible Network Services

Built-in support for routing, bridging, encapsulation, ACLs, NAT, and tunnel termination with high efficiency.

Conclusion

Integrating VPP with VyOS transforms the platform into a high-performance, cloud-ready network appliance capable of supporting demanding workloads across enterprise, service provider, and telco environments. The VPP dataplane accelerates packet processing, enhances scalability, and maintains the operational simplicity that VyOS users expect.

For organizations seeking an open-source, automation-friendly, and performance-optimized routing solution, VyOS with VPP represents a powerful and future-proof choice.