Tuning vCloud NFV for Data Plane Intensive Workloads

VMware vCloud NFV OpenStack Edition 3.0
About Tuning vCloud NFV for Data Plane Intensive Workloads

1 Introduction to vCloud NFV

2 Introduction
   Workload Definition
   The Importance of Workload Acceleration
   Workload Acceleration in vCloud NFV

3 Introduction to Workload Acceleration
   Workload Acceleration Building Blocks
   Workload Acceleration Example

4 Designing the NFV Infrastructure
   Host Configuration
   Configuring N-VDS Enhanced Data Path
   CPU Assignment for Network Packet Processing
   Achieving NUMA Vertical Alignment

5 Accelerating Data Plane Workloads
   VMXNET3 Paravirtualized NIC
   Virtual Machine Hardware Version
   Dedicating CPUs to a VNF Component
   Physical NIC and vNIC Buffer Size Tunings
   Huge Pages
   VNF Component vNIC Scaling

6 Conclusion

7 Summary of Recommendations

8 Acronyms and Definitions

9 References

10 Authors and Contributes
About Tuning vCloud NFV for Data Plane Intensive Workloads

The Tuning vCloud NFV for Data Plane Intensive Workloads white paper provides best practices and recommendations for designing and configuring the vCloud NFV platform for data plane intensive workloads. The document focuses on using the N-VDS logical switch in Enhanced datapath mode. N-VDS Enhanced is a new feature of NSX-T Data Center, which is the networking stack of the vCloud NFV platform starting from version 3.0. N-VDS Enhanced provides the NFV infrastructure operator the benefits of virtualization and maintains very 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.

The document focuses on the lowest layers in the NFV infrastructure, the physical server and its components, the ESXi hypervisor, the use of N-VDS Enhanced, and the configuration of the relevant Virtual Network Function Components (VNF-Cs).

The use of the Management and Orchestrations (MANO) components to onboard, configure and deploy Virtual Network Functions (VNFs), and the use of the Virtualized Infrastructure Management (VIM) are out of scope for this white paper.

Because this document builds on the information shared in the previous white paper, Tuning vCloud NFV for Data Plane Intensive Workloads, we recommend familiarizing yourself with that paper first.

Intended Audience

This document provides information about the data plane capabilities that are available in vCloud NFV starting from version 3.0. The paper is intended for two target groups:

- Communication Service Providers (CSPs). The document explains the design and considerations for building a vCloud NFV platform to support data plane intensive workloads.
- VNF vendors. This document assists the VNF vendors in understanding the configuration parameters that a VNF can use to leverage the capabilities of N-VDS Enhanced for workload acceleration.
Introduction to vCloud NFV

VMware vCloud® NFV™ is a modular, multi-tenant network functions virtualization (NFV) platform. It provides the compute, storage, networking, management, and operations capabilities to enable Communication Service Providers (CSPs) to provide virtualized network services.

NFV is an architectural framework that is first developed by the ETSI NFV Industry Specification Group (ISG). The framework provides a reference model where network functions are delivered through software virtualization with commercial off-the-shelf (COTS) hardware. This way, NFV moves away from the proprietary, purpose-built hardware that is dedicated to a single service. The result is a network that is agile, resilient, and equipped to deliver high quality of services.

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 the core NFV infrastructure components for compute, storage, networking, and Virtualized Infrastructure Manager (VIM), the vCloud NFV platform includes a fully-integrated suite for operational intelligence and monitoring. This suite can be used to further enhance the runtime environments with workflows for dynamic workload optimization and proactive issue avoidance.
Figure 1-1. vCloud NFV Components

The vCloud NFV OpenStack Edition components, their interactions with each other, and how they meet CSP requirements, are described in this reference architecture.
The VMware vCloud NFV platform is developed to support CSPs in operating a cost-effective and operationally robust NFV infrastructure. Version 3.0 of vCloud NFV introduces significant performance improvements in the platform’s data plane. For the first time, NSX-T Data Center is included with vCloud NFV 3.0 thus introducing a new networking stack to the vCloud NFV platform. NSX-T Data Center delivers substantial performance improvements to the switching fabric of the platform through a specific component that is called N-VDS Enhanced. The NSX-T Data Center networking stack requires less compute resources to operate compared to the previous versions of vCloud NFV. Furthermore, workloads can consume N-VDS in an easy and streamlined way.

This chapter includes the following topics:

- **Workload Definition**
- **The Importance of Workload Acceleration**
- **Workload Acceleration in vCloud NFV**

## Workload Definition

Many CSPs have adopted VMware vCloud NFV since the initial release of the platform. Numerous workloads have been deployed in production networks that are established over the vCloud NFV platform. As CSPs expand their NFV environments, VMware is observing an increased number of deployed data plane intensive workloads.

Two main categories for 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 customer or user traffic and do not generate a high networking load. With the growing user traffic and the increasing number of connections, control plane workloads are expected to generate a high transaction rate. These workloads are currently well served by the vCloud NFV platform and can easily benefit from shared resources.

- **Data plane workloads.** Depending on the area in the CSP network, data plane workloads are sometimes referred to as user plane or forwarding plane workloads. Data plane workloads carry user or customer traffic. As end-user applications consume more network bandwidth, and as network capacity grows, data plane workloads must support an ever-increasing load.
Data plane workloads will benefit from the information in this white paper. Examples of these workloads include mobile packet core, virtual routers, video optimization, traffic management, DPI engines, and generally any virtual function that processes a high volume of data plane packets.

The Importance of Workload Acceleration

A CSP’s main business is offering network services. CSPs have two distinct drives in virtualizing their data plane workloads.

- 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 the 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 in purpose-build hardware.

- Accelerating workloads. Workload acceleration plays a significant role in the business case behind data plane networking in several aspects.
  - Reducing resource consumption. When data plane VNFs and virtualization layer perform efficiently, CSPs need a smaller number of servers, less power and cooling in the data center, and fewer personnel to operate the environment.
  - Providing scale and predictability. Having knowledge about the amount of resources that are required to support a certain workload allows the CSP to build business models around the cost of deploying services. This predictability also helps the CSPs to gain a clear understanding about the costs that are involved in the service success. This is an important aspect of operating a virtual network service. Scale and predictability are therefore essential elements to any workload acceleration discussion.

Workload Acceleration in vCloud NFV

vCloud NFV 3.0 introduces a new networking fabric. It is called NSX-T Virtual Distributed Switch or N-VDS. N-VDS operates in two modes.

- Standard mode (N-VDS Standard). In the standard mode, N-VDS supports 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). This mode is aimed at high data-plane workloads that require accelerated networking performance and dedicated networking resources.

N-VDS Enhanced is an efficient virtual networking stack that leverages industry advances such as Data Plane Development Kit (DPDK). By using the DPDK principles, N-VDS Enhanced uses significantly less CPU resources to achieve higher packet throughput rates compared to previous releases of vCloud NFV. N-VDS Enhanced 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. The second benefit of this approach is that the amount of the resources that are
dedicated to the virtual networking stack are clearly separated between the virtualization platform and the
Virtual Network Function (VNF). The result of this approach is a simplified configuration for data plane
intensive VNF Components (VNF-Cs) as well as resource usage predictability.

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

- Security. The accelerated networking stack exists entirely within the hypervisor 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 leverage N-VDS.
Introduction to Workload Acceleration

Achieving accelerated performance for data plane intensive workloads requires a holistic approach to both the physical and virtual domains. Before we introduce specific configuration for the physical and virtual domains, we introduce the building blocks and the design principles that should be considered when creating an NFV platform and tuning VNFs for accelerated data plane performance.

This chapter includes the following topics:
- Workload Acceleration Building Blocks
- Workload Acceleration Example

Workload Acceleration Building Blocks

In the ETSI NFV Reference Architecture, the NFV infrastructure is split into two layers: the physical and the virtualization layer. To achieve workload acceleration, both layers should be designed in a specific way.

Physical Layer Design Best Practices

Following are best practices for building a physical NFV infrastructure aimed at hosting data plane intensive workloads.

- **Server Homogeneity.** To ease the operations in the environment, the server models and server configuration should be identical. For example, the same NIC and NIC driver should be used in all servers. Furthermore, the placement of the NIC in the PCIe slot should be also identical across all servers.

- **NUMA Node Design.** Servers intended for data plane intensive workloads typically have multiple NUMA nodes. A best practice is to configure the NUMA nodes identically. This means that if NUMA 0 has a single NIC, the rest of the NUMA nodes should have a single NIC too. Respectively, the NICs should be of the same make and model, use the same NIC driver, and should be placed in the same PCIe slot. This guideline covers 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 about the capabilities and performance of that NIC. Details such as throughput performance, offloading capabilities for Overlay technologies, PCIe speeds, and CPU offload influence the performance of the data plane intensive workloads. In addition, the NIC must have drivers for N-VDS Enhanced datapath mode.

CPU Speed and Density. Data plane intensive workloads consume CPU resources to transport network traffic from the physical NIC to the VNF Components (VNF-Cs). Operations performed inside the VNF-Cs also consume CPU resources. To deploy data plane intensive workloads successfully and efficiently, it is mandatory to 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 very high-end CPUs with. Their higher cost may not justify the additional CPUs cores. However, data plane intensive VNFs benefit from a high number of CPU cores.

Data Plane Traffic Path. The speed of any network is determined by its slowest link. To drive maximum performance out of the NFV infrastructure that runs data plane intensive workloads, fast NICs, Top of Rack (ToR) switches and End of Row (EoR) devices with high data plane performance should be selected. These paths can be completely dedicated to data plane workloads, while another set of components can be used for control and management plane traffic.

Virtualization Layer Design Best Practices

Once the choice for physical infrastructure is made, the next set of building blocks are the components in the virtual domain.

Virtual Infrastructure. Each VMware vCloud NFV release is accompanied with a Bill of Materials (BoM) that specifies the component versions that are used. The components and their versions are described in the release notes for every vCloud NFV platform version. Starting from vCloud NFV 3.0, the virtual networking stack of the platform is handled by NSX-T Data Center. NSX-T Data Center is an essential building block in the infrastructure that supports data plane intensive workloads. NSX-T introduces a new networking stack called NSX-Controlled virtual distributed switch (N-VDS). As best practice, N-VDS in Enhanced data path mode is used 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. For this reason and to ease the operations in such an environment, it is a best practice to dedicate a cluster for data plane workloads.

Data Plane VNF-C. VNF-Cs serve different functions such as management, control and data processing. In this white paper we focus on the data plane VNF-C. The Data plane VNF-C is designed with performance in mind which typically translates to the VNF-C leveraging 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 Enhanced for their data plane vNICs.
**Workload Acceleration Example**

The VNF used in this example is constructed from three VNF-Cs.

- A data plane VM
- A control plane VM
- An Administration, Operations and Management (OAM) VM

The data plane VM requires a large number of vCPUs to support a large number of users at high data rates. Since the VNF plays a central role in the CSP network, it must also provide a high level of availability. The control plane and OAM VMs have a small resource footprint and do not require acceleration.
The data plane VNF-C typically occupies a complete NUMA node. We recommend that the number of vCPUs used by the data plane VNF-C does not exceed the number of CPU cores that are available on the NUMA. To achieve networking resiliency, the NUMA node where the data plane VNF-C is installed, should have two physical NICs configured as redundant uplinks to an N-VDS Enhanced switch. The data plane VNF-C can also benefit from host resiliency by leveraging anti-affinity rules. The N-VDS Enhanced uplinks can also be bonded by using a teaming policy that will increase the availability.

If the VNF vendor follows the design best practices provided in this document, the other NUMA node on the server can be used to host other VNF-Cs. This is a typical VNF design since VNF vendors prefer to increase application-level resiliency by spreading functions between hosts. The result of such configuration is that all data plane traffic is flowing North-South. The vCloud NFV platform supports using both NUMA nodes for data plane intensive VNF-Cs if the VNF is designed in that way.

**Figure 3-2. Accelerated Workload on vCloud NFV Example**

Our recommendation is to use an N-VDS Standard to facilitate control plane and management communications between the various VNF-Cs. Another best practice is to connect the vNICs that are used to process data plane traffic to an N-VDS Enhanced. The benefits of this approach are the following:

- **Network traffic separation.** The approach provides clear separation between bursty management and control plane traffic from high throughput and constant data plane traffic.

- **Ease of networking design.** The physical NICs used to transport data plane traffic could use a different path and physical components than the NICs used to transport management and control plane traffic.
- **Benefits of shared resources.** Because N-VDS Enhanced is configured with dedicated CPU resources that are not accessible by other applications, it is important to leverage these resources where they are truly needed. Control and management plane traffic are perfectly served by N-VDS Standard.
The NFVI operator is the CSP. The CSP is responsible for creating the NVF environment as well as for its operation.

In a typical NFV environment, VNF vendors do not have access to the infrastructure management components such as NSX-T Manager or 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 VNF is placed. All interactions between the virtual infrastructure and the VNF are performed through the available cloud management interfaces from within the VNF tenancy. It is therefore the responsibility of the CSP to ensure that all host-level configurations are prepared for data plane intensive workloads. The CSP configures the hardware and software and assigns the appropriate resources to the virtual networking. These resources are then consumed by the VNFs as needed.

This chapter includes the following topics:

- Host Configuration
- Configuring N-VDS Enhanced Data Path
- CPU Assignment for Network Packet Processing
- Achieving NUMA Vertical Alignment

Host Configuration

To benefit from virtualization, a common compute platforms should be used. The right choice of servers and NICs is very important in supporting workload acceleration. Once the servers are procured and installed, server-level configuration is required. The generic or default BIOS settings are not always configured for high performance. The BIOS settings that control the CPU play a significant role in making every CPU cycle available to the demanding workloads. Because the names of BIOS settings differ between servers and BIOS manufacturers, generic terms are used here.

- **Power Management.** Set this option to high or maximum to ensure that the processor CPU cycles are always available, and the processor does not slow down to preserve energy. On some systems, this option is called High Performance Profile. In certain cases, setting the Power Management configuration to high or maximum might not be sufficient. The Extreme Performance Series: Performance Best Practice VMworld 2017 presentation provides excellent information about BIOS configuration that is related to CPUs.
- **Hyperthreading.** Enable this option in the systems that support it. Hyperthreading allows a single processor core to execute 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 this as a Logical Processor. More information about hyperthreading is available at [VMware vSphere 6.7 Documentation](https://www.vmware.com).  

- **Turbo Boost.** Enable this BIOS option. It allows the processor to operate faster than the rated frequency for peak loads. More information about Turbo Boost is available from the Frequently Asked Questions about Intel Turbo Boost Technology page of the Intel Web site.  

- **NUMA Node Interleaving.** Make sure that this option is disabled. With NUMA node interleaving enabled, the hypervisor sees the available memory as one contiguous area. Consequently, the ability to place memory pages local to the CPU is lost, that is CPU and memory pages on same the NUMA node.

### Recommendations

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

### Configuring N-VDS Enhanced Data Path

As a prerequisite to using N-VDS Enhanced, you should configure the virtual environment according to the relevant vCloud NFV version. To use N-VDS Enhanced, assign at least one physical NIC as an uplink to the switch. For high-availability, assign at least two uplinks to N-VDS Enhanced.

The physical NICs must use a Poll Mode Driver (PMD) for N-VDS Enhanced. You can download the NIC driver from the VMware Compatibility Guide. You must install the driver on every host in the vSphere cluster for use of the physical NICs that are dedicated to N-VDS Enhanced. You can identify the correct drivers by looking for N-VDS Enhanced Data Path in the feature column of the VMware Compatibility Guide. Examples of NICs and drivers supported at the time of publication are listed in the table below. Additional drivers are expected to be added over time.

<table>
<thead>
<tr>
<th>Intel Ethernet Controller Family</th>
<th>Enhanced Network Stack Driver</th>
</tr>
</thead>
<tbody>
<tr>
<td>82599, x520, x540, x550 and x552</td>
<td>VMware ESXi6.7 ixgben ENS 1.0.3 NIC Driver</td>
</tr>
<tr>
<td>X710, XL710, XXV710, and X722</td>
<td>VMware ESXi6.7 i40en ENS 1.0.4 NIC Driver</td>
</tr>
</tbody>
</table>

A physical NIC that is assigned to an N-VDS Enhanced cannot be used by other virtual switches such as N-VDS Standard or vSphere Distributed Switch. When designing the servers' networking configuration, account for the number of physical NICs that are used for non-accelerated workloads such as vMotion, ESXi management, vSAN networking, and so on. The non-accelerated data path is not expected to require high-performance physical NICs.
The steps to configure N-VDS are described in the NSX-T Data Center installation guide available at VMware Docs. Note that N-VDS Enhanced supports NIC teaming policies that you can use to increase the aggregate networking capacity and resiliency. Teaming policies are discussed in the NSX-T Data Center Installation Guide.

**Recommendations:**
- Physical NICs must use poll mode driver for N-VDS Enhanced on every host that uses that switch.
- Assign a minimum of two physical NICs to an N-VDS Enhanced for uplink availability.

**CPU Assignment for Network Packet Processing**

Because packet processing uses CPU cycles, the host CPU resources play crucial role in the network performance. To provide predictable throughput performance for data plane intensive workloads, N-VDS Enhanced uses dedicated CPU cores that are assigned to the switch during its creation. These dedicated CPU cores are referred to as logical cores. When you assign logical cores to N-VDS Enhanced, these cores are removed from the available CPU resources on the host. As a result, there is a reduced number of CPU cores and therefore host-level CPU cycles that are available to DRS and other vSphere.

You assign logical cores to N-VDS Enhanced per NUMA node. You assign the same number of logical cores to each NUMA node. This best practice allows the optimal utilization of the cluster that is dedicated to data plane intensive workloads.
The number of CPU cores assigned to N-VDS Enhanced depends on the speed of the physical NICs and the workload type. For example, an N-VDS Enhanced using 10GbE NIC and serving typical Web-based services where the traffic consists of packets larger than 1,000 bytes, can completely saturate the NIC with only one single logical core assigned to N-VDS Enhanced. Because each environment is different and workloads have unique characteristics, the exact number of logical cores assigned to N-VDS Enhanced must be evaluated by the CSP.

**Recommendations**

- Assign logical cores to N-VDS Enhanced based on the expected traffic volumes, the servers’ NIC speeds and the VNF’s traffic characteristics.
- Assign the same number of logical cores to N-VDS Enhanced on each NUMA node.
Achieving NUMA Vertical Alignment

NUMA vertical alignment ensures that all CPU cores that a particular data path uses for network traffic are located on the same NUMA node. When a VNF-C uses the `sched.cpu.latencySensitivity=HIGH` parameter and is placed on a NUMA node, N-VDS Enhanced ensures that the VNF-C uses the logical cores from the same NUMA node. This function provides the fastest access between CPUs, their caches, and memory pages for optimal performance. This function also helps avoiding the inefficiencies of crossing the QuickPath Interconnect (QPI) bus between the NUMA nodes.

By setting `sched.cpu.latencySensitivity` to HIGH, N-VDS Enhanced ensures that the virtual machine, and logical cores are vertically aligned as long as resources are available. The VM-specific configuration is discussed in Chapter 5 Accelerating Data Plane Workloads.

Recommendations

- Replicate the NUMA node configuration in terms of NIC placement and logical cores assignment to N-VDS Enhanced across all NUMA nodes.
- Set the latency sensitivity VM configuration to high for data plane intensive VNF-Cs.
- NUMA node affinity setting is not required.
Accelerating Data Plane Workloads

With N-VDS Enhanced, configuring a VNF-C for optimal data plane performance is simplified. VNF vendors must no longer account for the networking processes that operate on behalf of the VNF-C. N-VDS Enhanced is responsible for allocating optimal CPU cores to the networking processes from the available logical core pool that is configured to the switch.

According to the Tuning vCloud NFV for Data Plane Intensive Workloads white paper, there are three VNF-C parameters relevant to data plane intensive workloads: sched.cpu.latencySensitivity, cTxPerDev on each data plane intensive vNIC, and sched.cpu.latencySensitivity.sysContexts. When deploying data plane intensive VNFs on vCloud NFV 3.0, only one of these parameters remains relevant: sched.cpu.latencySensitivity.

Another VNF-C parameter that is no longer required is numa.nodeAffinity. One of the capabilities of N-VDS Enhanced is automatic resource alignment. If the NFVI is designed as discussed in the Host Configuration chapter, the configuration of all NUMA nodes will be identical. This means that from an operational perspective, the CSP or the application owner, can depend on the cloud management placement to identify the correct host where the VNF-C will be deployed.

Relevant performance tuning that must be considered for a data plane intensive workload is the following:

- Using VMXNET3 paravirtualized vNICs.
- Using the latest virtual machine hardware version.
- Dedicating CPU cores to the data plane intensive VNF-C.
- Physical NIC and vNIC buffer tuning.
- Using Huge Pages.
- Data plane intensive VNF-C vNIC scaling.

This chapter includes the following topics:

- VMXNET3 Paravirtualized NIC
- Virtual Machine Hardware Version
- Dedicating CPUs to a VNF Component
- Physical NIC and vNIC Buffer Size Tunings
- Huge Pages
VNF Component vNIC Scaling

VMXNET3 Paravirtualized NIC

The data plane vNIC should use the paravirtual VMXNET3 driver. The paravirtualized network interface card (NIC) VMXNET3 has improved the performance compared to other virtual network interfaces. VMXNET3 provides several advanced features including multi-queue support, Receive Side Scaling (RSS), Large Receive Offload (LRO), IPv4 and IPv6 offloads, and MSI and MSI-X interrupt delivery. By default, VMXNET3 also supports an interrupt coalescing algorithm. Virtual interrupt coalescing helps drive high throughput to VMs with multiple vCPUs with parallelized workloads, for example with multiple threads, while at the same time striving to minimize the latency of virtual interrupt delivery.

To benefit from the performance and efficiency offered by N-VDS Enhanced, a data plane intensive VNF-C must always use VMXNET3 poll mode driver (PMD) as its vNIC. VMXNET3 is included in several modern Linux distributions such as Red Hat Enterprise Linux and SUSE Linux since its release 2.6.32 in October of 2009. Although the VMXNET3 driver is included in the Linux distributions, you must also install VMware Tools on the VM. See VMware support for Linux inbox VMware drivers (2073804) for more information. For more information about the DPDK support in the VMXNET3 driver, please refer to the article Poll Mode Driver for Paravirtual VMXNET3 NIC.

Recommendations

- Use paravirtualized VMXNET3 vNIC.
- Install and use VMXNET3 PMD.
- Install VMware Tools.

Virtual Machine Hardware Version

It is important to 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.

You must use uniform VM hardware versions for all VMs comprising a VNF. This is especially important because different VM hardware versions support different components and different amounts of resources. For example, VM hardware version 14 supports a maximum of 6128 GB of RAM for a VM, whereas VM hardware version 11 supports 4080 GB of RAM. VM hardware versions also enable processor features, which is why the best practice is to use the latest virtual hardware version to expose new instruction sets to the VM. Mismatches between VNF-Cs configured with different VM hardware versions impact the performance and must be avoided.

- 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).

Upgrading the virtual hardware version is described in Upgrading a virtual machine to the latest hardware version (multiple versions) (1010675).

Recommendations

- Use the latest virtual machine hardware version.
- Use 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 of a 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 the 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 will be pinned and receive exclusive affinity to a complete CPU core as opposed to a hyperthread. This reduces the risk of two heavy processes competing for hyperthread resources on the same CPU core, therefore reducing the effective CPU cycles available to each vCPU.

Recommendations

- Set VM CPU and memory reservation to maximum and the CPU limit to unlimited.
- Set sched.cpu.latencySensitivity = High for data plane intensive workloads.
- Ensure VNF-C is compiled with DPDK and uses VMXNET3 DPDK poll mode driver.

Physical NIC and vNIC Buffer Size Tunings

Selecting buffer sizes strongly depends on the VNF workload. It is expected that VNF vendors benchmark their VNFs to measure the optimal buffer size configuration. The VMXNET3 DPDK buffers are typically compiled in the VNF-C which means that CSPs will not be able to change these values. Generally, higher buffer sizes provide improved performance, however, latency can also increase with larger buffers.
The following example buffer sizes are verified to support data plane intensive workloads across various network functions and environments. However, they might not be suitable for all workload types.

Table 5-1. Example Buffer Alignment

<table>
<thead>
<tr>
<th>Buffer Size (ESXi SoftQueue: 6000)</th>
<th>Rx</th>
<th>Tx</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical NIC</td>
<td>1024</td>
<td>2048</td>
</tr>
<tr>
<td>vNIC</td>
<td>3000</td>
<td>4000</td>
</tr>
</tbody>
</table>

Recommendation

- Use larger buffer sizes for better performance.

Huge Pages

The virtual infrastructure provides support for backing guest memory with 1GB pages for memory intensive and large memory VNF-Cs. Such VNF-Cs can potentially experience improved performance because of the increased translation lookaside buffers (TLB) access efficiency. A larger TLB covers a greater range of memory reducing the TLB misses.

To use 1GB pages to back VNF-C guest operating system memory, you apply the option sched.mem.lpage.enable1GPage = TRUE to the VM. Further information is available in Backing Guest vRAM with 1GB Pages.

Before you configure Huge Pages, make sure that the VM’s memory reservation is configured to the maximum value so that all the vRAM can be pre-allocated at the time of the VM’s power-on. You must also consider the resource availability for other workloads in the cluster such as HA and DRS.

To set Huge Pages, follow the 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.

Recommendation

- Set sched.mem.lpage.enable1GPage = True to enable Huge Pages.
VNF Component vNIC Scaling

The data plane performance of a VNF-C can scale up by increasing the number of vNICs that are used by the VM. When using high throughput physical NICs on the host, it is beneficial to 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 benefit from the CPU cores available on the host.

When scaling a VNF-C, you must find balance between the number of vNICs and CPU cores used by the VNF-C and the resources needed to fully utilize the physical NIC. As a best practice, consider scaling a VNF horizontally once a certain number of vCPUs have been consumed by the VNF-C.

Recommendations

- To scale the data plane performance of a VNF-C, increase the number of vNICs.
- Once the number of vNICs is larger than the number of logical cores used by N-VDS Enhanced, consider scaling horizontally.
Conclusion

With every release of the VMware vCloud NFV platform, we aim to improve the data plane performance and predictability. With this release, we introduce a major milestone towards increasing the efficiency and predictability of data plane workloads in NFV deployments. Not only does the platform now support simplified data plane workload configuration and predictable performance, the workload themselves require no changes to benefit from the new stack. Alongside these improvements, we also maintain the platform functionality that customers have learned to love. As more and more data plane workloads are deployed, we are committed to continuing the evolution of the platform.
Summary of Recommendations

This section provides configuration information, including the one discussed within the body of the document and the information provided within some of the additional resources included here.

**Note** Before applying any of the configuration parameters, it is strongly recommended that you test them in a lab while measuring the effect on the environment and the VNF performance.

### BIOS Configuration

Different BIOS manufacturers use different names for their BIOS functions. This document attempts to reflect general naming conventions where possible. Consult your server documentation for exact configuration details.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power Management</td>
<td>Maximum Performance / High Performance</td>
</tr>
<tr>
<td>Hyperthreading</td>
<td>Enable</td>
</tr>
<tr>
<td>Turbo Boost</td>
<td>Enable</td>
</tr>
<tr>
<td>C States</td>
<td>Disable</td>
</tr>
<tr>
<td>Intel VT Technology</td>
<td>Enable</td>
</tr>
<tr>
<td>QPI Power Management</td>
<td>Disable</td>
</tr>
<tr>
<td>Execute Disable Bit</td>
<td>Enable</td>
</tr>
<tr>
<td>Node Interleaving</td>
<td>Disable</td>
</tr>
</tbody>
</table>

### Hypervisor

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>N-VDS Enhanced NIC Driver</td>
<td>Search for your NIC’s driver in the VMware compatibility list. Search for N-VDS Enhanced Data Path in the feature column</td>
<td>Not relevant</td>
</tr>
<tr>
<td>Rx Descriptor</td>
<td>2048 (example)</td>
<td>nsxdp-cli ens uplink ring set -r 2048 --uplink &lt;vmnic#&gt;</td>
</tr>
</tbody>
</table>
### Table 7-1. Virtual CPU

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shares</td>
<td>High</td>
<td>VM Settings &gt; Virtual Hardware &gt; CPU</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CPU Shares set to High.</td>
</tr>
<tr>
<td>Reservation</td>
<td>Maximum</td>
<td>VM Settings &gt; Virtual Hardware &gt; CPU</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Reservation set to the number of vCPUs multiplied by the processor base frequency.</td>
</tr>
<tr>
<td>Limit</td>
<td>Unlimited</td>
<td>VM Settings &gt; Virtual Hardware &gt; CPU</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Limit set to Unlimited.</td>
</tr>
</tbody>
</table>

### Table 7-2. Memory

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shares</td>
<td>High</td>
<td>VM Settings &gt; Virtual Hardware &gt; Memory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Shares set to High.</td>
</tr>
<tr>
<td>Reservation</td>
<td>Maximum</td>
<td>VM Settings &gt; Virtual Hardware &gt; Memory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Reservation set to Maximum as needed.</td>
</tr>
<tr>
<td>Limit</td>
<td>Unlimited</td>
<td>VM Settings &gt; Virtual Hardware &gt; Memory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Limit set to Unlimited.</td>
</tr>
</tbody>
</table>

### Table 7-3. VM Level Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware Version</td>
<td>14 for ESXi 6.7</td>
<td>Right-click the VM, select Compatibility &gt; Upgrade VM Compatibility</td>
</tr>
<tr>
<td>Virtual NIC for Data Plane Interfaces</td>
<td>VMXNET3</td>
<td>Set VMXNET3 as the vNIC adapter type.</td>
</tr>
<tr>
<td>Latency Sensitivity</td>
<td>High</td>
<td>Edit Settings &gt; VM Options &gt; Advanced &gt; Configuration Parameters</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sched.cpu.latencySensitivity = High</td>
</tr>
</tbody>
</table>
Acronyms and Definitions

This document uses a specific set of NFV-related and industry-specific terms.

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPDK</td>
<td>Data Plane Development Kit, an Intel led packet processing acceleration technology.</td>
</tr>
<tr>
<td>MANO</td>
<td>Management and Orchestration components, a term coined by ETSI NFV.</td>
</tr>
<tr>
<td>NFVI</td>
<td>Network Functions Virtualization Infrastructure.</td>
</tr>
<tr>
<td>N-VDS Enhanced</td>
<td>Enhanced datapath mode when using the NSX-T Data Center N-VDS logical switch. This mode enables DPDK for workload acceleration.</td>
</tr>
<tr>
<td>N-VDS Standard</td>
<td>Standard datapath mode when using the NSX-T Data Center N-VDS logical switch.</td>
</tr>
<tr>
<td>VIM</td>
<td>Virtualized Infrastructure Manager.</td>
</tr>
<tr>
<td>PMD</td>
<td>Poll Mode Driver.</td>
</tr>
<tr>
<td>VNF</td>
<td>Virtual Network Function.</td>
</tr>
<tr>
<td>VNF-C</td>
<td>Virtual Network Function Component, the building block of VNFs. VNF-Cs are represented as VMs in vSphere.</td>
</tr>
</tbody>
</table>
References

The information in this white paper is based on a number of references.

- VMware vCloud NFV OpenStack Edition 3.0 Reference Architecture
- Tuning vCloud NFV for Data Plane Intensive Workloads
- Performance Best Practices for VMware vSphere 6.7
- NSX-T Data Center Installation Guide
This section provides a brief description about the Authors and Contributors.

- Jambi Ganbar, Sr. Technical Solutions Manager, NFV Solutions Engineering, VMware
- Sumit Verdi, Director Lighthouse Solutions, NFV Solutions, VMware
- Indranil Bal, Solution Consultant, NFV Solutions, VMware
- Pradip Kadam, Solution Consultant, NFV Solutions, VMware
- Mahesh Pandurangan, Staff Engineer, NFV Solutions, VMware
- Ramachandran Sathyanarayanan, Solutions Consultant, NFV Solutions, VMware
- Piyush Kumar Singh, Solution Consultant, NFV Solutions, VMware
- T. Sridhar, Principal Engineer, NFV Solutions, VMware
- Ramkumar Venketaramani, Director Product Management, NFV Solutions, VMware
- Dobrinka Boeva, Sr. Staff Technical Writer, NFV Solutions, VMware