Tuning vCloud NFV for Data Plane Intensive Workloads

VMware vCloud NFV OpenStack Edition 3.3
VMware vCloud NFV 3.2.1

vmware®

You can find the most up-to-date technical documentation on the VMware website at:

https://docs.vmware.com/

If you have comments about this documentation, submit your feedback to

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Contents

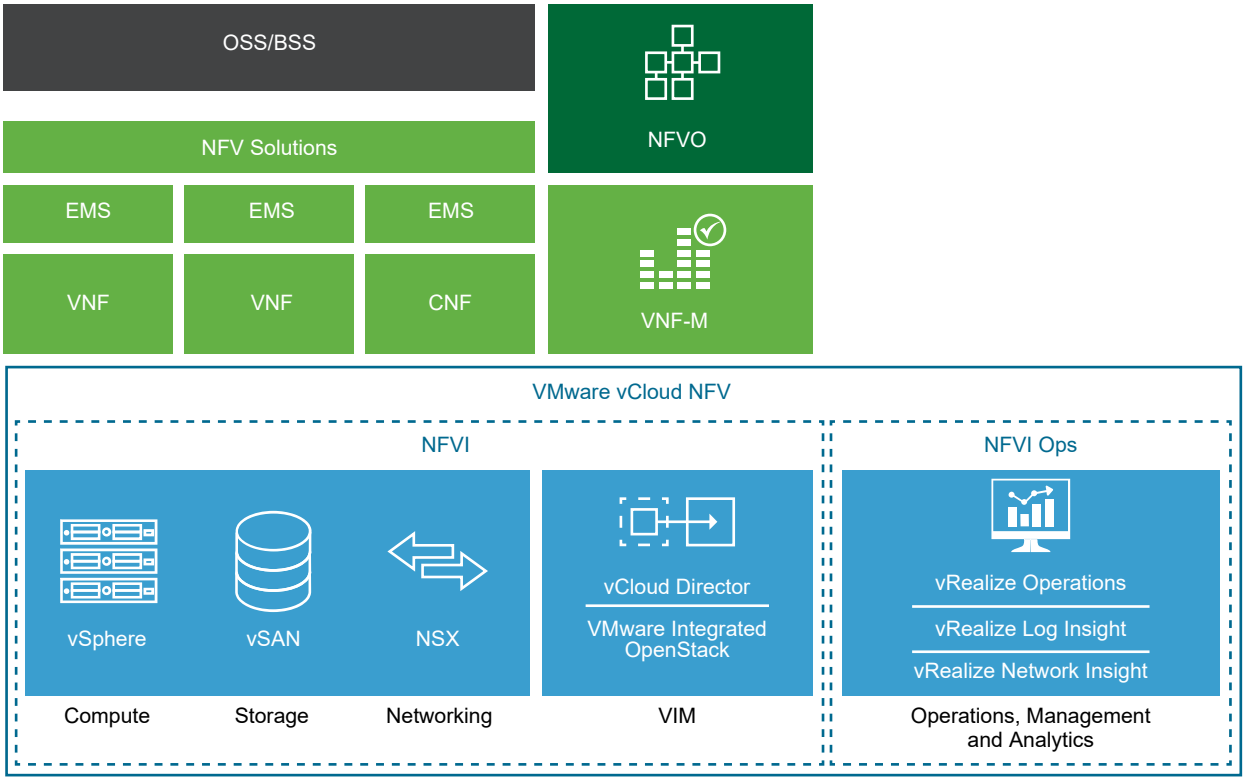
About vCloud NFV

1

VMware vCloud® NFV™ combines a carrier-grade NFV Infrastructure (NFVI) with Virtualized Infrastructure Manager (VIM). VIM comprises a rich set of APIs with stable and supportable vCloud NFVI. Thus, vCloud NFV provides a platform to support Communication Service Providers (CSPs) in realizing the goal for network modernization and business transformation.

The vCloud NFV platform implements a modular design with abstractions that enable multi-vendor, multi-domain, and hybrid (physical and virtual) execution environments. The platform also delivers an automation framework to interoperate with external functions for service orchestration and management. In addition to VIM and the core NFV infrastructure components for compute, storage, and networking, the vCloud NFV platform includes a fully integrated suite for operational intelligence and monitoring. This suite can be used to enhance the runtime environments with workflows for dynamic workload optimization and proactive issue avoidance.

Figure 1-1. vCloud NFV Platform



Introduction 2

The vCloud NFV platform supports CSPs in operating a cost-effective and operationally robust NFV infrastructure. vCloud NFV 3.0 introduced significant performance improvements to the platform's data plane capabilities. VMware NSX-T™ Data Center, which is included in the vCloud NFV platform from 3.0, introduced a new networking stack. NSX-T Data Center delivers substantial performance improvements to the switching fabric of the platform through the NSX-managed Virtual Distributed Switch (N-VDS) in Enhanced data path mode and NSX Bare Metal Edge (NSX BM Edge). The NSX-T Data Center networking stack is resource efficient and provides a clear separation between the virtual networking and the Virtual Network Functions (VNFs) resources.

This chapter includes the following topics:

Scope of this Guide

This 'Tuning vCloud NFV for Data Plane Intensive Workloads' guide focuses on the use of N-VDS in Enhanced data path mode, referred to as N-VDS (E), for both VLAN and Overlay networking. The use of N-VDS (E) provides the NFV infrastructure operator (CSP) the benefits of virtualization, while maintaining high data plane performance. For this reason, topics such as Single Root Input/Output Virtualization (SR-IOV) and other virtual switch bypassing technologies are not discussed in this guide.

This guide focuses on the lowest layers in the NFV infrastructure:

- Server and its components (CPUs, NICs)
- ESXi hypervisor
- N-VDS (E) in the NSX-T Data Center
- NSX Bare Metal Edge Server
- Relevant Virtual Network Function (VNF) components configuration

The use of the Management and Orchestration (MANO) components to onboard, configure, and deploy VNFs and the use of Virtual Infrastructure Management (VIM) are out of scope for this guide.

This guide focuses on data plane intensive workloads and their characteristics. We do not cover aspects that are not directly related to network acceleration. However, every data plane workload also has requirements for non-accelerated interfaces. These non-accelerated interfaces are not covered in this paper.

Intended Audience

This guide provides information about the data plane capabilities available in the VMware vCloud NFV platform. It builds on the information shared in the previous version of this guide. This guide also provides information about new data plane capabilities introduced in vCloud NFV 3.2.1 and vCloud NFV OpenStack Edition 3.3.

This guide is intended for two target groups:

- **Communication Service Providers**: Explains the design and considerations for building the vCloud NFV platform to support data plane intensive workloads.
- **Virtual Network Function vendors**: Assists the VNF vendors in understanding the configuration parameters a VNF can use to leverage the accelerated data plane in the platform.

We assume that the reader is familiar with the following, depending on the vCloud NFV edition that is used:

- vCloud NFV OpenStack Edition 3.3 Reference Architecture and Release Notes.
- vCloud NFV 3.2.1 Release Notes. Reference Architecture for vCloud NFV 3.0 is applicable and should be referred.

Acronyms

This table describes the NFV and industry-specific acronyms used in this guide:

Acronym	Definition
NSX BM Edge	NSX Bare Metal Edge
CSP	Communication Service Provider
DPDK	Data Plane Development Kit, an Intel led packet processing acceleration technology.
LACP	Link Aggregation Control Protocol
LAG	Link Aggregation Group
LRO	Large Receive Offload
MANO	Management and Orchestration components, a term coined by ETSI NFV.
NAT	Network Address Translation
NIC	Network Interface Card

Acronym	Definition
NF	Network Function, a generalized term to refer to virtualized and containerized packaging.
NFVI	Network Functions Virtualization Infrastructure
NUMA	Non-Uniform Memory Access
N-VDS	NSX-managed Virtual Distributed Switch
N-VDS (E)	N-VDS in Enhanced data path mode. This mode enables DPDK for workload acceleration.
N-VDS (S)	N-VDS in Standard mode
NSX DFW	NSX Distributed Firewall
OAM	Operations, Administration, and Management
PCIe	Peripheral Component Interconnect Express
PMD	Poll Mode Driver
RSS	Receive Side Scaling
VIM	Virtualized Infrastructure Manager
VNF	Virtual Network Function
VNF-C	VNF Component, the building block of VNFs.

Workload Categories

Many CSPs adopted vCloud NFV from early release versions of the platform. Hundreds of workloads have been deployed in production networks that are set up over the vCloud NFV platform. As CSPs expand their NFV environments, an increased number of data plane intensive workloads are being deployed.

Two main categories of telecommunication workloads exist:

- **Control plane workloads**: These workloads typically carry the signaling traffic between various components. Control plane workloads do not participate in the flow of user traffic and do not generate a high networking load. With the growth in user traffic and the increasing number of connections, control plane workloads are expected to generate a high transaction rate. The control plane workload can easily benefit from shared resources. However, operators can select the use of accelerated networking stack for control plane workloads.
- **Data plane workloads**: Depending on the area in the CSP network, data plane workloads are sometimes called user plane or forwarding plane. Data plane workloads carry the user traffic. The end-user applications such as streaming videos, augmented reality, and interactive gaming consume more network bandwidth especially in the context of 5G networks, so the data plane workloads must support an ever-increasing load.

Data plane workloads benefit from the information in this guide. Examples of these workloads include:

- Mobile packet core
- User Plane Function (UPF) in a 5G core

- Virtual routers
- Video optimizers
- Traffic management
- Load balancers

The Importance of Workload Acceleration

A CSP's main business is to offer network services. The components that offer network services have been virtualized in recent years. Standards organization, such as 3GPP, adopted software-focused approach to new development, firmly planting the telecommunication industry in a network that is constructed by general-purpose servers and telecommunications software deployed on those servers.

As CSPs turn to software, the following two goals must be met:

- **Keeping the level of performance in the virtual domain the same as in the physical domain**: Data plane workloads move network data from one place to another. Therefore, most of the processing power of applications is dedicated to some networking logic such as traffic forwarding, switching, encapsulating, decapsulating, and so on. Because data plane workloads actively participate in the services that CSPs offer, consumers immediately experience any inefficiencies or disruptions. As CSPs migrate their physical network functions to the virtual domain, they are looking to maintain the same performance they achieved when the function was implemented on a purpose-built hardware.
- **Accelerating workloads to meet new technology trends**: Workload acceleration plays a significant role in the business case behind data plane networking in several aspects such as:
  - **Reducing resource consumption**: When the data plane workloads and the virtualization layer perform efficiently, CSPs require fewer servers, less power, and less cooling in the data center, as well as fewer personnel to operate the environment.
  - **Providing resource scale and predictability**: Understanding the amount of resources that are required to support a certain workload, allows CSPs to build business models around the cost of deploying a service. The resource predictability also helps CSPs understand the costs that are involved in the success of a service. This is an important aspect of operating a virtual network service. Resource scale and predictability are therefore essential elements to any workload acceleration discussion.

Using NSX Virtual Distributed Switch to Accelerate Workloads

With the introduction of NSX-T Data Center version 2.5 into the vCloud NFV platform, additional capabilities have been added to the platform.

N-VDS is a distributed data plane component instantiated within each ESXi hypervisor kernel that is used for the creation of logical overlay networks, facilitating flexible workload placement of the VNF components. N-VDS is central to network virtualization as it enables logical networks that are independent of physical constructs such as VLANs. N-VDS is a multilayer switch and therefore supports Layer 3 functionality alongside Layer 2.

N-VDS operates in two modes:

- **Standard mode (N-VDS Standard)**: In the standard mode, N-VDS supports the overlay and VLAN-backed networking. In this mode, the resources that N-VDS uses are pooled together with other compute and networking resources that the virtualization layer uses.
- **Enhanced data path mode (N-VDS Enhanced)**: The enhanced mode is aimed at data-plane intensive workloads that require accelerated networking performance and dedicated networking resources. It supports both VLAN and Overlay modes.

N-VDS (E) is an efficient virtual networking stack that uses industry advances such as Data Plane Development Kit (DPDK). By using DPDK principles, N-VDS (E) uses significantly less CPU resources to achieve higher packet throughput rates than previous virtual networking stacks. N-VDS (E) uses dedicated CPU resources assigned to its networking processes. This ensures that the amount of resources that are dedicated to the platform's virtual networking is predetermined by the administrator. By dedicating resources to virtual networking, the resources are clearly separated between the virtualization platform and VNFs. The result of this approach is a simplified configuration for data plane intensive VNF Components (VNF-Cs) and resource usage predictability.

The workload acceleration capabilities of vCloud NFV are contained within the hypervisor kernel-space. This approach, as opposed to the user-space implementation, provides several important benefits to the user:

- **Security**: The accelerated networking stack exists entirely within the hypervisor kernel-space and is therefore not susceptible to the manipulations to which user-space processes are exposed.
- **Feature support**: Existing platform capabilities and features such as vMotion and DRS continue to function with the new networking stack.
- **Zero workload impact**: NSX-T Data Center and N-VDS are transparent to the workloads running on top of the virtual layer. Data plane intensive workloads that leverage DPDK are not required to make any changes to their applications to use N-VDS (E).

Designing the NFV Infrastructure 3

The Communication Service Provider (CSP) is the NFVI operator. The CSP creates the NFV environment and is usually responsible for its operation.

In a typical NFV environment, application (VM or container form) vendors do not have the access to the infrastructure management components such as NSX-T Manager and vCenter Server. The CSP cloud administrator creates logical resource entities for each tenant. In the deployment model that NFV customers often use, a tenant is the construct where a workload is deployed. All interactions between the virtual infrastructure and the VNF are performed through the available cloud management interfaces from within the workload tenancy. It is therefore the responsibility of the CSP to perform all host-level configurations for data plane intensive workloads. The CSP procures the servers, configures the hardware and software, and assigns the resources to virtual networking. VNFs consume these resources as needed.

This chapter includes the following topics:

Best Practices for the Physical Layer

The following best practices are for building a physical NFV infrastructure aimed at hosting data plane intensive workloads:

- **Server Homogeneity**: To ease the operations in the NFV environment, the server models and server configurations must be identical. For example, the same NIC and NIC driver must be used in all servers. For data plane workloads, using fast PCIe slots is advisable.

- **NUMA Node Design**: Servers intended for data plane intensive workloads typically have multiple NUMA nodes. The best practice is to align the NUMA nodes identically. If NUMA 0 has two NICs for the data plane, the other NUMA nodes should also have at least two NICs for the data plane applications. This guideline applies to all components that have NUMA locality such as memory and CPU.
- **Choice of NIC**: Before selecting a certain NIC type, we recommend gaining a thorough understanding of the capabilities and performance of that NIC. Details such as throughput performance, overlay offloading capabilities, PCIe speeds, and CPU offload, influence the performance of data plane intensive workloads. In addition, the NIC must have drivers for N-VDS (E). Drivers that support N-VDS (E) are listed in the VMware Compatibility Guide.
- **CPU Speed and Density**: Data plane intensive workloads consume CPU resources to transport the network traffic from the physical NIC to the VNF Components (VNF-Cs). Operations performed inside the VNF-Cs also consume CPU resources. To deploy the data plane intensive workloads successfully and efficiently, use CPUs with a high CPU core count and fast CPU clock frequency. It is also important to consider the performance gains from investing in high-end CPUs with high core count and high CPU clock rates. Sometimes, their cost may not justify the additional CPU cores.
- **Data Plane Traffic Path**: The speed of any network is determined by its slowest link. To drive the maximum performance out of the NFV infrastructure that runs the data plane intensive workloads, NICs, Top of Rack (ToR) switches, and End of Row (EoR) devices must be selected to support high data plane performance. We recommend reserving NIC ports and ToR switch ports for data plane traffic and isolating them from control and management plane traffic.

Accelerated Workloads Host Configuration

When designing the NFV infrastructure to support data plane intensive workloads, consider the following two functions:

- Hosts used to deploy workloads
- Hosts used to deploy the NSX BM Edge functionality

The hosts that support accelerated workloads are expected to use common compute platforms. You must make the correct choice of servers and NICs to support workload acceleration as described in the section ‘Best Practices for the Physical Layer’.

Server configuration, especially, BIOS configuration, greatly influences the data plane performance of the server. For example, the generic or default BIOS settings are often configured for power conservation as opposed to performance. Specifically, BIOS settings that control the CPU play a significant role in making every CPU cycle available to the demanding workloads. As the names of BIOS settings differ between servers and BIOS manufacturers, generic terms are used here.

Server-level configurations that must be tuned are:

- **Power Management**: Set this setting to "High" or "Maximum Performance" (verbiage depending on vendor) to ensure that the CPUs always run at least at the base frequency and use the shallowest idle. The VMworld 2019 “Extreme Performance Series: Performance Best Practices (HBI2526BE)” presentation is an excellent source of information about the Power Management technology. Its major conclusion, however, does not apply to this workload, which may not benefit from higher maximum Turbo Boost frequencies and could be at greater risk of jitter.
- **Hyperthreading**: Enable this setting on systems that support it. Hyperthreading allows a single processor core to run two independent threads simultaneously. On processors with Hyperthreading, each core can have two logical threads that share the core's resources such as memory caches and functional units. BIOS providers might refer to the hyperthreaded core as a ‘Logical Processor’. For more information about hyperthreading, see the VMware vSphere® 6.7 documentation. Hyperthreading can allow more logical CPU threads available for non-data plane intensive workloads and for hypervisor tasks.
- **Turbo Boost**: Enable this setting in the BIOS. It allows the processor to operate faster than the rated frequency for peak loads. For more information about Turbo Boost, see Frequently Asked Questions on the Intel Turbo Boost Technology page of the Intel website.
- **NUMA Node Interleaving**: Ensure that this setting is disabled. With the NUMA node interleaving setting enabled, the hypervisor sees the available memory as one contiguous area. Therefore, the ability to place memory pages local to the CPU is lost and the hypervisor sees all resources on the host as local.

Summary of Recommendations:

- Use common compute platforms.
- Configure BIOS for optimal performance.
- Enable CPU hyperthreading.
- Enable Turbo Boost.
- Disable NUMA node interleaving.

Accelerated Workloads and Overlay Topologies

To leverage the benefits of Overlay network topologies, the NSX BM Edge Node is used as the entry and exit point from the virtual to the physical domain. To achieve independence from the underlying physical network, GENEVE encapsulation is used. For more details about the NSX BM Edge and Edge Pod, see the vCloud NFV Reference Architecture.

The Edge Pod is expected to process all data traffic, so it is essential to choose a high-end server for this function. As a best practice, choose servers with fast CPU with a large number of CPU cores and high-speed Network Interface Cards (NICs). The NSX-T Data Center 2.5 documentation provides a definite list of servers, NICs, and CPU families supported by the NSX BM Edge.

From an operational perspective, the NSX BM Edge could be seen as a black box. The operator can configure services, IP addresses, and other operational aspects, but cannot make changes to hardware configurations such as CPU core allocation. For this reason, the hardware used for the NSX BM Edge is important.

Some key aspects that influence the data plane performance of the NSX BM Edge are:

- **RSS and Flow Cache**: The NSX BM Edge uses DPDK to accelerate its data path (or fast path). It leverages Receive Side Scaling (RSS) and Flow Cache technologies to maximize the throughput performance. By design, the NSX BM Edge expects a large number of active flows to distribute the traffic across the fast path cores.
- **RX/TX Buffer Ring**: The default NIC ring descriptor size for both TX and RX is 512 in the NSX BM Edge. A different configuration might be required for some use cases such as long flows (elephant flows) or bursty traffic.

Performance Scaling in the NSX Bare Metal Edge

As more data plane intensive workloads are added to the data plane cluster facing the Edge Pod, the Edge Pod must support high aggregate throughput. A straightforward approach is to proportionally add more NSX BM Edge nodes in the Edge Pod. Adding more NSX BM Edge servers to the Edge cluster enables higher data plane rates for the workloads. The ratio between the NSX BM Edge nodes and resource hosts is highly dependent on traffic patterns. As a general rule, the NSX BM Edge nodes should use as many high-speed NICs as their hardware can support.

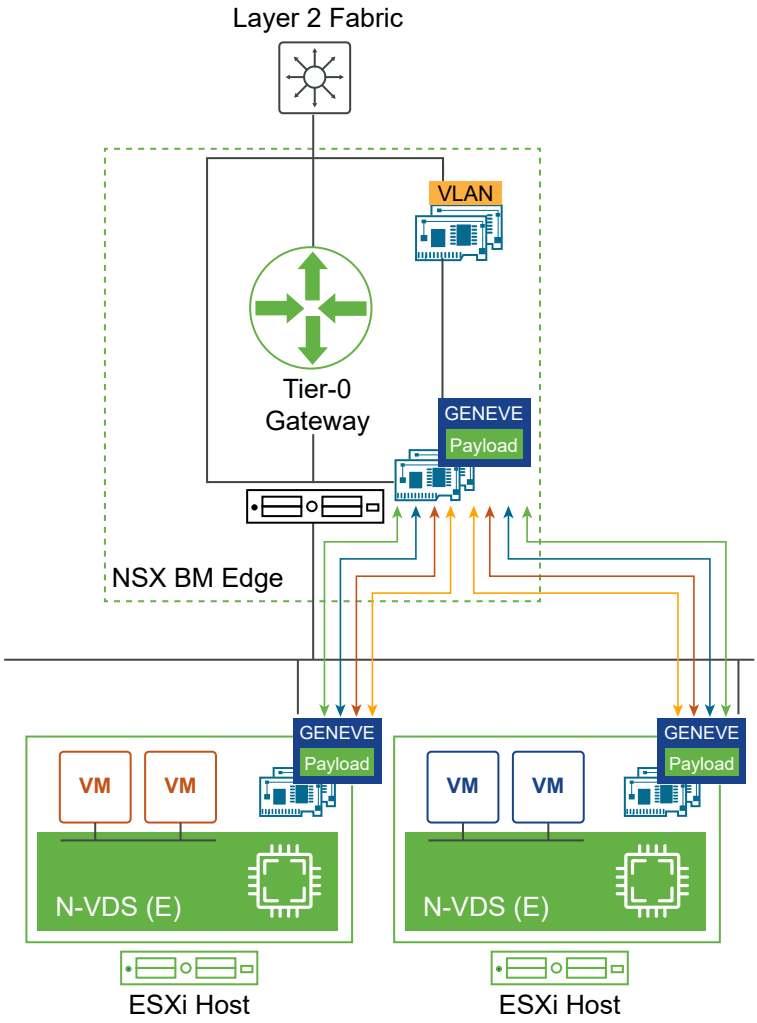
In this section, we focus on the design best practices for scaling the performance of the NSX BM Edge nodes themselves. The goal is to maximize the throughput of each node. The NSX BM Edge throughput can be scaled by adding more NICs to the NSX BM Edge host. It is important that all NICs used for the data plane are installed in single NUMA.

The NSX BM Edge host must have enough PCIe slots to install new NICs. Those PCIe slots should have enough bandwidth to support the maximum throughput of the NICs. We recommend using high capacity NICs in high-bandwidth PCIe slots as most servers have limited number of PCIe slots in a NUMA node.

When you add NICs, you can configure more fast path ports in the NSX BM Edge. This will help scale the NSX BM Edge throughput only if these design considerations are followed:

- Think of the NSX BM Edge uplink ports (towards the Provider network) and downlink ports (towards the ESXi hosts) as comprising a single channel in a NUMA. To scale the capacity of an NSX BM Edge, you must increase the capacity of this channel at both ends. So, if you configure additional fast path ports in the uplink, you must also configure similar capacity with additional ports in the downlink direction. The current release of NSX BM Edge can support such scaling of the channel with more NICs in a single NUMA only.
- When you configure more fast path cores in the NSX BM Edge, you must use link bonding (LACP) between the downlinks and the Top of the Rack switch. For example, if two NIC ports are used for uplink, two NIC ports should be used for downlink. These four NICs must be in the same NUMA. Lastly, the two NIC ports in the downlink direction must use LACP link bonding.

Figure 3-1. Data Plane Acceleration with Overlay Traffic



Summary of Recommendations:

- Use the NICs and CPUs that are supported by the NSX-T BM Edge as listed in the NSX-T Data Center 2.5 documentation.
- Install NICs in PCIe slots with enough bandwidth to support their maximum throughput (for example, PCI Express version 3.0 in an X16 slot).
- For maximum performance, install NICs for the data plane in single NUMA.
- Consider the NSX BM Edge uplink and downlink ports as a channel in a NUMA. Scaling the NSX BM Edge throughput should be considered as scaling the capacity of this channel, by adding more fast path ports to it at both ends.
- Use the LACP link bonding for scaling the bandwidth between downlink and TOR.
- Firewall rules on the NSX BM Edge data path can have a negative impact on performance and hence they should be carefully implemented or avoided.

Best Practices for the Virtualization Layer

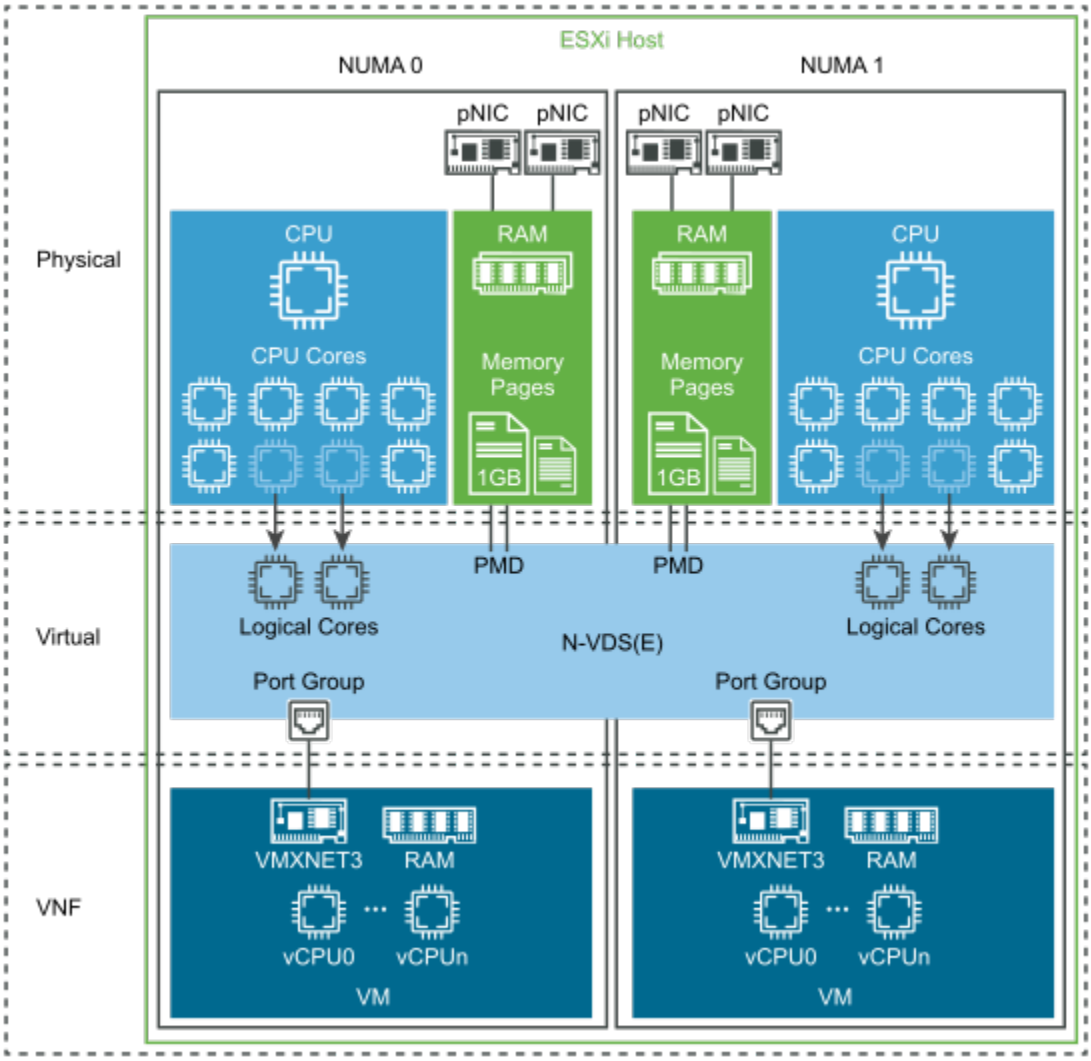
The data plane workloads are deployed in the Resource Pod as per the vCloud NFV reference architecture.

This section focuses on the best practices for data plane workload acceleration building blocks:

- **Virtual Infrastructure**: Each vCloud NFV release is accompanied with a Bill of Materials (BoM) that has been tested and validated. The components and their versions are described in the release notes of every Generally Available (GA) version of the vCloud NFV platform. As a best practice, you must use N-VDS (E) for data plane intensive virtual network interfaces.
- **Cluster**: Data plane intensive workloads typically require dedicated resources to deliver consistent and predictable performance. A cluster is an aggregate of physical servers. So, to make available dedicated resources and to easily manage operations, we recommend using a dedicated cluster for data plane workloads.
- **VNF**: After you set up the infrastructure and configure the virtual domain, a data plane VNF is installed. Data plane VNFs typically consist of multiple VNF Components (VNF-Cs). The VNF vendor provides recommendations for how to deploy the components and how they must communicate with each other, because the VNF vendor is responsible for the performance of the workload. The vendor also specifies the distribution of the components across servers, clusters, and physical racks.
- **Data Plane VNF-C**: VNF-Cs serve distinct functions, such as management, control, and data processing. In this guide, we focus on the data plane VNF-Cs. The data plane VNF-C is designed with performance in mind which typically means that the VNF-C is using a data plane acceleration technology such as Intel’s DPDK. These VNF-Cs are deployed into the data plane intensive cluster and use N-VDS (E) for their data plane vNICs.

The following figure shows an example of NFV workload acceleration building blocks:

Figure 3-2. vCloud NFV Workload Acceleration Building Blocks



Network Data Path Design

Before you configure N-VDS in the Enhanced Data Path mode, consider the following design points:

- Configure the virtual environment according to the relevant vCloud NFV version.
- Assign at least one physical NIC as an uplink to N-VDS.
- For high-availability environments, assign at least two uplinks to N-VDS.
- The physical NICs must use a Poll Mode Driver (PMD) for N-VDS (E).

  Note You can download the NIC driver from the VMware Compatibility Guide. You must install the driver on every host in the vSphere cluster for the physical NICs that are dedicated to N-VDS (E).

- Identify the correct drivers by looking for N-VDS Enhanced Data Path in the feature column of the VMware Compatibility Guide. A physical NIC that is assigned to an N-VDS (E) cannot be used by other virtual switches such as N-VDS (S) and vSphere Distributed Switch.

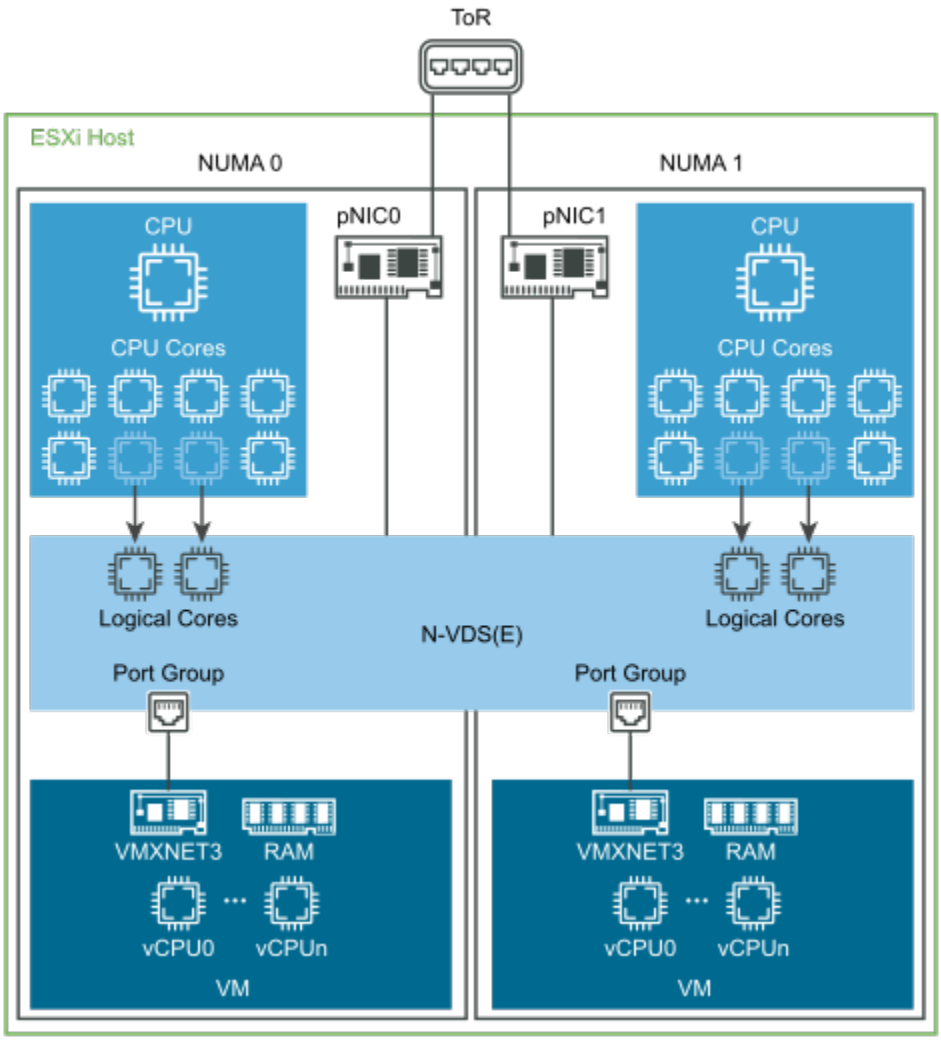
- When designing the networking setup of the server, account for the number of physical NICs that are used for non-accelerated workloads such as vMotion, ESXi management, vSAN network, and so on. The non-accelerated data path is not expected to require high-performance or high-capacity physical NICs.
- N-VDS (E) supports NIC teaming to increase the aggregate networking capacity and resiliency.

CPU Assignment for Network Packet Processing

The host CPU resources play a crucial role in the network performance because the packet processing requires CPU cycles. To provide a predictable throughput performance for data plane intensive workloads, you can assign dedicated CPU cores to an N-VDS (E). These dedicated CPU cores are called logical cores (lcores). The total number of cores remaining for other CPU workloads such as DRS and vSphere operations are reduced after you assign logical cores to N-VDS (E). At the time of publication, a single instance of N-VDS (E) can use up to eight logical cores.

The following figure demonstrates the best practices for NUMA node design. As a design consideration, you should assign logical cores to N-VDS (E) per NUMA node. If you follow the symmetrical design recommendation shown in this section, ensure that each NUMA node is assigned the same number of logical cores. This best practice allows the optimal use of the cluster that is dedicated to data plane intensive workloads.

Figure 3-3. NUMA Node Design for Accelerated Workloads



The number of CPU cores assigned to N-VDS (E) depends on the speed of the uplink physical NICs and the workload type. For example, N-VDS (E) using 40 GbE uplink and serving web-based services where the traffic consists of packets larger than 650 bytes, can completely saturate the NIC with only three logical cores assigned to N-VDS (E). Because each environment is unique and the workload requirements differ, the CSP must evaluate the exact number of logical cores assigned to N-VDS (E).

Automatic N-VDS (E) Logical Core Assignment

The logical cores assigned to N-VDS (E) are in turn assigned to its ports (vNICs). NSX-T Data Center 2.5 introduced new capabilities for automatic assignment of logical cores to vNICs such that dedicated logical cores manage the incoming traffic to and from vNICs.

Every vNIC has two directions for network traffic: one for incoming traffic (Rx) and another for outgoing traffic (Tx). A logical core can be assigned independently to a vNIC direction. When multiple logical cores are configured, the host automatically determines which logical core must be assigned to a vNIC direction.

Two modes are supported for automatic assignment of logical cores to vNIC directions:

- **vNIC-count**: This is the default mode. In this mode, host assumes that processing of incoming or outgoing traffic for any vNIC direction on N-VDS (E) requires the same amount of the CPU resources. Each logical core is assigned the same number of vNIC directions based on the available number of logical cores. The vNIC-count mode is reliable and should be used for most use cases. In cases where the workload is expected to use asymmetrical traffic distributions, the CPU-usage mode could be considered.
- **CPU-usage**: In this mode, host continuously monitors the CPU usage required to process incoming or outgoing traffic at each vNIC direction on N-VDS (E). The host predicts where more resources are needed and changes the logical core assignments to balance the load among logical cores. The CPU usage mode is dynamic and will reach an optimal logical core assignment.

In the CPU-usage mode, if the traffic pattern changes frequently, the predicted CPU resources requirement and the logical core assignment might also change frequently. Frequent logical core assignment changes could cause minimal packet drops.

In the vNIC-count mode, it is recommended to configure an appropriate number of logical cores. The number of logical cores to assign per NUMA, depends on the number of vNICs, the speed of the pNIC, the capabilities of the data plane VNF, and the call model. We recommend that the number of logical cores is always proportional to the number of vNICs so that each logical core can be assigned to a vNIC direction.

When a vNIC is connected or disconnected or when a logical core is added or removed, hosts automatically detect the changes and rebalance the logical core assignment. For information about setting the automatic logical core assignment mode, see the NSX-T Data Center 2.5 Administration Guide.

Summary of Recommendations:

- The number of logical cores assigned to N-VDS (E) should be based on the expected traffic volume, the server's NIC speeds, and the VNF's traffic characteristics.
- CSP must evaluate the exact number of logical cores that should be assigned to an N-VDS (E) per workload.
- Choose the appropriate automatic logical core assignment mode. If the traffic load is symmetric across all the vNIC directions (on the N-VDS (E) switch), use the default vNIC-count mode. If not, use the CPU-usage mode that can deliver more efficient throughput.

NUMA Vertical Alignment

NUMA vertical alignment means that all CPU cores assigned to the VNF-C are located on the same NUMA node as the N-VDS (E) logical cores and uplinks.

NUMA vertical alignment helps with the following aspects:

- Enables faster access between CPUs, caches, and memory pages for an optimal performance.

- Avoids inefficiencies of crossing the Intel Ultra Path Interconnect (UPI) bus between the NUMA nodes.

NSX-T Data Center 2.5 introduced NUMA-aware capability in the Load Balanced Source Teaming policy. Customers can leverage this capability to ensure automatic NUMA vertical alignment for VNF-C.

Load Balanced Source Teaming Policy Mode Aware of NUMA

The NSX Load Balanced (LB) Source Teaming Policy mode defined for N-VDS (E) becomes aware of NUMA when the following conditions for VNF-C are met:

- The Latency Sensitivity configuration for the VNF-C is set to High.
- The network adapter type used by the VNF-C is VMXNET3.

If you deploy and power on a VNF-C that has the above conditions met, the Load Balanced Source Teaming policy triggers NSX to check the VNF-C NUMA placement and aligns its logical cores with the VNF-C NUMA placement.

For deterministic performance of data plane workloads, uniform NUMA node design and NUMA-aware Load Balanced Source Teaming is crucial. This ensures that the same number of logical cores and NICs can be made available to VNF-C, irrespective of the NUMA node to which VNF-C is pinned by the ESXi scheduler.

The Load Balanced Source teaming policy does not consider NUMA awareness to align VNF-Cs and NICs in the following conditions:

- The LAG uplink is configured with NICs from multiple NUMA nodes.
- The VNF-C has affinity to or spans over multiple NUMA nodes.
- The ESXi host failed to determine NUMA information for either VNF-C or NICs.

Under the above conditions or if the Load Balanced Source Teaming policy is not used, NUMA vertical alignment could be enforced manually for the data plane VNF-Cs. This is done by setting the parameter `numa.nodeAffinity` for the VNF-C to the correct NUMA node.

Summary of Recommendations:

- For automatic NUMA vertical alignment, replicate the NUMA node configuration in terms of NIC placement and logical core assignment to N-VDS (E) across all NUMA nodes, and use Load Balanced Source teaming policy.
- If needed, set `numa.nodeAffinity` for VNF-C to the same NUMA node as that of logical cores and NIC.

NSX Distributed Firewall

NSX Distributed Firewall (DFW) provides stateful protection of the workload through the hypervisor-level firewall enforcement. The NSX host preparation operation activates the DFW with the default rule set to allow. This ensures that VM-to-VM communication is facilitated during staging or migration phases.

It is a best practice at the ESXi level to exclude data plane vNICs from the NSX DFW avoiding unnecessary DFW filter to be applied which can impact the data plane throughput. To achieve high data plane throughput, a VNF-C must exclude its data plane vNICs from the NSX DFW rules. This does not require disabling DFW at the global scope, instead this can be achieved at the N-VDS (E) logical port or logical switch level. The NSX DFW supports Exclusion List that allows a logical port, logical switch, or NSGroup to be excluded from the firewall rule.

NSGroups can be configured to contain a combination of IP sets, MAC sets, logical ports, logical switches, and other NSGroups. By adding a data plane intensive N-VDS (E) logical port or logical switch to the DFW Exclusion List, you ensure that only those ports and switches are excluded from the DFW. The DFW rules remain active for other workloads that can benefit from micro-segmentation.

Summary of Recommendations:

Add the data plane intensive N-VDS (E) logical ports or logical switches to the NSX DFW exclusion list.

Performance Tuning of Data Plane Workloads

4

With N-VDS (E), configuring a VNF-C for the optimal data plane performance is simplified. A VNF vendor must no longer account for the networking processes that operate on behalf of the VNF-C. N-VDS (E) allocates optimal CPU resources to the networking processes from the available logical cores that are configured at the time the host switch was created.

Starting from vCloud NFV 3.0, when deploying data plane intensive VNFs, only one VM-level parameter is relevant to data plane VMs: `sched.cpu.latencySensitivity`. Setting this parameter to High achieves both CPU pinning, defined as exclusive affinity between vCPU and CPU core, as well as core isolation. CPU core isolation removes the level 1 and level 2 cache contention and the hyperthread contention. Additional benefits to CPU scheduling of this configuration are NUMA awareness and pre-allocation of CPU capacity in oversubscribed hosts.

The `sched.cpu.latencySensitivity` benefits do not stop with CPU. To simplify the configuration of VNF-Cs meant for data plane usage, VNF-C memory is fully reserved and features such as ballooning are automatically disabled. This allows all memory pages to be mapped even before the guest operating system boots. Lastly, setting Latency Sensitivity to High automatically disables Large Receive Offload (LRO), disables vNIC interrupt coalescing, and always try to use a dedicated physical NIC queue for isolation.

This chapter includes the following topics:

VMXNET3 Paravirtualized NIC

The data plane vNIC must use the paravirtual VMXNET3 driver because it has improved performance as compared to other virtual network interfaces.

VMXNET3 provides several advanced features including multi-queue support, Receive Side Scaling (RSS), LRO, IPv4 and IPv6 offloads, and MSI and MSI-X interrupt delivery. By default, VMXNET3 also supports the interrupt coalescing algorithm. Virtual interrupt coalescing helps drive a high throughput to VMs with multiple vCPUs with parallelized workloads (for example, multiple threads), whereas at the same time striving to minimize the latency of the virtual interrupt delivery.

To benefit from the performance and efficiency offered by N-VDS (E), a data plane intensive VNF-C must always use a poll mode driver (PMD) for its VMXNET3 vNIC.

Since the release of Linux 2.6.32 in October 2009, VMXNET3 is included in several modern Linux distributions such as Red Hat Enterprise Linux and SUSE Linux. At the time this guide is written, the latest VMXNET3 driver version 1.4.17.0-k is up-streamed to Linux and is available from kernel version 5.3.

For more information about the DPDK support in the VMXNET3 driver, see the article Poll Mode Driver for Paravirtual VMXNET3 NIC.

Summary of Recommendations:

- Use the latest paravirtualized VMXNET3 vNIC driver.
- Install and use the latest DPDK PMD for VMXNET3.

Virtual Machine Hardware Version

Ensure that a VNF-C uses the most up-to-date virtual machine hardware version. The virtual machine hardware version reflects the supported virtual hardware features of the VM. These features correspond to the physical hardware that is available on the ESXi host where the VM runs.

Virtual hardware features include the BIOS and Extensible Firmware Interface (EFI), the available virtual PCI slots, the maximum number of CPUs, the maximum configurable memory, and other typical hardware characteristics.

Different VM hardware versions support different components and different amounts of resources, so you must use uniform VM hardware versions for all VMs comprising a VNF. For example, the VM hardware version 14 supports a maximum of 6128 GB of RAM for a VM, whereas the VM hardware version 11 supports up to 4080 GB of RAM.

VM hardware versions also enable processor features, so the best practice is to use the latest virtual hardware version to expose the new instruction sets to the VM. Avoid mismatches between VNF-Cs configured with different VM hardware versions because it impacts the performance.

- The differences in the virtual hardware versions are listed in Hardware features available with virtual machine compatibility settings (2051652).
- The latest VM virtual hardware versions and their matching hypervisor versions are listed in Virtual machine hardware versions (1003746).
- For information about upgrading a virtual hardware version, see Upgrading a virtual machine to the latest hardware version (multiple versions) (1010675).

Summary of Recommendations:

- Use the latest virtual machine hardware version.
- Use a uniform virtual machine hardware version across all the VMs that comprise a VNF.

Dedicating CPUs to a VNF Component

To ensure pinning and exclusive affinity of all vCPUs in the data plane VNF-C, set the VM configuration parameter `sched.cpu.latencySensitivity` to **`High`**. This configuration also disables the interrupt coalescing and Large Receive Offload (LRO) automatically, which is not an issue, because data plane intensive VNF-Cs use DPDK and VMXNET3 DPDK poll mode driver for their data plane intensive vNICs.

To set a VNF-C for CPU affinity, follow these steps:

1 In the **vSphere Web Client**, right click the VM and select **Edit Settings**.

2 Select **Virtual Hardware**.

3 Set the CPU and memory reservations of the VM to their maximal value.

4 Set the CPU limit to Unlimited.

5 Click **VM Options** and select **Advanced**.

6 Scroll down to **Latency Sensitivity** and select **High** from the drop-down menu.

Important The vCPUs are pinned and receive an exclusive affinity to a complete CPU core as opposed to a hyperthread. This functionality reduces the risk of two heavy processes competing for a hyperthread resource on the same CPU core, therefore reducing the effective CPU cycles available to each vCPU.

Summary of Recommendations:

- Set the VM CPU and memory reservation to maximum and the CPU limit to unlimited.
- Set `sched.cpu.latencySensitivity` to **`High`** for data plane intensive workloads.

Physical NIC and vNIC Ring Descriptor Size Tunings

Selecting ring sizes strongly depends on the VNF workload. It is expected that VNF vendors benchmark their VNFs to understand the optimal ring size configuration.

The VMXNET3 DPDK rings are intrinsic to VNF-C which means that CSPs may not be able to configure these parameters. Larger ring sizes provide an improved resiliency against the packet loss. However, using bigger ring size potentially increases the memory footprint which also potentially incurs a performance penalty.

Summary of Recommendations:

- Use the NIC ring descriptor size with which packet loss is minimized.
- Use ring sizes in ascending order through the traffic path.

Huge Pages

The virtual infrastructure supports backing VNF-C Guest OS memory with 1 GB Huge Pages for memory intensive and large memory VNF-Cs. VNF-Cs enabled with Huge Pages can benefit from reduced frequency of CPU cache miss and page fault, which can significantly improve the VNF-Cs performance.

To use 1 GB Huge Pages to back VNF-C guest operating system memory, set `sched.mem.lpage.enable1GPage = TRUE` to the VM. For more details, see the VMware vSphere 6.7 documentation.

Before you enable Huge Pages:

- Ensure that the VM has the full memory reservation so that all the vRAM can be pre-allocated at the time of power-on.
- Consider the resource availability for other workloads in the cluster and for cluster operations such as HA and DRS.

To set Huge Pages, follow these steps:

1 In the vSphere Web Client, power-off the VM.

2 Right click the VM and select **Edit Settings**.

3 Click **VM Options** and select **Advanced**.

4 Scroll down to **Configuration Parameters** and select **Edit Settings**.

5 Click **Add Configuration Params** and enter `sched.mem.lpage.enable1GPage = TRUE`.

Summary of Recommendations:

- Set `sched.mem.lpage.enable1GPage = True` to enable Huge Pages.

VNF-C vNIC Scaling

You can scale up the data plane performance of a VNF-C by increasing the number of vNICs used by the VM. Note the following design considerations:

- Increase the number of vNICs as opposed to trying to match the speed of the physical NIC with a virtual NIC. Spreading high traffic load across several vNICs allows the VNF-C to gain more efficiency from the CPU cores available on the host.
- When scaling a VNF-C, maintain a balance between the number of vNICs and CPU cores used by the VNF-C and the resources required to fully use the physical NIC.

Summary of Recommendations:

- To scale the data plane performance of a VNF-C, increase the number of vNICs and add two logical cores to N-VDS (E) per vNIC.

To summarize this section, the relevant performance tuning that must be considered for a data plane intensive workload are:

- Using the latest VMXNET3 paravirtualized vNICs driver
- Using the latest virtual machine hardware version
- Dedicating CPU cores to the data plane intensive VNF-C
- Configuring `sched.cpu.latencySensitivity = High` in the data plane VNF-C
- Using Huge Pages
- vNIC scaling in Data plane intensive VNF-C

Conclusion

5

With every release of the VMware vCloud NFV platform, the aim is to improve the data plane performance and predictability.

With the inclusion of NSX-T Data Center 2.5 in vCloud NFV 3.2.1 and vCloud NFV OpenStack Edition 3.3, VMware has:

- Added support for accelerated end-to-end Overlay networking.
- Provided more mechanisms to efficiently assign compute resources to the virtual networking stack.
- Further improved the ease of use by leveraging the automatic NUMA alignment.

We collaborate with several Network Equipment Providers (NEPs) to ensure that the data plane intensive workloads can benefit from the VMware Telco Platform. We confirmed that the VMware Telco Platform is indeed very capable of supporting data plane workloads. For more information, see the following VMworld presentations:

- Achieving High Performance with NFV Data Plane Workloads: The Secrets (T5G1105BE)
- Ericsson Virtual Packet Gateway on vCloud NFV: Cooperation and Optimization (T5G2264BE)

Summary of Recommendations:

This section provides the summary of configuration information discussed in this guide and from additional resources.

Note Before applying any of the configuration parameters, you must test them in a lab when measuring the effect on the environment and the VNF performance.

This chapter includes the following topics:

BIOS Configuration

Different BIOS manufacturers use different names for their BIOS functions. This guide uses general naming conventions wherever possible. Refer to your server documentation for exact configuration details.

The following table describes BIOS parameters and their configuration details:

Parameter	Value
Power Management	Maximum Performance / High Performance
Hyperthreading	Enable
Turbo Boost	Enable
C-States	Disable
Intel VT Technology	Enable
QPI Power Management	Disable
Execute Disable Bit	Enable
Node Interleaving	Disable

Hypervisor

The following table describes the Hypervisor parameters and their configuration details:

Parameter	Value	Configuration
N-VDS Enhanced NIC Driver	Identify your NIC's driver by searching for 'N-VDS Enhanced Data Path' in the feature column of the VMware Compatibility Guide.	Not relevant
Physical NIC's firmware	Identify your NIC's firmware in the VMware Compatibility Guide. **Note**: Check the hardware vendor for the latest version.	Not relevant. Based on hardware vendor instructions.

Virtual Machine (VNF-C) Configuration

The following table describes the VNF-C configuration details:

Parameter	Value	Configuration
	Virtual CPU	
Reservation	Maximum per number of vCPU allocated to VM	Go to **VM Settings** > **Virtual Hardware** > **CPU** > **Reservation**. Set the value to number of vCPUs multiplied by processor base frequency.
	Memory	

Parameter	Value	Configuration
Shares	High	Go to **VM Settings** > **Virtual Hardware** > **Memory** > **Shares**. Set the value to High.
Reservation	Maximum Necessary	Go to **VM Settings** > **Virtual Hardware** > **Memory** > **Reservation**. Set the value to Maximum, as required.
Limit	Unlimited	Go to **VM Settings** > **Virtual Hardware** > **Memory** > **Limit**. Set the value to Unlimited.
	VM-level Configuration	
Hardware Version	Version 15 for ESXi 6.7U2/U3	Right-click the VM, select **Compatibility** > **Upgrade VM Compatibility**.
Virtual NIC for Data Plane Interfaces	VMXNET3	Set VMXNET3 as the vNIC adapter type.
Latency Sensitivity	Set to High	Go to **Edit Settings** > **VM Options** > **Advanced** > **Configuration Parameters**. Set `sched.cpu.latencySensitivity = high`.
Huge Pages	Set to TRUE.	Go to **Edit Settings** > **VM Options** > **Advanced** > **Configuration Parameters**. Set `sched.mem.lpage.enable1GPage = TRUE`.

References

6

- vCloud NFV 3.2.1 Release Notes
- vCloud NFV 3.0 Reference Architecture
- vCloud NFV OpenStack Edition 3.3 Release Notes
- vCloud NFV OpenStack Edition 3.3 Reference Architecture
- Tuning vCloud NFV for Data Plane Intensive Workloads
- Performance Best Practices for VMware vSphere 6.7
- VMware NSX-T Data Center 2.5 Installation Guide

Authors and Contributors

7

The following authors co-wrote this guide:

- Jambi Ganbar, Director, Telco Partner Solutions, VMware
- Piyush Kumar Singh, Solution Architect, Telco Solutions, VMware
- Ramachandran Sathyanarayanan, Solution Consultant, Telco GTM, VMware
- Sorin Amihalachioaie, Solution Architect, Telco Solutions, VMware

Many thanks for contributions from:

- Ramkumar Venketaramani, Director, Solutions Management, Telco Edge Cloud, VMware
- Revathi Govindarajan, Technical Writer, Telco Solutions, VMware
- T. Sridhar, Principal Engineer & Chief Architect, Telco Solutions, VMware
- Valentin Bondzio, Senior Staff Technical Support Engineer, VMware