



NetScaler VPX 12-1

Contents

Support matrix and usage guidelines	4
Optimize Citrix ADC VPX performance on VMware ESX, Linux KVM, and Citrix Hypervisors	10
Install a Citrix ADC VPX instance on a bare metal server	23
Install a Citrix ADC VPX instance on XenServer	24
Configure VPX instances to use single root I/O virtualization (SR-IOV) network interfaces	27
Install a Citrix ADC VPX instance on VMware ESX	30
Configure a Citrix ADC VPX instance to use VMXNET3 network interface	35
Configure a Citrix ADC VPX instance to use SR-IOV network interface	47
Migrating the Citrix ADC VPX from E1000 to SR-IOV or VMXNET3 Network Interfaces	65
Configure a Citrix ADC VPX instance to use PCI passthrough network interface	66
Install a Citrix ADC VPX instance on Microsoft Hyper-V server	69
Install a Citrix ADC VPX instance on Linux-KVM platform	74
Prerequisites for installing a Citrix ADC VPX instance on Linux-KVM platform	75
Provision the Citrix ADC VPX instance by using OpenStack	80
Provision the Citrix ADC VPX instance by using the Virtual Machine Manager	89
Configure a Citrix ADC VPX instance to use SR-IOV network interfaces	103
Configure a Citrix ADC VPX instance to use PCI passthrough network interfaces	114
Provision the Citrix ADC VPX instance by using the virsh program	118
Manage the Citrix ADC VPX guest VMs	121
Provision the Citrix ADC VPX instance with SR-IOV, on OpenStack	124
Configure a Citrix ADC VPX instance on KVM to use OVS DPDK-based host interfaces	131
Deploy a Citrix ADC VPX instance on AWS	140
Limitations and usage guidelines	144

Prerequisites	146
Deploy a Citrix ADC VPX standalone instance on AWS	147
Scenario: standalone instance	152
Download a Citrix ADC VPX license	160
Load balancing servers in different availability zones	166
Deploy a high availability pair on AWS	166
High availability across AWS availability zones	171
Add back-end AWS Autoscaling service	176
Configure a Citrix ADC VPX instance to use SR-IOV network interface	182
Upgrade a Citrix ADC VPX instance on AWS	185
Troubleshoot a VPX instance on AWS	190
AWS FAQs	191
Deploy a Citrix ADC VPX instance on Microsoft Azure	191
Azure terminology	196
Network architecture for Citrix ADC VPX instances on Microsoft Azure	199
Configure a Citrix ADC VPX standalone instance	202
Configure multiple IP addresses for a Citrix ADC VPX standalone instance	216
Configure a high-availability setup with multiple IP addresses and NICs	222
Configure a high-availability setup with multiple IP addresses and NICs by using Power-Shell commands	231
Configure HA-INC nodes by using the Citrix high availability template with Azure ILB	243
Add Azure autoscale settings	256
Configure GSLB on Citrix ADC VPX instances	263
Configure GSLB on an active-standby high-availability setup	271

Configure address pools (IIP) for a Citrix Gateway appliance	275
Configure multiple IP addresses for a Citrix ADC VPX standalone instance by using PowerShell commands	277
Additional PowerShell scripts for Azure deployment	284
Azure FAQs	299
Deploy a Citrix ADC VPX instance on Google Cloud Platform	300
Jumbo frames on Citrix ADC VPX instances	310

Support matrix and usage guidelines

November 13, 2024

This document lists the different hypervisors and features supported on a Citrix ADC VPX instance. It also describes their usage guidelines and limitations.

Table 1. VPX instance on Citrix Hypervisor

Citrix Hypervisor version	SysID	VPX models
7.1	450000	VPX 10, VPX 25, VPX 200, VPX 1000, VPX 3000, VPX 5000, VPX 8000, VPX 10G, VPX 15G, VPX 25G, VPX 40G

Table 2. VPX instance on VMware ESXi server

The following VPX models with 450010 (Sys ID) supports the VMware ESX versions listed in the table.

VPX models: VPX 10, VPX 25, VPX 200, VPX 1000, VPX 3000, VPX 5000, VPX 8000, VPX 10G, VPX 15G, VPX 25G, VPX 40G, and VPX 100G.

ESXi version	ESXi release date in (YYYY/MM/DD) format	ESXi build number	Citrix ADC VPX version
ESXi 7.0 update 3m	2023/05/03	21686933	12.1-65.x and higher builds
ESXi 7.0 update 3f	2022/07/12	20036589	12.1-65.x and higher builds
ESXi 7.0 update 3d	2022/03/29	19482537	12.1-65.x and higher builds
ESXi 7.0 update 2d	2021/09/14	18538813	12.1-63.x and higher builds
ESXi 7.0 update 2a	2021/04/29	17867351	12.1-62.x and higher builds
ESXi 6.7 P04	2020/11/19	17167734	12.1-55.x and higher builds
ESXi 6.7 P03	2020/08/20	16713306	12.1-55.x and higher builds
ESXi 6.5 GA	2016/11/15	4564106	12.1-55.x and higher builds

ESXi version	ESXi release date in (YYYY/MM/DD) format	ESXi build number	Citrix ADC VPX version
ESXi 6.5 U1g	2018/3/20	7967591	12.1-55.x and higher builds

Note:

Each ESXi patch support is validated on the Citrix ADC VPX version specified in the preceding table and is applicable for all the higher builds of Citrix VPX 12.1 version.

Table 3. VPX on Microsoft Hyper-V

Hyper-V version	SysID	VPX models
2012, 2012R2	450020	VPX 10, VPX 25, VPX 200, VPX 1000, VPX 3000

VPX instance on Nutanix AHV

NetScaler VPX is supported on Nutanix AHV through the [Citrix Ready partnership](#). Citrix Ready is a technology partner program that helps software and hardware vendors develop and integrate their products with NetScaler technology for digital workspace, networking, and analytics.

For more information on a step-by-step method to deploy a NetScaler VPX instance on Nutanix AHV, see [Deploying a NetScaler VPX on Nutanix AHV](#).

Third-party support:

If you experience any issues with a particular third-party (Nutanix AHV) integration on a NetScaler environment, open a support incident directly with the third-party partner (Nutanix).

If the partner determines that the issue appears to be with NetScaler, the partner can approach NetScaler support for further assistance. A dedicated technical resource from partners works with the NetScaler support until the issue is resolved.

For more information, see [Citrix Ready Partner Program FAQs](#).

Table 4. VPX instance on generic KVM

Generic KVM version	SysID	VPX models
RHEL 7.4, RHEL 7.5 (from Citrix ADC version 12.1 50.x onwards) Ubuntu 16.04	450070	VPX 10, VPX 25, VPX 200, VPX 1000, VPX 3000, VPX 5000, VPX 8000, VPX 10G, VPX 15G. VPX 25G, VPX 40G, VPX 100G

Note:

The VPX instance is qualified for hypervisor release versions mentioned in table 1–4, and not for patch releases within a version. However, the VPX instance is expected to work seamlessly with patch releases of a supported version. If it does not, log a support case for troubleshooting and debugging.

Table 5. VPX instance on AWS

AWS version	SysID	VPX models
N/A	450040	VPX 10, VPX 200, VPX 1000, VPX 3000, VPX 5000, VPX 15G, VPX BYOL

Table 6. VPX instance on Azure

Azure version	SysID	VPX models
N/A	450020	VPX 10, VPX 200, VPX 1000, VPX 3000, VPX BYOL

Table 7. VPX feature matrix

Features	VPX on XenServer		VPX on VMware ESX				VPX on Microsoft Hyper-V	VPX on generic KVM			VPX on AWS	VPX on Azure	VPX on GCP
	PV	SR-IOV	PV	SR-IOV	Emulated	PCI Passthrough	PV	PV	SR-IOV	PCI Passthrough			
Multi-PE Support	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Clustering Support	Yes	Yes ¹	Yes	Yes ¹	Yes	Yes	Yes	Yes	Yes ¹	Yes	No	No	No
VLAN Tagging	Yes	Yes	Yes	Yes	Yes	Yes	Yes (only on 2012R2)	Yes	Yes	Yes	No	No	No
Detecting Link Events	No ²	Yes ³	No ²	Yes ³	No ²	Yes ³	No ²	No ²	Yes ³	Yes ³	No ²	No ²	No ²
Interface Parameter Configuration	No	No	No	No	No	Yes	No	No	No	Yes	No	No	No
Static LA	Yes ²	Yes ³	Yes ²	No	Yes ²	Yes ³	Yes ²	Yes ²	Yes ³	Yes ²	No	No	No
LACP	No	Yes ³	Yes ²	No	Yes ²	Yes ³	No	Yes ²	Yes ³	Yes ³	No	No	No
Static CLAG	No	No	No	No	No	No	No	No	No	No	No	No	No
LACP CLAG	No	No	Yes ²	No	Yes ²	Yes ³	No	Yes ²	Yes ³	Yes ³	No	No	No
Hot-plug	No	No	No	No	No	No	No	No	No	No	Yes	No	No

- Clustering support is available on SRIOV for client- and server-facing interfaces and not for the backplane.
- Interface DOWN events are not recorded in Citrix ADC VPX instances.
- For Static LA, traffic might still be sent on the interface whose physical status is DOWN.
- For LACP, peer device knows interface DOWN event based on LACP timeout mechanism.
 - Short timeout: 3 seconds
 - Long timeout: 90 seconds
- For LACP, interfaces should not be shared across VMs.
- For Dynamic routing, convergence time depends on the Routing Protocol since link events are not detected.
- Monitored static Route functionality fails if monitors are not bound to static routes since Route state depends on the VLAN status. The VLAN status depends on the link status.
- Partial failure detection does not happen in high availability if there's link failure. High availability-split brain condition might happen if there is link failure.
- When any link event (disable/enable, reset) is generated from a VPX instance, the physical status of the link does not change. For static LA, any traffic initiated by the peer gets dropped on the instance.
- For the VLAN tagging feature to work, do the following:

On the VMware ESX, set the port group's VLAN ID to 1–4095 on the vSwitch of the VMware ESX server. For more information about setting a VLAN ID on the vSwitch of VMware ESX server, see [VMware ESX Server 3 802.1Q VLAN Solutions](#).

Table 8. Supported browsers

Operating system	Browser and versions
Windows 7	Internet Explorer- 8, 9, 10, and 11; Mozilla Firefox 3.6.25 and above; Google Chrome- 15 and above
Windows 64 bit	Internet Explorer - 8, 9; Google Chrome - 15 and above
MAC	Mozilla Firefox - 12 and above; Safari - 5.1.3; Google Chrome - 15 and above

Usage guidelines

Follow these usage guidelines:

- See the **VMware ESXi CPU Considerations** section in the document [Performance Best Practices for VMware vSphere 6.5](#). Here's an extract:

It is not recommended that virtual machines with high CPU/Memory demand sit on a Host/Cluster that is overcommitted.

In most environments ESXi allows significant levels of CPU overcommitment (that is, running more vCPUs on a host than the total number of physical processor cores in that host) without impacting virtual machine performance.

If an ESXi host becomes CPU saturated (that is, the virtual machines and other loads on the host demand all the CPU resources the host has), latency-sensitive workloads might not perform well. In this case you might want to reduce the CPU load, for example by powering off some virtual machines or migrating them to a different host (or allowing DRS to migrate them automatically).

- Citrix recommends the latest hardware compatibility version to avail latest feature sets of the ESXi hypervisor for the virtual machine. For more information about the hardware and ESXi version compatibility, see [VMware documentation](#).
- The Citrix ADC VPX is a latency-sensitive, high-performance virtual appliance. To deliver its expected performance, the appliance requires vCPU reservation, memory reservation, vCPU pinning on the host. Also, hyper threading must be disabled on the host. If the host does not meet these requirements, issues such as high-availability failover, CPU spike within the VPX instance, sluggishness in accessing the VPX CLI, pitboss daemon crash, packet drops, and low throughput occur.

- A hypervisor is considered over-provisioned if one of the following two conditions is met:
 - The total number of virtual cores (vCPU) provisioned on the host is greater than the total number of physical cores (pCPUs).
 - The total number of provisioned VMs consume more vCPUs than the total number of pCPUs.

At times, if an instance is over-provisioned, the hypervisor might not be able to guarantee the resources reserved (such as CPU, memory, and others) for the instance due to hypervisor scheduling over-heads or bugs or limitations with the hypervisor. This can cause lack of CPU resource for Citrix ADC and might lead to issues mentioned in the first point under **Usage guidelines**. As administrators, you're recommended to reduce the tenancy on the host so that the total number of vCPUs provisioned on the host is lesser or equal to the total number of pCPUs.

Example

For ESX hypervisor, if the `%RDY%` parameter of a VPX vCPU is greater than 0 in the `esx top` command output, the ESX host is said to be having scheduling overheads, which can cause latency related issues for the VPX instance.

In such a situation, reduce the tenancy on the host so that `%RDY%` returns to 0 always. Alternatively, contact the hypervisor vendor to triage the reason for not honoring the resource reservation done.

- Hot adding is supported only for PV and SRIOV interfaces on Citrix ADC.
- Hot removing either through the AWS Web console or AWS CLI is not supported for PV and SRIOV interfaces on Citrix ADC. The behavior of the instances can be unpredictable if hot-removal is attempted.
- You can use two commands (`set ns vpxparam` and `show ns vpxparam`) to control packet engine(non-management) CPU usage behavior of VPX instances in hypervised and cloud environments:
 - `set ns vpxparam -cpuyield (YES | NO | DEFAULT)`
Allow each VM to use CPU resources that have been allocated to another VM but are not being used.
Set ns vpxparam parameters:
 - cpuyield**: Release or do not release of allocated but unused CPU resources.
 - * **YES**: Allow allocated but unused CPU resources to be used by another VM.
 - * **NO**: Reserve all CPU resources for the VM to which they have been allocated. This option shows higher percentage in hypervisor and cloud environments for VPX CPU usage.

* **DEFAULT:** No.

Note:

On all the Citrix ADC VPX platforms, the vCPU usage on the host system is 100 percent. Type the `set ns vpxparam -cpuyield YES` command to override this usage.

If you want to set the cluster nodes to “yield”, you must perform the following additional configurations on CCO:

- * If a cluster is formed, all the nodes come up with “yield=DEFAULT”.
- * If a cluster is formed using the nodes that are already set to “yield=YES”, then the nodes are added to cluster using “DEFAULT”yield.

Note:

If you want to set the cluster nodes to “yield=YES”, you can perform suitable configurations only after forming the cluster but not before the cluster is formed.

- `show ns vpxparam`
Display the current vpxparam settings.

Optimize Citrix ADC VPX performance on VMware ESX, Linux KVM, and Citrix Hypervisors

November 13, 2024

The Citrix ADC VPX performance greatly varies depending on the hypervisor, allocated system resources, and the host configurations. To achieve the desired performance, first follow the recommendations in the VPX data sheet, and then further optimize it using the best practices provided in this document.

Citrix ADC VPX instance on VMware ESX hypervisors

This section contains details of configurable options and settings, and other suggestions that help you achieve optimal performance of Citrix ADC VPX instance on VMware ESX hypervisors.

- [Recommended configuration on ESX hosts](#)
- [Citrix ADC VPX with E1000 network interfaces](#)
- [Citrix ADC VPX with VMXNET3 network interfaces](#)
- [Citrix ADC VPX with SR-IOV and PCI passthrough network interfaces](#)

Recommended configuration on ESX hosts

To achieve high performance for VPX with E1000, VMXNET3, SR-IOV, and PCI passthrough network interfaces, follow these recommendations:

- The total number of virtual CPUs (vCPUs) provisioned on the ESX host must be less than or equal to the total number of physical CPUs (pCPUs) on the ESX host.
- Non-uniform Memory Access (NUMA) affinity and CPU affinity must be set for the ESX host to achieve good results.

–To find the NUMA affinity of a Vmnic, log in to the host locally or remotely, and type:

```
1 #vsish -e get /net/pNics/vmnic7/properties | grep NUMA
2 Device NUMA Node: 0
```

- To set NUMA and vCPU affinity for a VM, see [VMware documentation](#).

Citrix ADC VPX with E1000 network interfaces

Perform the following settings on the VMware ESX host:

- On the VMware ESX host, create two vNICs from one pNIC vSwitch. Multiple vNICs create multiple Rx threads in the ESX host. This increases the Rx throughput of the pNIC interface.
- Enable VLANs on the vSwitch port group level for each vNIC that you have created.
- To increase vNIC transmit (Tx) throughput, use a separate Tx thread in the ESX host per vNIC. Use the following ESX command:

- For ESX version 5.5:

```
1 esxcli system settings advanced set -o /Net/NetTxWorldlet -i
```

- For ESX version 6.0 onwards:

```
1 esxcli system settings advanced set -o /Net/NetVMTxType -i 1
```

- To further increase the vNIC Tx throughput, use a separate Tx completion thread and Rx threads per device (NIC) queue. Use the following ESX command:

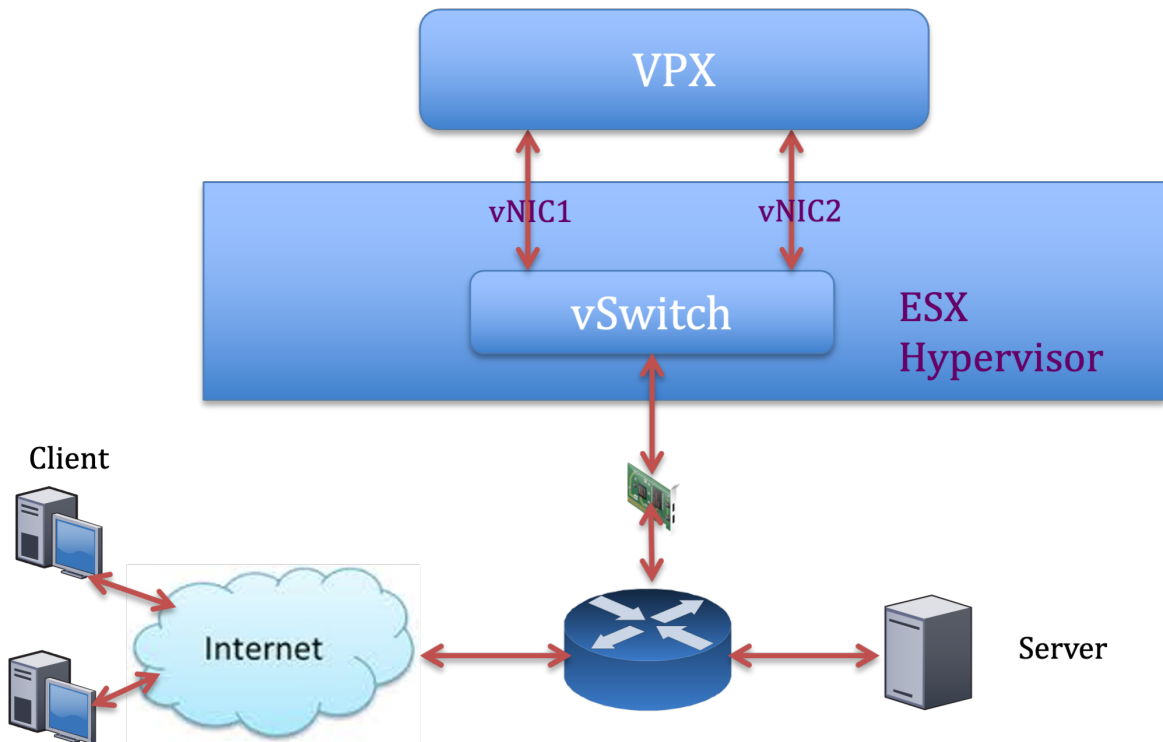
```
1 esxcli system settings advanced set -o /Net/NetNetqRxQueueFeatPairEnable -i 0
```

Note:

Make sure that you reboot the VMware ESX host to apply the updated settings.

Two vNICs per pNIC deployment

The following is a sample topology and configuration commands for the **Two vNICs per pNIC** model of deployment that delivers better network performance.



Citrix ADC VPX sample configuration:

To achieve the deployment shown in the preceding sample topology, perform the following configuration on the Citrix ADC VPX instance:

- On the client side, bind the SNIP (1.1.1.2) to network interface 1/1 and enable the VLAN tag mode.

```
1 bind vlan 2 -ifnum 1/1 -tagged
2 bind vlan 2 -IPAddress 1.1.1.2 255.255.255.0
```

- On the server side, bind the SNIP (2.2.2.2) to network interface 1/1 and enable the VLAN tag mode.

```
1 bind vlan 3 -ifnum 1/2 -tagged
2 bind vlan 3 -IPAddress 2.2.2.2 255.255.255.0
```

- Add an HTTP virtual server (1.1.1.100) and bind it to a service (2.2.2.100).

```

1  add lb vserver v1 HTTP 1.1.1.100 80 -persistenceType NONE -
    Listenpolicy None -cltTimeout 180
2  add service s1 2.2.2.100 HTTP 80 -gslb NONE -maxClient 0 -maxReq
    0 -cip DISABLED -usip NO -useproxyport YES -sp ON -cltTimeout
    180 -svrTimeout 360 -CKA NO -TCPB NO -CMP NO
3  bind lb vserver v1 s1

```

Note:

Make sure that you include the following two entries in the route table:

- 1.1.1.0/24 subnet with gateway pointing to SNIP 1.1.1.2
- 2.2.2.0/24 subnet with gateway pointing to SNIP 2.2.2.2

Citrix ADC VPX with VMXNET3 network interfaces

To achieve high performance for VPX with VMXNET3 network interfaces, do the following settings on the VMware ESX host:

- Create two vNICs from one pNIC vSwitch. Multiple vNICs create multiple Rx threads in the ESX host. This increases the Rx throughput of the pNIC interface.
- Enable VLANs on the vSwitch port group level for each vNIC that you have created.
- To increase vNIC transmit (Tx) throughput, use a separate Tx thread in the ESX host per vNIC. Use the following ESX commands:

- For ESX version 5.5:

```
1  esxcli system settings advanced set -o /Net/NetTxWorldlet -i
```

- For ESX version 6.0 onwards:

```
1  esxcli system settings advanced set -o /Net/NetVMTxType -i 1
```

On the VMware ESX host, perform the following configuration:

- On the VMware ESX host, create two vNICs from 1 pNIC vSwitch. Multiple vNICs create multiple Tx and Rx threads in the ESX host. This increases the Tx and Rx throughput of the pNIC interface.
- Enable VLANs on the vSwitch port group level for each vNIC that you have created.
- To increase Tx throughput of a vNIC, use a separate Tx completion thread and Rx threads per device (NIC) queue. Use the following command:

```
1  esxcli system settings advanced set -o /Net/
    NetNetqRxQueueFeatPairEnable -i 0
```

- Configure a VM to use one transmit thread per vNIC, by adding the following setting to the VM's configuration:

```
1 ethernetX.ctxPerDev = "1"
```

For more information, see [Best Practices for Performance Tuning of Telco and NFV Workloads in vSphere](#)

Note:

Make sure that you reboot the VMware ESX host to apply the updated settings.

You can configure VMXNET3 as a **Two vNICs per pNIC** deployment. For more information, see [Two vNICs per pNIC deployment](#).

Citrix ADC VPX with SR-IOV and PCI passthrough network interfaces

To achieve high performance for VPX with SR-IOV and PCI passthrough network interfaces, see [Recommended configuration on ESX hosts](#).

Citrix ADC VPX instance on Linux-KVM platform

This section contains details of configurable options and settings, and other suggestions that help you achieve optimal performance of Citrix ADC VPX instance on Linux-KVM platform.

- [Performance settings for KVM](#)
- [Citrix ADC VPX with PV network interfaces](#)
- [Citrix ADC VPX with SR-IOV and Fortville PCIe passthrough network interfaces](#)

Performance settings for KVM

Perform the following settings on the KVM host:

Find the NUMA domain of the NIC using the `lsstopo` command:

Make sure that memory for the VPX and the CPU is pinned to the same location.

In the following output, the 10G NIC “ens2” is tied to NUMA domain #1.

```
[root@localhost ~]# lstopo-no-graphics
Machine (128GB)
  NUMANode L#0 (P#0 64GB)
    Socket L#0 + L3 L#0 (20MB)
      L2 L#0 (256KB) + L1d L#0 (32KB) + L1i L#0 (32KB) + Core L#0 + PU L#0 (P#0)
      L2 L#1 (256KB) + L1d L#1 (32KB) + L1i L#1 (32KB) + Core L#1 + PU L#1 (P#1)
      L2 L#2 (256KB) + L1d L#2 (32KB) + L1i L#2 (32KB) + Core L#2 + PU L#2 (P#2)
      L2 L#3 (256KB) + L1d L#3 (32KB) + L1i L#3 (32KB) + Core L#3 + PU L#3 (P#3)
      L2 L#4 (256KB) + L1d L#4 (32KB) + L1i L#4 (32KB) + Core L#4 + PU L#4 (P#4)
      L2 L#5 (256KB) + L1d L#5 (32KB) + L1i L#5 (32KB) + Core L#5 + PU L#5 (P#5)
      L2 L#6 (256KB) + L1d L#6 (32KB) + L1i L#6 (32KB) + Core L#6 + PU L#6 (P#6)
      L2 L#7 (256KB) + L1d L#7 (32KB) + L1i L#7 (32KB) + Core L#7 + PU L#7 (P#7)
    HostBridge L#0
      PCI 8086:1521
        Net L#0 "eno1"
      PCI 8086:1521
        Net L#1 "eno2"
      PCI 8086:1584
        Net L#2 "ens3"
      PCI 8086:1584
        Net L#3 "ens4"
      PCI 8086:8d52
        Block L#4 "sda"
        Block L#5 "sdb"
      PCI 8086:2000
        GPU L#6 "card0"
        GPU L#7 "control064"
      PCI 8086:8d82
    NUMANode L#1 (P#1 64GB)
      Socket L#1 + L3 L#1 (20MB)
        L2 L#8 (256KB) + L1d L#8 (32KB) + L1i L#8 (32KB) + Core L#8 + PU L#8 (P#8)
        L2 L#9 (256KB) + L1d L#9 (32KB) + L1i L#9 (32KB) + Core L#9 + PU L#9 (P#9)
        L2 L#10 (256KB) + L1d L#10 (32KB) + L1i L#10 (32KB) + Core L#10 + PU L#10 (P#10)
        L2 L#11 (256KB) + L1d L#11 (32KB) + L1i L#11 (32KB) + Core L#11 + PU L#11 (P#11)
        L2 L#12 (256KB) + L1d L#12 (32KB) + L1i L#12 (32KB) + Core L#12 + PU L#12 (P#12)
        L2 L#13 (256KB) + L1d L#13 (32KB) + L1i L#13 (32KB) + Core L#13 + PU L#13 (P#13)
        L2 L#14 (256KB) + L1d L#14 (32KB) + L1i L#14 (32KB) + Core L#14 + PU L#14 (P#14)
        L2 L#15 (256KB) + L1d L#15 (32KB) + L1i L#15 (32KB) + Core L#15 + PU L#15 (P#15)
      HostBridge L#6
        PCI 8086:1584
          Net L#8 "ens2"
      PCI 8086:10fb
        Net L#9 "ens1f0"
      PCI 8086:10fb
        Net L#10 "ens1f1"
      PCI ffff:ffff
        Net L#11 "enp131s16"
    [root@localhost ~]# modprobe kvm-intel acpienv=N
```

Allocate the VPX memory from the NUMA domain.

The `numactl` command indicates the NUMA domain from which the memory is allocated. In the following output, around 10 GB RAM is allocated from NUMA node #0.

```
[root@localhost ~]# numactl --hardware
available: 2 nodes (0-1)
node 0 cpus: 0 1 2 3 4 5 6 7
node 0 size: 65429 MB
node 0 free: 55854 MB
node 1 cpus: 8 9 10 11 12 13 14 15
node 1 size: 65536 MB
node 1 free: 52388 MB
node distances:
node  0  1
  0:  10 21
  1:  21 10
[root@localhost ~]#
```

To change the NUMA node mapping, follow these steps.

1. Edit the .xml of the VPX on the host.

```
1 /etc/libvirt/qemu/<VPX_name>.xml
```


2. Add the following tag:

```
1 <numatune>
2 <memory mode="strict" nodeset="1"/>   ☒ This is the NUMA domain
   name
3 </numatune>
```

3. Shut down the VPX.
4. Run the following command:

```
1 virsh define /etc/libvirt/qemu/<VPX_name>.xml
```

This command updates the configuration information for the VM with the NUMA node mappings.

5. Power on the VPX. Then check the `numactl --hardware` command output on the host to see the updated memory allocations for the VPX.

```
[root@localhost ~]# numactl --hardware
available: 2 nodes (0-1)
node 0 cpus: 0 1 2 3 4 5 6 7
node 0 size: 65429 MB
node 0 free: 65429 MB
node 1 cpus: 8 9 10 11 12 13 14 15
node 1 size: 65536 MB
node 1 free: 55854 MB
node distances:
node 0 1
  0: 10 21
  1: 21 10
[root@localhost ~]#
```

Pin vCPUs of VPX to physical cores.

- To view the vCPU to pCPU mappings of a VPX, type the following command

```
1 virsh vcpupin <VPX name>
```

```
root@localhost qemu]# virsh vcpupin NS-VPX-DVR
CPU: CPU Affinity
-----
0: 8
1: 9
2: 10
3: 11
```

The vCPUs 0–4 are mapped to physical cores 8–11.

- To view the current pCPU usage, type the following command:

```
1 mpstat -P ALL 5
```

```
[root@localhost qemu]# mpstat -P ALL 5
Linux 3.10.0-123.el7.x86_64 (localhost.localdomain) 05/17/2016 _x86_64_ (16 CPU)

02:26:20 PM CPU      %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
02:26:25 PM all      0.24    0.00    1.67   0.00     0.00   0.00   0.00   17.32   0.00   80.78
02:26:25 PM 0        0.20    0.00    1.00   0.00     0.00   0.00   0.00   0.00   0.00   98.80
02:26:25 PM 1        0.20    0.00    0.20   0.00     0.00   0.00   0.00   0.00   0.00   99.60
02:26:25 PM 2        0.20    0.00    0.40   0.00     0.00   0.00   0.00   0.00   0.00   99.40
02:26:25 PM 3        0.00    0.00    0.20   0.00     0.00   0.00   0.00   0.00   0.00   99.80
02:26:25 PM 4        0.20    0.00    0.20   0.00     0.00   0.00   0.00   0.00   0.00   99.60
02:26:25 PM 5        0.60    0.00    0.20   0.00     0.00   0.00   0.00   0.00   0.00   99.20
02:26:25 PM 6        0.40    0.00    0.00   0.00     0.00   0.00   0.00   0.00   0.00   99.60
02:26:25 PM 7        1.62    0.00    1.42   0.00     0.00   0.00   0.00   0.00   0.00   96.96
02:26:25 PM 8        0.00    0.00    0.00   0.00     0.00   0.00   0.00   0.00   0.00  100.00
02:26:25 PM 9        0.00    0.00    7.60   0.00     0.00   0.00   0.00   92.40   0.00   0.00
02:26:25 PM 10       0.20    0.00    7.00   0.00     0.00   0.00   0.00   92.80   0.00   0.00
02:26:25 PM 11       0.00    0.00    8.60   0.00     0.00   0.00   0.00   91.40   0.00   0.00
02:26:25 PM 12       0.00    0.00    0.00   0.00     0.00   0.00   0.00   0.00   0.00  100.00
02:26:25 PM 13       0.00    0.00    0.00   0.00     0.00   0.00   0.00   0.00   0.00  100.00
02:26:25 PM 14       0.00    0.00    0.00   0.00     0.00   0.00   0.00   0.00   0.00  100.00
02:26:25 PM 15       0.00    0.00    0.00   0.00     0.00   0.00   0.00   0.00   0.00  100.00
```

In this output, 8 is management CPU, and 9–11 are packet engines.

- To change the vCPU to pCPU pinning, there are two options.
 - Change it at runtime after the VPX boots up using the following command:

```
1 virsh vcpupin <VPX name> <vCPU id> <pCPU number>
2 virsh vcpupin NetScaler-VPX-XML 0 8
3 virsh vcpupin NetScaler-VPX-XML 1 9
4 virsh vcpupin NetScaler-VPX-XML 2 10
5 virsh vcpupin NetScaler-VPX-XML 3 11
```

- To make static changes to the VPX, edit the `.xml` file as before with the following tags:

1. Edit the `.xml` file of the VPX on the host

```
1   ...
2   /etc/libvirt/qemu/<VPX_name>.xml
3   ...
```

2. Add the following tag:

```
1   ...
2   <vcpu placement='static' cpuset='8-11'>4</vcpu>
3     <cputune>
4       <vcpupin vcpu='0' cpuset='8'/>
5       <vcpupin vcpu='1' cpuset='9'/>
6       <vcpupin vcpu='2' cpuset='10'/>
7       <vcpupin vcpu='3' cpuset='11'/>
8     </cputune>
9   ...
```

3. Shut down the VPX.
4. Update the configuration information for the VM with the NUMA node mappings using the following command:

```

1  ``
2  virsh define /etc/libvirt/qemu/ <VPX_name>.xml
3  ``

```

5. Power on the VPX. Then check the `virsh vcpupin <VPX name>` command output on the host to see the updated CPU pinning.

Eliminate host interrupt overhead.

- Detect VM_EXITS using the `kvm_stat` command.

At the hypervisor level, host interrupts are mapped to the same pCPUs on which the vCPUs of the VPX are pinned. This might cause vCPUs on the VPX to get kicked out periodically.

To find the VM exits done by VMs running the host, use the `kvm_stat` command.

```

1  [root@localhost ~]# kvm_stat -1 | grep EXTERNAL
2  kvm_exit(EXTERNAL_INTERRUPT) 1728349 27738
3  [root@localhost ~]#

```

A higher value in the order of 1+M indicates an issue.

If a single VM is present, the expected value is 30–100 K. Anything more than that can indicate that there are one or more host interrupt vectors mapped to the same pCPU.

- Detect host interrupts and migrate host interrupts.

When you run the `concatenate` command for the “/proc/interrupts” file, it displays all the host interrupt mappings. If one or more active IRQs map to the same pCPU, its corresponding counter increments.

Move any interrupts that overlap with your Citrix ADC VPX’s pCPUs to unused pCPUs:

```

1  echo 0000000f > /proc/irq/55/smp_affinity
2  0000000f -- > it is a bitmap, LSBs indicates that IRQ 55 can
   only be scheduled on pCPUs 0 - 3

```

- Disable IRQ balance.

Disable IRQ balance daemon, so that no rescheduling happens on the fly.

```

1  service irqbalance stop
2  service irqbalance show - To check the status
3  service irqbalance start - Enable if needed

```

Make sure you run the `kvm_stat` command to ensure that there are not many counters.

Citrix ADC VPX with PV network interfaces

You can configure para-virtualization (PV), SR-IOV, and PCIe passthrough network interfaces as a **Two vNICs per pNIC** deployment. For more information, see [Two vNICs per pNIC deployment](#).

For optimal performance of PV (virtio) interfaces, follow these steps:

- Identify the NUMA domain to which the PCIe slot/NIC is tied to.
- The Memory and vCPU for the VPX must be pinned to the same NUMA domain.
- Vhost thread must be bound to the CPUs in the same NUMA domain.

Bind the virtual host threads to the corresponding CPUs:

1. Once the traffic is started, run the `top` command on the host.

```

top - 14:49:08 up 6 days, 17 min, 4 users, load average: 1.46, 0.42, 0.65
taskrt: 486 total, 3 running, 483 sleeping, 0 stopped, 0 zombie
%cpu(s): 4.1 us, 5.1 sy, 0.0 ni, 89.2 id, 0.0 wa, 0.0 hi, 1.7 si, 0.0 st
KiB Mem: 13175540+total, 6496624 used, 12525878+free, 884 buffers
KiB Swap: 4194300 total, 0 used, 4194300 free. 2088468 cached Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM     TIME+ COMMAND
 29824 qemu     20   0 12.786g 742864 8040 S 139.2 0.6  8789.04 qemu-kvm
 29838 root      20   0 0         0      0 R 100.0 0.0   5659.06 vhost-29824
 29837 root      20   0 0         0      0 R 99.7 0.0   5659.25 vhost-29824
 3063  root     20   0 1073944 23992 9396 S 1.7 0.0 111:58.18 libvirtd
 1070  root     39  19 0         0      0 S 1.0 0.0 91:35.98 kpmio
 27439 test     20   0 2710032 1.159g 25868 S 0.7 0.9 45:35.56 virt-manager
16500 root     20   0 0         0      0 S 0.3 0.0 0:16.96 kworker/25:0
 1 root     20   0 53704    7724 2536 S 0.0 0.0 0:13.69 systemd
 2 root     20   0 0         0      0 S 0.0 0.0 0:00.22 kthreadd
 3 root     20   0 0         0      0 S 0.0 0.0 384:17.42 ksotfired/0
 5 root     0 -20 0         0      0 S 0.0 0.0 0:00.00 kworker/0:0H
 6 root     20   0 0         0      0 S 0.0 0.0 0:00.00 kworker/u64:0
 8 root     rt  0 0         0      0 S 0.0 0.0 0:03.02 migration/0
 9 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcu bh
10 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/0
11 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/1
12 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/2
13 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/3
14 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/4
15 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/5
16 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/6
17 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/7
18 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/8
19 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/9
20 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/10
21 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/11
22 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/12
23 root     20   0 0         0      0 S 0.0 0.0 0:00.00 rcuob/13
    
```

2. Identify the virtual host process (named as `vhost-<pid-of-qemu>`) affinity.
3. Bind the vHost processes to the physical cores in the NUMA domain identified earlier using the following command:

```
1 taskset -pc <core-id> <process-id>
```

Example:

```
1 taskset -pc 12 29838
```

4. The processor cores corresponding to the NUMA domain can be identified with the following command:

```
1 [root@localhost ~]# virsh capabilities | grep cpu
2 <cpu>
3 </cpu>
```

```

4      <cpus num='8'>
5          <cpu id='0' socket_id='0' core_id='0' siblings='0' />
6          <cpu id='1' socket_id='0' core_id='1' siblings='1' />
7          <cpu id='2' socket_id='0' core_id='2' siblings='2' />
8          <cpu id='3' socket_id='0' core_id='3' siblings='3' />
9          <cpu id='4' socket_id='0' core_id='4' siblings='4' />
10         <cpu id='5' socket_id='0' core_id='5' siblings='5' />
11         <cpu id='6' socket_id='0' core_id='6' siblings='6' />
12         <cpu id='7' socket_id='0' core_id='7' siblings='7' />
13     </cpus>
14
15     <cpus num='8'>
16         <cpu id='8' socket_id='1' core_id='0' siblings='8' />
17         <cpu id='9' socket_id='1' core_id='1' siblings='9' />
18         <cpu id='10' socket_id='1' core_id='2' siblings='10' />
19         <cpu id='11' socket_id='1' core_id='3' siblings='11' />
20         <cpu id='12' socket_id='1' core_id='4' siblings='12' />
21         <cpu id='13' socket_id='1' core_id='5' siblings='13' />
22         <cpu id='14' socket_id='1' core_id='6' siblings='14' />
23         <cpu id='15' socket_id='1' core_id='7' siblings='15' />
24     </cpus>
25
26     <cpuselection />
27     <cpuselection />

```

Bind the QEMU process to the corresponding physical core:

1. Identify the physical cores on which the QEMU process is running. For more information, see the preceding output.
2. Bind the QEMU process to the same physical cores to which you bind the vCPUs, using the following command:

```
1 taskset -pc 8-11 29824
```

Citrix ADC VPX with SR-IOV and Fortville PCIe passthrough network interfaces

For optimal performance of the SR-IOV and Fortville PCIe passthrough network interfaces, follow these steps:

- Identify the NUMA domain to which the PCIe slot/NIC is tied to.
- The Memory and vCPU for the VPX must be pinned to the same NUMA domain.

Sample VPX XML file for vCPU and memory pinning for Linux KVM:

```

1     <domain type='kvm'>
2         <name>NetScaler-VPX</name>
3         <uuid>138f7782-1cd3-484b-8b6d-7604f35b14f4</uuid>
4         <memory unit='KiB'>8097152</memory>
5         <currentMemory unit='KiB'>8097152</currentMemory>

```

```
6     <vcpu placement='static'>4</vcpu>
7
8     <cputune>
9         <vcupin vcpu='0' cpuset='8' />
10        <vcupin vcpu='1' cpuset='9' />
11        <vcupin vcpu='2' cpuset='10' />
12        <vcupin vcpu='3' cpuset='11' />
13    </cputune>
14
15    <numatune>
16    <memory mode='strict' nodeset='1' />
17    </numatune>
18
19    </domain>
```

Citrix ADC VPX instance on Citrix Hypervisors

This section contains details of configurable options and settings, and other suggestions that help you achieve optimal performance of Citrix ADC VPX instance on Citrix Hypervisors.

- [Performance settings for Citrix Hypervisors](#)
- [Citrix ADC VPX with SR-IOV network interfaces](#)
- [Citrix ADC VPX with para-virtualized interfaces](#)

Performance settings for Citrix Hypervisors

Find the NUMA domain of the NIC using the “xl” command:

```
1 xl info -n
```

Pin vCPUs of VPX to physical cores.

```
1 xl vcpu-pin <Netsclaer VM Name> <vCPU id> <physical CPU id>
```

Check binding of vCPUs.

```
1 xl vcpu-list
```

Allocate more than 8 vCPUs to Citrix ADC VMs.

For configuring more than 8 vCPUs, run the following commands from the Citrix Hypervisor console:

```
1 xe vm-param-set uuid=your_vms_uuid VCPUs-max=16
2 xe vm-param-set uuid=your_vms_uuid VCPUs-at-startup=16
```

Citrix ADC VPX with SR-IOV network interfaces

For optimal performance of the SR-IOV network interfaces, follow these steps:

- Identify the NUMA domain to which the PCIe slot or NIC is tied to.
- Pin the Memory and vCPU for the VPX to the same NUMA domain.
- Bind the Domain-0 vCPU to the remaining CPU.

Citrix ADC VPX with para-virtualized interfaces

For optimal performance, two vNICs per pNIC and one vNIC per pNIC configurations are advised, as in other PV environments.

To achieve optimal performance of para-virtualized (netfront) interfaces, follow these steps:

- Identify the NUMA domain to which the PCIe slot or NIC is tied to.
- Pin the memory and vCPU for the VPX to the same NUMA domain.
- Bind the Domain-0 vCPU to the remaining CPU of the same NUMA domain.
- Pin host Rx/Tx threads of vNIC to Domain-0 vCPUs.

Pin host threads to Domain-0 vCPUs:

1. Find Xen-ID of the VPX by using the `xl list` command on the Citrix Hypervisor host shell.
2. Identify host threads by using the following command:

```
1 ps -ax | grep vif <Xen-ID>
```

In the following example, these values indicate:

- **vif5.0** - The threads for first interface allocated to VPX in XenCenter (management interface).
- **vif5.1** - The threads for second interface assigned to VPX and so on.

```
[root@xenserver-uuffyqlx ~]# xl list
Name                               ID    Mem  VCPUs    State    Time(s)
Domain-0                           0    4092    8    r----- 633321.0
Sai_VPX                             5    8192    4    r----- 1529471.0
[root@xenserver-uuffyqlx ~]#
[root@xenserver-uuffyqlx ~]#
[root@xenserver-uuffyqlx ~]# ps -ax | grep "vif5"
Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.7/FAQ
20447 pts/6      S+      0:00  grep vif5
29187 ?           S       1:09  [vif5.0-guest-rx]
29188 ?           S       0:00  [vif5.0-dealloc]
29189 ?           S      201:33 [vif5.1-guest-rx]
29190 ?           S      80:51  [vif5.1-dealloc]
29191 ?           S       0:20  [vif5.2-guest-rx]
29192 ?           S       0:00  [vif5.2-dealloc]
[root@xenserver-uuffyqlx ~]#
```

3. Pin the threads to Domain-0 vCPUs using the following command:

```
1 taskset -pc <core-id> <process-id>
```

Example:

```
1 taskset -pc 1 29189
```

Install a Citrix ADC VPX instance on a bare metal server

November 13, 2024

A bare metal is a fully dedicated physical server that delivers physical isolation, fully integrated into the cloud environment. It is also known as a single-tenant server. Single tenancy allows you to avoid the noisy neighbor effect. With bare metal, you do not witness the noisy neighbor effect because you are the sole user.

A bare metal server installed with a hypervisor provides you a management suite to create virtual machines on the server. The hypervisor does not run applications natively. Its purpose is to virtualize your workloads into separate virtual machines to gain the flexibility and reliability of virtualization.

Prerequisites for installing Citrix ADC VPX instance on bare metal servers

A bare metal server must be obtained from a cloud vendor that meets all the system requirements for the respective hypervisor.

Install the Citrix ADC VPX instance on bare metal servers

To install Citrix ADC VPX instances on a bare metal server, you must first obtain a bare metal server with adequate system resources from a cloud vendor. On that bare metal server, any of the supported hypervisors such as Linux KVM, VMware ESX, Citrix Hypervisor, or Microsoft Hyper-V must be installed and configured before deploying the ADC VPX instance.

For more information on the list of different hypervisors and features supported on a Citrix ADC VPX instance, see [Support matrix and usage guidelines](#).

For more information on installing Citrix ADC VPX instances on different hypervisors, see the respective documentation.

- **Citrix Hypervisor:** See [Install a Citrix ADC VPX instance on Citrix Hypervisor](#).
- **VMware ESX:** See [Install a Citrix ADC VPX instance on VMware ESX](#).

- **Microsoft Hyper-V:** See [Install a Citrix ADC VPX instance on Microsoft Hyper-V server.](#)
- **Linux KVM platform:** See [Install a Citrix ADC VPX instance on Linux-KVM platform.](#)

Install a Citrix ADC VPX instance on XenServer

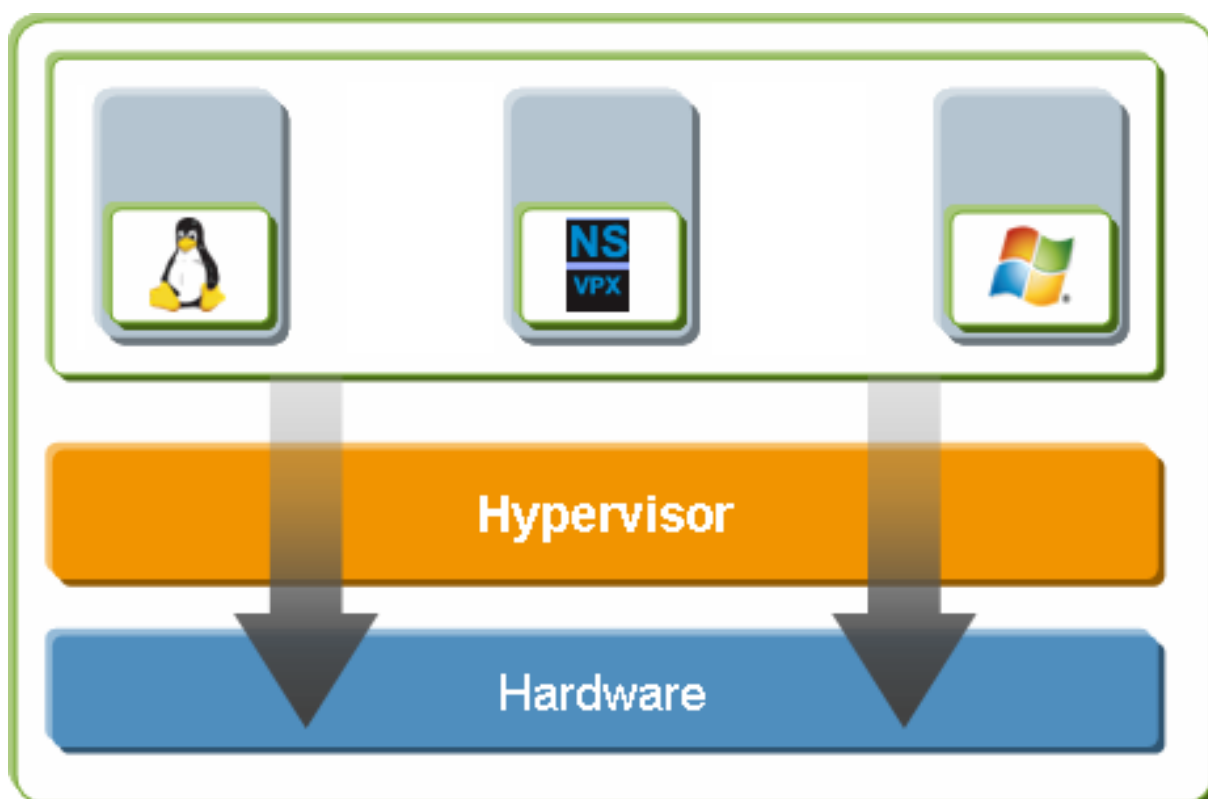
November 18, 2024

To install VPX instances on Citrix XenServer, you must first install XenServer on a machine with adequate system resources. To perform the Citrix ADC VPX instance installation, you use Citrix XenCenter, which must be installed on a remote machine that can connect to the XenServer host through the network.

For more information about XenServer, see [XenServer documentation.](#)

The following figure shows the bare-metal solution architecture of Citrix ADC VPX instance on XenServer.

Figure. A Citrix ADC VPX instance on XenServer



Prerequisites for installing a Citrix ADC VPX instance on XenServer

Before you begin installing a virtual appliance, do the following:

- Install XenServer version 6.0 or later on hardware that meets the minimum requirements.
- Install XenCenter on a management workstation that meets the minimum system requirements.
- Obtain virtual appliance license files. For more information about virtual appliance licenses, see the [Citrix ADC Licensing Guide](#).

XenServer hardware requirements

The following table describes the minimum hardware requirements for a XenServer platform running a Citrix ADC VPX instance.

Table 1. Minimum system requirements for XenServer running a nCore VPX instance

Component	Requirement
CPU	2 or more 64-bit x86 CPUs with virtualization assist (Intel-VT) enabled. AMD processor is not supported. To run Citrix ADC VPX instance, hardware support for virtualization must be enabled on the XenServer host. Make sure that the BIOS option for virtualization support is not disabled. For more details, see BIOS documentation.
RAM	3 GB
Disk space	Locally attached storage (PATA, SATA, SCSI) with 40 GB of disk space. Note: XenServer installation creates a 4 GB partition for the XenServer host control domain; the remaining space is available for Citrix ADC VPX instance and other virtual machines.
NIC	One 1-Gbps NIC; recommended: two 1-Gbps NICs

For information about installing XenServer, see the XenServer documentation at <http://support.citrix.com/product/xens/>.

The following table lists the virtual computing resources that XenServer must provide for each nCore VPX virtual appliance.

Table 2. Minimum virtual computing resources required for running a ncore VPX instance

Component	Requirement
Memory	2 GB
Virtual CPU (VCPU)	2
Virtual network interfaces	2

Note:

For production use of Citrix ADC VPX instance, Citrix recommends that CPU priority (in virtual machine properties) be set to the highest level, in order to improve scheduling behavior and network latency.

XenCenter system requirements

XenCenter is a Windows client application. It cannot run on the same machine as the XenServer host. For more information about minimum system requirements and installing XenCenter, see the following XenServer documents:

- [System requirements](#)
- [Install](#)

Install Citrix ADC VPX instances on XenServer by using XenCenter

After you have installed and configured XenServer and XenCenter, you can use XenCenter to install virtual appliances on XenServer. The number of virtual appliances that you can install depends on the amount of memory available on the hardware that is running XenServer.

To install Citrix ADC VPX instances on XenServer by using XenCenter, follow these steps:

1. Start XenCenter on your workstation.
2. On the Server menu, click **Add**.
3. In the Add New Server dialog box, in the Hostname text box, type the IP address or DNS name of the XenServer that you want to connect to.
4. In the User Name and Password text boxes, type the administrator credentials, and then click Connect. The XenServer name appears in the navigation pane with a green circle, which indicates that the XenServer is connected.

5. In the navigation pane, click the name of the XenServer on which you want to install Citrix ADC VPX instance.
6. On the VM menu, click **Import**.
7. In the Import dialog box, in Import file name, browse to the location at which you saved the Citrix ADC VPX instance .xva image file. Make sure that the Exported VM option is selected, and then click Next.
8. Select the XenServer on which you want to install the virtual appliance, and then click Next.
9. Select the local storage repository in which to store the virtual appliance, and then click Import to begin the import process.
10. You can add, modify, or delete virtual network interfaces as required. When finished, click Next.
11. Click **Finish** to complete the import process.

Note:

To view the status of the import process, click the **Log** tab.

12. If you want to install another virtual appliance, repeat steps 5 through 11.

Note:

After the initial configuration of the VPX instance, if you want to upgrade the appliance to the latest software release, see [Upgrading or Downgrading the System Software](#).

Configure VPX instances to use single root I/O virtualization (SR-IOV) network interfaces

November 13, 2024

After you have installed and configured a Citrix ADC VPX instance on Citrix Hypervisor, you can configure the virtual appliance to use SR-IOV network interfaces.

Limitations

Citrix Hypervisor does not support the following features on SRIOV interfaces:

- L2 mode switching
- Clustering
- Admin partitioning [Shared VLAN mode]

- High Availability [Active - Active mode]
- Jumbo frames
- IPv6 protocol in Cluster environment

Prerequisites

On the Citrix Hypervisor host, ensure that you:

- Add the Intel 82599 Network Interface Card (NIC) to the host.
- Blacklist the ixgbevf driver by adding the following entry to the **/etc/modprobe.d/blacklist.conf** file:
blacklist ixgbevf
- Enable SR-IOV Virtual Functions (VFs) by adding the following entry to the **/etc/modprobe.d/ixgbe** file:
options ixgbe max_vfs=<number_of_VFs>
where *<number_of_VFs>* is the number of SR-IOV VFs that you want to create.
- Verify that SR-IOV is enabled in BIOS.

Note:

IXGBE driver version 3.22.3 is recommended.

Assign SR-IOV VFs to the VPX instance by using the Citrix Hypervisor host

To assign SR-IOV network interfaces to Citrix ADC VPX instance, follow these steps:

1. On the Citrix Hypervisor host, use the following command to assign the SR-IOV VFs to the Citrix ADC VPX instance:

```
xe host-call-plugin plugin=iovirt host-uuid=<Xen host UUID> fn=assign_free_vf args:uuid=<Netscaler VM UUID> args:ethdev=<interface name> args:mac=<mac addr>
```

Where:

- *<Xen host UUID>* is the UUID of the Citrix Hypervisor host.
- *<Netscaler VM UUID>* is the UUID of the Citrix ADC VPX instance.
- *<interface name>* is the interface for the SR-IOV VFs.
- *<mac addr >* is the mac address of the SR-IOV VF.

Note:

Specify the mac address that you want use in the `args:mac=` parameter, if not specified, the `iovirt` script randomly generates and assigns a mac address. Also, if you want use the SR-IOV VFs in Link

Aggregation mode, make sure that you specify the mac address as 00:00:00:00:00:00.

2. Boot the Citrix ADC VPX instance.

Unassign SR-IOV VFs to the VPX instance by using the Citrix Hypervisor host

If you have assigned an incorrect SR-IOV VFs or if you want modify the a assigned SR-IOV VFs, you need to unassign and reassign the SR-IOV VFs to the Citrix ADC VPX instance.

To unassign SR-IOV network interface assigned to a Citrix ADC VPX instance, follow these steps:

1. On the Citrix Hypervisor host, use the following command to assign the SR-IOV VFs to the Citrix ADC VPX instance and reboot the Citrix ADC VPX instance:

```
xe host-call-plugin plugin=iovirt host-uuid=<Xen_host_UUID> fn=unassign_all args:uuid=<Netscaler_VM_UUID>
```

Where:

- <Xen_host_UUID> - The UUID of the Citrix Hypervisor host.
- <Netscaler_VM_UUID> - The UUID of the Citrix ADC VPX instance

2. Boot the Citrix ADC VPX instance.

Configure link aggregation on the SR-IOV interface

To use the SR-IOV virtual functions in link aggregation mode, you need to disable spoof checking for virtual functions that you have created. On the Citrix Hypervisor host, use the following command to disable spoof checking:

```
ip link set <interface_name> vf <VF_id> spoofchk off
```

Where:

- <interface_name> is the interface name.
- <VF_id> is the virtual function ID.

After disabling spoof checking for all the virtual functions that you have created, restart the Citrix ADC VPX instance and configure link aggregation. For instructions, see [Configure link aggregation](#).

Important:

While you are assigning the SR-IOV VFs to the Citrix ADC VPX instance, make sure that you specify MAC address 00:00:00:00:00:00 for the VFs.

Configure VLAN on the SR-IOV interface

You can configure VLAN on the SR-IOV Virtual Functions, for instructions, see [Configuring a VLAN](#).

Important:

Make sure that the Citrix Hypervisor host does not contain VLAN settings for the VF interface.

Install a Citrix ADC VPX instance on VMware ESX

November 18, 2024

Before installing Citrix ADC VPX instances on VMware ESX, make sure that VMware ESX Server is installed on a machine with adequate system resources. To install a Citrix ADC VPX instance on VMware ESXi, you use the VMware vSphere client. The client or tool must be installed on a remote machine that can connect to VMware ESX through the network.

This section includes the following topics:

- Prerequisites
- Installing a Citrix ADC VPX instance on VMware ESX

Important:

You cannot install standard VMware Tools or upgrade the VMware Tools version available on a Citrix ADC VPX instance. VMware Tools for a Citrix ADC VPX instance are delivered as part of the Citrix ADC software release.

Prerequisites

Before you begin installing a virtual appliance, do the following:

- Install VMware ESX on hardware that meets the minimum requirements.
- Install VMware Client on a management workstation that meets the minimum system requirements.
- Download the Citrix ADC VPX appliance setup files.
- Label the physical network ports of VMware ESX.
- Obtain VPX license files. For more information about Citrix ADC VPX instance licenses, see [Licensing overview](#).

VMware ESX hardware requirements

The following table describes the minimum system requirements for VMware ESX servers running Citrix ADC VPX ncore virtual appliance.

Table 1. Minimum system requirements for a VMware ESX server running a Citrix ADC VPX instance

Component	Requirement
-	2 or more 64-bit x86 CPUs with virtualization assist (Intel-VT) enabled. To run Citrix ADC VPX instance, hardware support for virtualization must be enabled on the VMware ESX host. Make sure that the BIOS option for virtualization support is not disabled. For more information, see your BIOS documentation.
RAM	2 GB VPX. For critical deployments, we do not recommend 2 GB RAM for VPX because the system operates in a memory-constrained environment. This might lead to scale, performance, or stability related issues. Recommended is 4 GB RAM or 8 GB RAM.
Disk space	20 GB more than the minimum server requirements from VMware for setting up ESXi. See VMware documentation for minimum server requirements.
Network	One 1-Gbps NIC. Two 1-Gbps NICs recommended

For information about installing VMware ESX, see [VMware documentation](#).

To enable SR-IOV or PCI passthrough support, ensure that the following processors are supported:

- Intel processors support Intel-VT.
- I/O Memory Management Unit (IOMMU) or SR-IOV is enabled in BIOS.

The following table lists the virtual computing resources that the VMware ESX server must provide for each VPX ncore virtual appliance.

Table 2. Minimum virtual computing resources required for running a Citrix ADC VPX instance

Component	Requirement
Memory	2 GB
Virtual CPU (vCPU)	2
Virtual network interfaces	1. With ESX, you can install a maximum of 10 virtual network interfaces if the VPX hardware is upgraded version to 7 or higher.
Disk space	20 GB

This is in addition to any disk requirements for the hypervisor.

For production use of VPX virtual appliance, the full memory allocation must be reserved. CPU cycles (in MHz) equal to at least the speed of one CPU core of the ESX must be reserved.

VMware vSphere client system requirements

VMware vSphere is a client application that can run on Windows and Linux operating systems. It cannot run on the same machine as the VMware ESX server. The following table describes the minimum system requirements.

Table 3. Minimum system requirements for VMware vSphere client installation

Component	Requirement
Operating system	For detailed requirements from VMware, search for the “vSphere Compatibility Matrixes” PDF file at http://kb.vmware.com/ .
CPU	750 MHz; 1 gigahertz (GHz) or faster recommended
RAM	1 GB. 2 GB recommended
Network Interface Card (NIC)	100 Mbps or faster NIC

OVF Tool 1.0 system requirements

OVF Tool is a client application that can run on Windows and Linux systems. It cannot run on the same machine as the VMware ESX server. The following table describes the minimum system requirements.

Table 4. Minimum system requirements for OVF tool installation

Component	Requirement
Operating system	For detailed requirements from VMware, search for the “OVF Tool User Guide” PDF file at http://kb.vmware.com/ .
CPU	750 MHz minimum, 1 GHz or faster recommended
RAM	1 GB Minimum, 2 GB recommended
Network Interface Card (NIC)	100 Mbps or faster NIC

For details on installing OVF, search for the “OVF Tool User’s Guide” PDF available at [VMware documentation](#).

Downloading the Citrix ADC VPX setup files

The Citrix ADC VPX instance setup package for VMware ESX follows the Open Virtual Machine (OVF) format standard. You can download the files from the Citrix website. You need a Citrix account to log on. If you do not have a Citrix account, access the home page at <http://www.citrix.com>, click the **New Users link**, and follow the instructions to create a new Citrix account.

Once logged on, navigate the following path from the Citrix home page:

Citrix.com > **Downloads** > **Citrix ADC** > **Virtual Appliances**.

Copy the following files to a workstation on the same network as the ESX server. Copy all three files into the same folder.

- NSVPX-ESX-<release number>-<build number>-disk1.vmdk (for example, NSVPX-ESX-9.3-39.8-disk1.vmdk)
- NSVPX-ESX-<release number>-<build number>.ovf (for example, NSVPX-ESX-9.3-39.8.ovf)
- NSVPX-ESX-<release number>-<build number>.mf (for example, NSVPX-ESX-9.3-39.8.mf)

Label the physical network ports of VMware ESX

Before installing a VPX virtual appliance, label all the interfaces that you plan to assign to virtual appliances, in a unique format, for example, NS_NIC_1_1, NS_NIC_1_2, and so on. In large deployments, labeling in a unique format helps in quickly identifying the interfaces that are allocated to the VPX virtual appliance among other interfaces used by other virtual machines, such as Windows and Linux. Such labeling is especially important when different types of virtual machines share interfaces.

To label the physical network ports of VMware ESX server, follow these steps:

1. Log on to the VMware ESX server by using the vSphere client.
2. On the vSphere client, select the **Configuration** tab, and then click Networking.
3. At the top-right corner, click Add Networking.
4. In the Add Network Wizard, for **Connection Type**, select **Virtual Machine**, and then click Next.
5. Scroll through the list of vSwitch physical adapters, and choose the physical port that maps to interface 1/1 on the virtual appliances.
6. Enter the label of the interface, for example, **NS_NIC_1_1** as the name of the vSwitch that is associated with interface 1/1 of the virtual appliances.
7. Click **Next** to finish the vSwitch creation. Repeat the procedure, beginning with step 2, to add any additional interfaces to be used by your virtual appliances. Label the interfaces sequentially, in the correct format (for example, NS_NIC_1_2).

Installing a Citrix ADC VPX instance on VMware ESX

After you have installed and configured VMware ESX, you can use the VMware vSphere client to install virtual appliances on the VMware ESX server. The number of virtual appliances that you can install depends on the amount of memory available on the hardware that is running VMware ESX.

To install Citrix ADC VPX instances on VMware ESX by using VMware vSphere Client, follow these steps:

1. Start the VMware vSphere client on your workstation.
2. In the **IP address / Name** text box, type the IP address of the VMware ESX server that you want to connect to.
3. In the **User Name** and **Password** text boxes, type the administrator credentials, and then click Login.
4. On the **File** menu, click **Deploy OVF Template**.
5. In the **Deploy OVF Template** dialog box, in **Deploy from file**, browse to the location at which you saved the Citrix ADC VPX instance setup files, select the .ovf file, and click **Next**.
6. Map the networks shown in the virtual appliance OVF template to the networks that you configured on the ESX host. Click **Next** to start installing a virtual appliance on VMware ESX. When installation is complete, a pop-up window informs you of the successful installation.
7. You are now ready to start the Citrix ADC VPX instance. In the navigation pane, select the Citrix ADC VPX instance that you have installed and, from the right-click menu, select **Power On**.
8. After the VM is booted, from the console, configure the Citrix ADC IP, Netmask, and Gateway addresses. When you complete the configuration, select the **Save and Quit** option in the console.
9. If you want to install another virtual appliance, repeat through step 6.

Note:

By default, the Citrix ADC VPX instance uses E1000 network interfaces.

After the installation, you can use vSphere client or vSphere Web Client to manage virtual appliances on VMware ESX.

Note:

For the VLAN tagging feature to work, on the VMware ESX, set the port group's VLAN ID to 1–4095 on the vSwitch of VMware ESX server.

Migrate a Citrix ADC VPX instance by using VMware vMotion

You can migrate a Citrix ADC VPX instance by using VMware vSphere vMotion.

Follow these usage guidelines:

- VMware does not support the vMotion feature on virtual machines configured with PCI Passthrough and SR-IOV interfaces.
- Supported interfaces are E1000 and VMXNET3. To use vMotion on your VPX instance, ensure that the instance is configured with a supported interface.
- For more information about how to migrate an instance by using VMware vMotion, see VMware documentation.

Configure a Citrix ADC VPX instance to use VMXNET3 network interface

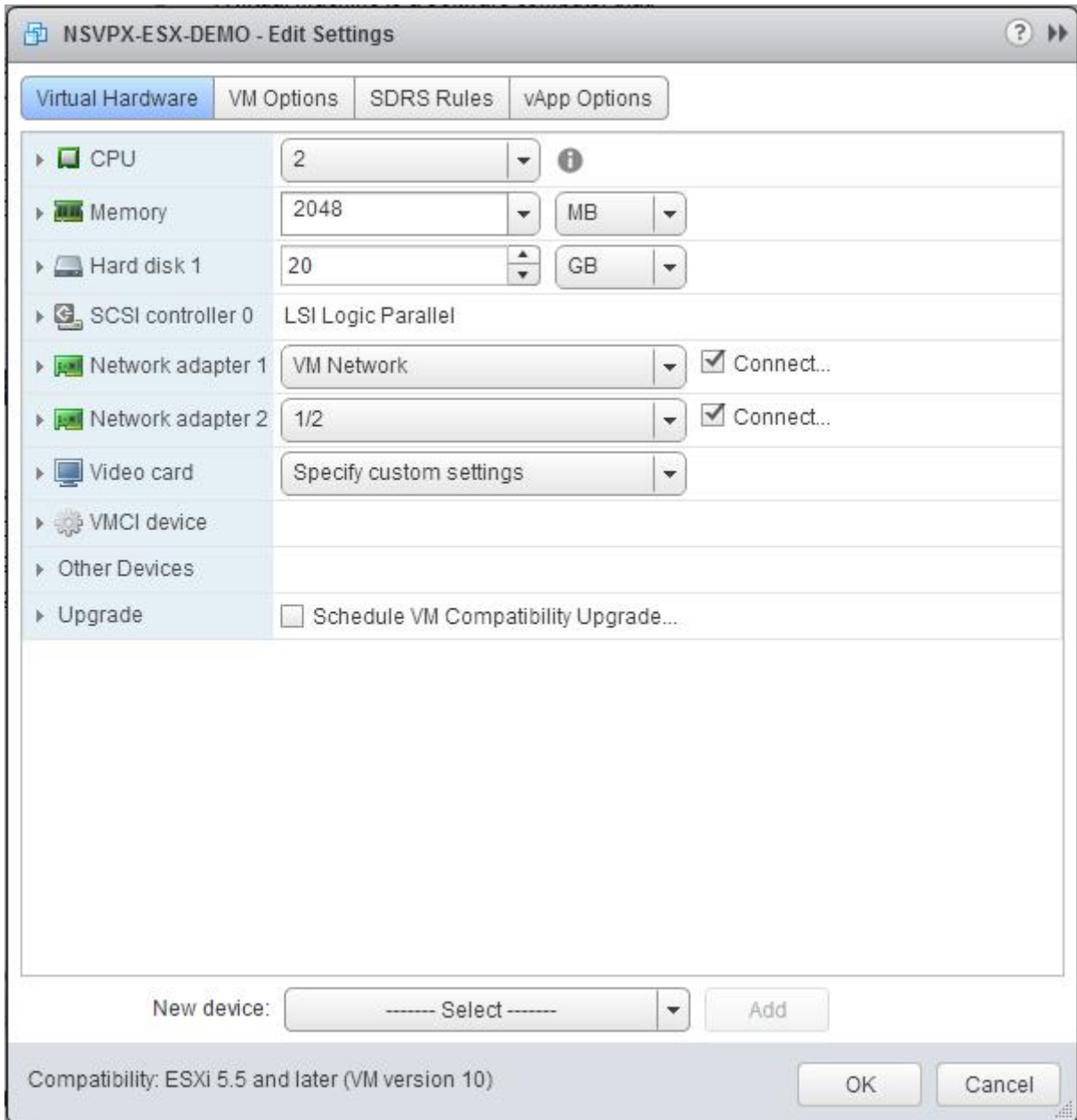
November 14, 2024

After you have installed and configured the Citrix ADC VPX instance on the VMware ESX, you can use the VMware vSphere web client to configure the virtual appliance to use VMXNET3 network interfaces.

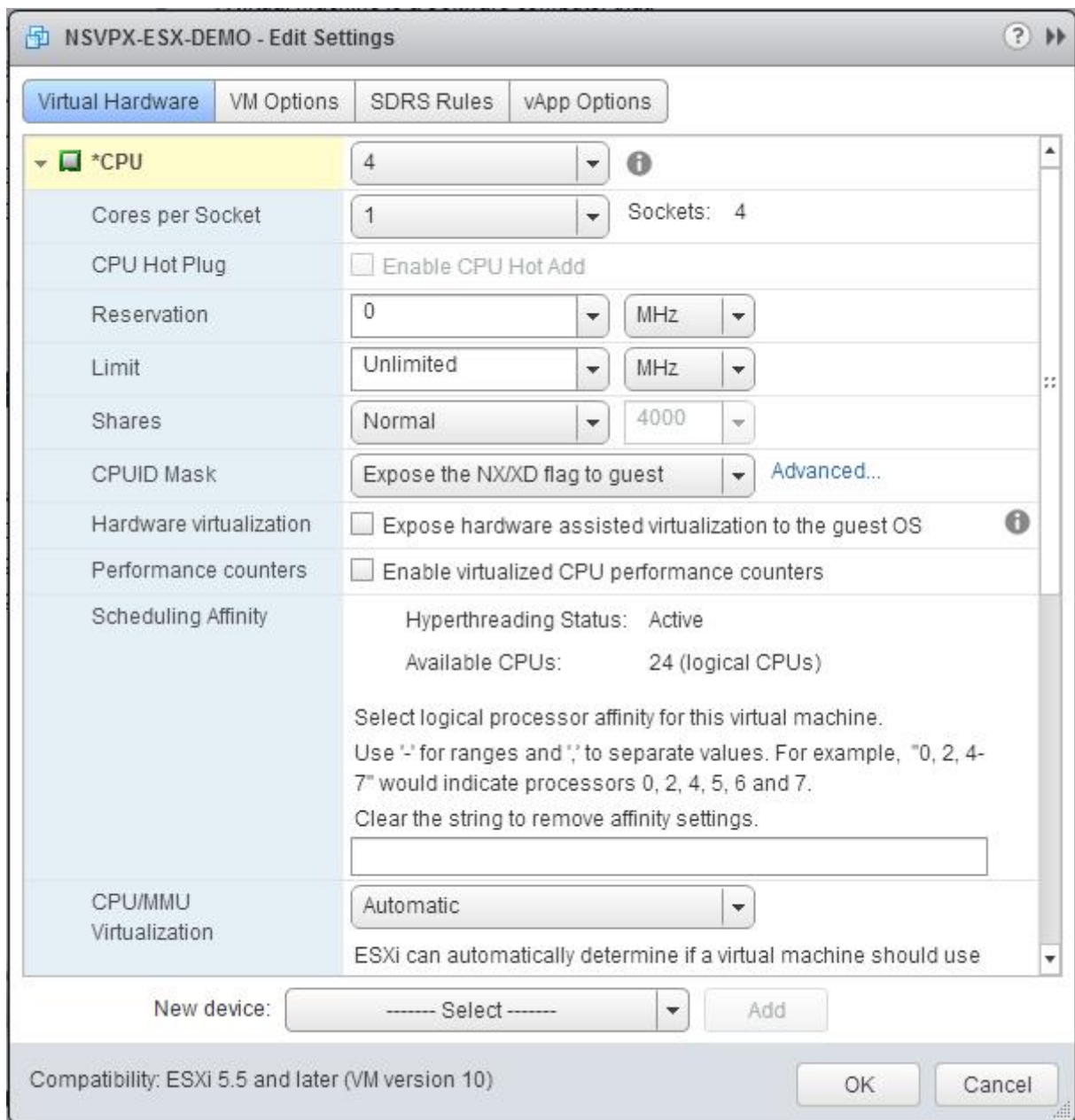
To configure Citrix ADC VPX instances to use VMXNET3 network interfaces by using the VMware vSphere Web Client:

1. In the vSphere Web Client, select Hosts and Clusters.
1. Upgrade the Compatibility setting of the Citrix ADC VPX instance to ESX, as follows:
 1. Power off the Citrix ADC VPX instance.
 1. Right-click the Citrix ADC VPX instance and select Compatibility > Upgrade VM Compatibility.
 1. In the Configure VM Compatibility dialog box, select ESXi 5.5 and later from the Compatible with drop-down list and click OK.

3. Right-click on the Citrix ADC VPX instance and click Edit Settings.



4. In the <virtual_appliance> - Edit Settings dialog box, click the CPU section.



5. In the CPU section, update the following:

- Number of CPUs
- Number of Sockets
- Reservations
- Limit
- Shares

Set the values as follows:

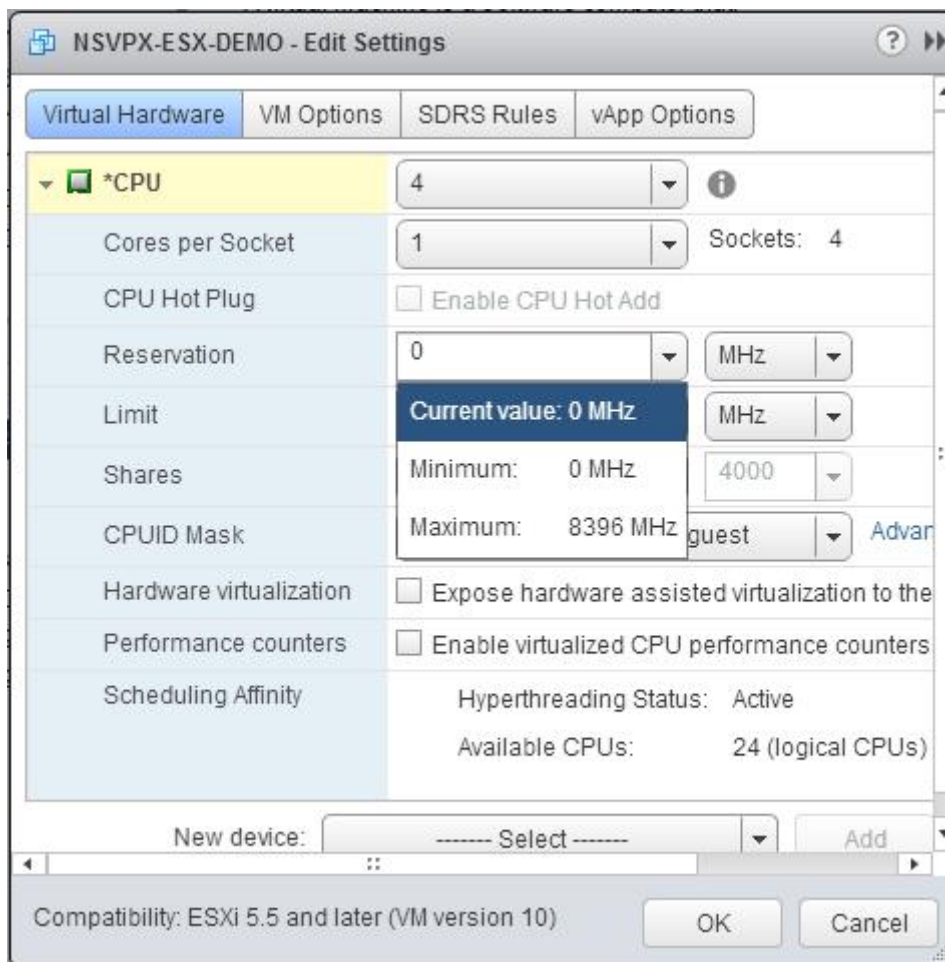
1. In the CPU drop-down list, select the number of CPUs to assign to the virtual appliance.

1. In the Cores per Socket drop-down list, select the number of sockets.

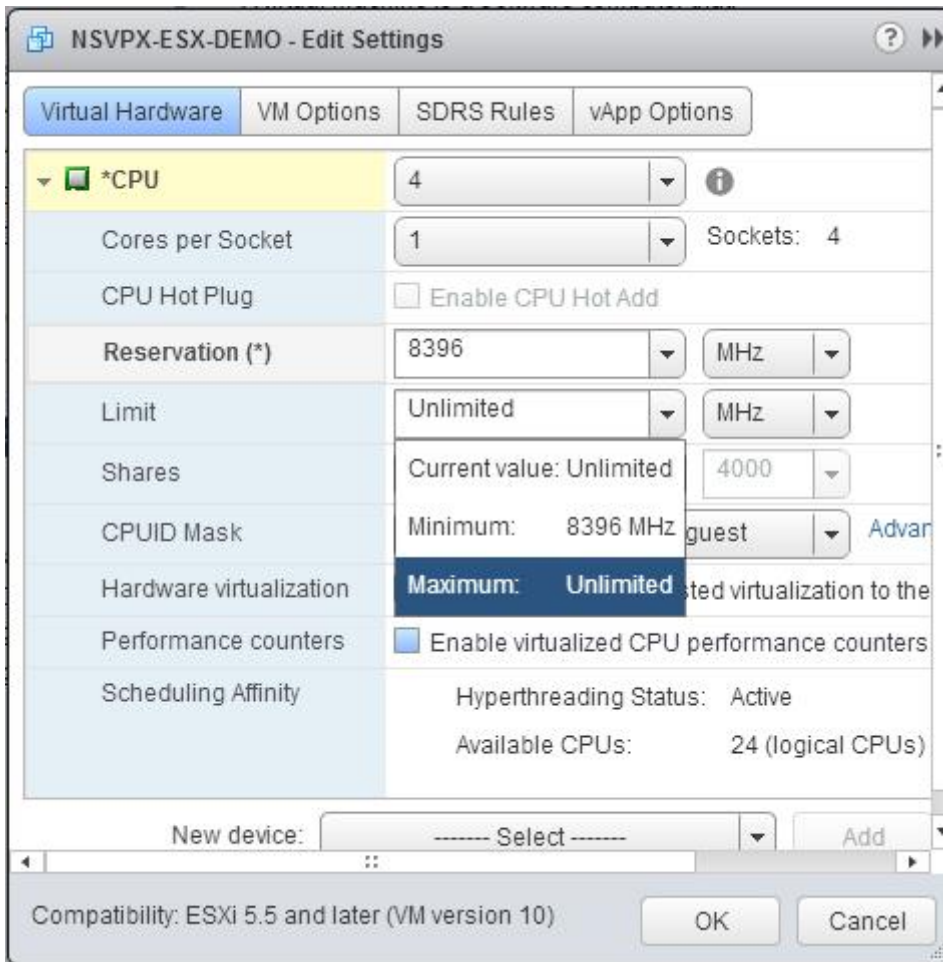
1. (Optional) In the CPU Hot Plug field, select or unselect the Enable CPU Hot Add checkbox.

Note: Citrix recommends accepting the default (disabled).

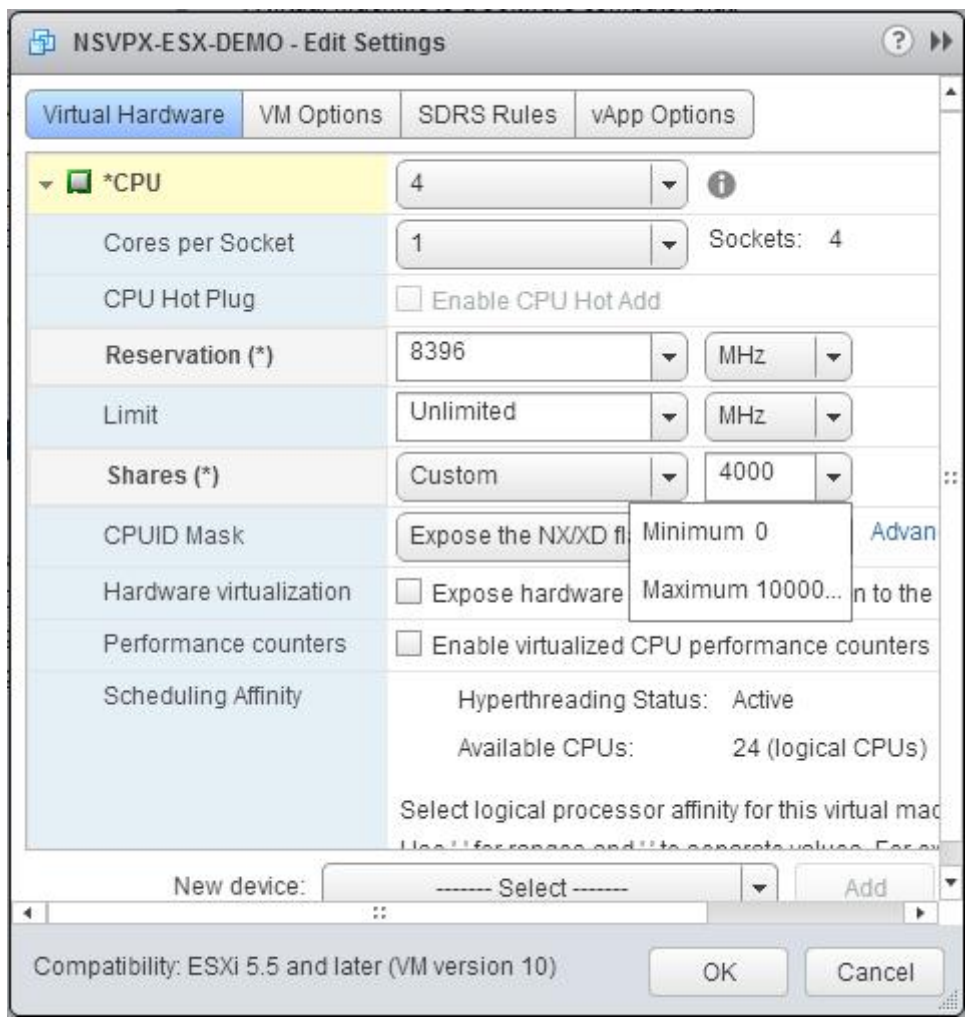
1. In the Reservation drop-down list, select the number that is shown as the maximum value.



- 1. In the Limit drop-down list, select the number that is shown as the maximum value.



1. In the Shares drop-down lists, select Custom and the number that is shown as the maximum value.



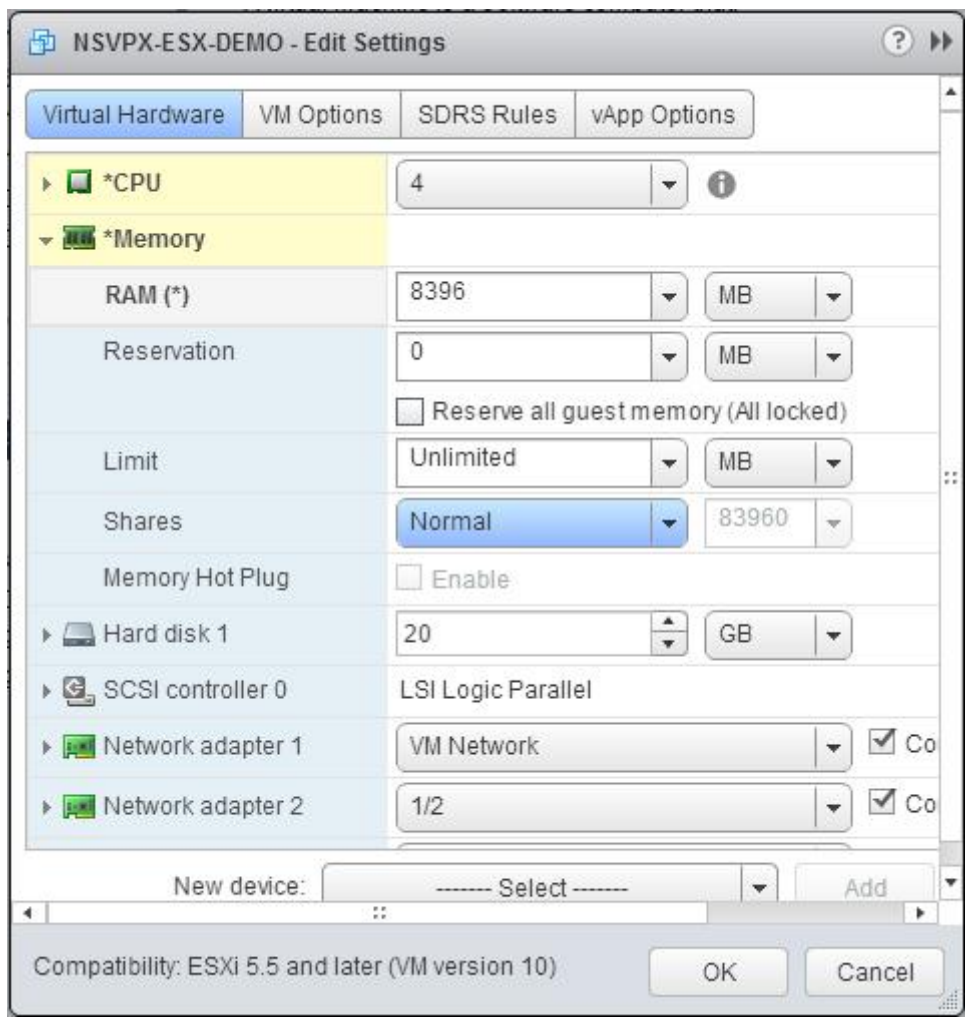
6. In the Memory section, update the following:

- Size of RAM
- Reservations
- Limit
- Shares

Set the values as follows:

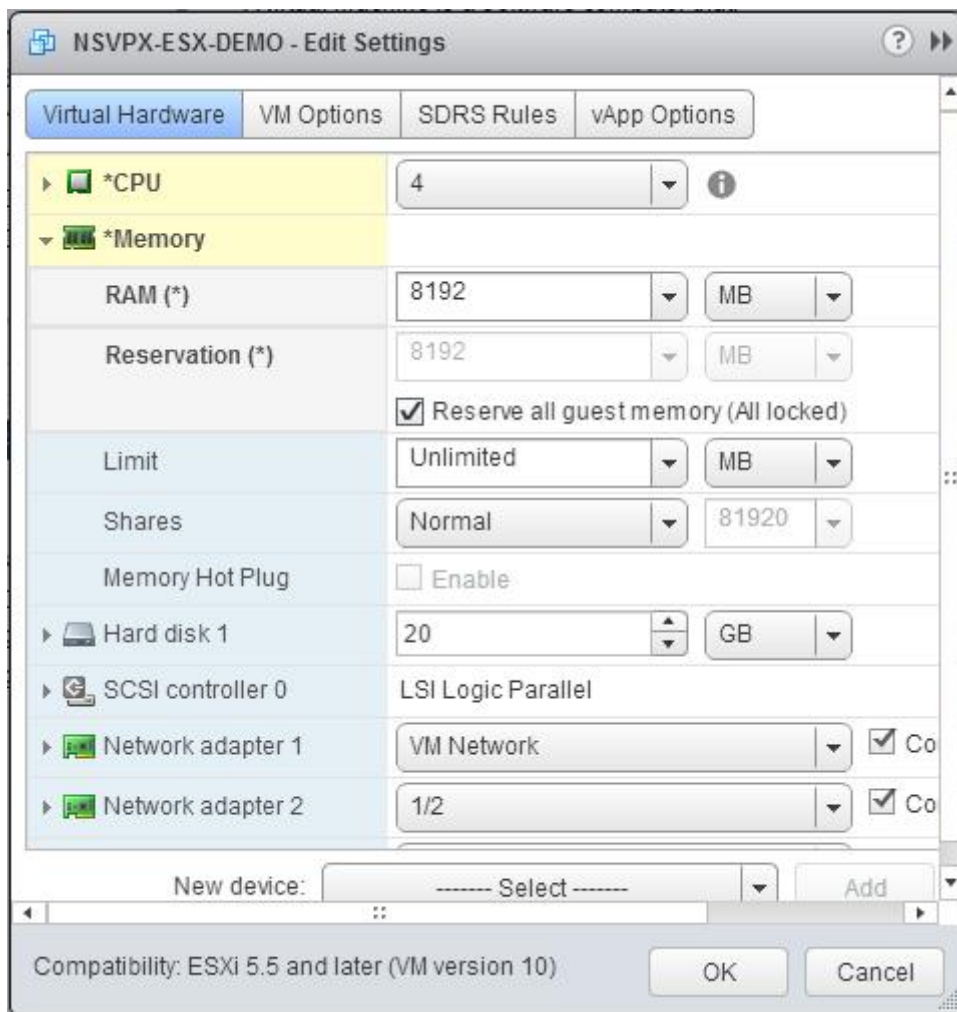
1. In the RAM drop-down list, select the size of the RAM. It should be number of vCPUs x 2 GB. For example, if the number of vCPUs is 4, the RAM should be 4 x 2 GB = 8 GB.

Note: For an Advanced or Premium edition of the Citrix ADC VPX appliance, make sure that you allocate 4 GB of RAM to each vCPU. For example, if the number of vCPU is 4 then RAM = 4 x 4 GB = 16 GB.

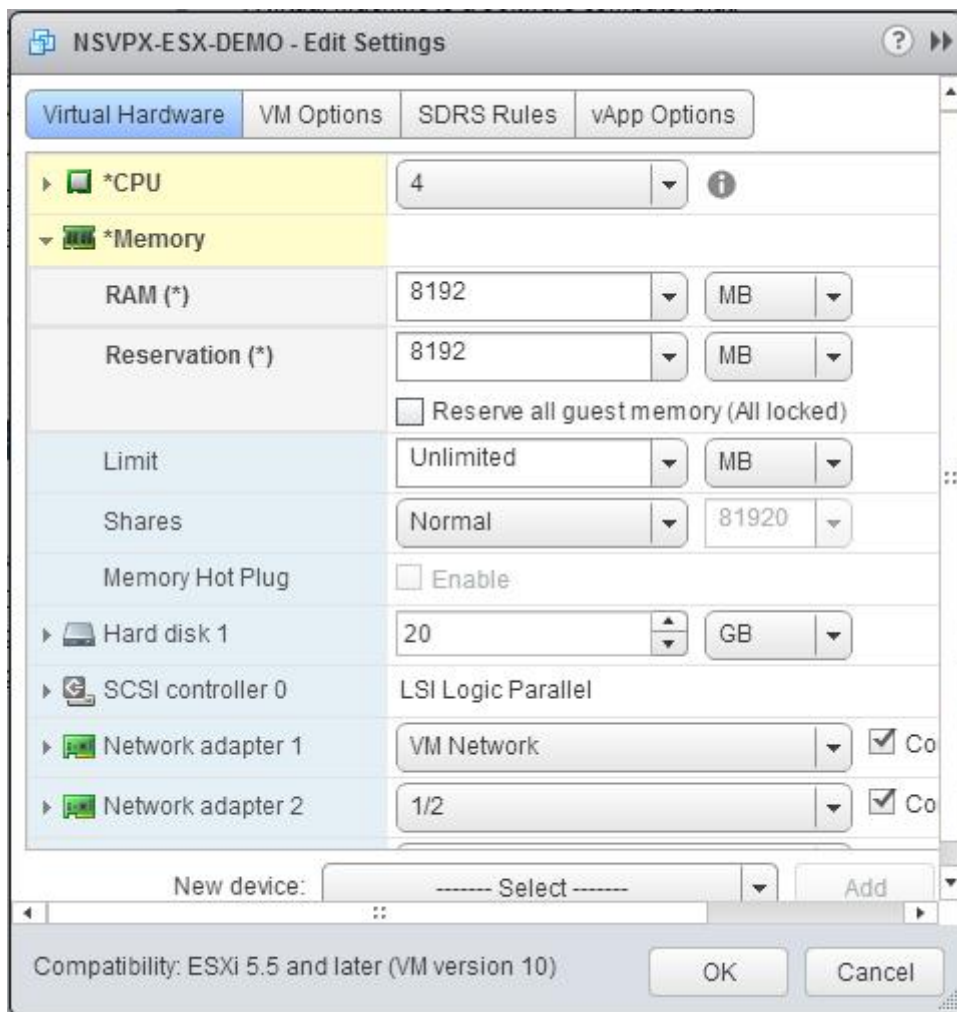


b. In the Reservation drop-down list, enter the value for the memory reservation, and select the Reserve all guest memory (All locked) checkbox. The memory reservation should be the number of vCPUs x 2 GB. For example, if the number of vCPUs is 4, the memory reservation should be 4 x 2 GB = 8 GB.

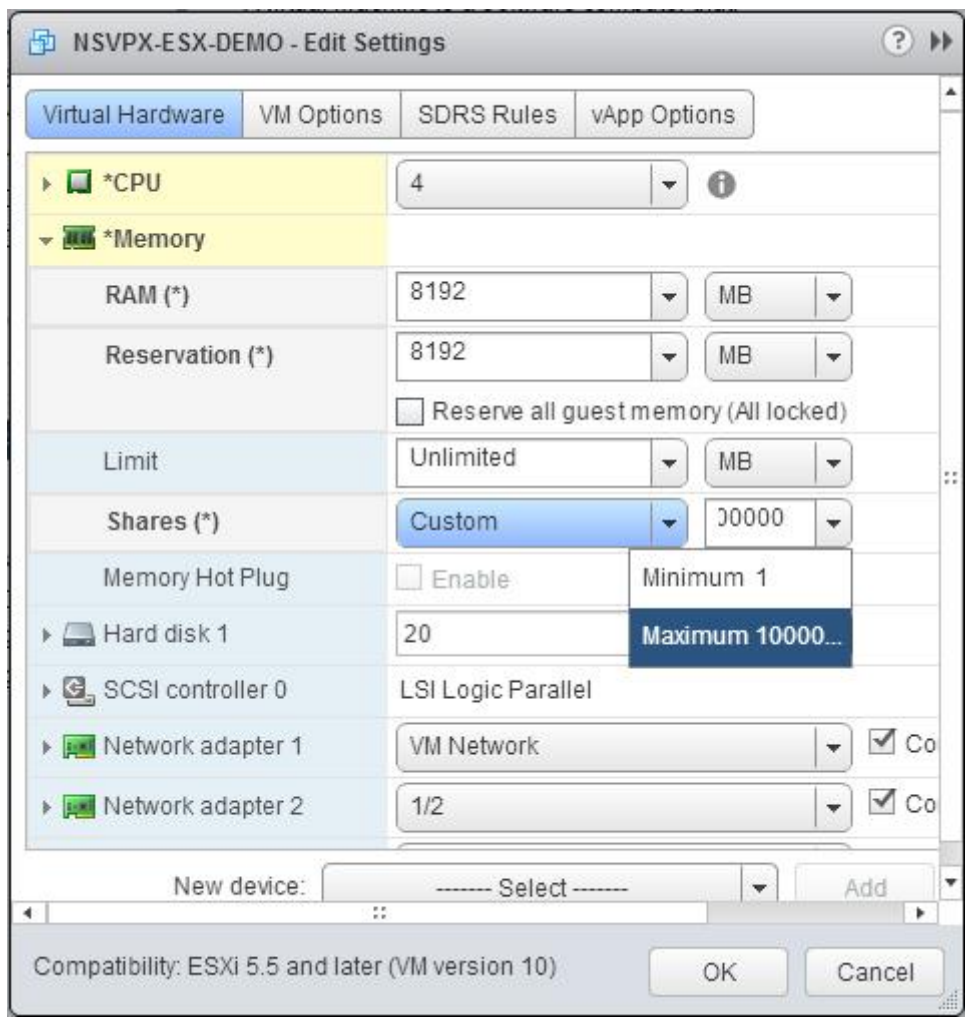
Note: For an Advanced or Premium edition of the Citrix ADC VPX appliance, make sure that you allocate 4 GB of RAM to each vCPU. For example, if the number of vCPU is 4 then RAM = 4 x 4 GB = 16 GB.



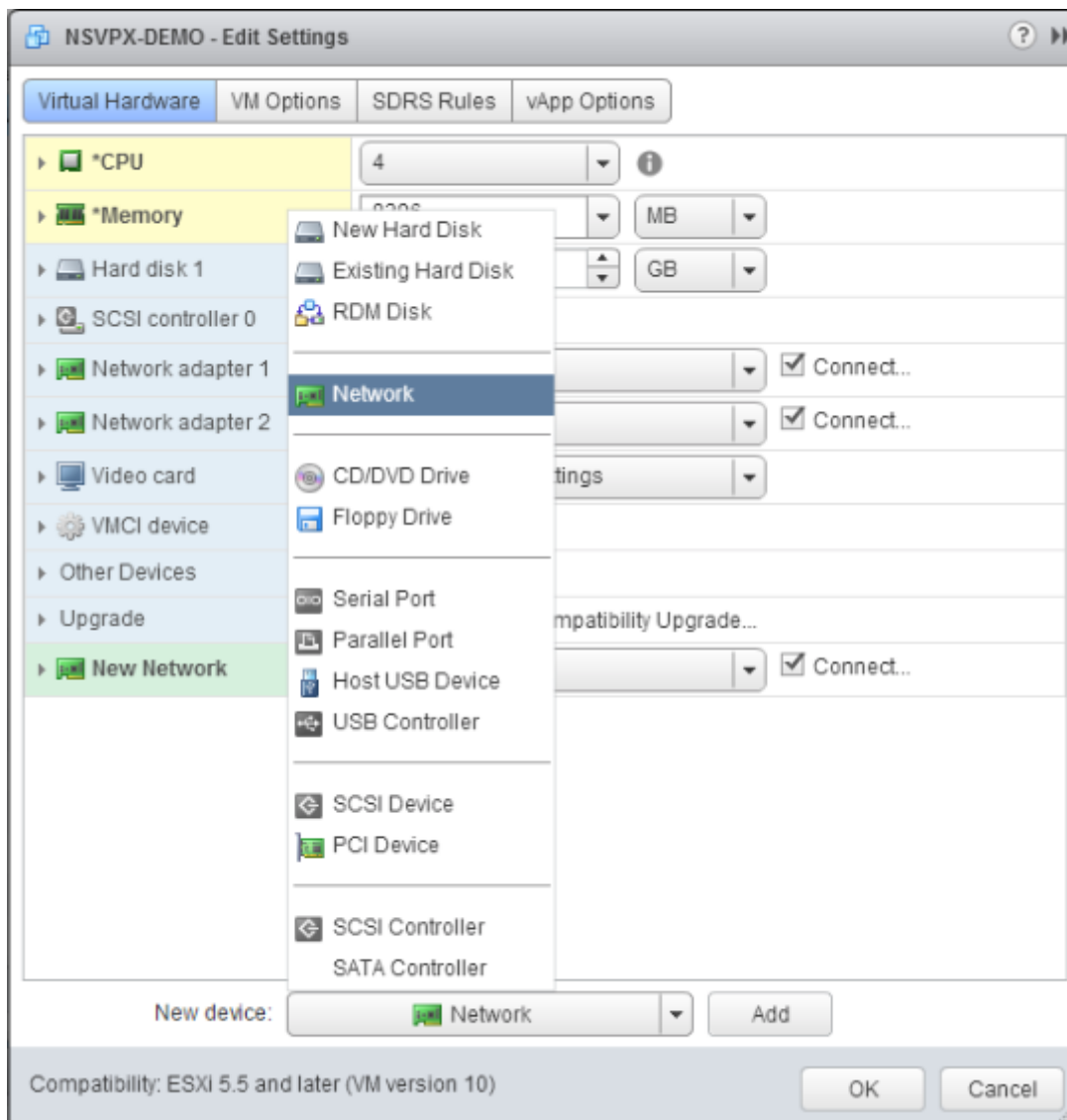
c. In the Limit drop-down list, select the number that is shown as the maximum value.



d. In the Shares drop-down lists, select Custom and the number that is shown as the maximum value.



7. Add a VMXNET3 network interface. From the New device drop-down list, select Network and click Add.

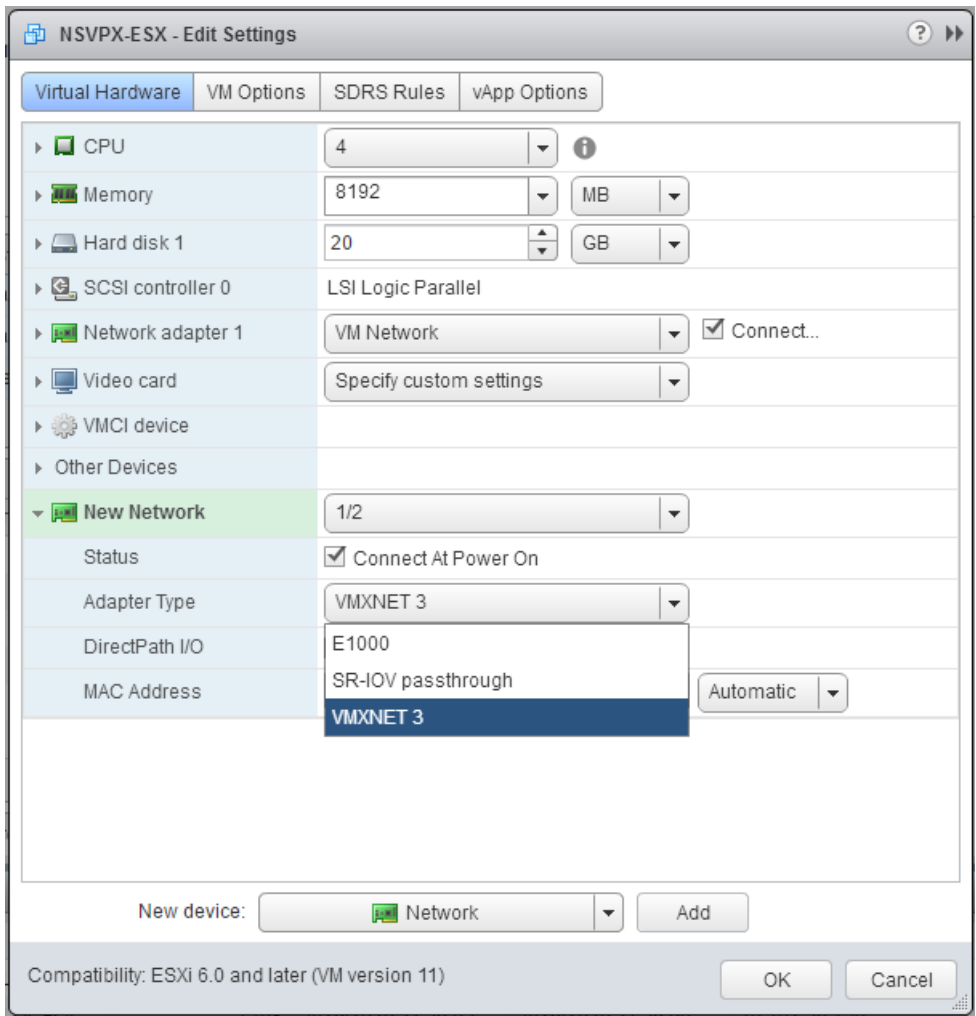


8. In the New Network section, from the drop-down list, select the network interface, and do the following:

a. In the Adapter Type drop-down list, select VMXNET3.

Important

The default E1000 network interface and VMXNET3 cannot coexist, make sure that you remove the E1000 network interface and use VMXNET3 (0/1) as the management interface.



9. Click OK.

10. Power on the Citrix ADC VPX instance.

11. Once the Citrix ADC VPX instance powers on, you can use the following command to verify the configuration:

```
1 > show interface summary
```

The output should show all the interfaces that you configured:

```
1 > show interface summary
2 -----
3      Interface  MTU      MAC                      Suffix
4 -----
5 1      0/1        1500     00:0c:29:89:1d:0e       NetScaler Vir...rface,
6      VMXNET3
7 2      1/1        9000     00:0c:29:89:1d:18       NetScaler Vir...rface,
8      VMXNET3
```

7	3	1/2 VMXNET3	9000	00:0c:29:89:1d:22	NetScaler Vir...rface,
8	4	LO/1 interface	9000	00:0c:29:89:1d:0e	Netscaler Loopback

Note

After you add a VMXNET3 interface and restart the Citrix ADC VPX appliance, the VMWare ESX hypervisor might change the order in which the NIC is presented to the VPX appliance. So, network adapter 1 might not always remain 0/1, resulting in loss of management connectivity to the VPX appliance. To avoid this issue, change the virtual network of the network adapter accordingly.

This is a VMWare ESX hypervisor limitation.

Configure a Citrix ADC VPX instance to use SR-IOV network interface

November 13, 2024

After you have installed and configured the Citrix ADC VPX instance on VMware ESX, you can use the VMware vSphere web client to configure the virtual appliance to use single root I/O virtualization (SR-IOV) network interfaces.

Limitations

A Citrix ADC VPX configured with SR-IOV network interface has the following limitations:

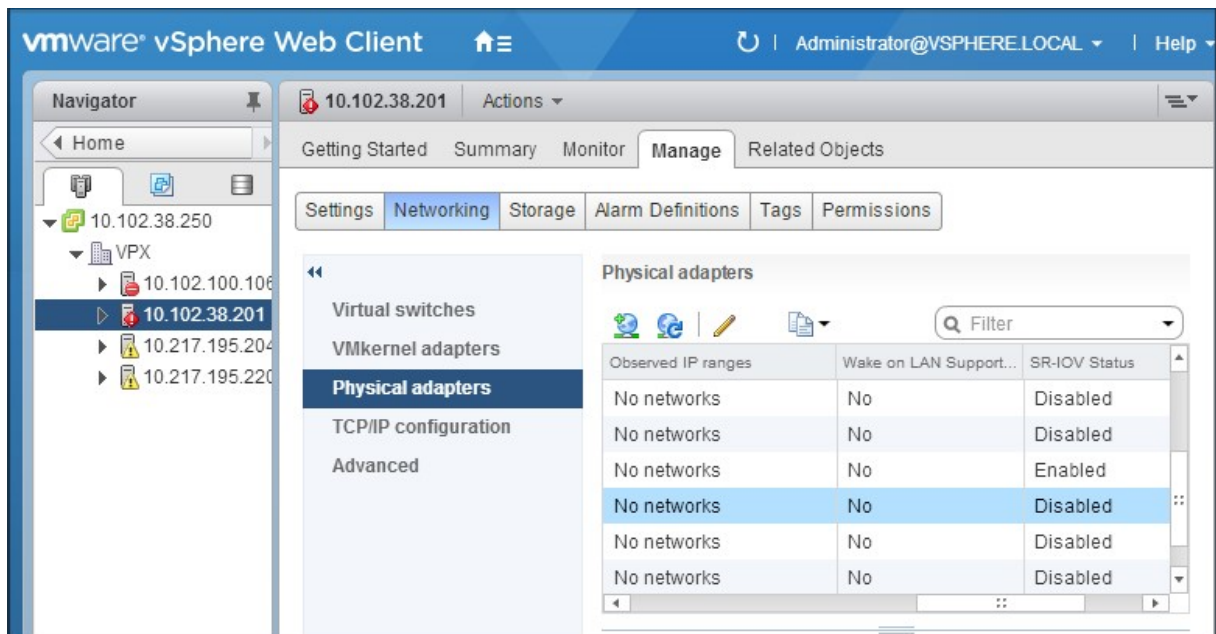
- The following features are not supported on SR-IOV interfaces using Intel 82599 10G NIC on ESX VPX:
 - L2 mode switching
 - Static Link Aggregation and LACP
 - Clustering
 - Admin partitioning [Shared VLAN mode]
 - High Availability [Active - Active mode]
 - Jumbo frames
 - IPv6
- The following features are not supported for on SR-IOV interface with an Intel 82599 10G NIC on KVM VPX:
 - Static Link Aggregation and LACP
 - L2 mode switching

- Clustering
- Admin partitioning [Shared VLAN mode]
- High Availability [Active –Active mode]
- Jumbo frames
- IPv6
- VLAN configuration on Hypervisor for SR-IOV VF interface through “ip link” command is not supported

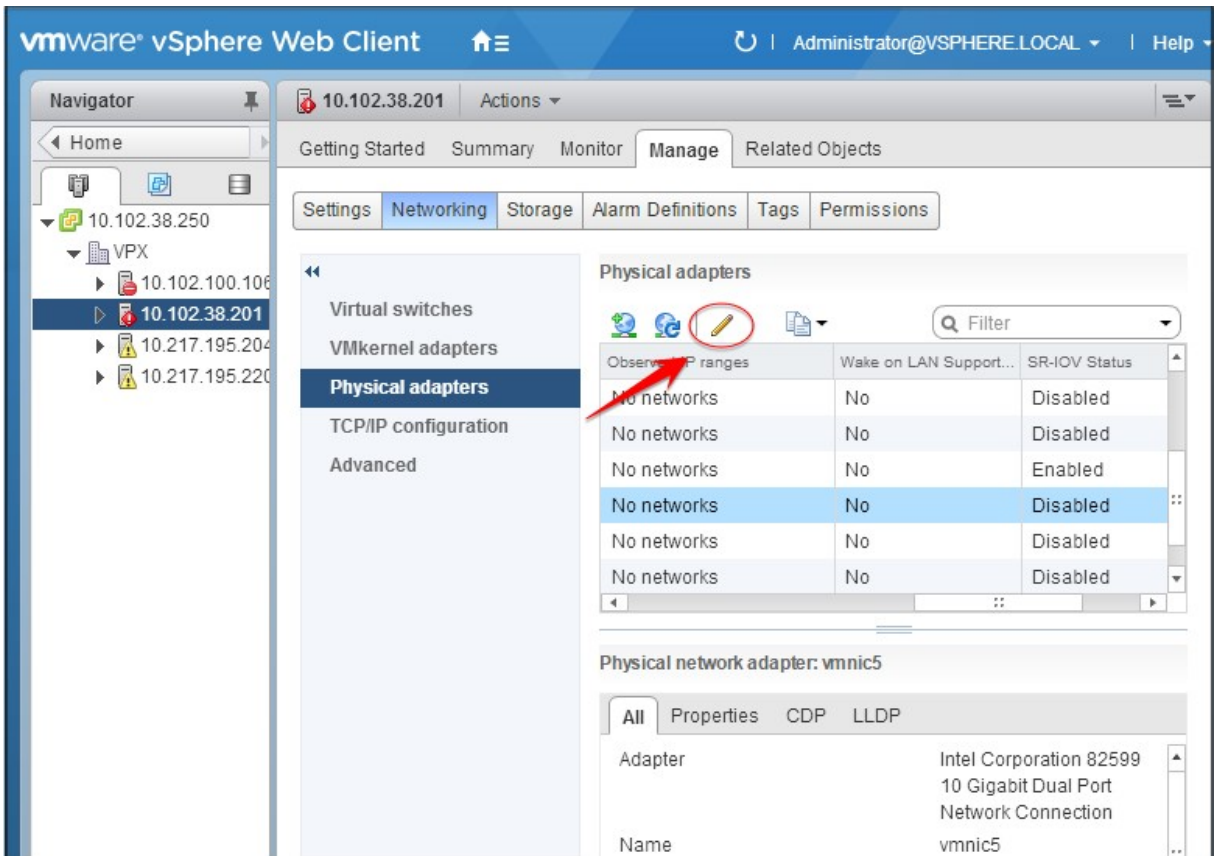
Prerequisite

Make sure that you:

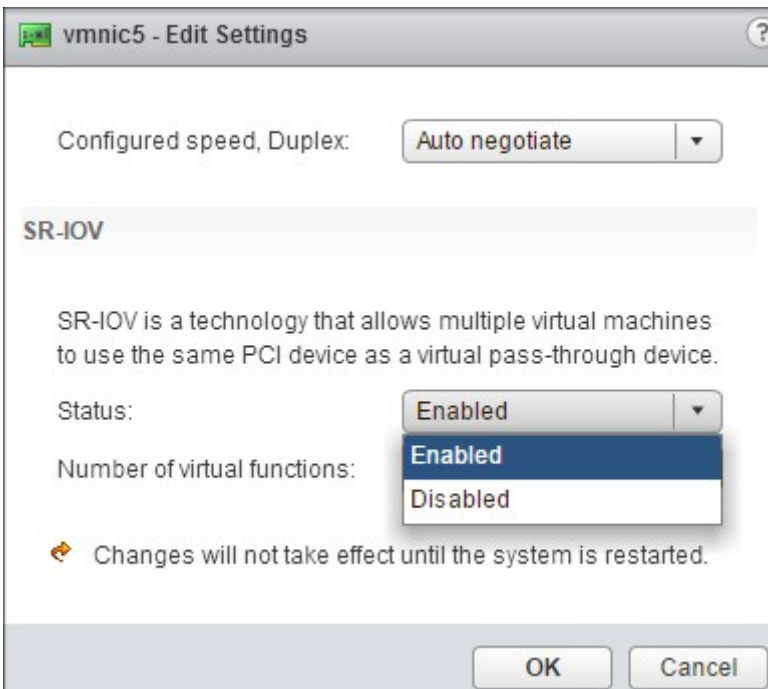
- Add the Intel 82599 Network Interface Card (NIC) to the ESX Host. IXGBE driver version 3.7.13.7.14io is recommended.
- Enable SR-IOV on the host physical adapter, as follows:
 1. In the vSphere Web Client, navigate to the Host.
 2. On the **Manage > Networking** tab, select **Physical adapters**. The SR-IOV Status field shows whether a physical adapter supports SR-IOV.



3. Select the physical adapter, and then click the pencil icon to open the **Edit Settings** dialog box.

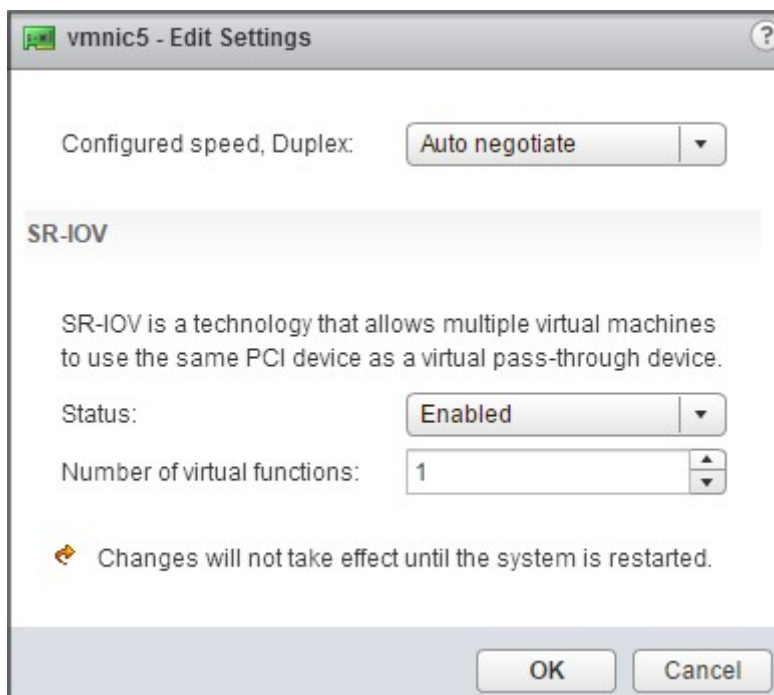


4. Under SR-IOV, select **Enabled** from the **Status** drop-down list.



5. In the **Number of virtual functions** field, enter the number of virtual functions that you want to

configure for the adapter.



6. Click **OK**.

7. Restart the host.

- Create a Distributed Virtual Switch (DVS) and Portgroups. For instructions, see the VMware Documentation.

Note:

Citrix has qualified the SR-IOV configuration on DVS and Portgroups only.

To configure Citrix ADC VPX instances to use SR-IOV network interface by using VMware vSphere Web Client:

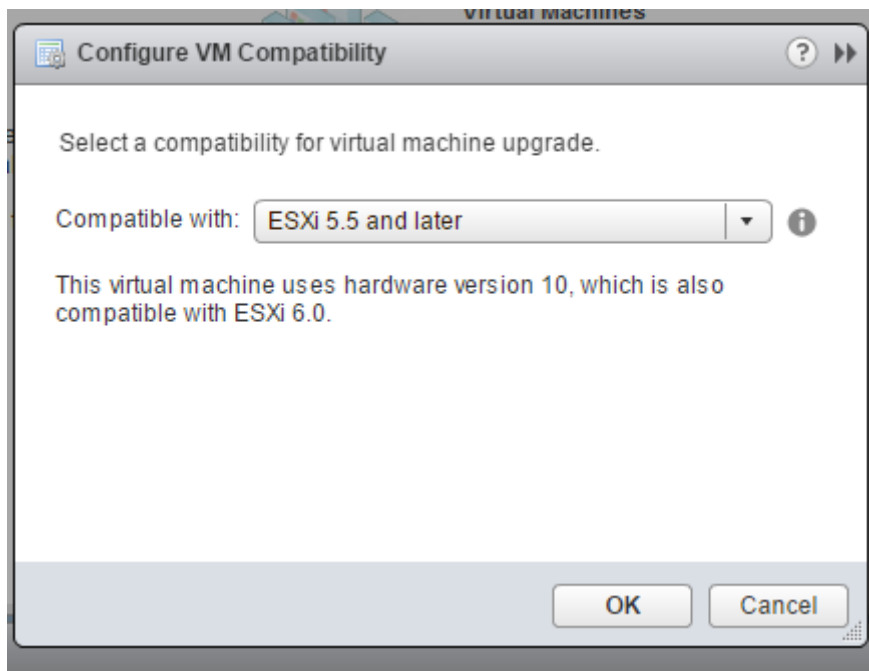
1. In the vSphere Web Client, select **Hosts and Clusters**.

1. Upgrade the Compatibility setting of the Citrix ADC VPX instance to ESX 5.5 or later, as follows:

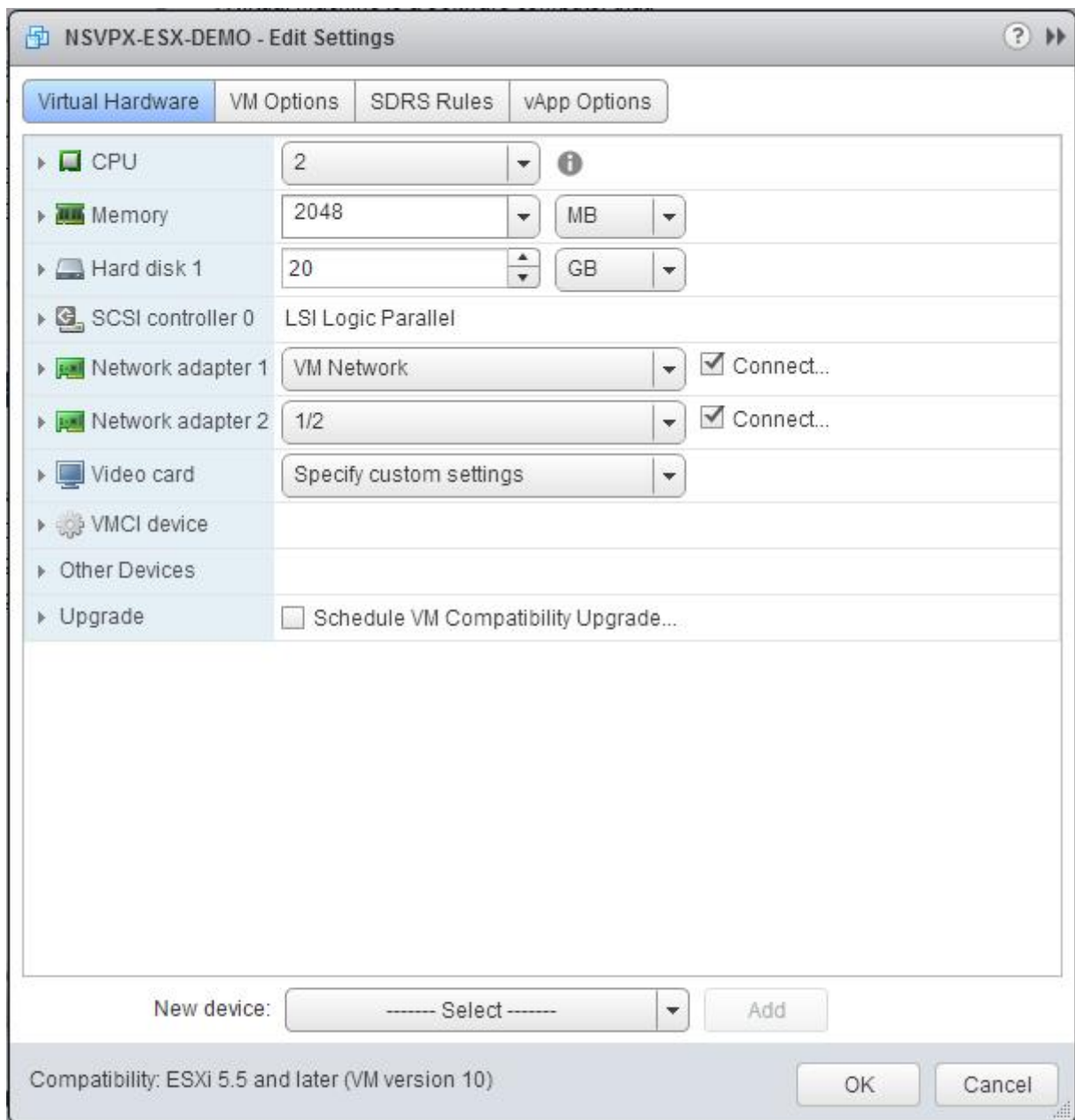
1. Power off the Citrix ADC VPX instance.

1. Right-click the Citrix ADC VPX instance and select **Compatibility > Upgrade VM Compatibility**.

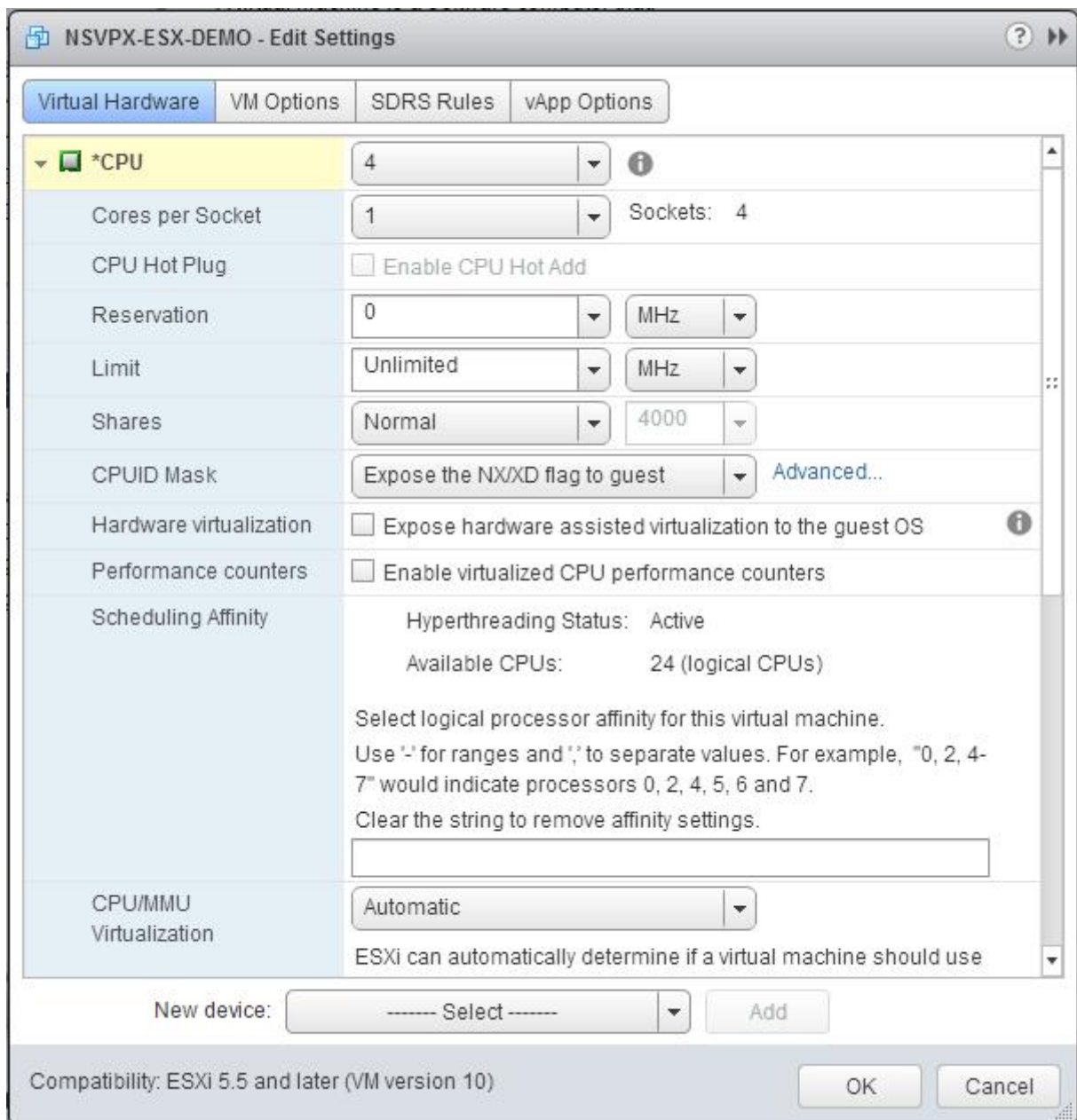
1. In the **Configure VM Compatibility** dialog box, select **ESXi 5.5 and later** from the **Compatible with** drop-down list and click **OK**.



1. Right-click on the Citrix ADC VPX instance and click **Edit Settings**.



1. In the **<virtual_appliance> - Edit Settings** dialog box, click the **CPU** section.



1. In the **CPU** section, update the following settings:

- Number of CPUs
- Number of Sockets
- Reservations
- Limit
- Shares

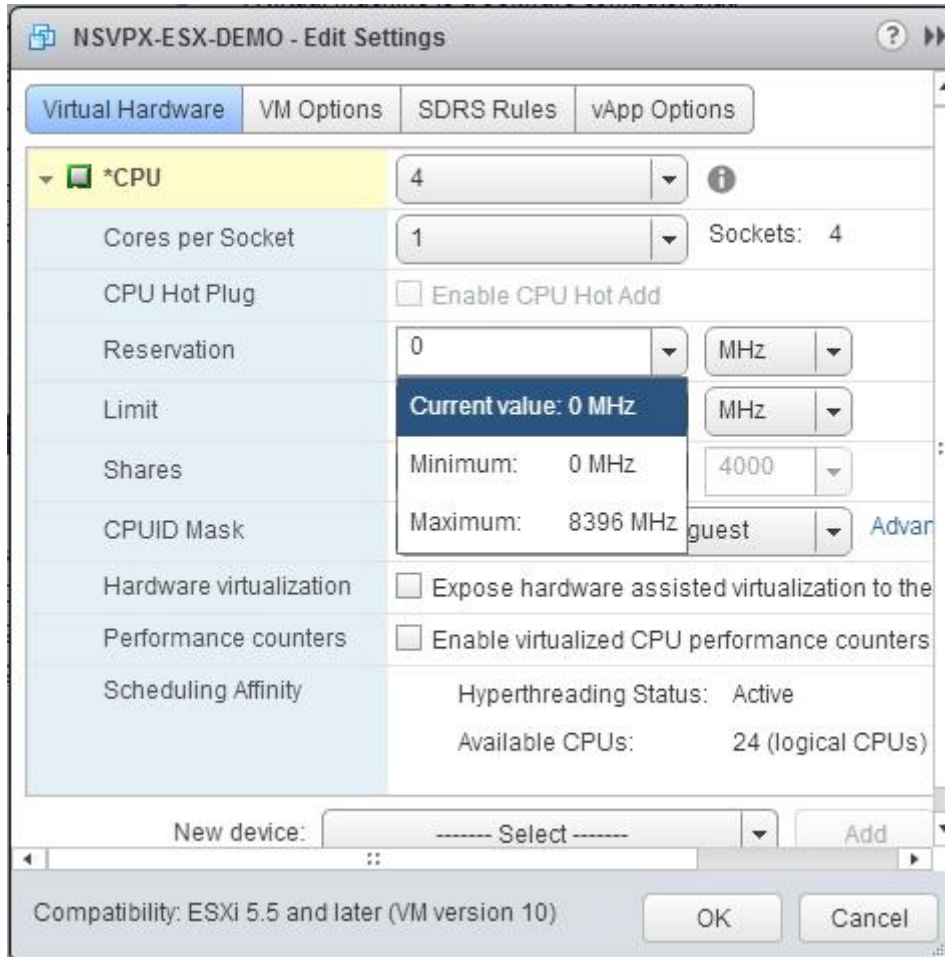
Set the values as follows:

1. In the **CPU** drop-down list, select the number of CPUs to assign to the virtual appliance.

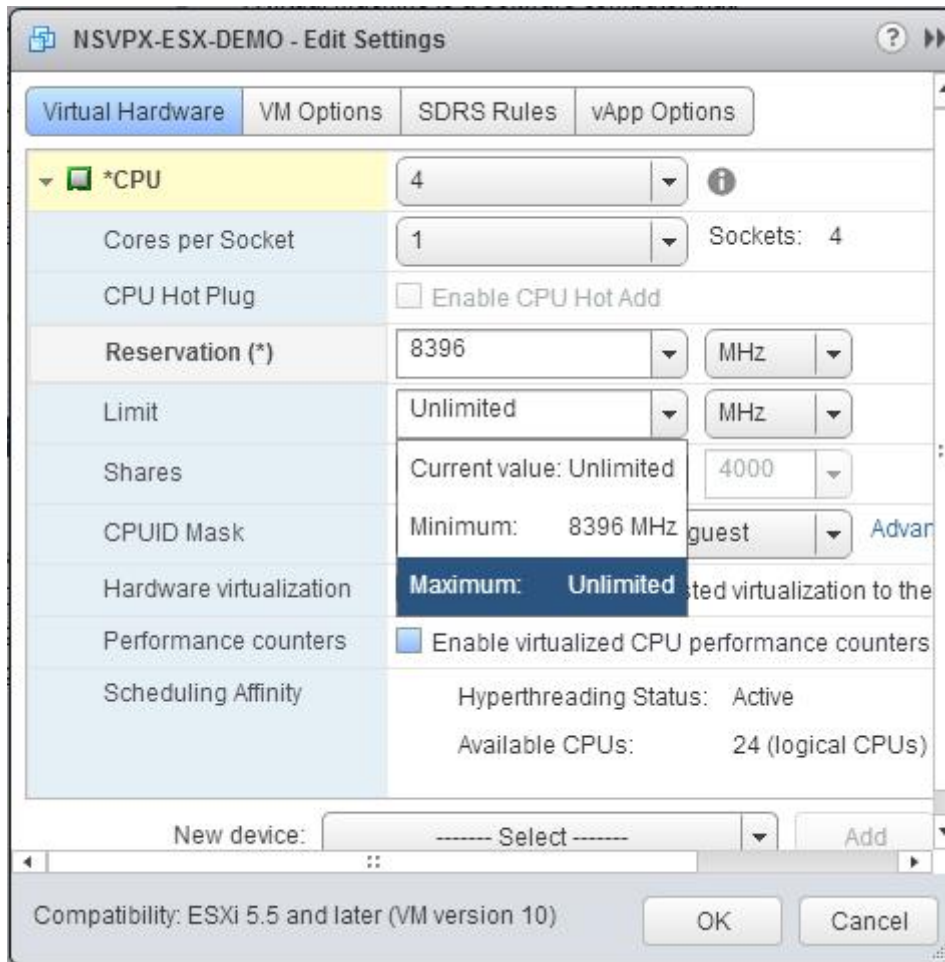
1. In the **Cores per Socket** drop-down list, select the number of sockets.
1. (Optional) In the **CPU Hot Plug** field, select or clear the **Enable CPU Hot Add** check box.

Note: Citrix recommends accepting the default (disabled).

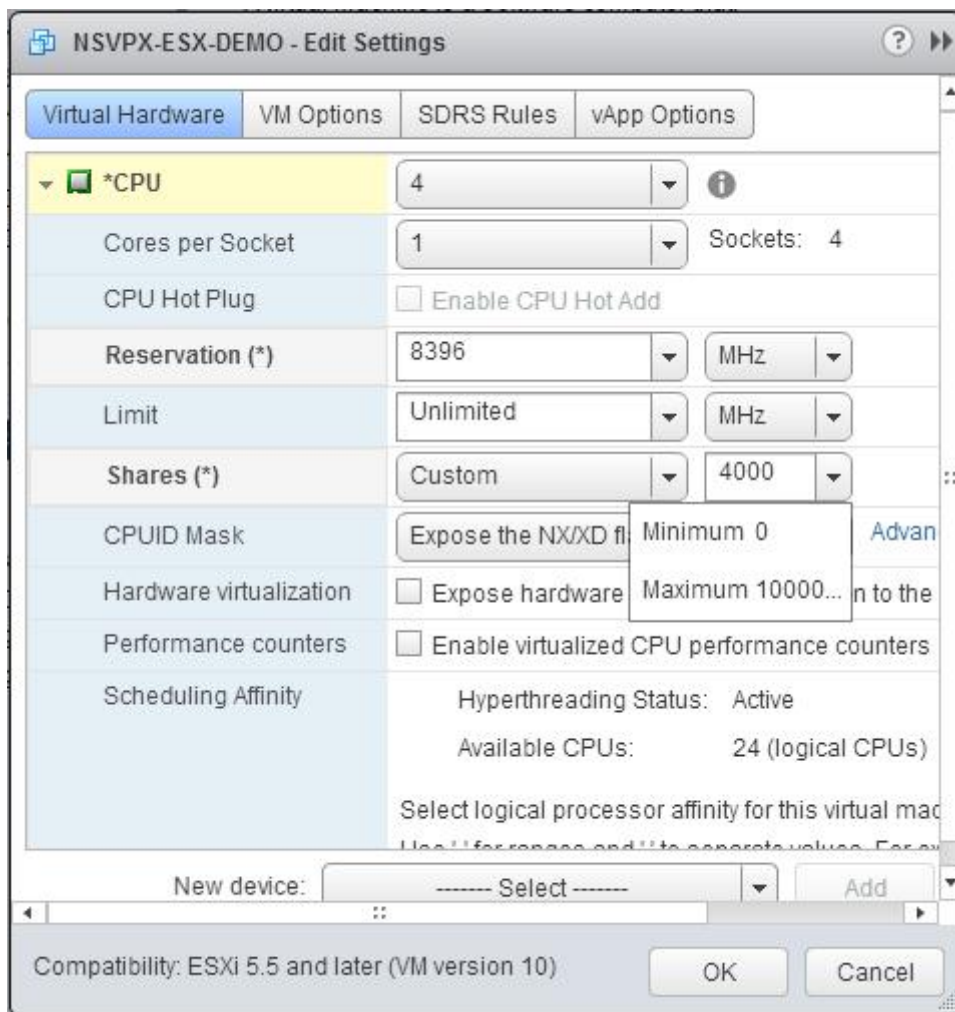
1. In the **Reservation** drop-down list, select the number that is shown as the maximum value.



1. In the **Limit** drop-down list, select the number that is shown as the maximum value.



1. In the **Shares** drop-down lists, select **Custom** and the number that is shown as the maximum value.



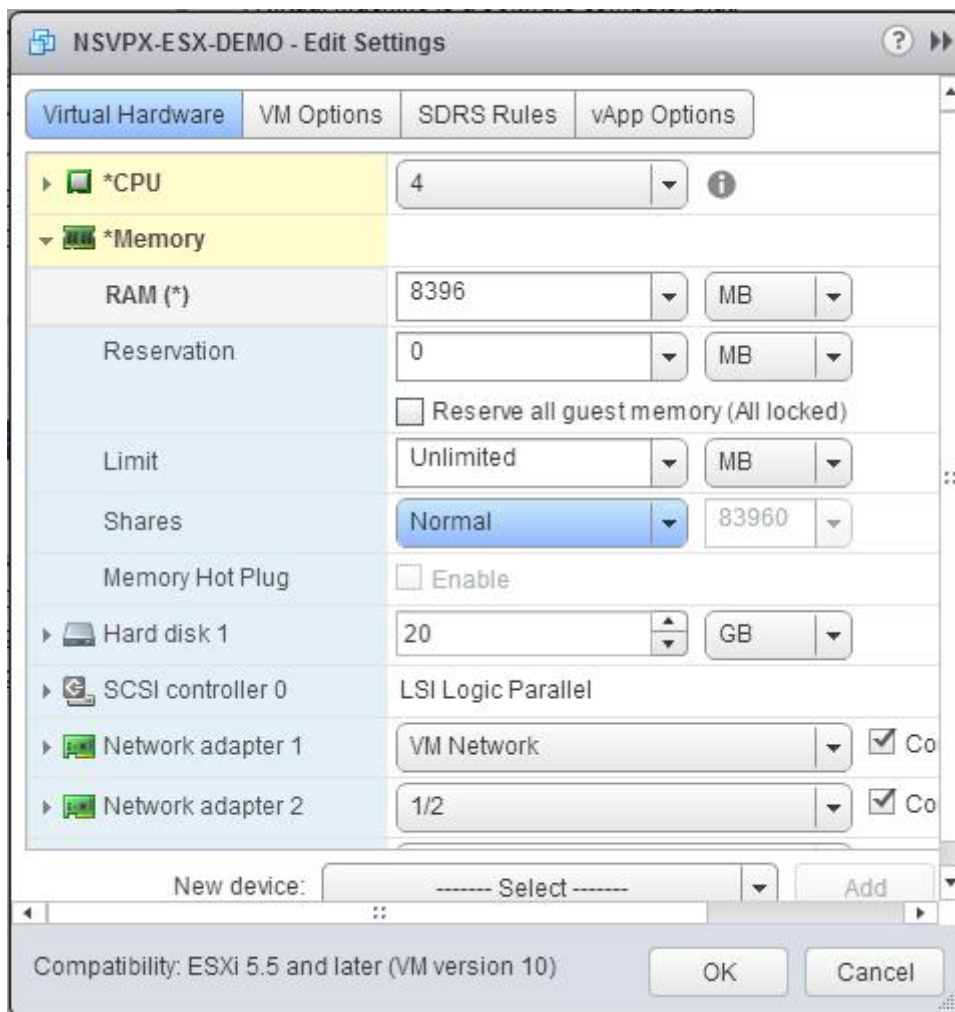
1. In the **Memory** section, update the following settings:

- Size of RAM
- Reservations
- Limit
- Shares

Set the values as follows:

1. In the **RAM** drop-down list, select the size of the RAM. It should be number of vCPUs x 2 GB. For example, if the number of vCPU is 4 then RAM = 4 x 2 GB = 8 GB.

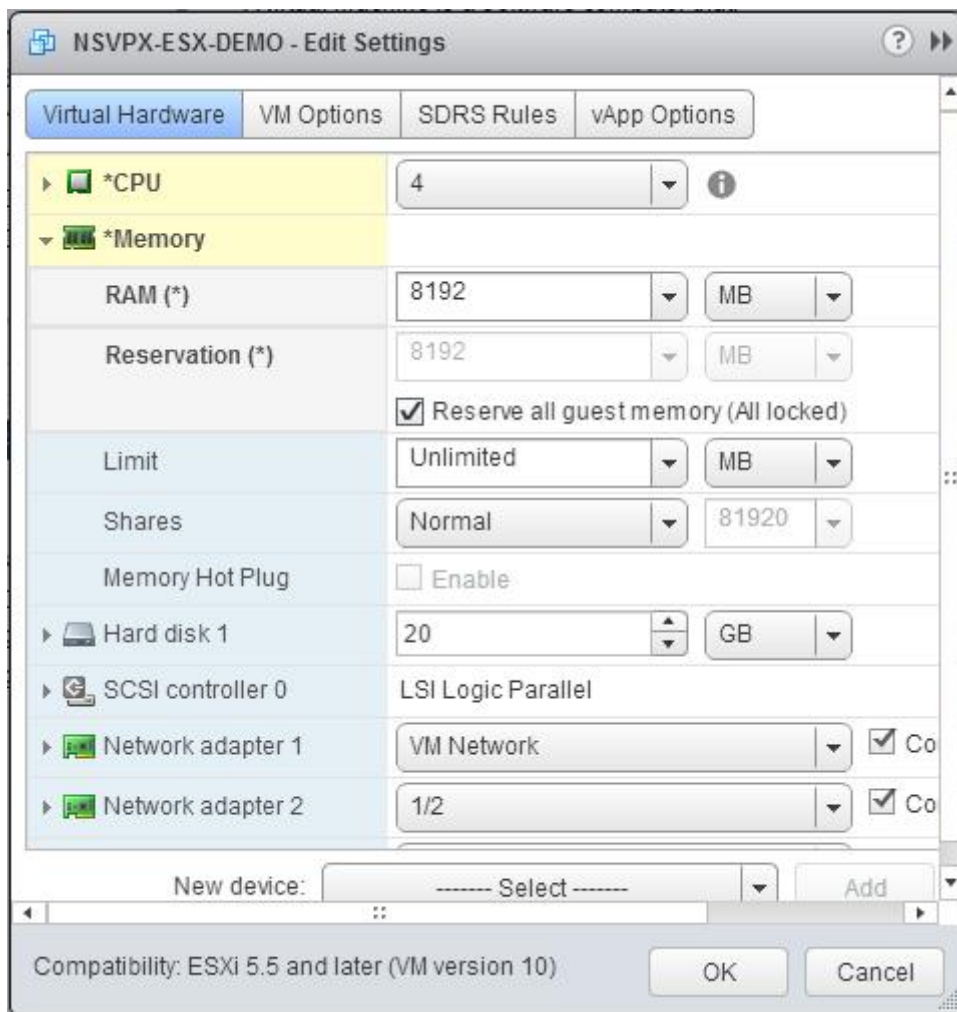
Note: For Advanced or Premium edition of the Citrix ADC VPX appliance, make sure that you allocate 4 GB of RAM to each vCPU. For example, if the number of vCPU is 4 then RAM = 4 x 4 GB = 16 GB.



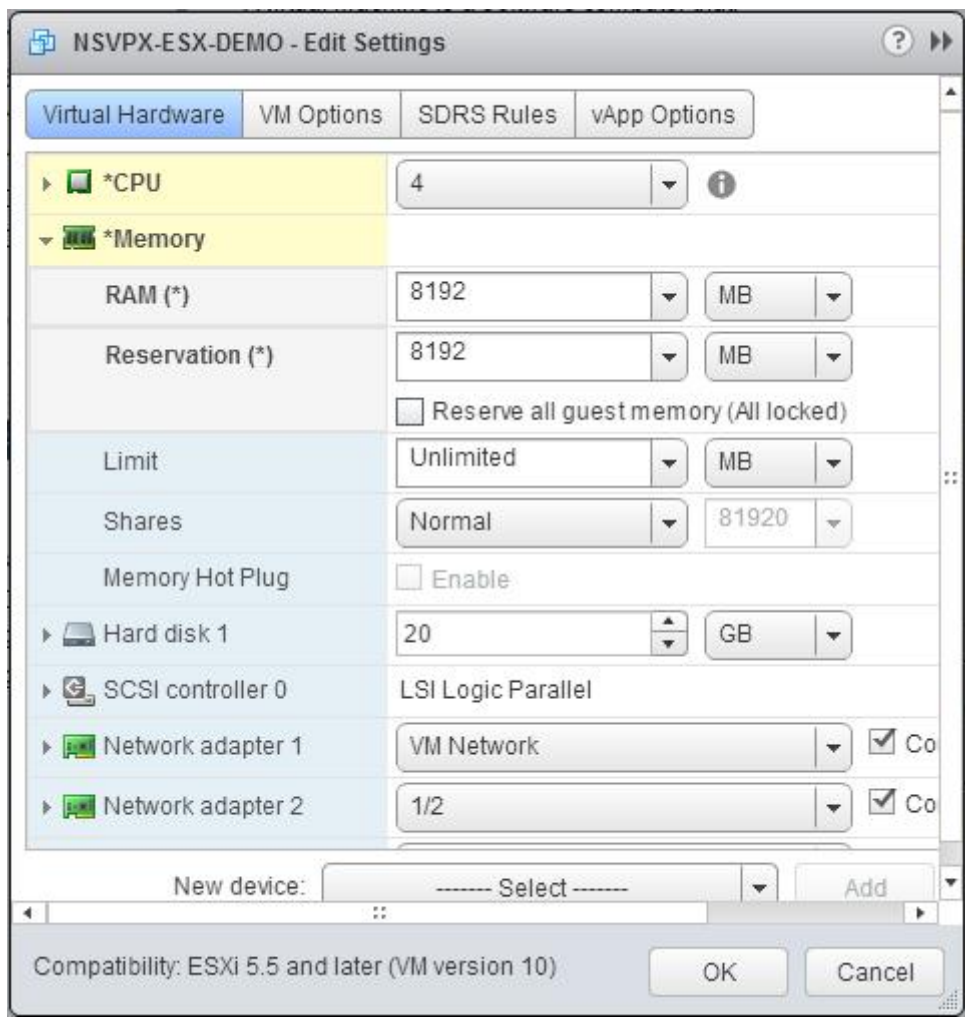
1. In the **Reservation** drop-down list, enter the value for the memory reservation, and select the **Reserve all guest memory (All locked)** check box. The memory reservation should be number of vCPUs x 2 GB. For example, if the number of vCPUs is 4, the memory reservation should be 4 x 2 GB = 8 GB.

Note:

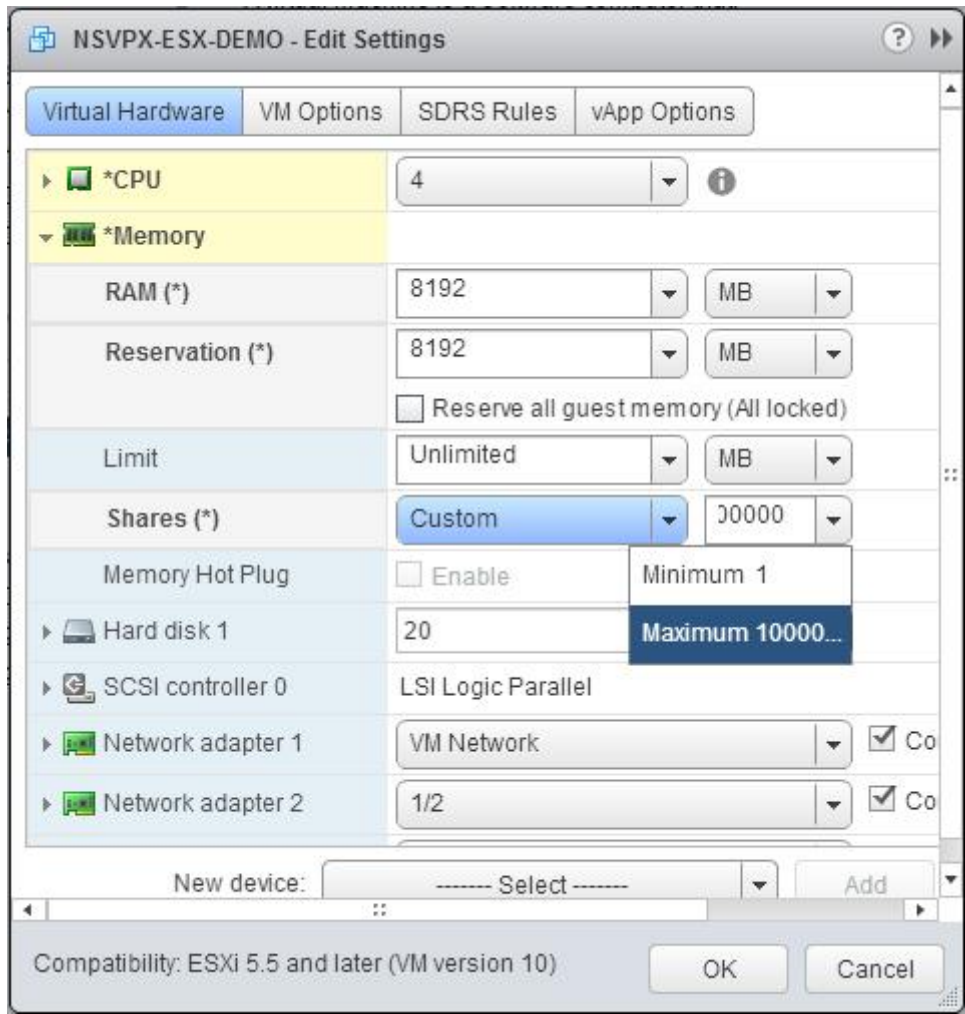
For Advanced or Premium edition of the Citrix ADC VPX appliance, make sure that you allocate 4 GB of RAM to each vCPU. For example, if the number of vCPU is 4 then RAM = 4 x 4 GB = 16 GB.



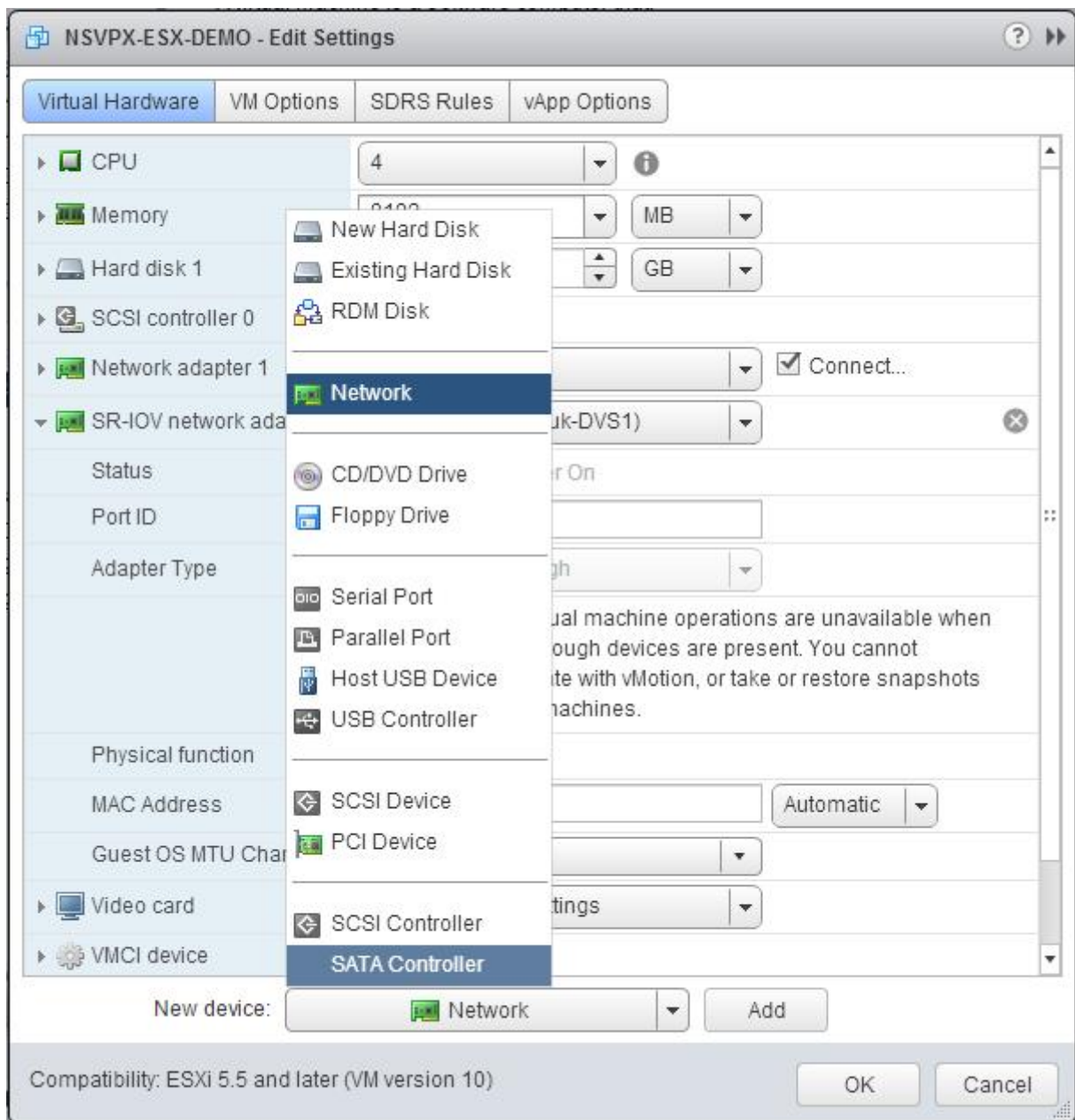
1. In the **Limit** drop-down list, select the number that is shown as the maximum value.



1. In the **Shares** drop-down lists, select **Custom**, and select the number that is shown as the maximum value.

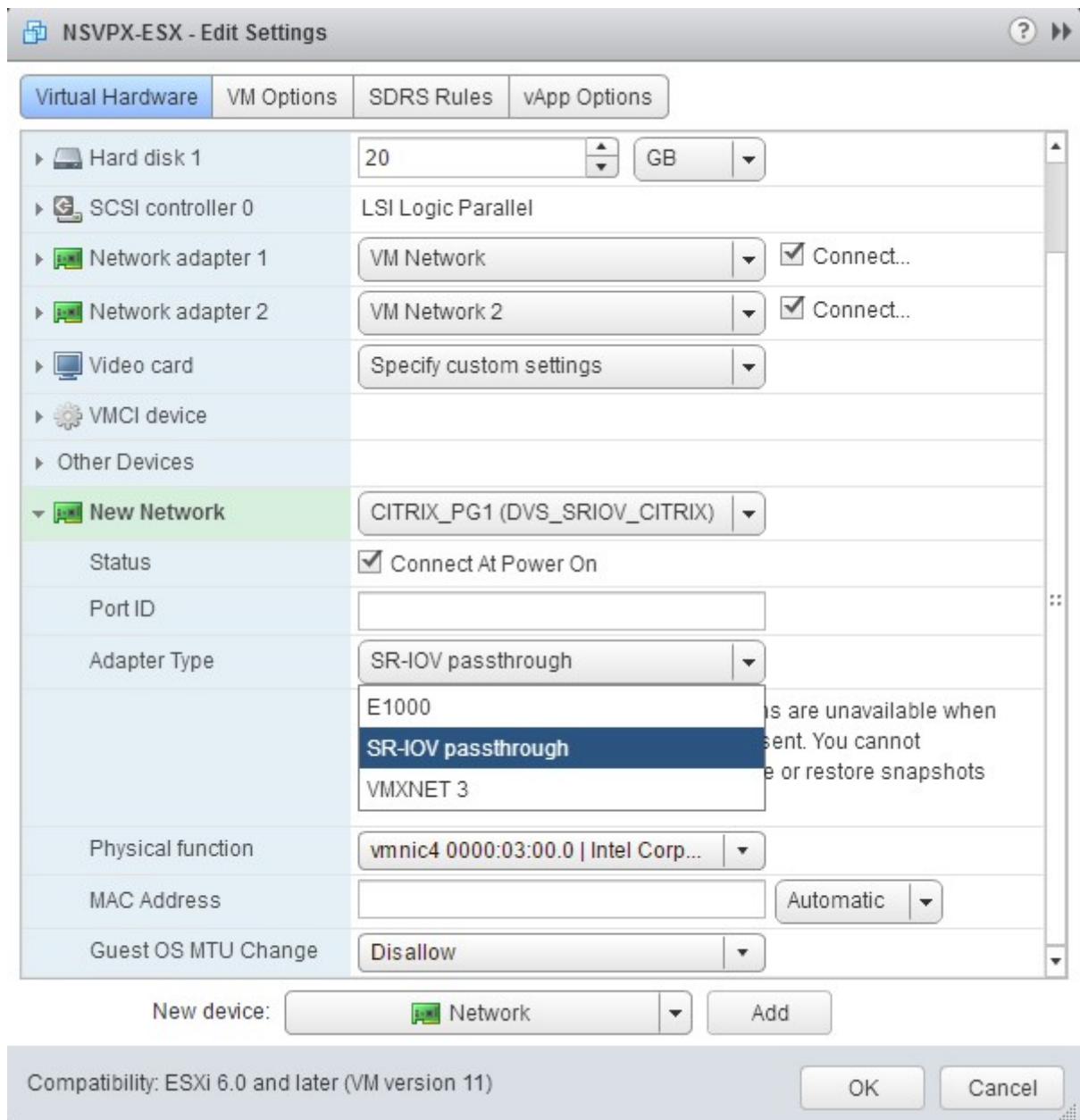


1. Add an SR-IOV network interface. From the **New device** drop-down list, select **Network** and click **Add**.

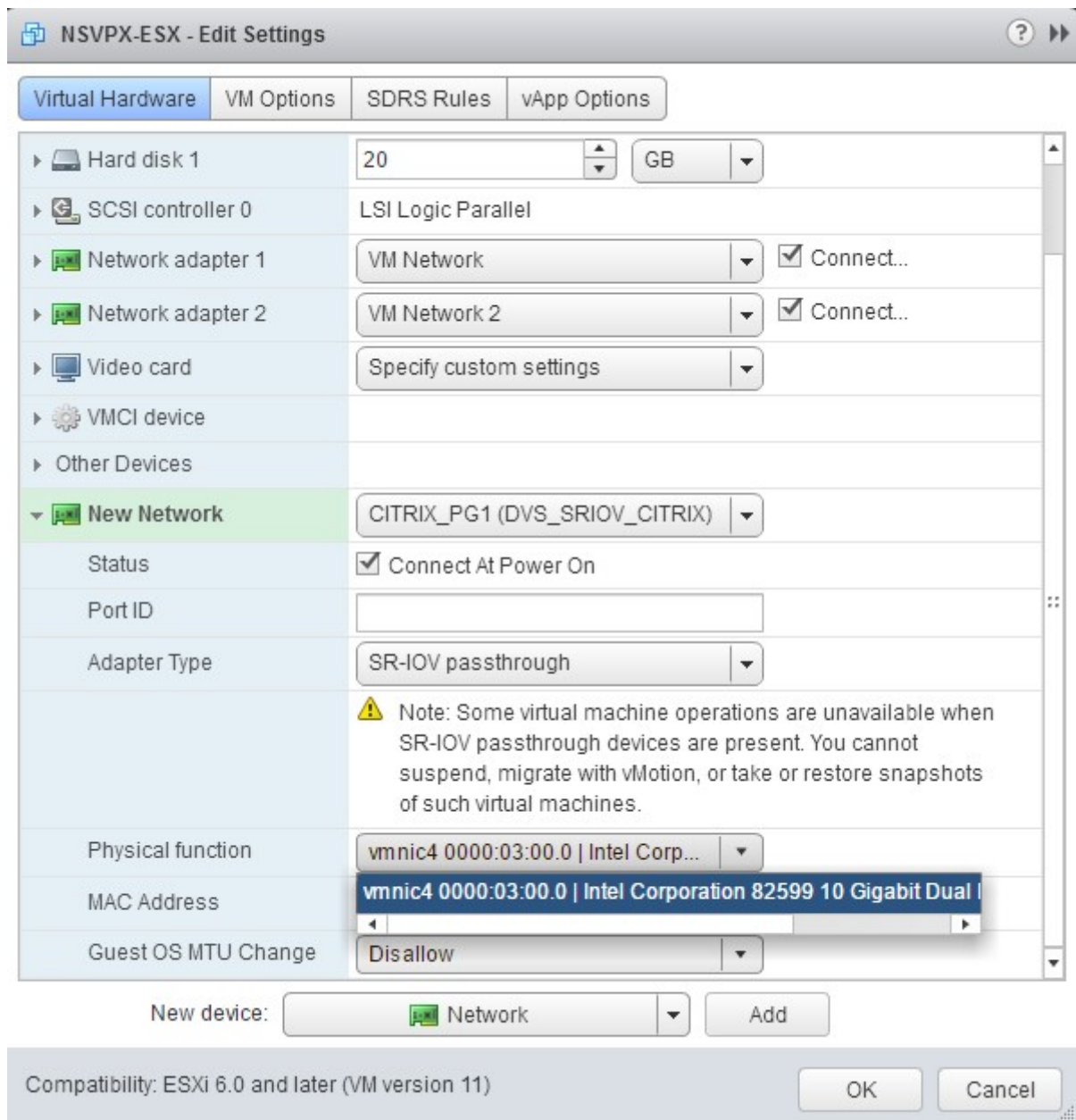


1. In the **New Network** section. From the drop-down list, select the Portgroup that you created, and do the following:

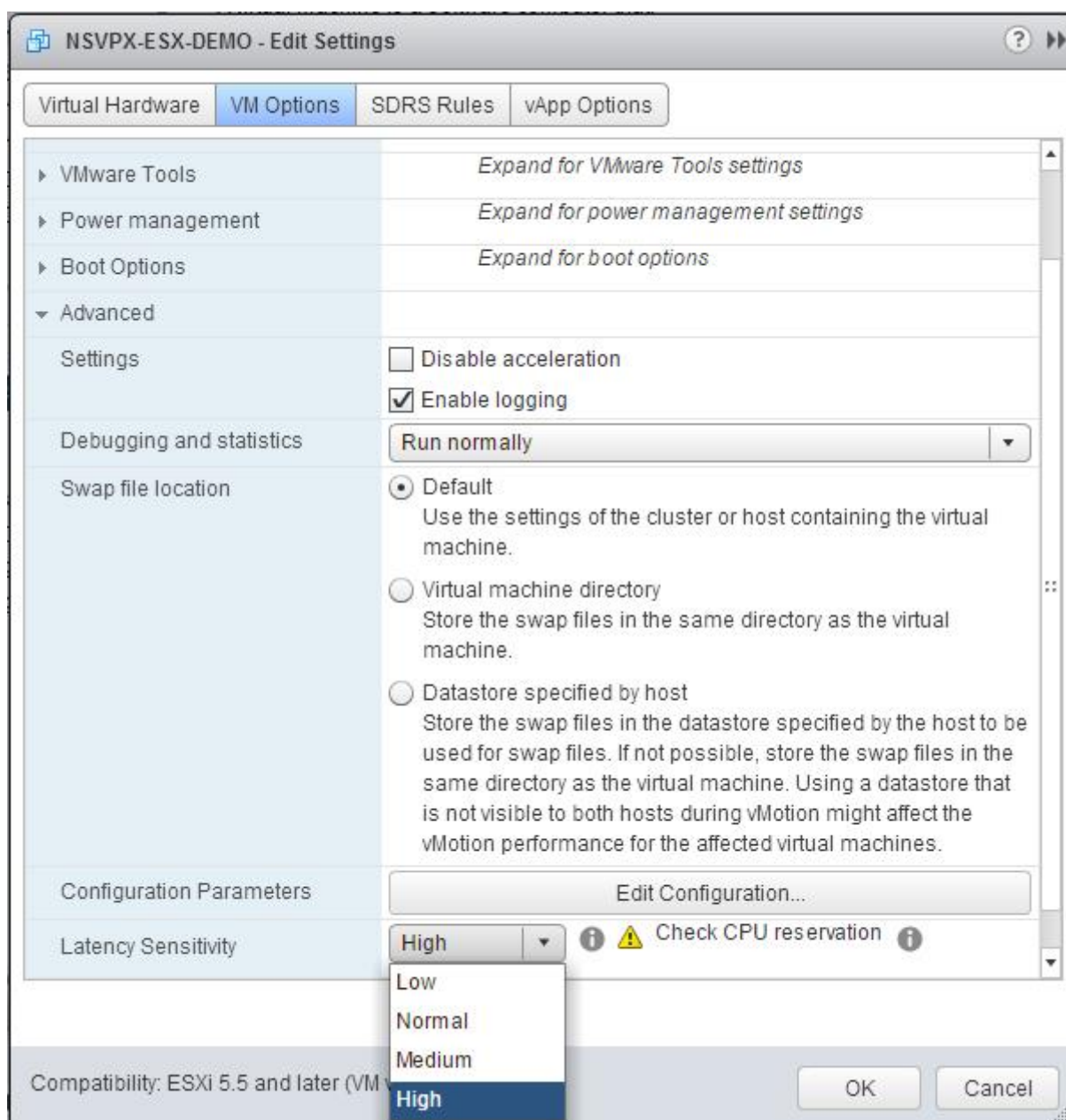
1. In the **Adapter Type** drop-down list, select **SR-IOV passthrough**.



1. In the **Physical function** drop-down list, select the physical adapter mapped with the Port-group.



1. In the **Guest OS MTU Change** drop-down list, select **Disallow**.
1. In the **<virtual_appliance> - Edit Settings** dialog box, click the **VM Options** tab.
1. On the **VM Options** tab, select the **Advanced** section. From the **Latency Sensitivity** drop-down list, select **High**.



1. Click **OK**.

1. Power on the Citrix ADC VPX instance.

1. Once the Citrix ADC VPX instance powers on, you can use the following command to verify the configuration:

```
1 > show interface summary
```

The output should show all the interfaces that you configured:

```
1 > show interface summary
2 -----
```

```

3      Interface  MTU      MAC      Suffix
4  -----
5  1      0/1      1500     00:0c:29:1b:81:0b  NetScaler Virtual
6      Interface
7  2      10/1     1500     00:50:56:9f:0c:6f  Intel 82599 10G VF
8      Interface
9  3      10/2     1500     00:50:56:9f:5c:1e  Intel 82599 10G VF
10     Interface
11  4      10/3     1500     00:50:56:9f:02:1b  Intel 82599 10G VF
12     Interface
13  5      10/4     1500     00:50:56:9f:5a:1d  Intel 82599 10G VF
14     Interface
15  6      10/5     1500     00:50:56:9f:4e:0b  Intel 82599 10G VF
16     Interface
17  7      LO/1     1500     00:0c:29:1b:81:0b  Netscaler Loopback
18     interface
19  Done
20 > show inter 10/1
21  1)      Interface 10/1 (Intel 82599 10G VFInterface) #1
22      flags=0xe460 <ENABLED, UP, UP, HAMON, 802.1q>
23      MTU=1500, native vlan=55, MAC=00:50:56:9f:0c:6f, uptime 0
24      h21m53s
25      Actual: media FIBER, speed 10000, duplex FULL, fctl NONE,
26      throughput 10000
27      LLDP Mode: NONE,          LR Priority: 1024
28
29      RX: Pkts(838020742) Bytes(860888485431) Errs(0) Drops(2527)
30      Stalls(0)
31      TX: Pkts(838149954) Bytes(860895860507) Errs(0) Drops(0) Stalls
32      (0)
33      NIC: InDisc(0) OutDisc(0) Fctls(0) Stalls(0) Hangs(0) Muted(0)
34      Bandwidth thresholds are not set.
35  Done

```

Migrating the Citrix ADC VPX from E1000 to SR-IOV or VMXNET3 Network Interfaces

November 12, 2024

May 24, 2018

You can configure your existing Citrix ADC VPX instances that use E1000 network interfaces to use SR-IOV or VMXNET3 network interfaces.

To configure an existing Citrix ADC VPX instance to use SR-IOV network interfaces, see [Configure a Citrix ADC VPX instance to use SR-IOV network interface](#).

To configure an existing Citrix ADC VPX instance to use VMXNET3 network interfaces, see [Configure a Citrix ADC VPX instance to use VMXNET3 network interface](#).

Configure a Citrix ADC VPX instance to use PCI passthrough network interface

November 13, 2024

Overview

After you have installed and configured a Citrix ADC VPX instance on VMware ESX Server, you can use the vSphere Web Client to configure the virtual appliance to use PCI passthrough network interfaces.

The PCI passthrough feature allows a guest virtual machine to directly access physical PCI and PCIe devices connected to a host.

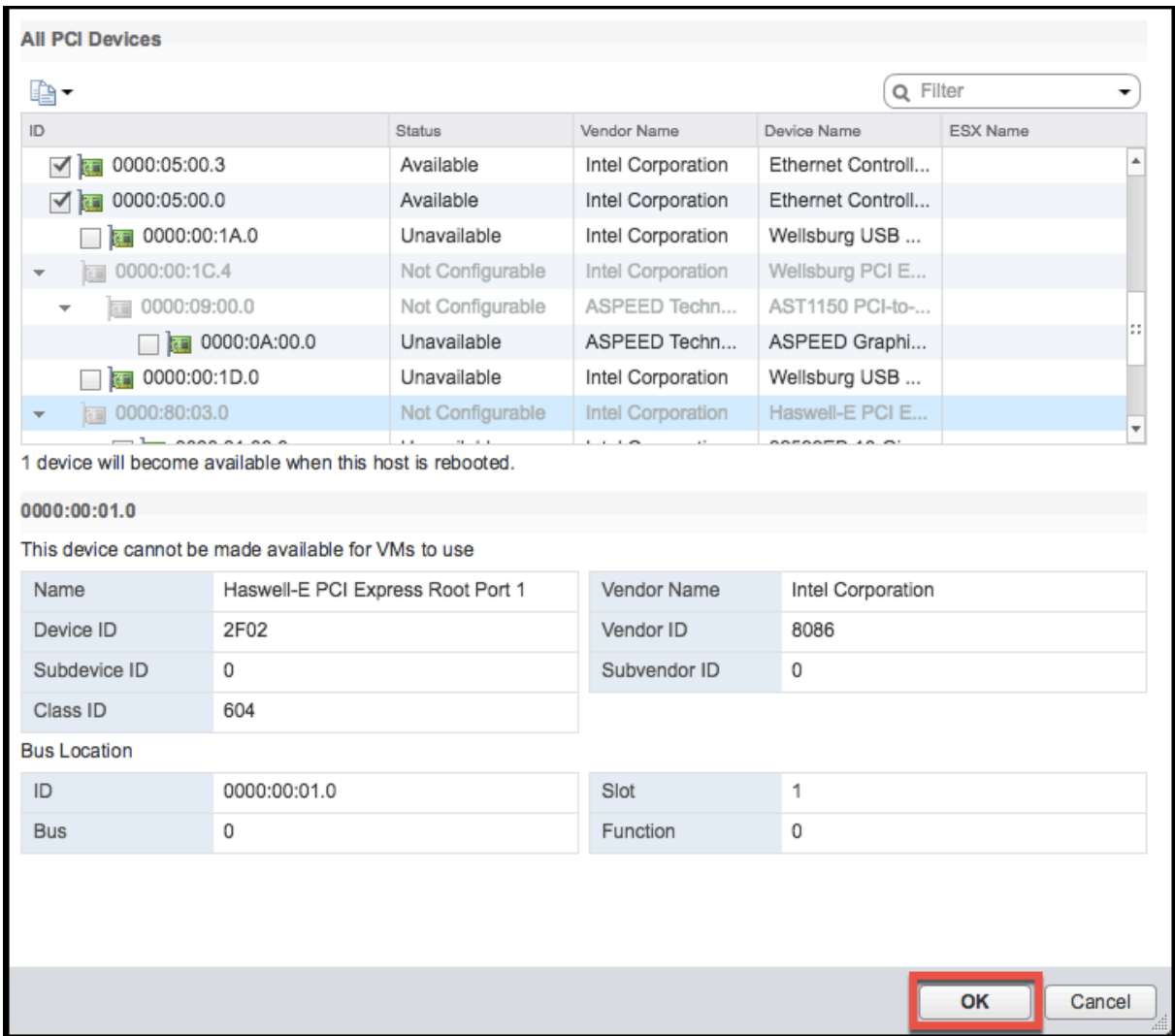
Prerequisites

- The firmware version of the Intel XL710 NIC on the host is 5.04.
- A PCI passthrough device connected to and configured on the host
- Supported NICs:
 - Intel X710 10G NIC
 - Intel XL710 Dual Port 40G NIC
 - Intel XL710 Single Port 40G NIC

Configure passthrough devices on a host

Before configuring a passthrough PCI device on a virtual machine, you should configure it on the host machine. Follow these steps to configure passthrough devices on a host.

1. Select the host from the Navigator panel of the vSphere Web Client.
2. Click **Manage > Settings > PCI Devices**. All available passthrough devices are displayed.
3. Right-click the device that you want to configure and click **Edit**.
4. The **Edit PCI Device Availability** window appears.
5. Select the devices to be used for passthrough and click **OK**.

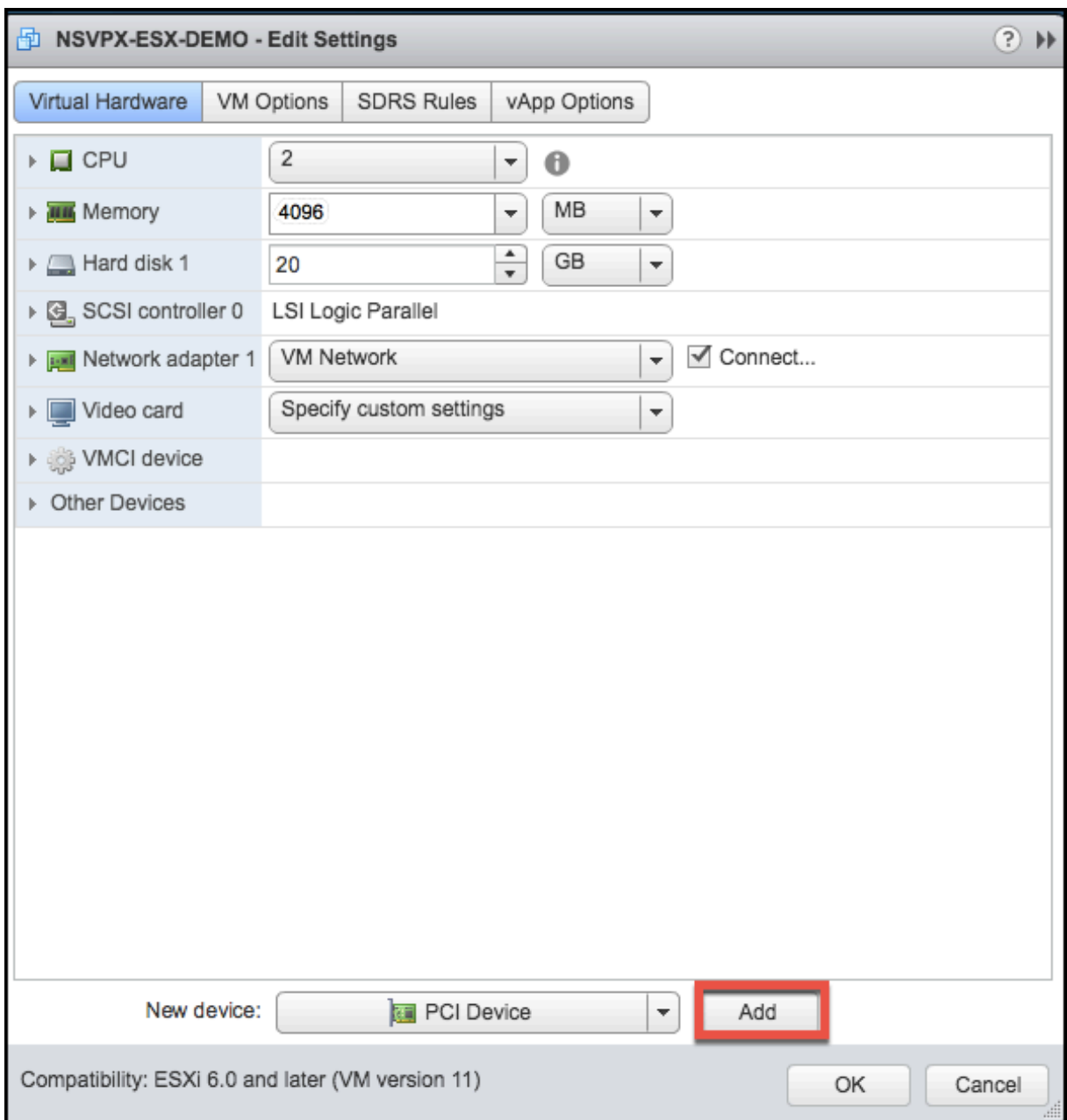


1. Restart the host machine.

Configure passthrough devices on a Citrix ADC VPX instance

Follow these steps to configure a passthrough PCI device on a Citrix ADC VPX instance.

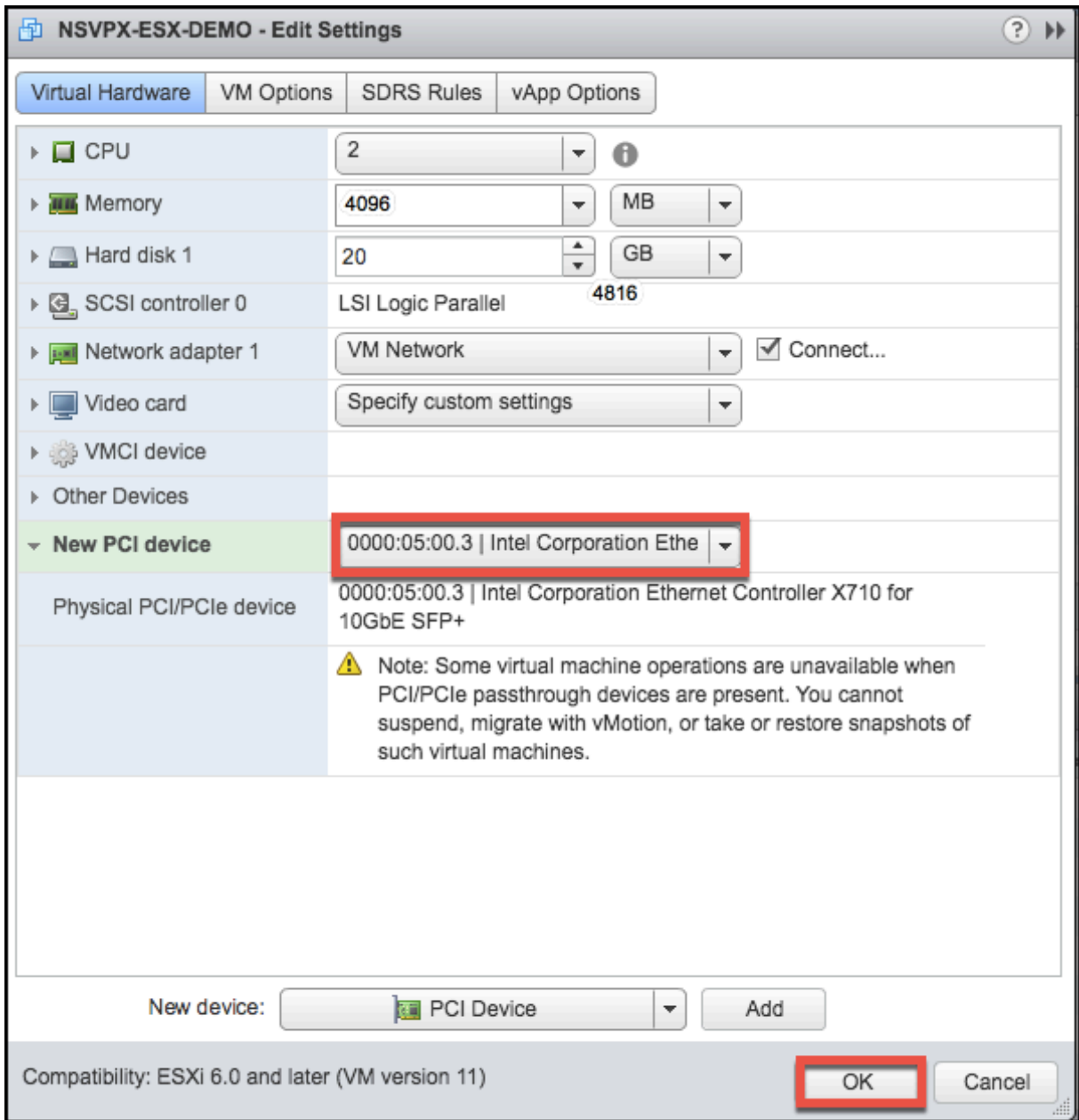
1. Power off the virtual machine.
2. Right-click the virtual machine and select **Edit Settings**.
3. On the **Virtual Hardware** tab, select **PCI Device** from the **New Device** drop-down menu, and click **Add**.



1. Expand **New PCI device** and select the passthrough device to connect to the virtual machine from the drop-down list and click **OK**.

Note:

VMXNET3 network interface and PCI Passthrough Network Interface cannot coexist.



6. Power on the guest virtual machine.

You have completed the steps to configuring Citrix ADC VPX to use PCI passthrough network interfaces.

Install a Citrix ADC VPX instance on Microsoft Hyper-V server

November 18, 2024

To install Citrix

Citrix ADC VPX instances on Microsoft Windows Server, you must first install Windows Server, with the Hyper-V role enabled, on a machine with adequate system resources. While installing the Hyper-V role, be sure to specify the network interface cards (NICs) on the server that Hyper-V will use to create the virtual networks. You can reserve some NICs for the host. Use Hyper-V Manager to perform the Citrix ADC VPX instance installation.

Citrix ADC VPX instance for Hyper-V is delivered in virtual hard disk (VHD) format. It includes the default configuration for elements such as CPU, network interfaces, and hard-disk size and format. After you install Citrix ADC VPX instance, you can configure the network adapters on virtual appliance, add virtual NICs, and then assign the Citrix ADC IP address, subnet mask, and gateway, and complete the basic configuration of the virtual appliance.

After the initial configuration of the VPX instance, if you want to upgrade the appliance to the latest software release, see [Upgrade a Citrix ADC VPX standalone appliance](#)

Note:

Intermediate System-to-Intermediate System (ISIS) protocol is not supported on the Citrix ADC VPX virtual appliance hosted on the HyperV-2012 platform.

Prerequisites for installing Citrix ADC VPX instance on Microsoft servers

Before you begin installing a virtual appliance, do the following:

- Enable the Hyper-V role on Windows Servers . For more information, see [http://technet.microsoft.com/en-us/library/ee344837\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/ee344837(WS.10).aspx).
- Download the virtual appliance setup files.
- Obtain Citrix ADC VPX instance license files. For more information about Citrix ADC VPX instance licenses, see the *Citrix ADC VPX Licensing Guide* at <http://support.citrix.com/article/ctx131110>.

Microsoft server hardware requirements

The following table describes the minimum system requirements for Microsoft Servers.

Table 1. Minimum system requirements for Microsoft servers

Component	Requirement
CPU	1.4 GHz 64-bit processor
RAM	3 GB
Disk Space	32 GB or greater

The following table lists the virtual computing resources for each Citrix ADC VPX instance.

Table 2. Minimum virtual computing resources required for running a Citrix ADC VPX instance

Component	Requirement
RAM	2 GB
Virtual CPU	2
Disk Space	20 GB
Virtual Network Interfaces	1

Download the Citrix ADC VPX setup files

The Citrix ADC VPX instance for Hyper-V is delivered in virtual hard disk (VHD) format. You can download the files from the Citrix website. You need a Citrix account to log in. If you do not have a Citrix account, access the home page at <http://www.citrix.com>, click **Sign In > My account > Create Citrix Account**, and follow the instructions to create a Citrix account.

To download the Citrix ADC VPX instance setup files, follow these steps:

1. In a web browser, go to <http://www.citrix.com/>.
2. Sign in with your user name and password.
3. Click **Downloads**.
4. In **Select a Product** drop-down menu, select **Citrix ADC (NetScaler ADC)**.
5. Under **Citrix ADC Release X.X > Virtual Appliances**, click **Citrix ADC VPX Release X.X**
6. Download the compressed file to your server.

Install the Citrix ADC VPX instance on Microsoft servers

After you have enabled the Hyper-V role on Microsoft Server and extracted the virtual appliance files, you can use Hyper-V Manager to install Citrix ADC VPX instance. After you import the virtual machine, you need to configure the virtual NICs by associating them to the virtual networks created by Hyper-V.

You can configure a maximum of eight virtual NICs. Even if the physical NIC is DOWN, the virtual appliance assumes that the virtual NIC is UP, because it can still communicate with the other virtual appliances on the same host (server).

Note:

You cannot change any settings while the virtual appliance is running. Shut down the virtual appliance and then make changes.

To install Citrix ADC VPX instance on Microsoft Server by using Hyper-V Manager:

1. To start Hyper-V Manager, click **Start**, point to **Administrative Tools**, and then click **Hyper-V Manager**.
2. In the navigation pane, under **Hyper-V Manager**, select the server on which you want to install Citrix ADC VPX instance.
3. On the **Action** menu, click **Import Virtual Machine**.
4. In the **Import Virtual Machine** dialog box, in **Location**, specify the path of the folder that contains the Citrix ADC VPX instance software files, and then select **Copy the virtual machine (create a new unique ID)**. This folder is the parent folder that contains the Snapshots, Virtual Hard Disks, and Virtual Machines folders.

Note:

If you received a compressed file, make sure that you extract the files into a folder before you specify the path to the folder.

5. Click **Import**.
6. Verify that the virtual appliance that you imported is listed under **Virtual Machines**.
7. To install another virtual appliance, repeat steps **2** through **6**.

Important:

Make sure that you extract the files to a different folder in step **4**.

Auto-provision a Citrix ADC VPX instance on Hyper-V

Auto-provisioning of Citrix ADC VPX instance is optional. If auto-provisioning is not done, the virtual appliance provides an option to configure the IP address and so on.

To auto-provision Citrix ADC VPX instance on Hyper-V, follow these steps.

1. Create an ISO9660 compliant ISO image using the xml file as depicted in the example. Make sure that the name of the xml file is **userdata**.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

```
<Environment xmlns:oe="http://schemas.dmtf.org/ovf/environment/1"
```

```
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
oe:id=""
xmlns="http://schemas.dmtf.org/ovf/environment/1">
<PlatformSection>
<Kind>HYPER-V</Kind>
<Version>2013.1</Version>
<Vendor>CISCO</Vendor>
<Locale>en</Locale>
</PlatformSection>
<PropertySection>
<Property oe:key="com.citrix.netscaler.ovf.version"oe:value="1.0"/>
<Property oe:key="com.citrix.netscaler.platform"oe:value="NS1000V"/>
<Property oe:key="com.citrix.netscaler.orch_env"oe:value="cisco-orch-env"/>
<Property oe:key="com.citrix.netscaler.mgmt.ip"oe:value="10.102.100.122"/>
<Property oe:key="com.citrix.netscaler.mgmt.netmask"oe:value="255.255.255.128"/>
<Property oe:key="com.citrix.netscaler.mgmt.gateway"oe:value="10.102.100.67"/></PropertySection>
</Environment>
```

2. Copy the ISO image to hyper-v server.
3. Select the virtual appliance that you imported, and then on the **Action** menu, select **Settings**. You can also select the virtual appliance and then right click and select **Settings**. The **Settings** window for the selected virtual appliance is displayed.
4. In the **Settings** window, under the hardware section, click on **IDE Controller**.
5. In the right window pane, select **DVD Drive** and click on **Add**. The DVD Drive is added under the **IDE Controller** section in the left window pane.
6. Select the **DVD Drive** added in step 5. In the right window pane, select the Image file radio button and click on Browse and select the ISO image that you copied on Hyper-V server, in step 2.
7. Click **Apply**.

Note:

The virtual appliance instance comes up in the default IP address, when:

- The DVD drive is attached and the ISO file is not provided.
- The ISO file does not include the userdata file.
- The userdata file name or format is not correct.

To configure virtual NICs on the Citrix ADC VPX instance, follow these steps:

1. Select the virtual appliance that you imported, and then on the **Action** menu, select **Settings**.
2. In the **Settings for <virtual appliance name>** dialog box, click **Add Hardware** in the left pane.
3. In the right pane, from the list of devices, select **Network Adapter**.
4. Click **Add**.
5. Verify that **Network Adapter (not connected)** appears in the left pane.
6. Select the network adapter in the left pane.
7. In the right pane, from the **Network** drop-down list, select the virtual network to connect the adapter to.
8. To select the virtual network for additional network adapters that you want to use, repeat steps **6** and **7**.
9. Click **Apply**, and then click **OK**.

To configure the Citrix ADC VPX instance:

1. Right-click the virtual appliance that you previously installed, and then select **Start**.
2. Access the console by double-clicking the virtual appliance.
3. Type the Citrix ADC IP address, subnet mask, and gateway for your virtual appliance.

You have completed the basic configuration of your virtual appliance. Type the IP address in a Web browser to access the virtual appliance.

Note:

You can also use virtual machine (VM) template to provision Citrix ADC VPX instance using SCVMM. If you use Microsoft Hyper-V NIC teaming solution with NetScaler VPX instances, see article [CTX224494](#) for more information.

Install a Citrix ADC VPX instance on Linux-KVM platform

November 13, 2024

To set up a Citrix ADC VPX for the Linux-KVM platform, you can use the graphical Virtual Machine Manager (Virt-Manager) application. If you prefer the Linux-KVM command line, you can use the `virsh` program.

The host Linux operating system must be installed on suitable hardware by using virtualization tools such as KVM Module and QEMU. The number of virtual machines (VMs) that can be deployed on the hypervisor depends on the application requirement and the chosen hardware.

After you provision a Citrix ADC VPX instance, you can add additional interfaces.

Limitations and usage guidelines

General recommendations

To avoid unpredictable behavior, apply the following recommendations:

- Do not change the MTU of the vnet interface associated with the VPX VM. Shut down the VPX VM before modifying any configuration parameters, such as Interface modes or CPU.
- Do not force a shutdown of the VPX VM. That is, do not use the `Force off` command.
- Any configurations done on the host Linux might or might not be persistent, depending on your Linux distribution settings. You can choose to make these configurations persistent to ensure consistent behavior across reboots of host Linux operating system.
- The Citrix ADC package has to be unique for each of the Citrix ADC VPX instance provisioned.

Limitations

- Live migration of a VPX instance that runs on KVM is not supported.

Prerequisites for installing a Citrix ADC VPX instance on Linux-KVM platform

November 13, 2024

Check the minimum system requirements for a Linux-KVM server running a Citrix ADC VPX instance.

CPU requirement:

- 64-bit x86 processors with the hardware virtualization features included in the AMD-V and Intel VT-X processors.

To test whether your CPU supports Linux host, enter the following command at the host Linux shell prompt:

```
1 *.egrep '^flags.*(vmx|svm)' /proc/cpuinfo*
```

If the BIOS settings for the above extension are disabled, you must enable them in BIOS.

- Provide at least 2 CPU cores to Host Linux.
- There is no specific recommendation for processor speed, but higher the speed, the better the performance of the VM application.

Memory (RAM) requirement:

Minimum 4 GB for the host Linux kernel. Add additional memory as required by the VMs.

Hard disk requirement:

Calculate the space for Host Linux kernel and VM requirements. A single Citrix ADC VPX VM requires 20 GB of disk space.

Software requirements

The Host kernel used must be a 64-bit Linux kernel, release 2.6.20 or later, with all virtualization tools. Citrix recommends newer kernels, such as 3.6.11-4 and later.

Many Linux distributions such as Red Hat, Centos, and Fedora, have tested kernel versions and associated virtualization tools.

Guest VM hardware requirements

Citrix ADC VPX supports IDE and virtIO hard disk type. The Hard Disk Type has been configured in the XML file, which is a part of the Citrix ADC package.

Networking requirements

Citrix ADC VPX supports virtIO para-virtualized, SR-IOV, and PCI Passthrough network interfaces.

For more information about the supported network interfaces, see:

- [Provision the Citrix ADC VPX instance by using the Virtual Machine Manager](#)
- [Configure a Citrix ADC VPX instance to use SR-IOV network interfaces](#)
- [Configure a Citrix ADC VPX instance to use PCI passthrough network interfaces](#)

Source Interface and Modes

The source device type can be either Bridge or MacVTap. In case of MacVTap, four modes are possible - VEPA, Bridge, Private and Pass-through.

Check the types of interfaces that you can use and the supported traffic types, as given below.

Bridge:

- Linux Bridge.
- Ebtables and iptables settings on host Linux might filter the traffic on the bridge if you do not choose the correct setting or disable IPTable services.

MacVTap (VEPA mode):

- Better performance than a bridge.
- Interfaces from the same lower device can be shared across the VMs.
- Inter-VM communication using the same lower device is possible only if upstream or downstream switch supports VEPA mode.

MacVTap (private mode):

- Better performance than a bridge.
- Interfaces from the same lower device can be shared across the VMs.
- Inter-VM communication using the same lower device is not possible.

MacVTap (bridge mode):

- Better as compared to bridge.
- Interfaces out of same lower device can be shared across the VMs.
- Inter-VM communication using the same lower device is possible, if lower device link is UP.

MacVTap (Pass-through mode):

- Better as compared to bridge.
- Interfaces out of same lower device cannot be shared across the VMs.
- Only one VM can use the lower device.

Note:

For best performance by the VPX instance, ensure that the gro and lro capabilities are switched off on the source interfaces.

Properties of source interfaces

Make sure that you switch off the generic-receive-offload (gro) and large-receive-offload (lro) capabilities of the source interfaces. To switch off the gro and lro capabilities, run the following commands at the host Linux shell prompt.

```
ethtool -K eth6 gro off
```

```
ethtool -K eth6 lro off
```

Example:

```
1 [root@localhost ~]# ethtool -K eth6
2
3         Offload parameters for eth6:
4
5             rx-checksumming: on
6
7             tx-checksumming: on
8
9         scatter-gather: on
10
11        tcp-segmentation-offload: on
12
13        udp-fragmentation-offload: off
14
15        generic-segmentation-offload: on
16
17        generic-receive-offload: off
18
19        large-receive-offload: off
20
21        rx-vlan-offload: on
22
23        tx-vlan-offload: on
24
25        ntuple-filters: off
26
27        receive-hashing: on
28
29 [root@localhost ~]#
```

Example:

If the host Linux bridge is used as a source device, as in the following example, gro and lro capabilities must be switched off on the vnet interfaces, which are the virtual interfaces connecting the host to the guest VMs.

```
1 [root@localhost ~]# brctl show eth6_br
2
3 bridge name      bridge id          STP enabled interfaces
4
5 eth6_br          8000.00e0ed1861ae  no          eth6
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

In the above example, the two virtual interfaces are derived from the eth6_br and are represented as vnet0 and vnet2. Run the following commands to switch off gro and lro capabilities on these interfaces.

```
1 ethtool -K vnet0 gro off
2 ethtool -K vnet2 gro off
3 ethtool -K vnet0 lro off
4 ethtool -K vnet2 lro off
```

Promiscuous mode

The promiscuous mode has to be enabled for the following features to work:

- L2 mode
- Multicast traffic processing
- Broadcast
- IPV6 traffic
- Virtual MAC
- Dynamic routing

Use the following command to enable the promiscuous mode.

```
1 [root@localhost ~]# ifconfig eth6 promisc
2 [root@localhost ~]# ifconfig eth6
3 eth6      Link encap:Ethernet  HWaddr 78:2b:cb:51:54:a3
4           inet6 addr: fe80::7a2b:cbff:fe51:54a3/64 Scope:Link
5           UP BROADCAST RUNNING PROMISC MULTICAST  MTU:9000  Metric
6           :1
7           RX packets:142961 errors:0 dropped:0 overruns:0 frame:0
8           TX packets:2895843 errors:0 dropped:0 overruns:0 carrier
9           :0
10          collisions:0 txqueuelen:1000
11          RX bytes:14330008 (14.3 MB)  TX bytes:1019416071 (1.0 GB)
12 [root@localhost ~]#
```

Module required

For better network performance, make sure the `vhost_net` module is present in the Linux host. To check the existence of `vhost_net` module, run the following command on the Linux host :

```
1 lsmod | grep "vhost\_net"
```

If `vhost_net` is not yet running, enter the following command to run it:

```
1 modprobe vhost\_net
```


Provision the Citrix ADC VPX instance by using OpenStack

November 13, 2024

You can provision a Citrix ADC VPX instance in an Openstack environment either by using the Nova boot command (OpenStack CLI) or Horizon (OpenStack dashboard) .

Provisioning a VPX instance, optionally involves using data from the config drive. Config drive is a special configuration drive that attaches to the instance as a CD-ROM device when it boots. This configuration drive can be used to pass networking configuration such as management IP address, network mask, default gateway, and to inject customer scripts.

In a Citrix ADC appliance, the default authentication mechanism is password based. Now, SSH key-pair authentication mechanism is supported for Citrix ADC VPX instances on OpenStack environment.

The key-pair (public key and private key) needs to be generated before using Public Key Cryptography mechanism. You can use different mechanisms, such as Horizon, Puttygen.exe for Windows, and ssh-keygen for Linux environment, to generate the key pair. Refer to online documentation of respective mechanisms for more information about generating key pair.

Once a key pair is available, copy the private key to a secure location to which authorized persons will have access. In OpenStack, public key can be deployed on a VPX instance by using Horizon or Nova boot command. When a VPX instance is provisioned by using OpenStack, it first detects that the instance is booting in an OpenStack environment by reading a specific BIOS string. This string is “OpenStack Foundation” and for Red Hat Linux distributions it is stored in /etc/nova/release. This is a standard mechanism that is available in all OpenStack implementations based on KVM hypervisor platform. The drive should have a specific OpenStack label.

If the config drive is detected, the instance attempts to read the network configuration, custom scripts, and SSH key pair if provided.

Userdata file

The Citrix ADC VPX instance uses a customized OVF file, also known as userdata file, to inject network configuration, custom scripts. This file is provided as part of config drive. Here is an example of a customized OVF file.

```
1 <?xml version="1.0" encoding="UTF-8" standalone="no"?>
2 <Environment xmlns:oe="http://schemas.dmtf.org/ovf/environment/1"
3 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4 oe:id=""
5 xmlns="http://schemas.dmtf.org/ovf/environment/1"
6 xmlns:cs="http://schemas.citrix.com/openstack">
```

```

 7 <PlatformSection>
 8 <Kind></Kind>
 9 <Version>2016.1</Version>
10 <Vendor>VPX</Vendor>
11 <Locale>en</Locale>
12 </PlatformSection>
13 <PropertySection>
14 <Property oe:key="com.citrix.netscaler.ovf.version" oe:value="1.0"/>
15 <Property oe:key="com.citrix.netscaler.platform" oe:value="NSVPX"/>
16 <Property oe:key="com.citrix.netscaler.orch_env" oe:value="openstack-
    orch-env"/>
17 <Property oe:key="com.citrix.netscaler.mgmt.ip" oe:value="10.1.2.22"/>
18 <Property oe:key="com.citrix.netscaler.mgmt.netmask" oe:value="
    255.255.255.0"/>
19 <Property oe:key="com.citrix.netscaler.mgmt.gateway" oe:value="10.1.2.1
    "/>
20 </PropertySection>
21 <cs:ScriptSection>
22 <cs:Version>1.0</cs:Version>
23 <ScriptSettingSection xmlns="http://schemas.citrix.com/openstack"
    xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
24 <Scripts>
25 <Script>
26 <Type>shell</Type>
27 <Parameter>X Y</Parameter>
28 <Parameter>Z</Parameter>
29 <BootScript>before</BootScript>
30 <Text>
31 <Text>#!/bin/bash
32 <Text>echo "Hi, how are you" $1 $2 >> /var/sample.txt
33 </Text>
34 </Script>
35 <Script>
36 <Type>python</Type>
37 <BootScript>after</BootScript>
38 <Text>
39 <Text>#!/bin/python
40 print("Hello");
41 </Text>
42 </Script>
43 <Script>
44 <Type>perl</Type>
45 <BootScript>before</BootScript>
46 <Text>
47 <Text>!/usr/bin/perl
48 my $name = "VPX";
49 print "Hello, World $name !\n" ;
50 </Text>
51 </Script>
52 <Script>
53 <Type>nscli</Type>
54 <BootScript>after</BootScript>
55 <Text>

```

```
56         add vlan 33
57     bind vlan 33 -ifnum 1/2
58         </Text>
59     </Script>
60 </Scripts>
61 </ScriptSettingSection>
62 </cs:ScriptSection>
63 </Environment>
```

In the OVF file above “PropertySection” is used for NetScaler networking configuration while `<cs:ScriptSection>` is used to enclose all scripts. `<Scripts></Scripts>` tags are used to bundle all scripts together. Each script is defined in between `<Script>` `</Script>` tags. Each script tag has following fields/tags:

- a) `<Type>`: Specifies value for script type. Possible values: Shell/Perl/Python/NSLCI (for NetScaler CLI scripts)
- b) `<Parameter>`: Provides parameters to the script. Each script can have multiple `<Parameter>` tags.
- c) `<BootScript>`: Specifies script execution point. Possible values for this tag: before/after. “before” specifies script will be executed before PE comes up. “after” specifies that the script will be executed after PE comes up.
- d) `<Text>`: Pastes content of a script.

Note

Currently the VPX instance does not take care of sanitization of scripts. As an administrator, you should check the validity of the script.

Not all sections need to be present. Use an empty “PropertySection” to only define scripts to execute on first boot or an empty `<cs:ScriptSection>` to only define networking configuration.

After the required sections of the OVF file (userdata file) are populated, use that file to provision the VPX instance.

Network configuration

As part of network configuration, the VPX instance reads:

- Management IP address
- Network mask
- Default gateway

After the parameters are successfully read, they are populated in the NetScaler configuration, to allow managing the instance remotely. If the parameters are not read successfully or the config drive is not available, the instance transitions to the default behavior, which is:

- The instance attempts to retrieve the IP address information from DHCP.
- If DHCP fails or times-out, the instance comes up with default network configuration (192.168.100.1/16).

Customer script

The VPX instance allows to execute a custom script during initial provisioning. The appliance supports script of type Shell, Perl, Python, and Citrix ADC CLI commands.

SSH key pair authentication

The VPX instance copies public key, available within the configuration drive as part of instance meta data, into its “authorized_keys”file. This allows the user to access the instance with private key.

Note:

When an SSH key is provided, the default credentials (nsroot/nsroot) no longer work. If password-based access is needed, log on with the respective SSH private key and manually set a password.

Before you begin

Before you provision a VPX instance on OpenStack environment, extract the .qcow2 file from the .tgz file and build

an OpenStack image from the qcow2 image. Follow these steps:

1. Extract the .qcow2 file from the .tgz file by typing the following command.

```
1 tar xvzf <TAR file>
2
3 tar xvzf <NSVPX-KVM-12.0-26.2_nc.tgz>
4 NSVPX-KVM.xml
5 NSVPX-KVM-12.0-26.2_nc.qcow2
```

2. Build an OpenStack image using the .qcow2 file extracted in step 1 by typing the following command.

```
1 openstack image create --container-format bare --property hw_disk_bus=
  ide --disk-format qcow2 --file <path to qcow2 file> --public <name
  of the OpenStack image>
2
3 glance image-create --name="NS-VPX-12-0-26-2" --property hw_disk_bus=
  ide --ispublic=
4 true --container-format=bare --disk-format=qcow2< NSVPX-KVM-12.0-26.2
  _nc.qcow2
```

Figure 1: The following illustration provides a sample output for the glance image-create command.

Field	Value
checksum	154ade3fc7dca7d1706b1d03d7d97552
container_format	bare
created_at	2017-03-13T08:52:31Z
disk_format	qcow2
file	/v2/images/322c1e0f-cce8-4b7b-b53e-bd8152c388ed/file
id	322c1e0f-cce8-4b7b-b53e-bd8152c388ed
min_disk	0
min_ram	0
name	VPX-KVM-12.0-26.2
owner	58d17d81df5d4406afbb4fdab3a58d79
properties	hw_disk_bus='ide'
protected	False
schema	/v2/schemas/image
size	784338944
status	active
updated_at	2017-03-13T08:52:43Z
virtual_size	None
visibility	public

Provisioning the VPX instance

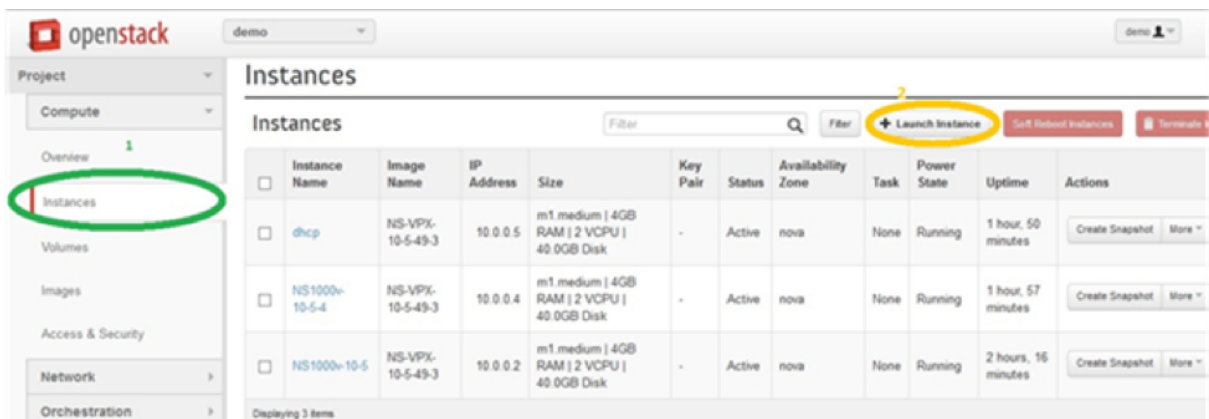
You can provision a VPX instance in two ways by using one of the options:

- Horizon (OpenStack dashboard)
- Nova boot command (OpenStack CLI)

Provision a VPX instance by using the OpenStack dashboard

Follow these steps to provision the VPX instance by using Horizon:

1. Log on to the OpenStack dashboard.
2. In the Project panel on the left hand side of the dashboard, select **Instances**.
3. In the Instances panel, click **Launch Instance** to open the Instance Launching wizard.



4. In the Launch Instance wizard, fill in the details, like:

1. Instance Name
2. Instance Flavor
3. Instance Count
4. Instance Boot Source
5. Image Name

Launch Instance ✕

Details *
Access & Security *
Networking *
Post-Creation
Advanced Options

Availability Zone:
nova ▼

Instance Name: *
NSVPX_10_1

Flavor: *
m1.medium ▼

Instance Count: *
1

Instance Boot Source: *
Boot from image ▼

Image Name:
NS-VPX-10-1-130-11 (20.0 GB) ▼

Specify the details for launching an instance.

The chart below shows the resources used by this project in relation to the project's quotas.

Flavor Details

Name	m1.medium
VCPUs	2
Root Disk	40 GB
Ephemeral Disk	0 GB
Total Disk	40 GB
RAM	4,096 MB

Project Limits

Number of Instances 6 of 10 Used

Number of VCPUs 12 of 20 Used

Total RAM 24,576 of 51,200 MB Used

Cancel
Launch

5. Deploy a new key pair or an existing key pair through Horizon by completing the following steps:
 - a) If you don't have an existing key pair, create the key by using any existing mechanisms. If you've an existing key, skip this step.
 - b) Copy the content of public key.
 - c) Go to **Horizon > Instances > Create New Instances**.
 - d) Click **Access & Security**.
 - e) Click the + sign next to the **Key Pair** drop-down menu and provide values for shown parameters.
 - f) Paste public key content in *Public key* box, give a name to the key and click **Import Key Pair**.

Import Key Pair ✕

Key Pair Name *

Description:

Key Pairs are how you login to your instance after it is launched.

Choose a key pair name you will recognise and paste your SSH public key into the space provided.

SSH key pairs can be generated with the ssh-keygen command:

```
ssh-keygen -t rsa -f cloud.key
```

This generates a pair of keys: a key you keep private (cloud.key) and a public key (cloud.key.pub). Paste the contents of the public key file here.

After launching an instance, you login using the private key (the username might be different depending on the image you launched):

```
ssh -i cloud.key <username>@<instance_ip>
```

Public Key *

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQACjZih
mFducHd8elrm/6RXQfvVuaQPOM92dyNOw74J7
03te1FrwL38iGXbjlByc2+qBV7ZIFRiYQEik2UfM+
EtJJIcx92m4aln1RlgFvukXECHIXGqfQXVI06pyim
KRWiqXhl+h+tvPGS4iltJ3uWKwfh1PDGYkmgAik
osA955L+W9ngVloVyaK40OuAqYCTwfQNBKVuZ
GBOAH9eJejim0Lo8w5uA58/Jbjl8gNCzQYw5S2w
EcvsxOvhdb3LW9YADAVnihVK4NLeBc4HlsFeHl
5UY0IYyGk7aW/2SXjzkWRqZ8cX1Oba0XoDiCYN
apRVOT6FB//ykrwu+BSVF4v0og3
```

6. Click the **Post Creation** tab in the wizard. In Customization Script, add the content of the userdata file. The userdata file contains the IP address, Netmask and Gateway details, and customer scripts of the VPX instance.

7. After a key pair is selected or imported, check config-drive option and click **Launch**.

Launch Instance ✕

Details *
Access & Security
Networking *
Post-Creation
Advanced Options

Disk Partition ⓘ

Specify advanced options to use when launching an instance.

Configuration Drive ⓘ

Provision the VPX instance by using OpenStack CLI

Follow these steps to provision a VPX instance by using OpenStack CLI.

1. To create an image from qcow2, type the following command:

```
1 openstack image create --container-format bare --property hw_disk_bus=
   ide --diskformat qcow2 --file NSVPX-OpenStack.qcow2 --public VPX-
   ToT-Image
```

2. To select an image for creating an instance, type the following command:

```
1 openstack image list | more
```

3. To create an instance of a particular flavor, type the following command to choose a flavor ID/Name of from a list:

```
1 openstack flavor list
```

4. To attach a NIC to a particular network, type the following command to choose a network ID from a network list:

```
1 openstack network list
```

5. To create an instance, type the following command:

```
1 openstack server create --flavor FLAVOR_ID --image IMAGE_ID --key-name
   KEY_NAME
2 --user-data USER_DATA_FILE_PATH --config-drive True --nic net-id=net-
   uuid
3 INSTANCE_NAME
4 openstack server create --image VPX-ToT-Image --flavor m1.medium --user
   -data
5 ovf.xml --config-drive True --nic net-id=2734911b-ee2b-48d0-a1b6-3
   efd44b761b9
6 VPX-ToT
```

Figure 2: The following illustration provides a sample output.

Field	Value
OS-DCF:diskConfig	MANUAL
OS-EXT-AZ:availability_zone	
OS-EXT-SRV-ATTR:host	None
OS-EXT-SRV-ATTR:hypervisor_hostname	None
OS-EXT-SRV-ATTR:instance_name	instance-000001c2
OS-EXT-STS:power_state	0
OS-EXT-STS:task_state	scheduling
OS-EXT-STS:vm_state	building
OS-SRV-USG:launched_at	None
OS-SRV-USG:terminated_at	None
accessIPv4	
accessIPv6	
addresses	
adminPass	pFVvMtq7N8Z6
config_drive	True
created	2017-03-13T10:32:59Z
flavor	m1.medium (3)
hostId	
id	a1fe991e-3604-43a0-9dd6-59fa0f3749df
image	VPX-ToT-Image (f0c2f9d1-08f2-4b2e-9943-2ee6bc2edbc7)
key_name	None
name	VPX-ToT
os-extended-volumes:volumes_attached	[]
progress	0
project_id	58d17d81df5d4406afbb4fdab3a58d79
properties	
security_groups	[{'name': 'u'default'}]
status	BUILD
updated	2017-03-13T10:33:00Z
user_id	a6347b33916b4eb1b1f76360a9c8f935

Provision the Citrix ADC VPX instance by using the Virtual Machine Manager

November 13, 2024

The Virtual Machine Manager is a desktop tool for managing VM guests. It enables you to create new VM guests and various types of storage, and manage virtual networks. You can access the graphical console of VM guests with the built-in VNC viewer and view performance statistics, either locally or remotely.

After installing your preferred Linux distribution, with KVM virtualization enabled, you can proceed with provisioning virtual machines.

While using the Virtual Machine Manager to provision a Citrix ADC VPX instance, you have two options:

- Enter the IP address, gateway, and netmask manually
- Assign the IP address, gateway, and netmask automatically (auto-provisioning)

You can use two kinds of images to provision a Citrix ADC VPX instance:

- RAW
- QCOW2

You can convert a Citrix ADC VPX RAW image to a QCOW2 image and provision the Citrix ADC VPX instance. To convert the RAW image to a QCOW2 image, type the following command:

```
qemu-img convert -O qcow2 original-image.raw image-converted.qcow
```

For example:

```
qemu-img convert -O qcow2 NSVPX-KVM-11.1-12.5_nc.raw NSVPX-KVM-11.1-12.5_nc.qcow
```

A typical Citrix ADC VPX deployment on KVM includes the following steps:

- Checking prerequisites for Auto-Provisioning a Citrix ADC VPX Instance
- Provisioning the Citrix ADC VPX Instance by Using a RAW Image
- Provisioning the Citrix ADC VPX Instance by Using a QCOW2 Image
- Adding Additional Interfaces to a VPX Instance by using Virtual Machine Manager

Check prerequisites for auto-provisioning a Citrix ADC VPX instance

Auto-provisioning is an optional feature, and it involves using data from the CDROM drive. If this feature is enabled, you need not enter the management IP address, network mask, and default gateway of the Citrix ADC VPX instance during initial setup.

You need to complete the following tasks before you can auto-provision a VPX instance:

1. Create a customized Open Virtualization Format (OVF) XML file or userdata file.
2. Convert the OVF file into an ISO image by using an online application (for example PowerISO).
3. Mount the ISO image on the the KVM host by using any secure copy (SCP)-based tools.

Sample OVF XML file:

Here's is an example of the contents an OVF XML file, which you can use as a sample to create your file.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<Environment xmlns:oe="http://schemas.dmtf.org/ovf/environment/1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
oe:id=""
xmlns="http://schemas.dmtf.org/ovf/environment/1"
xmlns:cs="http://schemas.citrix.com/openstack"\>
<PlatformSection>
<Kind></Kind>
<Version>2016.1</Version>
<Vendor>VPX</Vendor>
```

```
<Locale>en</Locale>
</PlatformSection>
<PropertySection>
<Property oe:key="com.citrix.netscaler.ovf.version"oe:value="1.0"/>
<Property oe:key="com.citrix.netscaler.platform"oe:value="NSVPX"/>
<Property oe:key="com.citrix.netscaler.orch_env"oe:value="KVM"/>
<Property oe:key="com.citrix.netscaler.mgmt.ip"oe:value="10.1.2.22"/>
<Property oe:key="com.citrix.netscaler.mgmt.netmask"oe:value="255.255.255.0"/>
<Property oe:key="com.citrix.netscaler.mgmt.gateway"oe:value="10.1.2.1"/>
</PropertySection>
</Environment>
```

In the OVF XML file above, “PropertySection” is used for NetScaler networking configuration. When you create the file, specify values for the parameters that are highlighted at the end of the example:

- Management IP address
- Netmask
- Gateway

Important:

If the OVF file is not properly XML formatted, the VPX instance is assigned the default network configuration, not the values specified in the file.

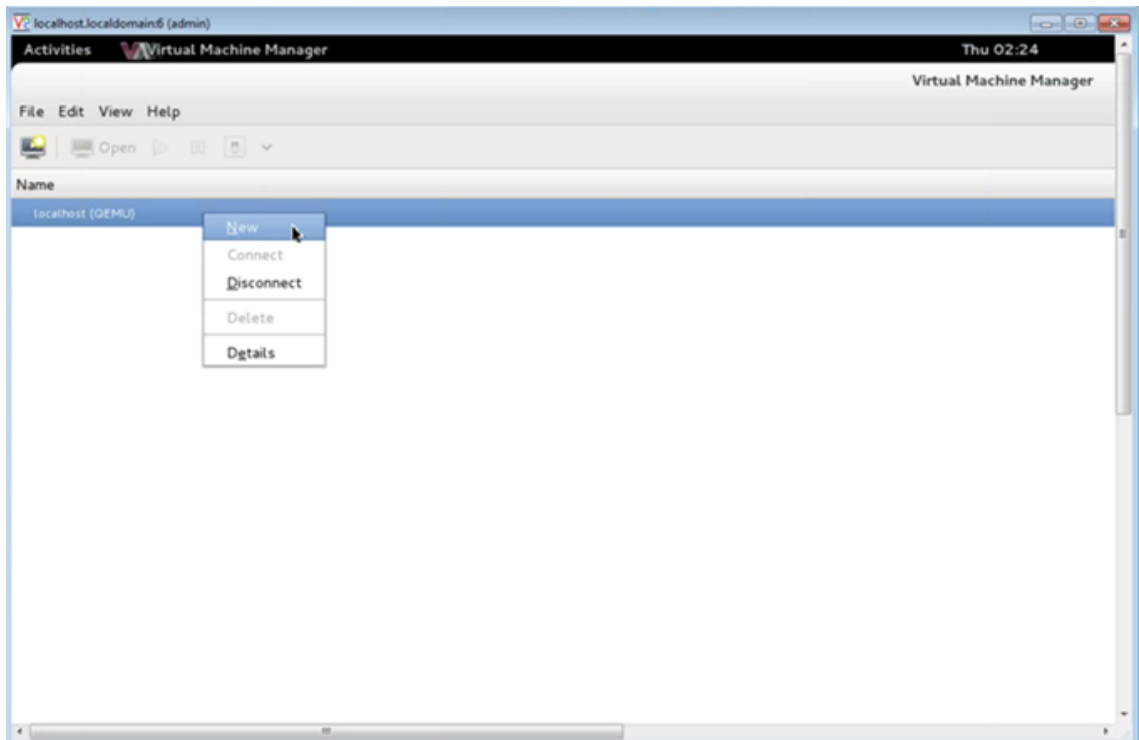
Provision the Citrix ADC VPX instance by using a RAW image

The Virtual Machine Manager enables you to provision a Citrix ADC VPX instance by using a RAW image.

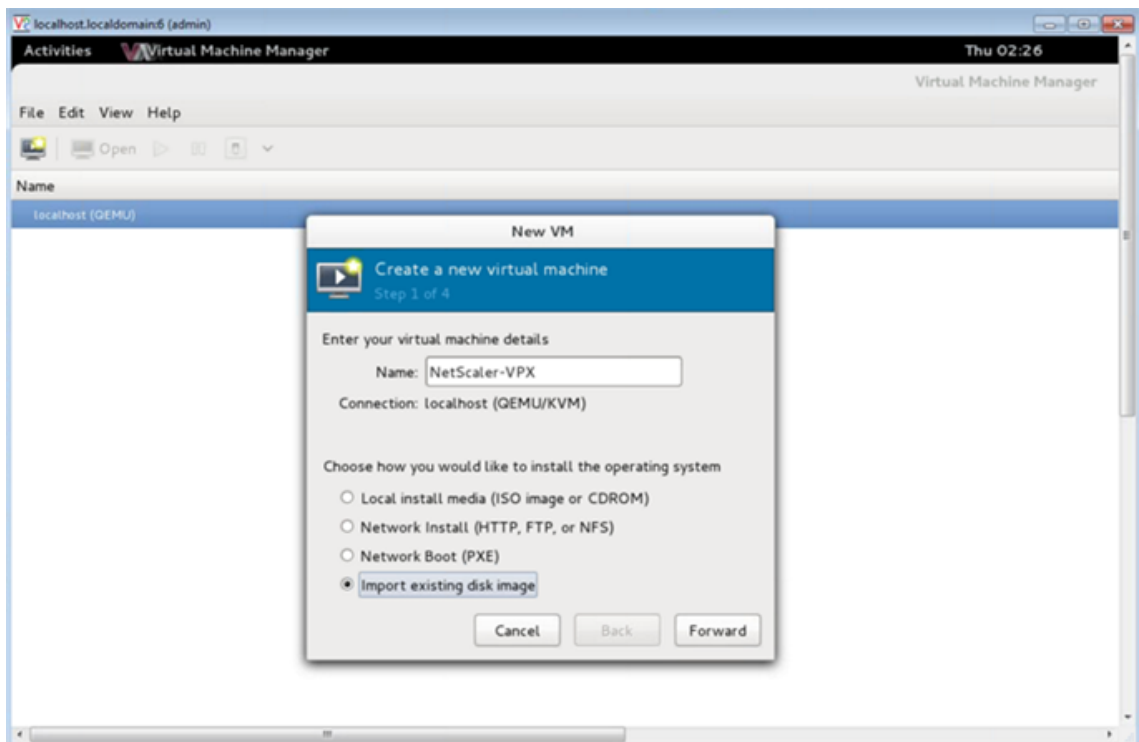
To provision a Citrix ADC VPX instance by using the Virtual Machine Manager, follow these steps:

1. Open the Virtual Machine Manager (**Application > System Tools > Virtual Machine Manager**) and enter the logon credentials in the **Authenticate** window.

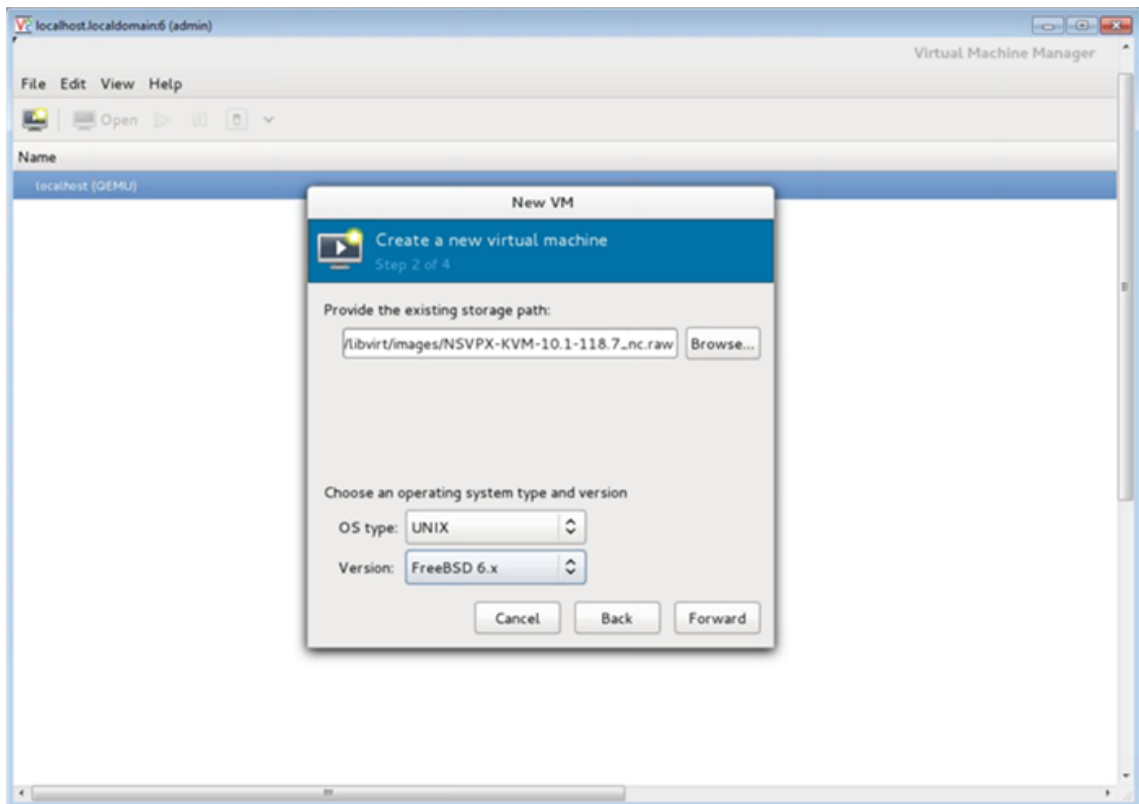
2. Click the  icon or right-click **localhost (QEMU)** to create a new Citrix ADC VPX instance.



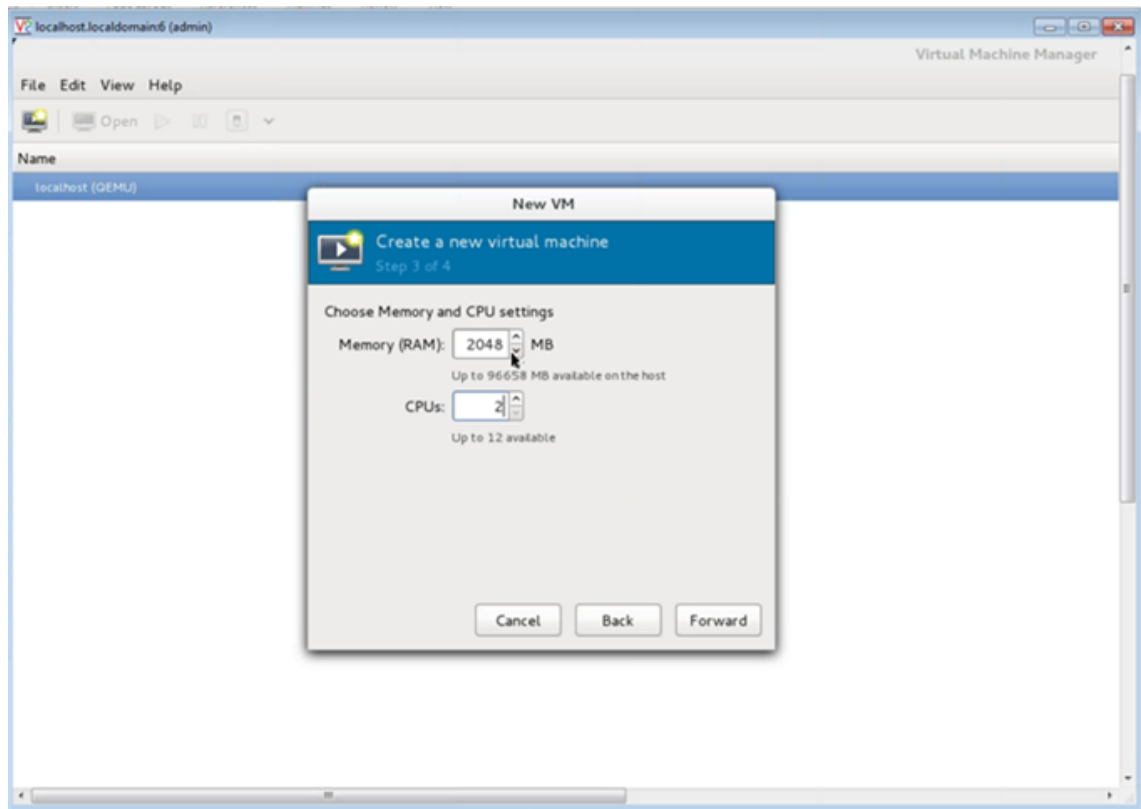
3. In the **Name** text box, enter a name for the new VM (for example, NetScaler-VPX).
4. In the **New VM** window, under “Choose how you would like to install the operating system,” select **Import existing disk image**, and then click **Forward**.



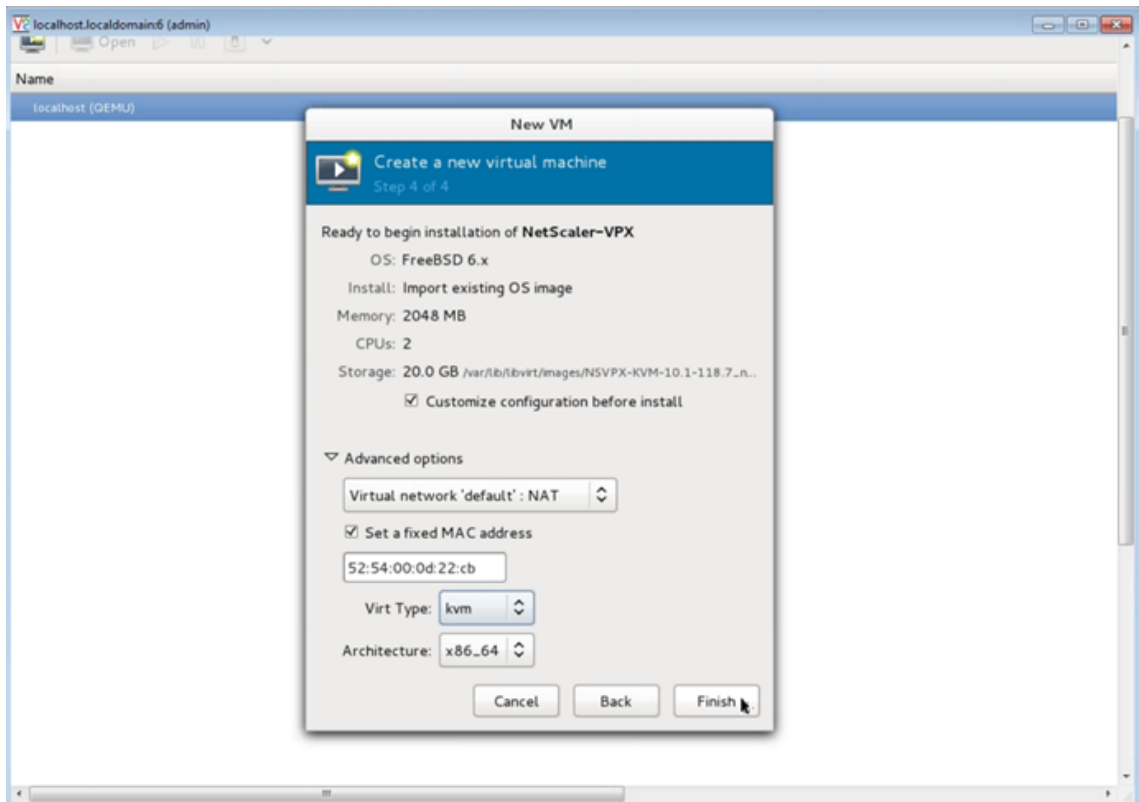
5. In the **Provide the existing storage path** field, navigate the path to the image. Choose the OS type as UNIX and Version as FreeBSD 6.x. Then, click **Forward**.



6. Under **Choose Memory and CPU** settings select the following settings, and then click **Forward**:
 - Memory (RAM)—2048 MB
 - CPUs—2

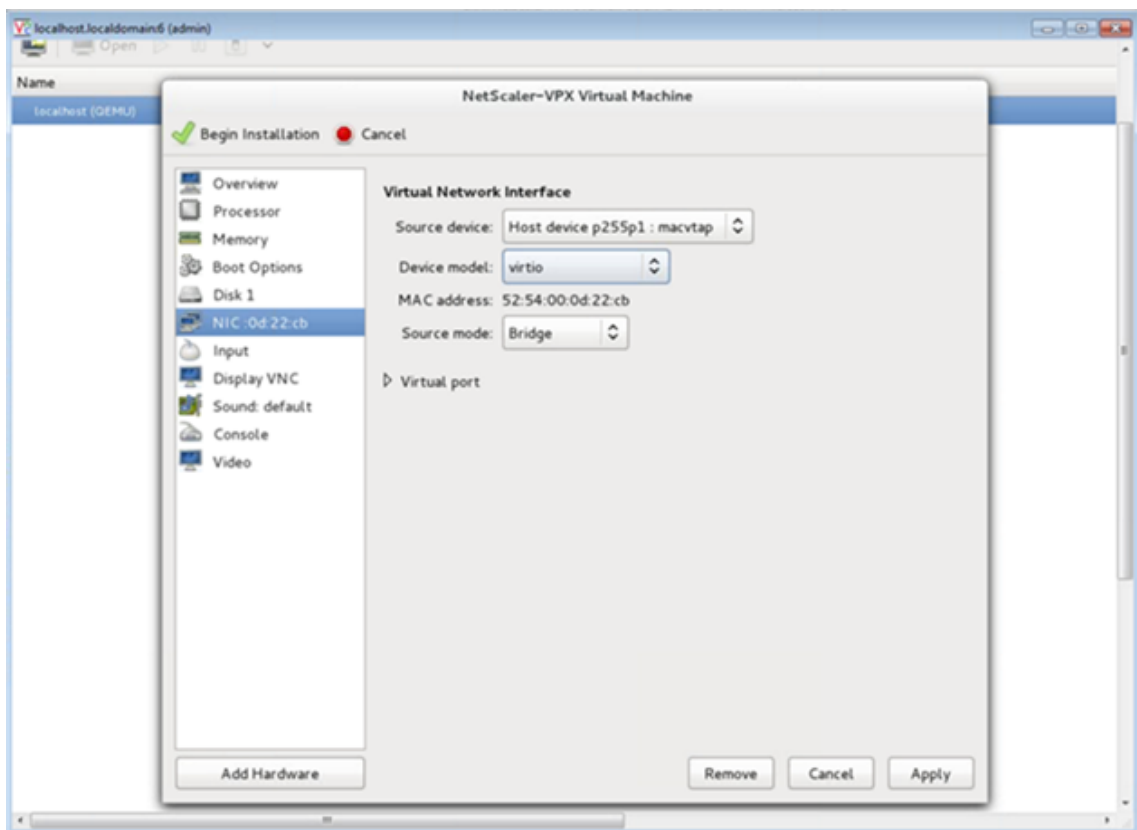


7. Select the **Customize configuration before install** check box. Optionally, under **Advanced options** you can you can customize the MAC address. Make sure the **Virt Type** selected is kvm and the Architecture selected is x86_64. Click **Finish**.



8. Select a NIC and provide the following configuration:

- Source device—ethX macvtap or Bridge
- Device model—virtio
- Source mode—Bridge



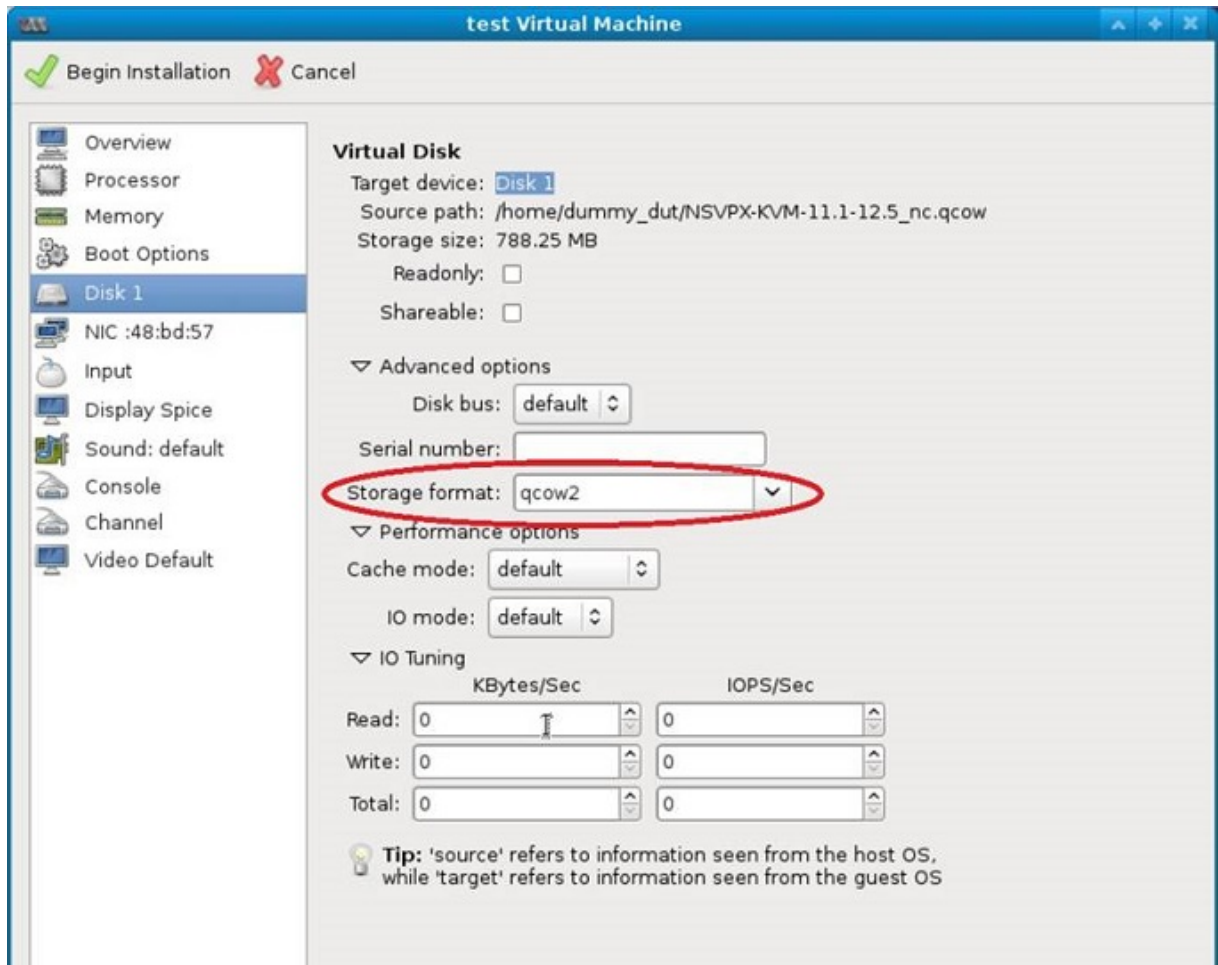
9. Click **Apply**.
10. If you want to auto-provision the VPX instance, see the section “**Enabling Auto-Provisioning by Attaching a CDROM Drive**” in this document. Otherwise, click **Begin Installation**. After you have provisioned the Citrix ADC VPX on KVM, you can add additional interfaces.

Provision the Citrix ADC VPX instance by using a QCOW2 image

Using the Virtual Machine Manager, you can provision the Citrix ADC VPX instance by using a QCOW2 image.

To provision a Citrix ADC VPX instance by using a QCOW2 image, follow these steps:

1. Follow **step 1 to step 8** in Provision the Citrix ADC VPX instance by using a RAW image.
Note: Ensure that you select **qcow2** image in **step 5**.
2. Select **Disk 1** and click **Advanced options**.
3. Select **qcow2** from the Storage format drop-down list.

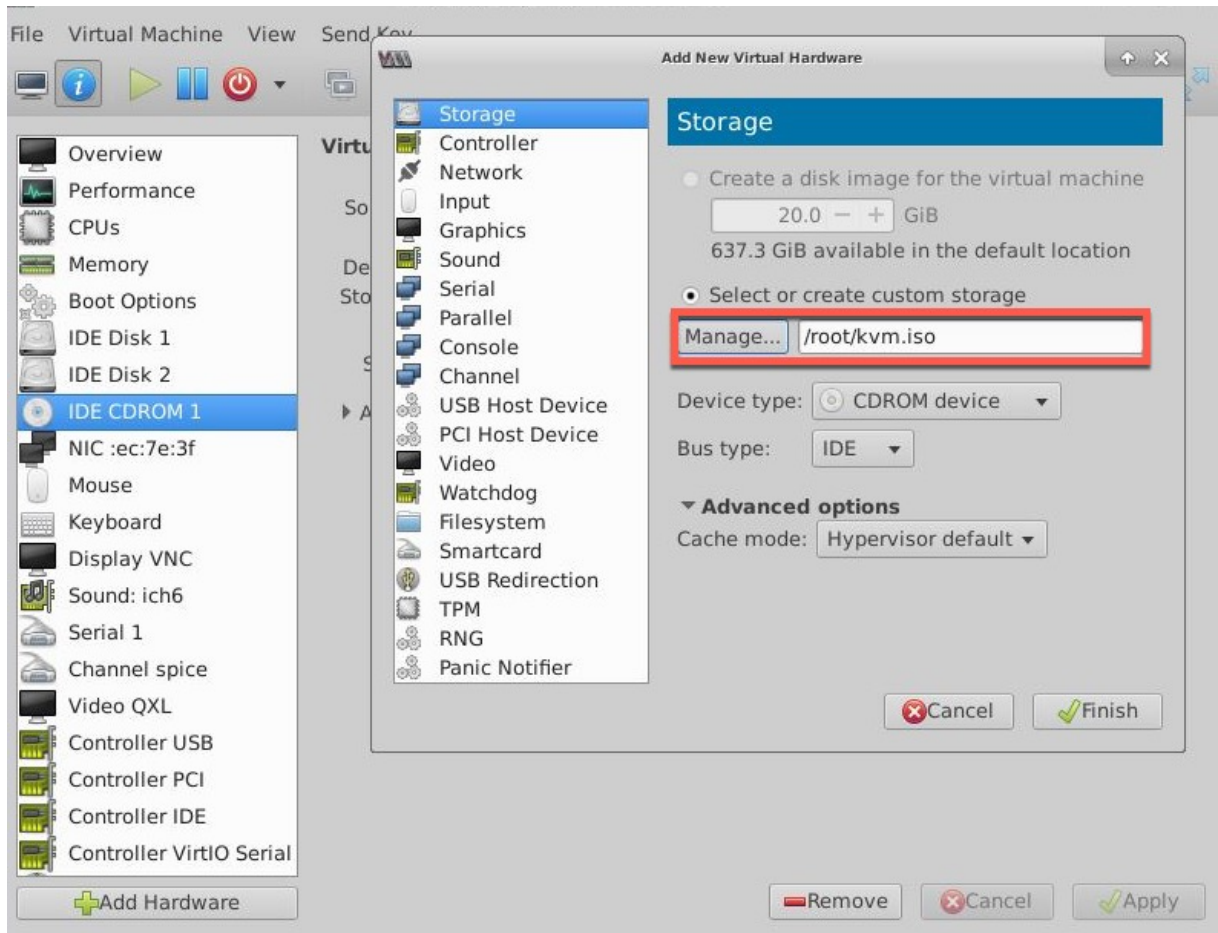


4. Click **Apply**, and then click

Begin Installation. After you have provisioned the Citrix ADC VPX on KVM, you can add additional interfaces.

Enable auto-provisioning by attaching a CDROM drive

1. Click Add **Hardware > Storage > Device type > CDROM device.**
2. Click **Manage** and select the correct ISO file that you mounted in the “Prerequisites for Auto-Provisioning a Citrix ADC VPX Instance” section, and click **Finish.** A new CDROM under Resources on your Citrix ADC VPX instance is created.



3. Power on the VPX instance, and it auto-provisions with the network configuration provided in the OVF file, as shown in the example screen capture.

```

File Virtual Machine View Send Key
Aug 11 10:14:55 <local0.alert> ns restart[2578]: Restart: /netscaler/nsstart.sh
exited normally. Exit code (0)
Aug 11 10:14:55 <local0.alert> ns restart[2578]: Successfully deregistered with
h Pitboss ...

login: nsroot
Password:
Aug 11 10:15:04 <auth.notice> ns login: ROOT LOGIN (nsroot) ON ttyv0
Copyright (c) 1992-2013 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
The Regents of the University of California. All rights reserved.

Done
> sh ip
      Ippaddress      Traffic Domain  Type          Mode      Arp      Icmp
  Userver  State
  -----
1)      10.1.2.22      0              NetScaler IP  Active    Enabled  Enab
led NA      Enabled
Done
> Aug 11 10:15:13 <local0.alert> ns restart[2578]: Nsshutdown lock released !

```

4. If auto-provision fails, the instance comes up with the default IP address (192.168.100.1). In that case, you need to complete the initial configuration manually. For more information, see [Configuring a NetScaler for the First Time](#).


Add additional interfaces to the Citrix ADC VPX instance by using Virtual Machine Manager

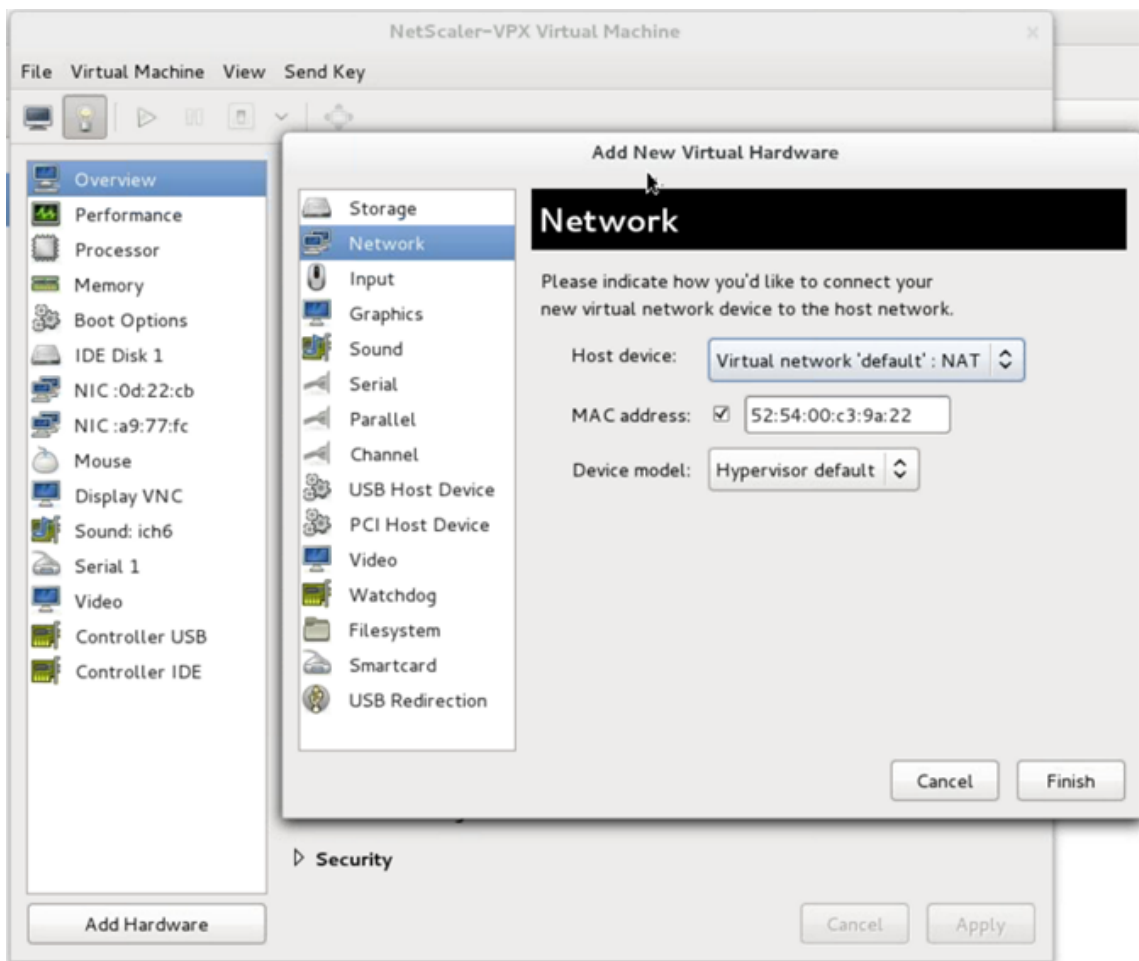
After you have provisioned the NetScaler VPX instance on KVM, you can add additional interfaces.

To add additional interfaces, follow these steps.

1. Shut down the NetScaler VPX instance running on the KVM.
2. Right-click the VPX instance and choose **Open** from the pop-up menu.



3. Click the  icon in the header to view the virtual hardware details.
4. Click **Add Hardware**. In the **Add New Virtual Hardware window**, select **Network** from the navigation menu.

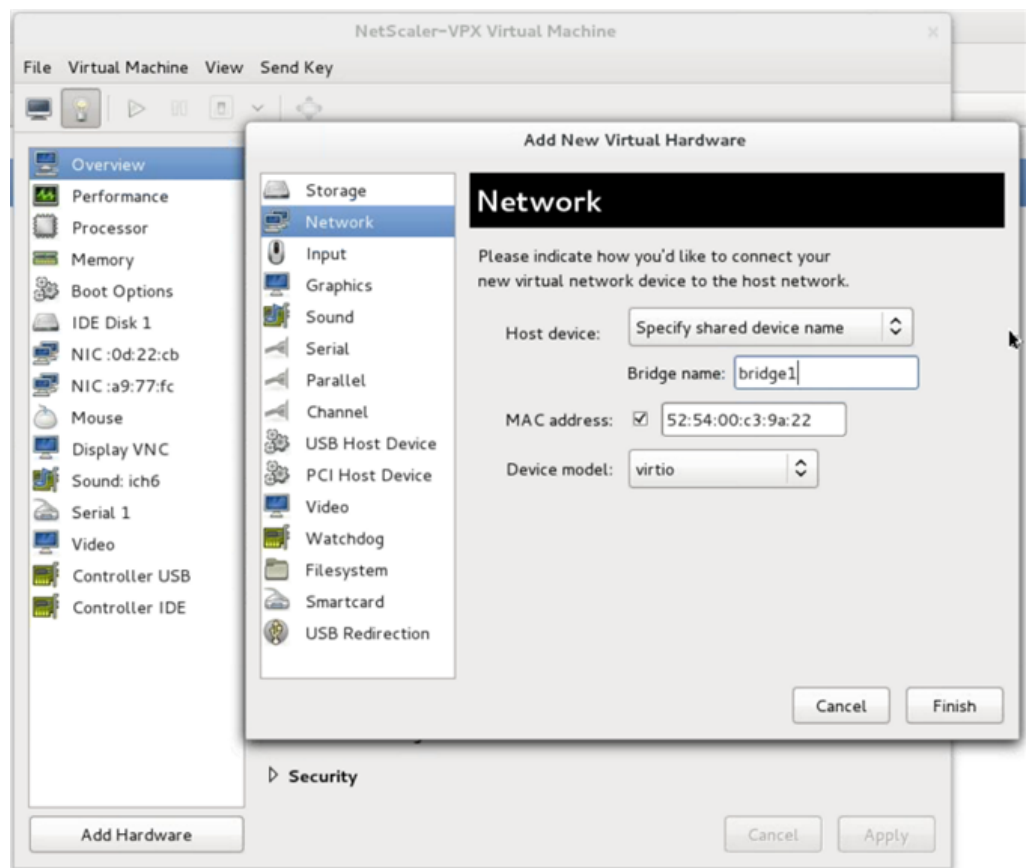


5. In **Host Device** field, select the physical interface type. The host device type can be either Bridge or MacVTap. In case of MacVTap, four modes possible are VEPA, Bridge, Private and Pass-through.

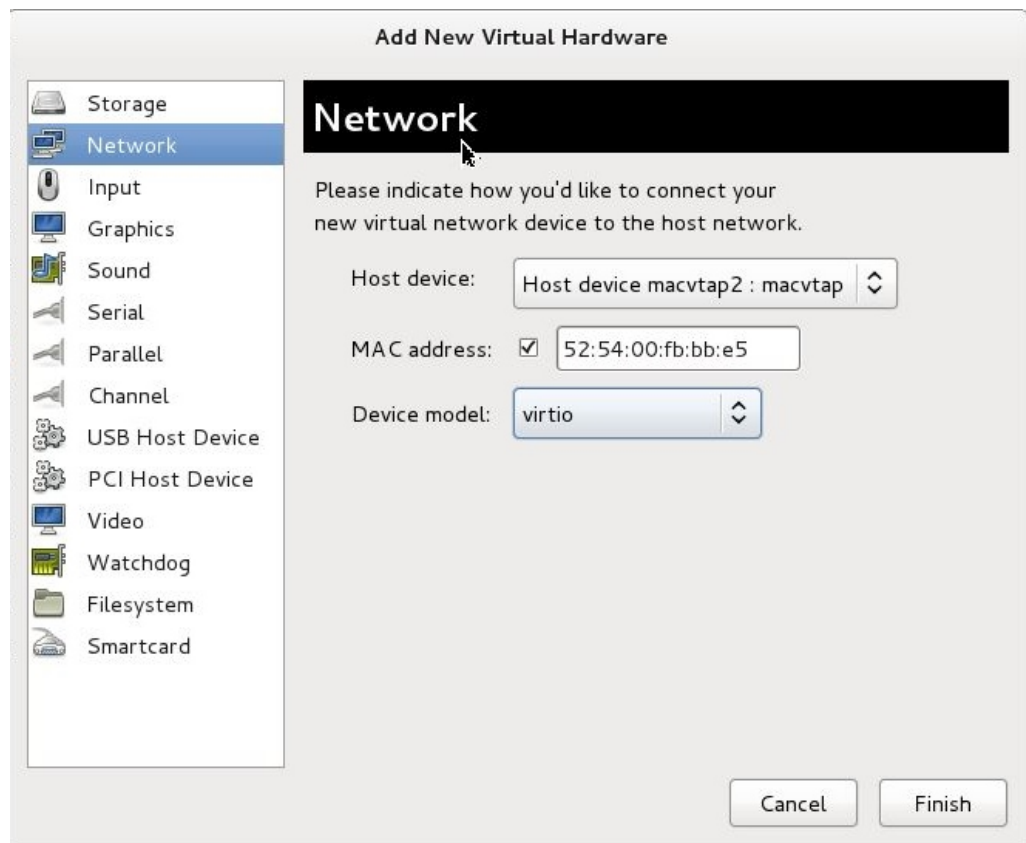
a) For Bridge

- i. Host device—Select the “Specify shared device name” option.
- ii. Provide the Bridge name that is configured in the KVM host.

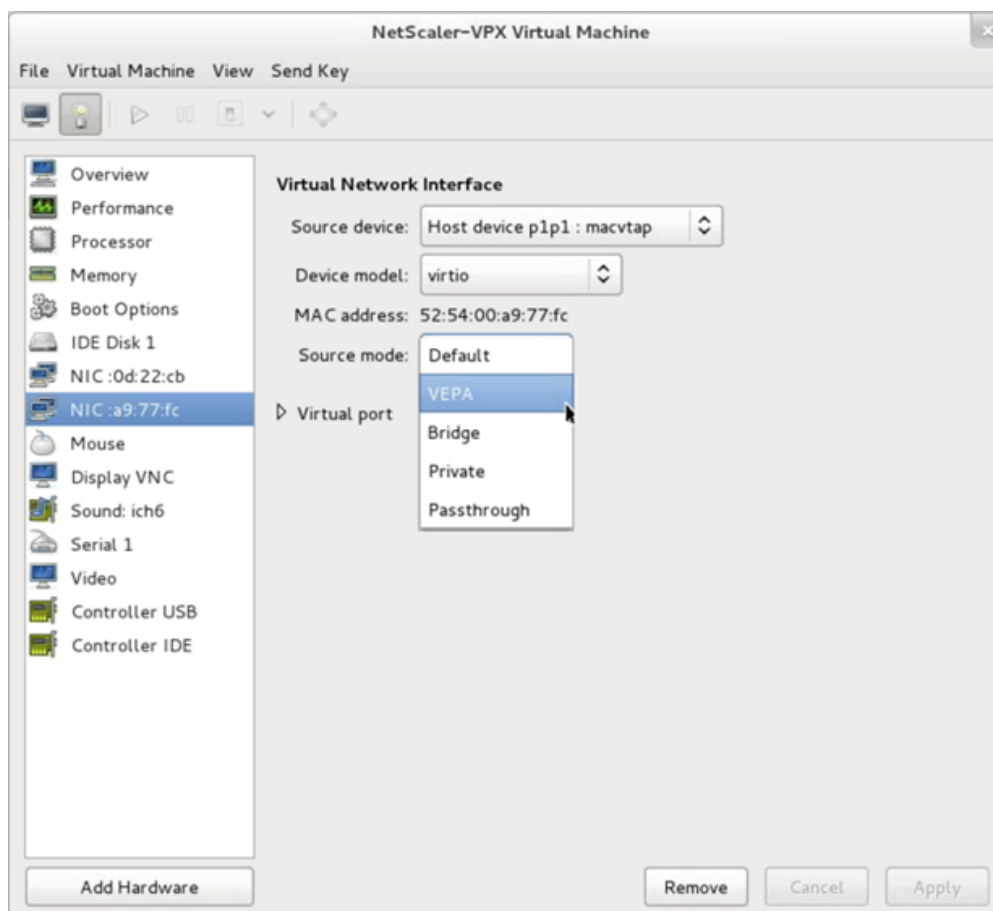
Note: Make sure that you have configured a Linux bridge in the KVM host, bound the physical interface to the bridge, and put the bridge in the UP state.



- iii. Device model—virtio.
 - iv. Click **Finish**.
- b) For MacVTap
- i. Host device—Select the physical interface from the menu.
 - ii. Device model—virtio.



iii. Click **Finish**. You can view the newly added NIC in the navigation pane.



iv. Select the newly added NIC and select the Source mode for this NIC. The available modes are VEPA, Bridge, Private, and Passthrough. For more details on the interface and modes, see Source Interface and Modes.

v. Click **Apply**.

6. If you want to auto-provision the VPX instance, see the section “Adding a Config Drive to Enable Auto-Provisioning” in this document. Otherwise, power on the VPX instance to complete the initial configuration manually.

Important:

Interface parameter configurations such as speed, duplex, and autonegotiation are not supported.

Configure a Citrix ADC VPX instance to use SR-IOV network interfaces

November 13, 2024

You can configure a Citrix ADC VPX instance running on Linux-KVM platform using single root I/O virtualization (SR-IOV) with the following NICs:

- Intel 82599 10G
- Intel X710 10G
- Intel XL710 40G
- Intel X722 10G

This section describes how to:

- Configure a Citrix ADC VPX Instance to Use SR-IOV Network Interface
- Configure Static LA/LACP on the SR-IOV Interface
- Configure VLAN on the SR-IOV Interface

Limitations

Keep the limitations in mind while using Intel 82599, X710, XL710, and X722 NICs. The following features not supported.

Limitations for Intel 82599 NIC:

- L2 mode switching.
- Admin partitioning (shared VLAN mode).
- High availability (active-active mode).
- Jumbo frames.
- IPv6: You can configure only up to 30 unique IPv6 addresses in a VPX instance if you've at least one SR-IOV interface.
- VLAN configuration on Hypervisor for SRIOV VF interface through "ip link" command is not supported.
- Interface parameter configurations such as speed, duplex, and autonegotiations are not supported.

Limitations for Intel X710 10G, Intel XL710 40G, and Intel X722 10G NICs:

- L2 mode switching.
- Admin partitioning (shared VLAN mode).
- In a cluster, Jumbo frames are not supported when the XL710 NIC is used as a data interface.
- Interface list reorders when interfaces are disconnected and reconnected.
- Interface parameter configurations such as speed, duplex, and auto negotiations are not supported.
- Interface name is 40/X for Intel X710 10G, Intel XL710 40G, and Intel X722 10G NICs
- Up to 16 Intel XL710/X710/X722 SRIOV or PCI passthrough interfaces can be supported on a VPX instance.

Note:

For Intel X710 10G, Intel XL710 40G, and Intel X722 10G NICs to support IPv6, you need to enable trust mode on the Virtual Functions (VFs) by typing the following command on the KVM host:

```
# ip link set <PNIC> <VF> trust on
```

Example:

```
# ip link set ens785f1 vf 0 trust on
```

Prerequisites

Before you configure a Citrix ADC VPX instance to use SR-IOV network interfaces, complete the following prerequisite tasks. See the NIC column for details about how to complete the corresponding tasks.

Task	Intel 82599 NIC	Intel X710, XL710, and X722 NICs
1. Add the NIC to the KVM host.	-	-
1. Download and install the latest Intel driver.	IXGBE driver	I40E driver
1. Blacklist the driver on the KVM host.	Add the following entry in the <code>/etc/modprobe.d/blacklist.conf</code> file: <code>blacklist ixgbevf</code> . Use IXGBE driver version 4.3.15 (recommended).	Add the following entry in the <code>/etc/modprobe.d/blacklist.conf</code> file: <code>blacklist i40evf</code> . Use i40e driver version 2.0.26 (recommended).
4. Enable SR-IOV Virtual Functions (VFs) on the KVM host. In both the commands in the next two columns: <code>number_of_VFs</code> = the number of Virtual VFs that you want to create. <code>device_name</code> = the interface name.	If you are using earlier version of kernel 3.8, then add the following entry to the <code>/etc/modprobe.d/ixgbe</code> file and restart the KVM host: <code>*options ixgbe max_vfs=*</code> . If you are using kernel 3.8 version or later, create VFs using the following command: <code>*echo > /sys/class/net//device/s-riov_numvfs*</code> . See example in figure 1.	If you are using earlier version of kernel 3.8, then add the following entry to the <code>/etc/modprobe.d/i40e.conf</code> file and restart the KVM host: <code>*options i40e max_vfs=*</code> . If you are using kernel 3.8 version or later, create VFs using the following command: <code>*echo > /sys/class/net//device/s-riov_numvfs*</code> . See example in figure 2.

Task	Intel 82599 NIC	Intel X710, XL710, and X722 NICs
1. Make the VFs persistent by adding the commands that you used to create VFs, to the rc.local file.	See example in figure 3.	See example in figure 3.

Important:
When you create the SR-IOV VFs, ensure that you do not assign MAC addresses to the VFs.

Figure 1: Enable SR-IOV VFs on the KVM host for Intel 82599 10G NIC.

```

root@ubuntu:/etc# echo 1 > /sys/class/net/ens3f0/device/sriov_numvfs
root@ubuntu:/etc# echo 1 > /sys/class/net/ens3f1/device/sriov_numvfs
root@ubuntu:/etc# lspci | grep 82599
02:00.0 Ethernet controller: Intel Corporation 82599ES 10-Gigabit SFI/SFP+ Network Connection (rev 01)
02:00.1 Ethernet controller: Intel Corporation 82599ES 10-Gigabit SFI/SFP+ Network Connection (rev 01)
02:10.0 Ethernet controller: Intel Corporation 82599 Ethernet Controller Virtual Function (rev 01)
02:10.1 Ethernet controller: Intel Corporation 82599 Ethernet Controller Virtual Function (rev 01)
root@ubuntu:/etc#
    
```

Figure 2: Enable SR-IOV VFs on the KVM host for Intel X710 10G and XL710 40G NICs.

```

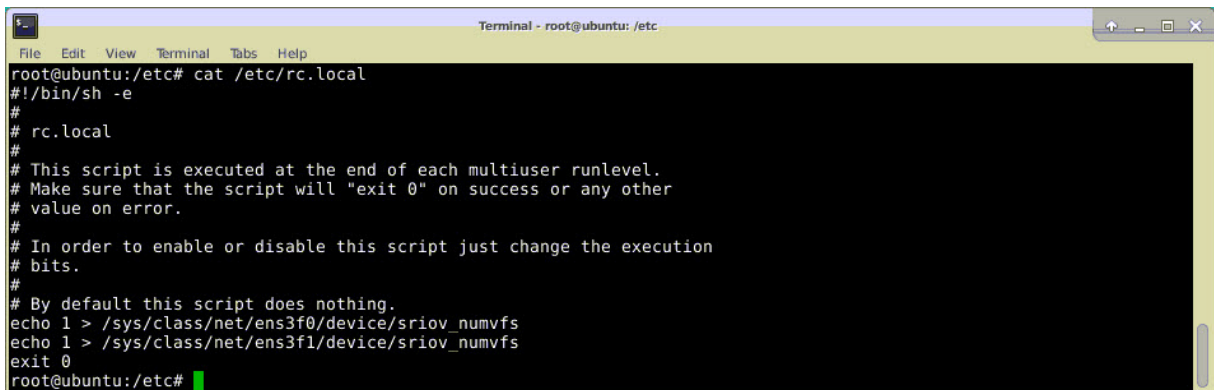
root@ubuntu:~# lspci | grep 710
03:00.0 Ethernet controller: Intel Corporation Ethernet Controller X710 for 10GbE SFP+ (rev 01)
03:00.1 Ethernet controller: Intel Corporation Ethernet Controller X710 for 10GbE SFP+ (rev 01)
03:00.2 Ethernet controller: Intel Corporation Ethernet Controller X710 for 10GbE SFP+ (rev 01)
03:00.3 Ethernet controller: Intel Corporation Ethernet Controller X710 for 10GbE SFP+ (rev 01)
03:06.0 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 01)
03:06.1 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 01)
03:0a.0 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 01)
03:0a.1 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 01)
03:0a.2 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 01)
03:0a.3 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 01)
03:0e.0 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 01)
03:0e.1 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 01)
03:0e.2 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 01)
03:0e.3 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 01)
81:00.0 Ethernet controller: Intel Corporation Ethernet Controller XL710 for 40GbE QSFP+ (rev 01)
82:00.0 Ethernet controller: Intel Corporation Ethernet Controller XL710 for 40GbE QSFP+ (rev 02)
82:00.1 Ethernet controller: Intel Corporation Ethernet Controller XL710 for 40GbE QSFP+ (rev 02)
82:02.0 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 02)
82:02.1 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 02)
82:0a.0 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 02)
82:0a.1 Ethernet controller: Intel Corporation XL710/X710 Virtual Function (rev 02)
root@ubuntu:~#
    
```

Figure 3: Enable SR-IOV VFs on the KVM host for Intel X722 10G NIC.

```

root@ubuntu:~# lspci | grep "37cd"
84:02.0 Ethernet controller: Intel Corporation Device 37cd (rev 04)
84:0a.0 Ethernet controller: Intel Corporation Device 37cd (rev 04)
    
```

Figure 4: Make the VFs persistent.

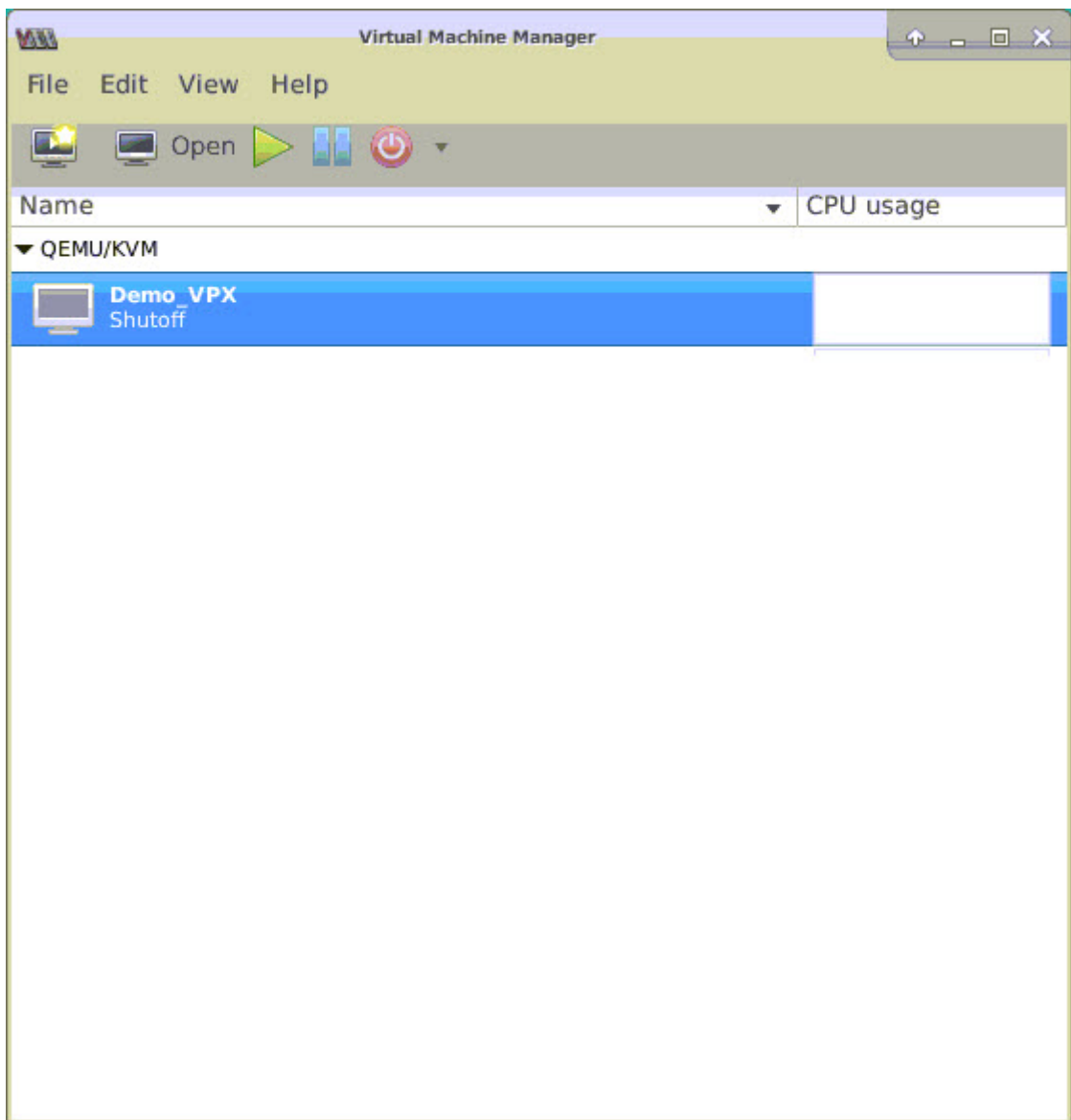
A terminal window titled "Terminal - root@ubuntu: /etc" showing the output of the command "cat /etc/rc.local". The output displays the standard rc.local script header and two echo commands that set the number of SR-IOV virtual functions (VFs) to 1 for network interfaces ens3f0 and ens3f1.

```
root@ubuntu:/etc# cat /etc/rc.local
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
echo 1 > /sys/class/net/ens3f0/device/sriov_numvfs
echo 1 > /sys/class/net/ens3f1/device/sriov_numvfs
exit 0
root@ubuntu:/etc#
```

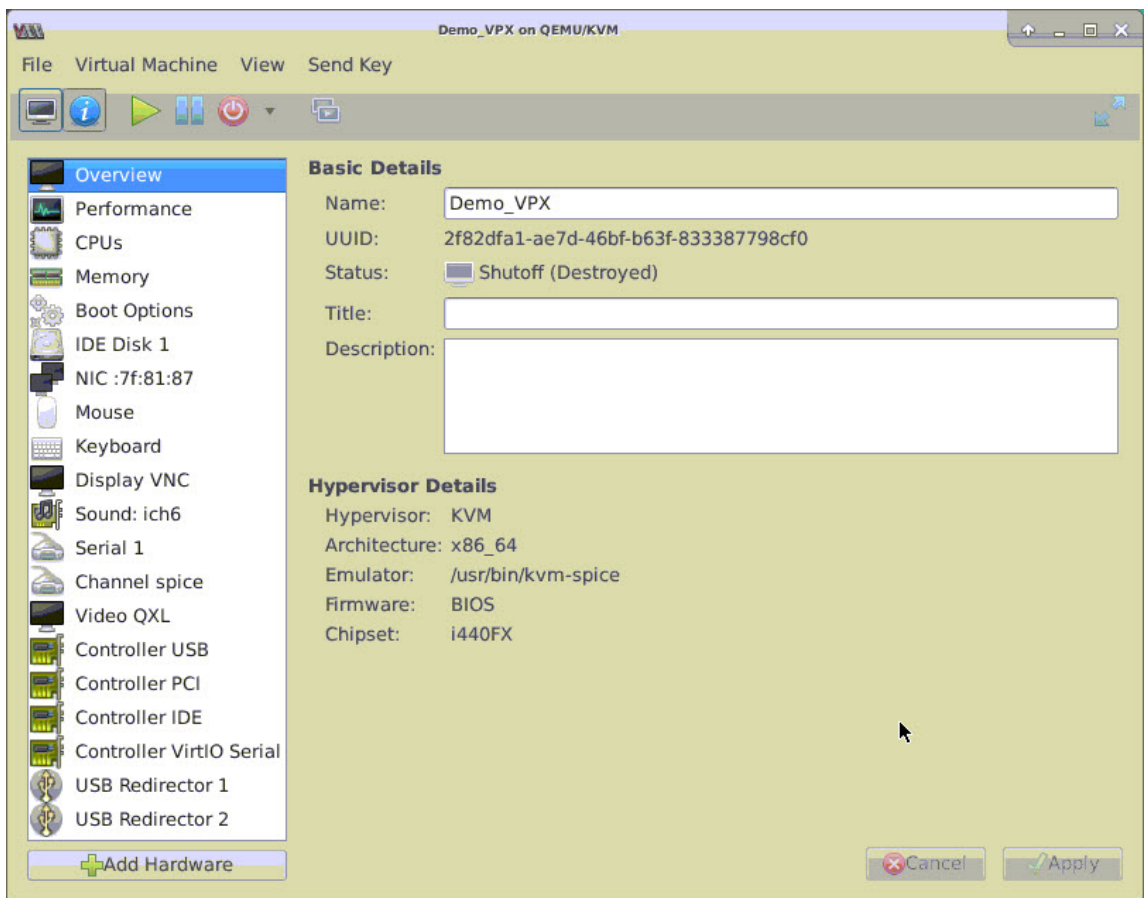
Configure a Citrix ADC VPX instance to use SR-IOV network interface

To configure Citrix ADC VPX instance to use SR-IOV network interface by using Virtual Machine Manager, complete these steps:

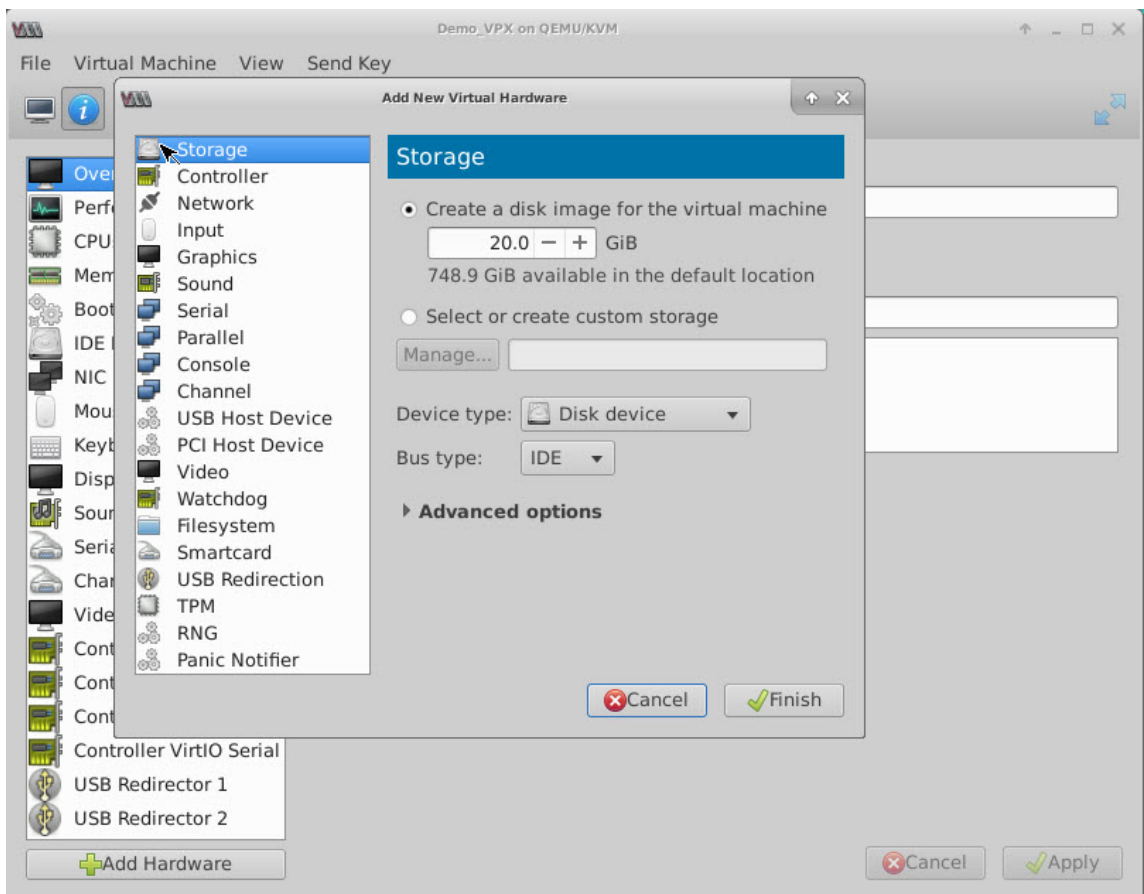
1. Power off the Citrix ADC VPX instance.
2. Select the Citrix ADC VPX instance and then select Open.



3. In the <virtual_machine on KVM> window, select the **i** icon.



4. Select **Add Hardware**.



5. In the **Add New Virtual Hardware** dialog box, do the following:
 - a) Select PCI Host Device.
 - b) In the Host Device section, select the VF you have created and click Finish.

Figure 4: VF for Intel 82599 10G NIC

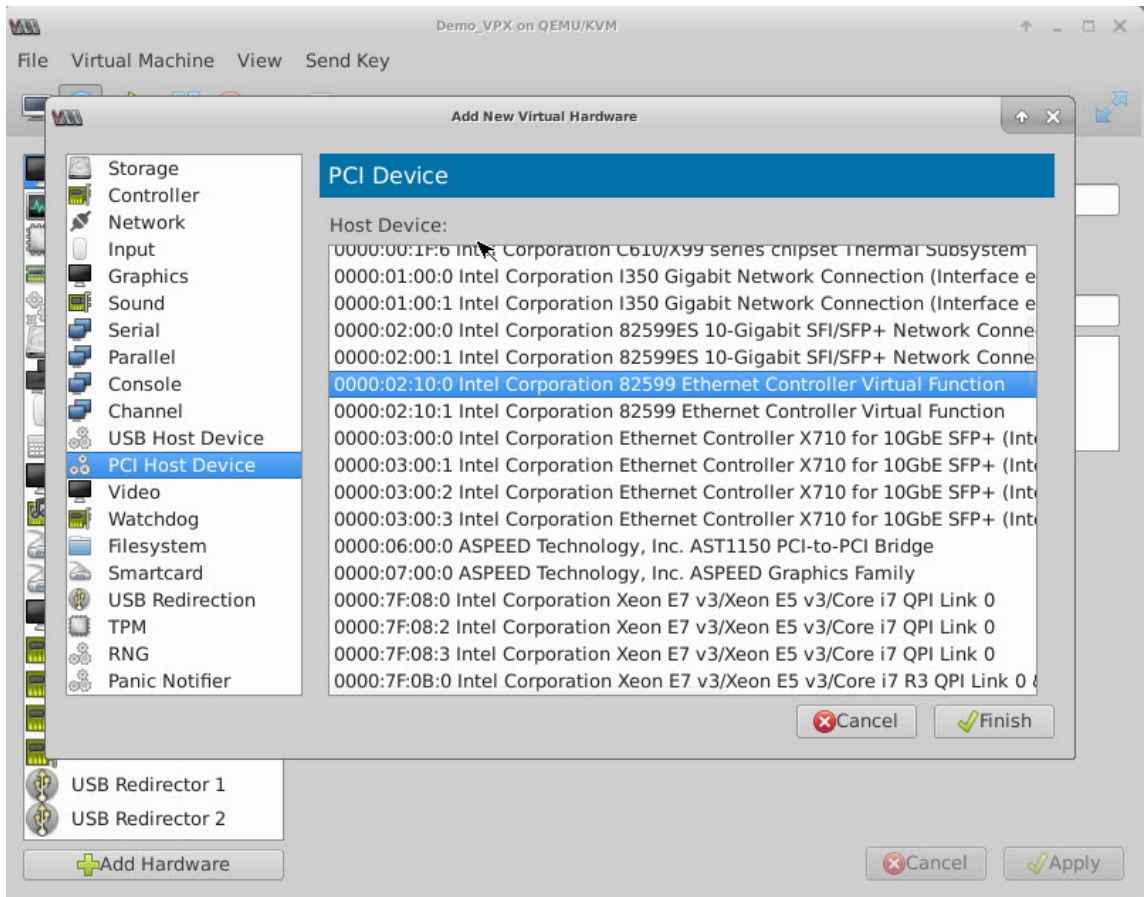


Figure 5: VF for Intel XL710 40G NIC

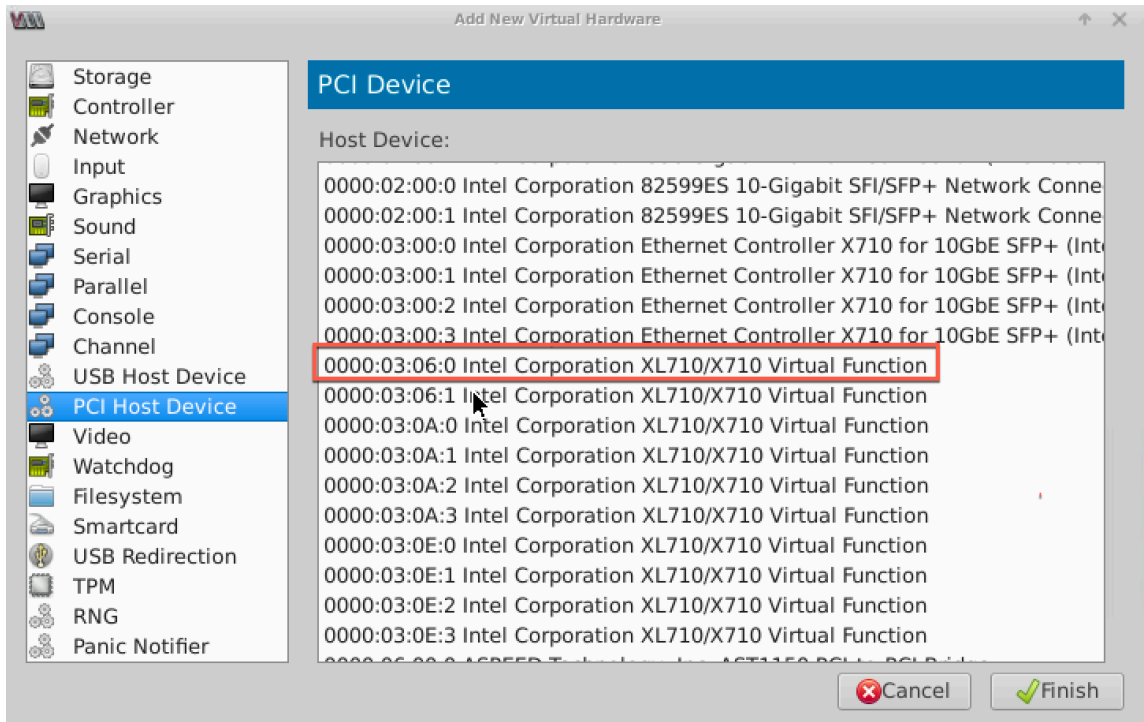
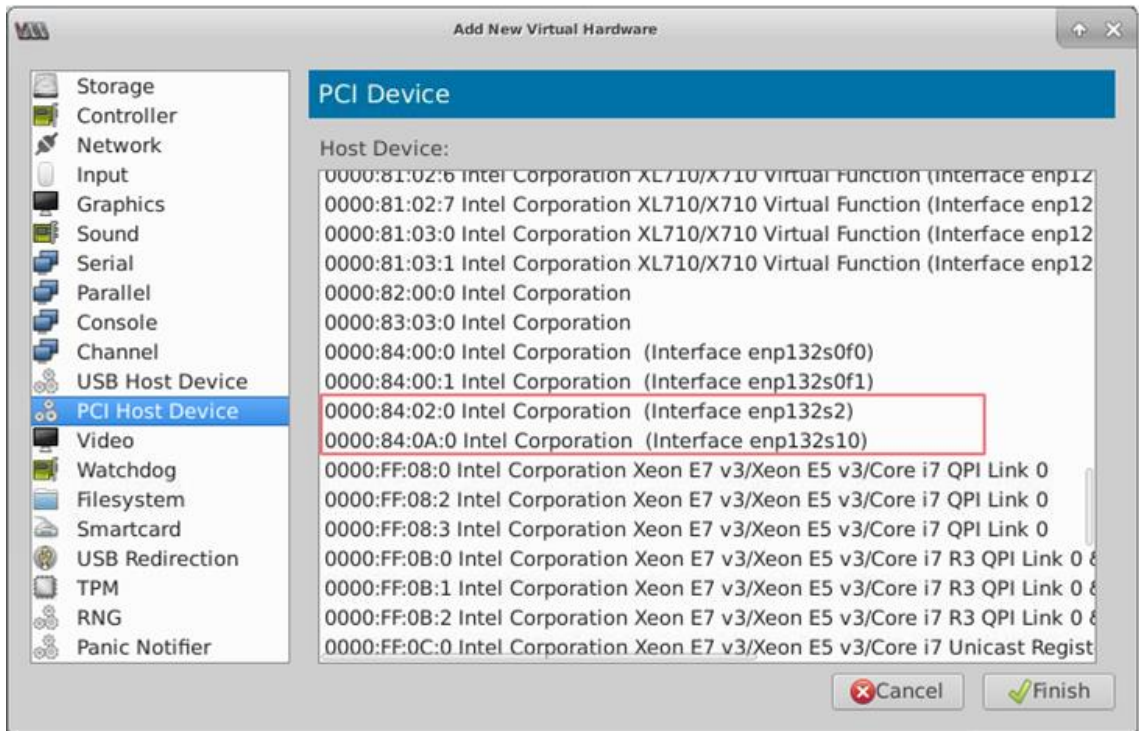


Figure 6: VF for Intel X722 10G NIC



6. Repeat Step 4 and 5 to add the VFs that you have created.
7. Power on the Citrix ADC VPX instance.
8. After the Citrix ADC VPX instance powers on, use the following command to verify the configuration:

```
1 show interface summary
```

The output shows all the interfaces that you configured.

Figure 6: output summary for Intel 82599 NIC.

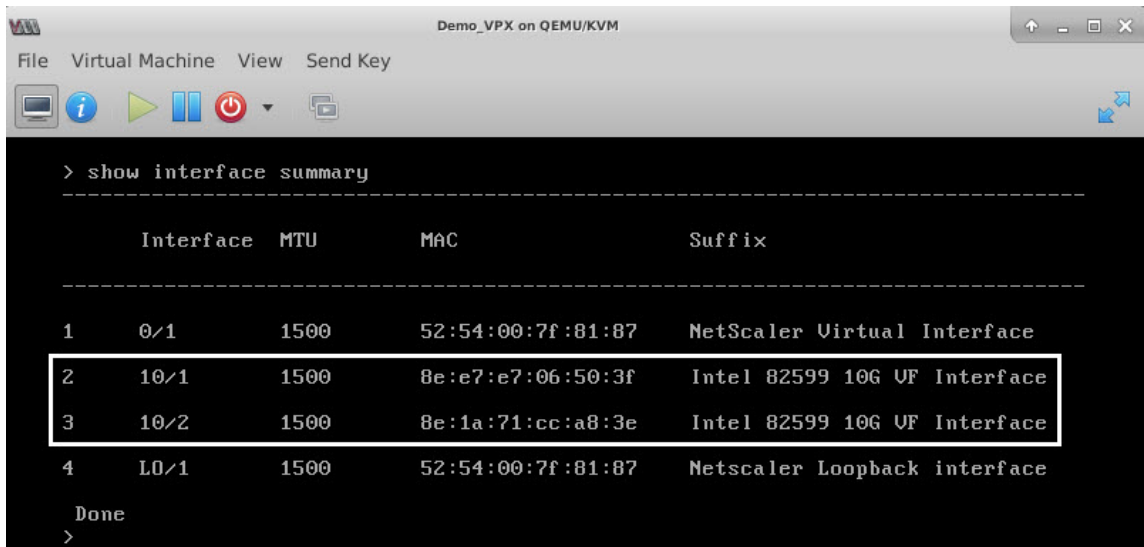
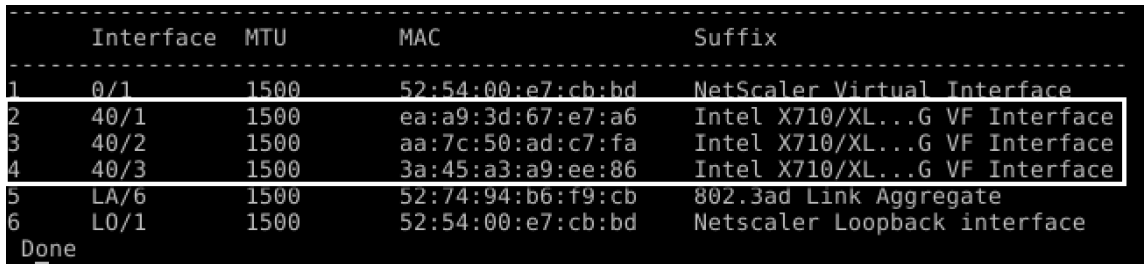


Figure 7. Output summary for Intel X710 and XL710 NICs.



Configure static LA/LACP on the SR-IOV interface

Important:

When you are creating the SR-IOV VFs, ensure that you do not assign MAC addresses to the VFs.

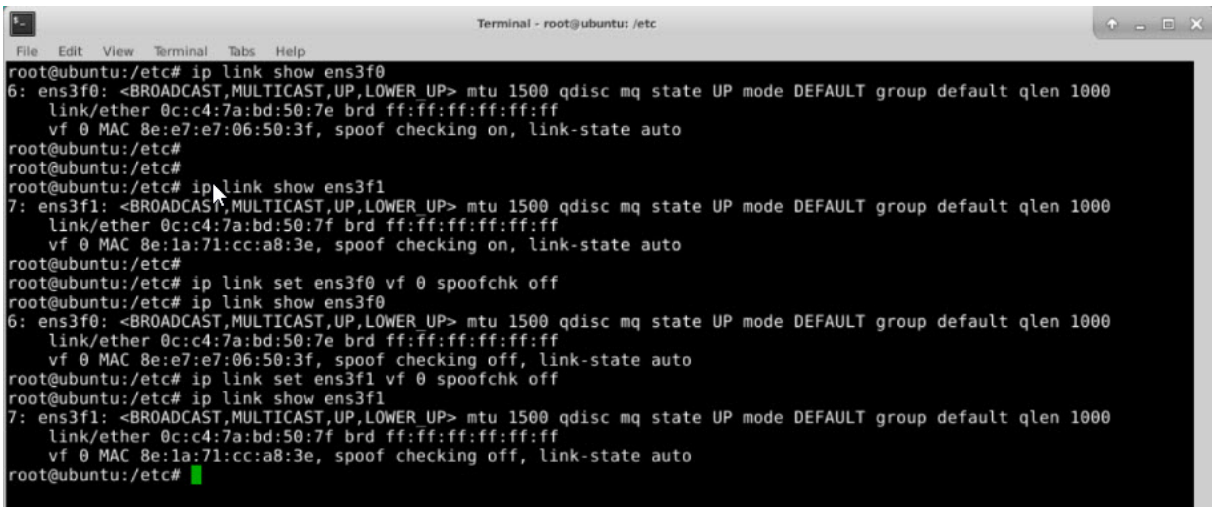
To use the SR-IOV VFs in link aggregation mode, disable spoof checking for VFs that you have created. On the KVM host, use the following command to disable spoof checking:

```
*ip link set \<interface\_name\> vf \<VF\_id\> spoofchk off*
```

Where:

- Interface_name –is the interface name.
- VF_id –is the Virtual Function id.

Example:



```
Terminal - root@ubuntu: /etc
File Edit View Terminal Tabs Help
root@ubuntu:/etc# ip link show ens3f0
6: ens3f0: <BROADCAST,MULTICAST,UP,LOWER UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000
    link/ether 0c:c4:7a:bd:50:7e brd ff:ff:ff:ff:ff:ff
    vf 0 MAC 8e:e7:e7:06:50:3f, spoof checking on, link-state auto
root@ubuntu:/etc#
root@ubuntu:/etc#
root@ubuntu:/etc# ip link show ens3f1
7: ens3f1: <BROADCAST,MULTICAST,UP,LOWER UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000
    link/ether 0c:c4:7a:bd:50:7f brd ff:ff:ff:ff:ff:ff
    vf 0 MAC 8e:1a:71:cc:a8:3e, spoof checking on, link-state auto
root@ubuntu:/etc#
root@ubuntu:/etc# ip link set ens3f0 vf 0 spoofchk off
root@ubuntu:/etc# ip link show ens3f0
6: ens3f0: <BROADCAST,MULTICAST,UP,LOWER UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000
    link/ether 0c:c4:7a:bd:50:7e brd ff:ff:ff:ff:ff:ff
    vf 0 MAC 8e:e7:e7:06:50:3f, spoof checking off, link-state auto
root@ubuntu:/etc# ip link set ens3f1 vf 0 spoofchk off
root@ubuntu:/etc# ip link show ens3f1
7: ens3f1: <BROADCAST,MULTICAST,UP,LOWER UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000
    link/ether 0c:c4:7a:bd:50:7f brd ff:ff:ff:ff:ff:ff
    vf 0 MAC 8e:1a:71:cc:a8:3e, spoof checking off, link-state auto
root@ubuntu:/etc#
```

After you disable spoof checking for all the VFs that you have created. Restart the Citrix ADC VPX instance and configure link aggregation. For detailed instructions, see [Configuring Link Aggregation](#).

Configuring VLAN on the SR-IOV Interface

You can configure VLAN on SR-IOV VFs. For detailed instructions, see [Configuring a VLAN](#).

Important:

Ensure that the KVM host does not contain VLAN settings for the VF interface.

Configure a Citrix ADC VPX instance to use PCI passthrough network interfaces

November 13, 2024

After you have installed and configured a Citrix ADC VPX instance on the Linux-KVM platform, you can use the Virtual Machine Manager to configure the virtual appliance to use PCI passthrough network interfaces.

Prerequisites

- The firmware version of the Intel XL710 Network Interface Card (NIC) on the KVM Host is 5.04.
- The KVM Host supports input–output memory management unit (IOMMU) and Intel VT-d, and they are enabled in the BIOS of the KVM Host. On the KVM Host, to enable IOMMU, add the

following entry to the **/boot/grub2/grub.cfg** file:

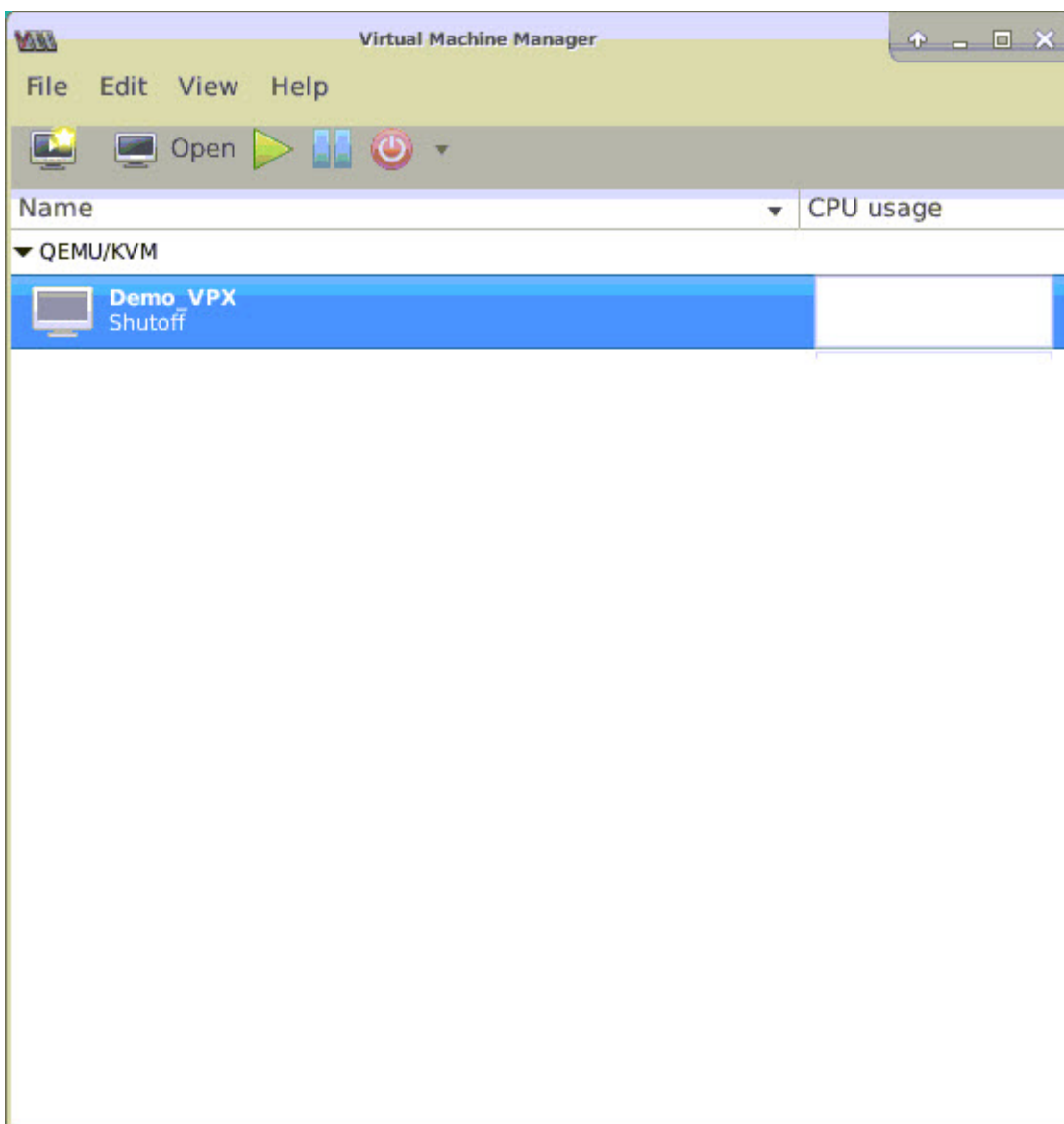
intel_iommu=1

- Execute the following command and reboot the KVM Host:

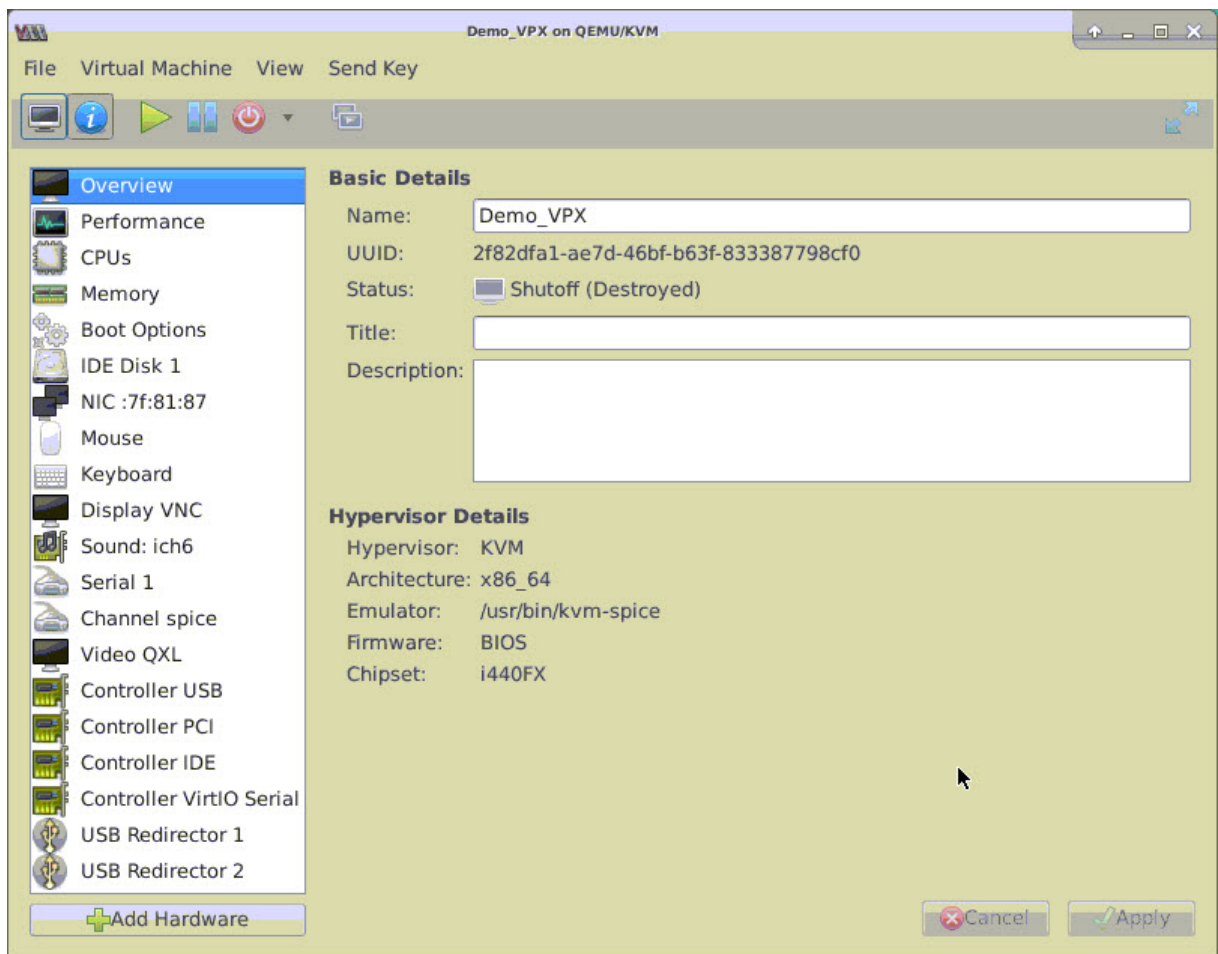
Grub2-mkconfig -o /boot/grub2/grub.cfg

To configure Citrix ADC VPX instances to use PCI passthrough network interfaces by using the Virtual Machine Manager:

1. Power off the Citrix ADC VPX instance.
2. Select the Citrix ADC VPX instance and click **Open**.



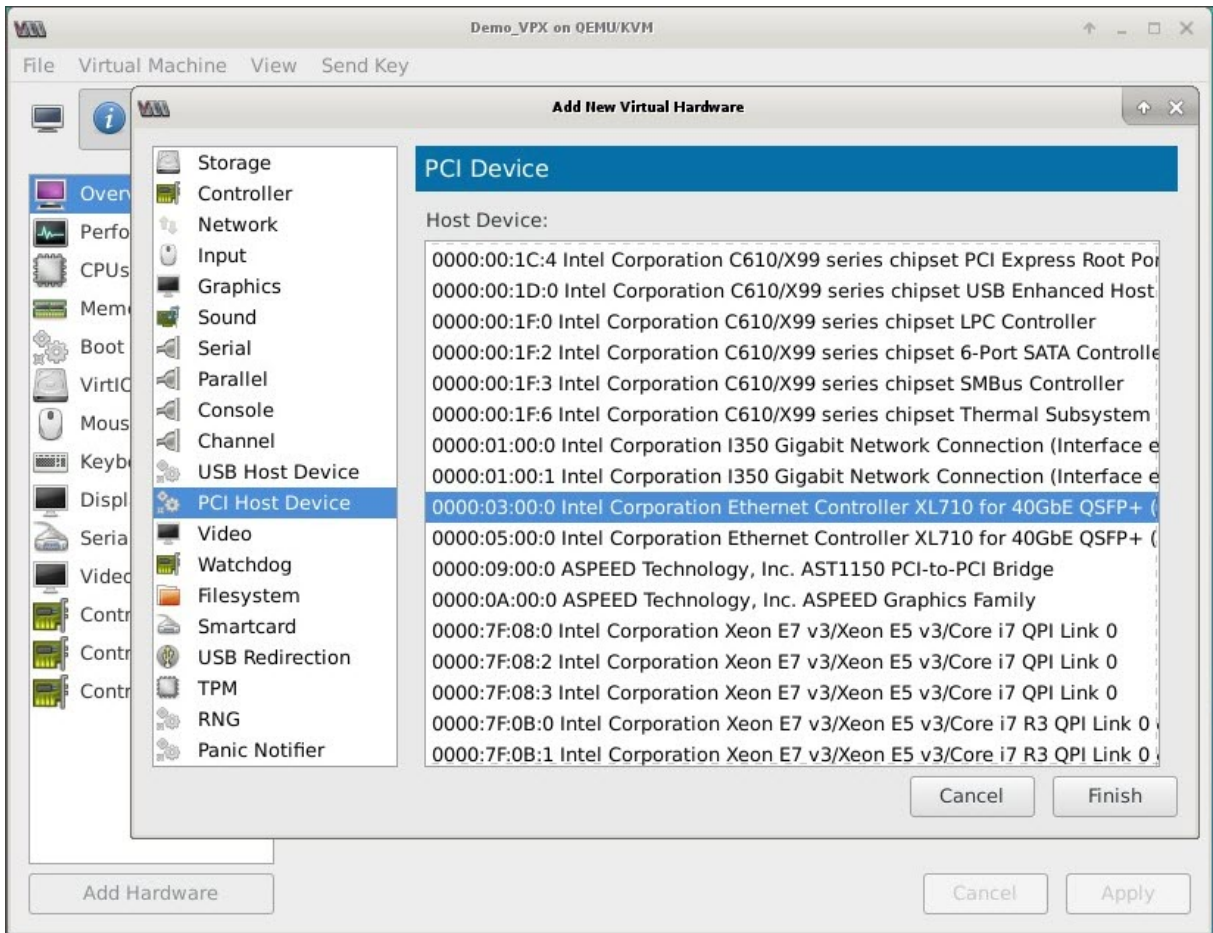
3. In the **<virtual_machine on KVM>** window, click the **i** icon.



4. Click **Add Hardware**.

5. In the **Add New Virtual Hardware** dialog box, do the following:

- a. Select **PCI Host Device**.
- b. In the **Host Device** section, select the Intel XL710 physical function.
- c. Click **Finish**.

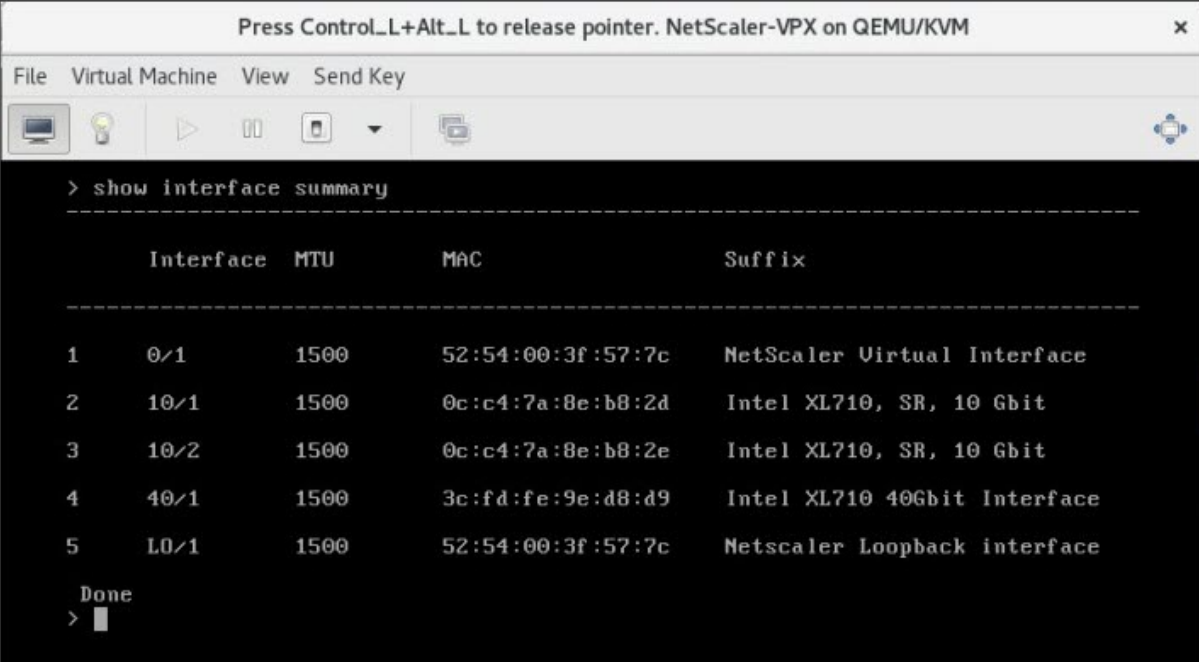


6. Repeat steps 4 and 5 to add any additional Intel XL710 physical functions.
7. Power on the Citrix ADC VPX instance.
8. Once the Citrix ADC VPX instance powers on, you can use the following command to verify the configuration:

```

COMMAND
> show interface summary
    
```

The output should show all the interfaces that you configured:



```

> show interface summary
-----
      Interface  MTU      MAC                               Suffix
-----
1      0/1        1500    52:54:00:3f:57:7c    NetScaler Virtual Interface
2      10/1        1500    0c:c4:7a:8e:b8:2d    Intel XL710, SR, 10 Gbit
3      10/2        1500    0c:c4:7a:8e:b8:2e    Intel XL710, SR, 10 Gbit
4      40/1        1500    3c:fd:fe:9e:d8:d9    Intel XL710 40Gbit Interface
5      L0/1        1500    52:54:00:3f:57:7c    Netscaler Loopback interface

Done
> █

```

Provision the Citrix ADC VPX instance by using the virsh program

November 13, 2024

The virsh program is a command line tool for managing VM Guests. Its functionality is similar to that of Virtual Machine Manager. It enables you to change a VM Guest's status (start, stop, pause, and so on), to set up new Guests and devices, and to edit existing configurations. The virsh program is also useful for scripting VM Guest management operations.

To provision Citrix ADC VPX by using the virsh program, follow these steps:

1. Use the tar command to untar the the Citrix ADC VPX package. The NSVPX-KVM-*_nc.tgz package contains following components:
 - The Domain XML file specifying VPX attributes [NSVPX-KVM-*_nc.xml]
 - Check sum of NS-VM Disk Image [Checksum.txt]
 - NS-VM Disk Image [NSVPX-KVM-*_nc.raw]

Example:

```

1 tar -xvzf NSVPX-KVM-10.1-117_nc.tgz
2 NSVPX-KVM-10.1-117_nc.xml
3 NSVPX-KVM-10.1-117_nc.raw
4 checksum.txt

```

- Copy the NSVPX-KVM-*_nc.xml XML file to a file named <DomainName>-NSVPX-KVM-*_nc.xml. The <DomainName> is also the name of the virtual machine. Example:

```
1 cp NSVPX-KVM-10.1-117_nc.xml NetScaler-VPX-NSVPX-KVM-10.1-117_nc.xml
```

- Edit the <DomainName>-NSVPX-KVM-*_nc.xml file to specify the following parameters:

- name—Specify the name.
- mac—Specify the MAC address.
Note: The domain name and the MAC address have to be unique.
- sourcefile—Specify the absolute disk-image source path. The file path has to be absolute. You can specify the path of the RAW image file or a QCOW2 image file.
If you want to specify a RAW image file, specify the disk image source path as shown in the following example:

Example:

```
1 *      \<name\>NetScaler-VPX\</name\>
2          \<mac address='52:54:00:29:74:b3' /\>
3          \<source file='/root/NSVPX-KVM-10.1-117\_nc.raw'
          /\>*
```

Specify the absolute QCOW2 disk-image source path and define the driver type as **qcow2**, as shown in the following example:

Example:

```
1 *      \<name\>NetScaler-VPX\</name\>
2          \<mac address='52:54:00:29:74:b3' /\>
3          \<driver name='qemu' type='qcow2' /\>
4          \<source file='/root/NSVPX-KVM-10.1-117\_nc.qcow'
          /\>*
```

- Edit the \<DomainName\>-NSVPX-KVM-*_nc.xml file to configure the networking details:

- source dev—specify the interface.
- mode—specify the mode. The default interface is **Macvtap Bridge**.

Example: Mode: MacVTap Bridge Set target interface as ethx and mode as bridge Model type as virtio

```
1 <interface type='direct'>
2   <mac address='52:54:00:29:74:b3' /\>
3   <source dev='eth0' mode='bridge' /\>
4   <target dev='macvtap0' /\>
5   <model type='virtio' /\>
6   <alias name='net0' /\>
```



```

7     <address type='pci' domain='0x0000' bus='0x00' slot='0x03'
      function='0x0' />
8     </interface>

```

Here, eth0 is the physical interface attached to the VM.

2. Define the VM attributes in the <DomainName>-NSVPX-KVM-*_nc.xml file by using the following command: `virsh define <DomainName>-NSVPX-KVM-*_nc.xml` Example:

```
1 virsh define NS-VPX-NSVPX-KVM-10.1-117_nc.xml
```

Start the VM by entering following command:

] Example

virsh start [

```
1 virsh start NetScaler-VPX
```

Connect the Guest VM through
the console `virsh console` [

] Example

```

2
1 virsh console NetScaler-VPX

```

Add additional interfaces to Citrix ADC VPX instance using virsh program

After you have provisioned the Citrix ADC VPX on KVM, you can add additional interfaces.

To add additional interfaces, follow these steps:

1. Shut down the Citrix ADC VPX instance running on the KVM.

Edit the -NSVPX-KVM-*_nc.xml file using the
command: `virsh edit` [

2.

3. In the <DomainName>-NSVPX-KVM-*_nc.xml file, append the following parameters:

a) For MacVTap

- Interface type—Specify the interface type as 'direct'.
- Mac address—Specify the Mac address and make sure the MAC address is unique across the interfaces.
- source dev—Specify the interface name.

- mode—Specify the mode; the modes supported are - Bridge, VEPA, Private, and Pass-through
- model type—Specify the model type as virtio

Example:

Mode: MacVTap Pass-through

Set target interface as

ethx, Mode as

bridge, and model type as

virtio

```
1 <interface type='direct'>
2   <mac address='52:54:00:29:74:b3' />
3   <source dev='eth1' mode='passthrough' />
4   <model type='virtio' />
5 </interface>
```

Here eth1 is the physical interface attached to the VM.

b) For Bridge Mode

Note: Make sure that you have configured a Linux bridge in the KVM host, bound the physical interface to the bridge, and put the bridge in the UP state.

- Interface type—Specify the interface type as 'bridge'.
- Mac address—Specify the Mac address and make sure the MAC address is unique across the interfaces.
- source bridge—Specify the bridge name.
- model type—Specify the model type as virtio

Example: Bridge Mode

```
1 <interface type='bridge'>
2   <mac address='52:54:00:2d:43:a4' />
3   <source bridge='br0' />
4   <model type='virtio' />
5 </interface>
```

Manage the Citrix ADC VPX guest VMs

November 13, 2024

You can use the Virtual Machine Manager and the virsh program to perform management tasks such as starting or stopping a VM Guest, setting up new guests and devices, editing existing configurations, and connecting to the graphical console through Virtual Network Computing (VNC).

Manage the VPX guest VMs by using Virtual Machine Manager

- List the VM guests

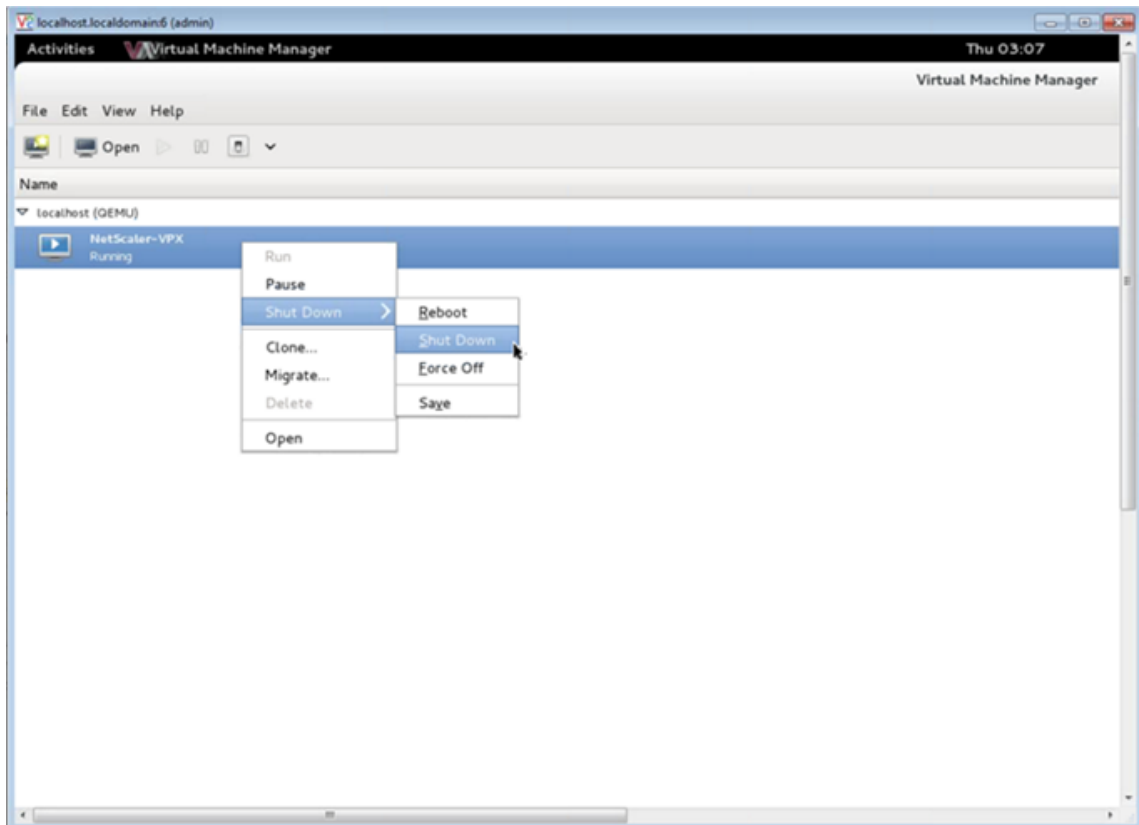
The main Window of the Virtual Machine Manager displays a list of all the VM Guests for each VM host server it is connected to. Each VM Guest entry contains the virtual machine's name, along with its status (Running, Paused, or Shutoff) displayed as icon.

- Open a graphical console

Opening a Graphical Console to a VM Guest enables you to interact with the machine like you would with a physical host through a VNC connection. To open the graphical console in the Virtual Machine Manager, right-click the VM Guest entry and select the Open option from the pop-up menu.

- Start and shut down a guest

You can start or stop a VM Guest from the Virtual Machine Manager. To change the state of the VM, right-click the VM Guest entry and select Run or one of the Shut Down options from the pop-up menu.



- Reboot a guest

You can reboot a VM Guest from the Virtual Machine Manager. To reboot the VM, right-click the VM Guest entry, and then select Shut Down > Reboot from the pop-up menu.

- Delete a guest

Deleting a VM Guest removes its XML configuration by default. You can also delete a guest's storage files. Doing so completely erases the guest.

1. In the Virtual Machine Manager, right-click the VM Guest entry.
2. Select Delete from the pop-up menu. A confirmation window opens.

Note:

The **Delete** option is enabled only when the VM Guest is shut down.

3. Click **Delete**.
4. To completely erase the guest, delete the associated .raw file by selecting the Delete Associated Storage Files check box.

Manage the Citrix ADC VPX guest VMs using the virsh program

- List the VM Guests and their current states.

To use virsh to display information about the Guests

```
virsh list --all
```

The command output displays all domains with their states.

Example output:

1	Id	Name	State
2	-----		
3	0	Domain-0	running
4	1	Domain-1	paused
5	2	Domain-2	inactive
6	3	Domain-3	crashed

- Open a virsh console.

Connect the Guest VM through the console

```
virsh console [<DomainID> | <DomainName> | <DomainUUID>]
```

Example:

```
virsh console NetScaler-VPX
```

- Start and shut down a guest.

Guests can be started using the DomainName or Domain-UUID.

```
virsh start [<DomainName> | <DomainUUID>]
```

Example

```
virsh start NetScaler-VPX
```

To shut down a guest:

```
virsh shutdown [<DomainID> | <DomainName> | <DomainUUID>]
```

Example:

```
virsh shutdown NetScaler-VPX
```

- Reboot a guest

```
virsh reboot [<DomainID> | <DomainName> | <DomainUUID>]
```

Example:

```
virsh reboot NetScaler-VPX
```

Delete a guest

To delete a Guest VM you need to shut-down the Guest and un-define the <DomainName>-NSVPX-KVM-*_nc.xml before you run the delete command.

```
1 virsh shutdown [<DomainID> | <DomainName> | <DomainUUID>]
2 virsh undefine [<DomainName> | <DomainUUID>]
```

Example:

```
1 virsh shutdown NetScaler-VPX
2 virsh undefine NetScaler-VPX
```

Note:

The delete command doesn't remove disk image file which needs to be removed manually.

Provision the Citrix ADC VPX instance with SR-IOV, on OpenStack

November 13, 2024

You can deploy high-performance Citrix ADC VPX instances that use single-root I/O virtualization (SR-IOV) technology, on OpenStack.

You can deploy a Citrix ADC VPX instance that uses SR-IOV technology, on OpenStack, in three steps:

- Enable SR-IOV Virtual Functions (VFs) on the host.
- Configure and make the VFs available to OpenStack.
- Provision the Citrix ADC VPX on OpenStack.

Prerequisites

Ensure that you:

- Add the Intel 82599 Network Interface Card (NIC) to the host.
- Download and Install the latest IXGBE driver from Intel.
- Blacklist the IXGBEVF driver on the host. Add the following entry in the `/etc/mod-probe.d/blacklist.conf` file: `blacklist ixgbevf`

Note:

The ixgbe driver version should be minimum 5.0.4.

Enable SR-IOV VFs on the host

Do one of the following steps to enable SR-IOV VFs:

- If you are using a kernel version earlier than 3.8, add the following entry to the `/etc/mod-probe.d/ixgbe` file and restart the host: `options ixgbe max_vfs=<number_of_VFs>`
- If you are using kernel 3.8 version or later, create VFs by using the following command:

```
1 echo <number_of_VFs> > /sys/class/net/<device_name>/device/sriov_numvfs
```

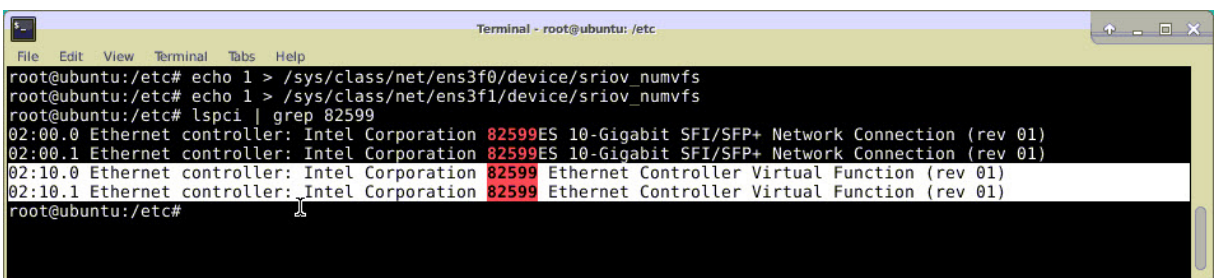
Where:

- `number_of_VFs` is the number of Virtual Functions that you want to create.
- `device_name` is the interface name.

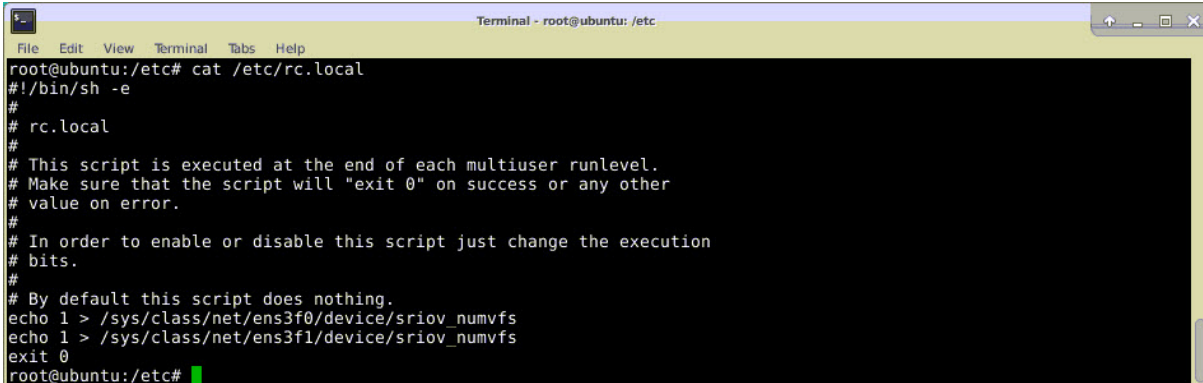
Important:

While you are creating the SR-IOV VFs, make sure that you do not assign MAC addresses to the VFs.

Here is an example of four VFs being created.



Make the VFs persistent, add the commands that you used to created VFs to the **rc.local** file. Here is an example showing contents of rc.local file.

A terminal window titled "Terminal - root@ubuntu: /etc" showing the output of the command "cat /etc/rc.local". The output is a shell script for rc.local with the following content:

```
root@ubuntu:/etc# cat /etc/rc.local
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
echo 1 > /sys/class/net/ens3f0/device/sriov_numvfs
echo 1 > /sys/class/net/ens3f1/device/sriov_numvfs
exit 0
root@ubuntu:/etc#
```

For more information, see this [Intel SR-IOV Configuration Guide](#).

Configure and make the VFs available to OpenStack

Follow the steps given at the link below to configure SR-IOV on OpenStack: <https://wiki.openstack.org/wiki/SR-IOV-Passthrough-For-Networking>.

Provision the Citrix ADC VPX instance on OpenStack

You can provision a Citrix ADC VPX instance in an OpenStack environment by using the OpenStack CLI.

Provisioning a VPX instance, optionally involves using data from the config drive. The config drive is a special configuration drive that attaches to the instance when it boots. This configuration drive can be used to pass networking configuration information such as management IP address, network mask, and default gateway etc. to the instance before you configure the network settings for the instance.

When OpenStack provisions a VPX instance, it first detects that the instance is booting in an OpenStack environment, by reading a specific BIOS string (OpenStack Foundation) that indicates OpenStack. For Red Hat Linux distributions, the string is stored in `/etc/nova/release`. This is a standard mechanism that is available in all OpenStack implementations based on KVM hyper-visor platform. The drive should have a specific OpenStack label. If the config drive is detected, the instance attempts to read the following information from the file name specified in the nova boot command. In the procedures below, the file is called “`userdata.txt`.”

- Management IP address
- Network mask
- Default gateway

Once the parameters are successfully read, they are populated in the NetScaler stack. This helps in managing the instance remotely. If the parameters are not read successfully or the config drive is not available, the instance transitions to the default behavior, which is:

- The instance attempts to retrieve the IP address information from DHCP.
- If DHCP fails or times-out, the instance comes up with default network configuration (192.168.100.1/16).

Provision the Citrix ADC VPX instance on OpenStack through CLI

You can provision a VPX instance in an OpenStack environment by using the OpenStack CLI. Here's the summary of the steps to provision a Citrix ADC VPX instance on OpenStack:

1. Extracting the .qcow2 file from the .tgz file
2. Building an OpenStack image from the qcow2 image
3. Provisioning a VPX instance

To provision a VPX instance in an OpenStack environment, do the following steps.

1. Extract the .qcow2 file from the .tgz file by typing the command:

```
1 tar xvzf <TAR file>
2
3 tar xvzf NSVPX-KVM-12.0-26.2_nc.tgz
4 NSVPX-KVM.xml
5 NSVPX-KVM-12.0-26.2_nc.qcow2
```

2. Build an OpenStack image using the .qcow2 file extracted in step 1 by typing the following command:

```
1 glance image-create --name="<name of the OpenStack image>" --
2 property hw_disk_bus=ide --is-public=
3 true --container-format=bare --disk-format=qcow2< <name of the
4 qcow2 file>
5
6 glance image-create --name="NS-VPX-12-0-26-2" --property
7 hw_disk_bus=ide --is-public=
8 true --container-format=bare --disk-format=qcow2< NSVPX-KVM-12.0-
9 26.2_nc.qcow2
```

The following illustration provides a sample output for the `glance image-create` command.

Property	Value
checksum	735dae4ea6e46e39ed3f0acfba02e755
container_format	bare
created_at	2017-02-16T10:03:29Z
disk_format	qcow2
hw_disk_bus	ide
id	aeaa13e9-b49b-411c-ab54-c61820a8e2f3
min_disk	0
min_ram	0
name	NSVPX-KVM-12.0-26.2
owner	06c41a73b32f4b48af55359fd7d3502c
protected	False
size	717946880
status	active
tags	[]
updated_at	2017-02-16T10:03:38Z
virtual_size	None
visibility	private

3. After an OpenStack image is created, provision the Citrix ADC VPX instance instance.

```

1 nova boot --image NSVPX-KVM-12.0-26.2 --config-drive=true --
  userdata
2 ./userdata.txt --flavor m1.medium --nic net-id=3b258725-eaae-
3 455e-a5de-371d6d1f349f --nic port-id=218ba819-9f55-4991-adb6-
4 02086a6bdee2 NSVPX-10

```

In the above command, userdata.txt is the file which contains the details like, IP address, netmask, and default gateway for the VPX instance. The userdata file is a user customizable file. NSVPX-KVM-12.0-26.2 is the name of the virtual appliance that you want to provision. --nic port-id=218ba819-9f55-4991-adb6-02086a6bdee2 is the OpenStack VF.

The following illustration gives a sample output of the nova boot command.

Property	Value
OS-DCF:diskConfig	MANUAL
OS-EXT-AZ:availability_zone	
OS-EXT-SRV-ATTR:host	-
OS-EXT-SRV-ATTR:hypervisor_hostname	-
OS-EXT-SRV-ATTR:instance_name	instance-0000003c
OS-EXT-STS:power_state	0
OS-EXT-STS:task_state	scheduling
OS-EXT-STS:vm_state	building
OS-SRV-USG:launched_at	-
OS-SRV-USG:terminated_at	-
accessIPv4	
accessIPv6	
adminPass	43EjPdM5shLz
config_drive	True
created	2017-02-20T11:53:37Z
flavor	m1.medium (3)
hostId	
id	6b9f6968-aab9-463c-b619-d58c73db3187
image	NSVPX-KVM-12.0-26.2 (a5478b8a-8435-48d1-b4a0-1494e2c8f8b1)
key_name	-
metadata	{}
name	NSVPX-10
os-extended-volumes:volumes_attached	[]
progress	0
security_groups	default
status	BUILD
tenant_id	06c41a73b32f4b48af55359fd7d3502c
updated	2017-02-20T11:53:38Z
user_id	418524f7101b4f0389ecbb36da9916b5

The following illustration shows a sample of the userdata.txt file. The values within the <PropertySection></PropertySection> tags are the values which is user configurable and holds the information like, IP address, netmask, and default gateway.

```

1    <?xml version="1.0" encoding="UTF-8" standalone="no"?>
2    <Environment xmlns:oe="http://schemas.dmtf.org/ovf/environment/1"
3    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4    oe:id=""
5    xmlns="http://schemas.dmtf.org/ovf/environment/1">
6    <PlatformSection>
7    <Kind>NOVA</Kind>
8    <Version>2013.1</Version>
9    <Vendor>Openstack</Vendor>
10   <Locale>en</Locale>
11   </PlatformSection>
12   <PropertySection>
13   <Property oe:key="com.citrix.netscaler.ovf.version" oe:value="1.0"
14   />
15   <Property oe:key="com.citrix.netscaler.platform" oe:value="vpx"/>
16   citrix.com 4
17   <Property oe:key="com.citrix.netscaler.orch_env"
18   oe:value="openstack-orch-env"/>
19   <Property oe:key="com.citrix.netscaler.mgmt.ip"
20   oe:value="10.1.0.100"/>
21   <Property oe:key="com.citrix.netscaler.mgmt.netmask"
22   oe:value="255.255.0.0"/>

```

```
22 <Property oe:key="com.citrix.netscaler.mgmt.gateway"  
23 oe:value="10.1.0.1"/>  
24 </PropertySection>  
25 </Environment>
```

Additional supported Configurations: Creating and Deleting VLANs on SR-IOV VFs from the Host

Type the following command to create a VLAN on the SR-IOV VF:

```
1 ip link show enp8s0f0 vf 6 vlan 10
```

In the above command “enp8s0f0” is the name of the physical function.

Example: vlan 10, created on vf 6

```
4: enp8s0f0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT qlen 1000  
link/ether 00:1b:21:7b:d7:88 brd ff:ff:ff:ff:ff:ff  
vf 0 MAC 00:00:00:00:00:00, spoof checking on, link-state auto, trust off  
vf 1 MAC 00:00:00:00:00:00, spoof checking on, link-state auto, trust off  
vf 2 MAC 00:00:00:00:00:00, spoof checking on, link-state auto, trust off  
vf 3 MAC fa:16:3e:1e:0b:ee, spoof checking on, link-state auto, trust off  
vf 4 MAC fa:16:3e:0d:05:62, spoof checking on, link-state auto, trust off  
vf 5 MAC 5e:46:0d:79:de:f8, spoof checking on, link-state auto, trust off  
vf 6 MAC fa:16:3e:db:ea:b3, vlan 10, spoof checking on, link-state auto, trust off  
vf 7 MAC 00:00:00:00:00:00, spoof checking on, link-state auto, trust off
```

Type the following command to delete a VLAN on the SR-IOV VF:

```
1 ip link show enp8s0f0 vf 6 vlan 0
```

Example: vlan 10, removed from vf 6

```
[root@localhost ~]# ip link show enp8s0f0  
4: enp8s0f0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT qlen 1000  
link/ether 00:1b:21:7b:d7:88 brd ff:ff:ff:ff:ff:ff  
vf 0 MAC 00:00:00:00:00:00, spoof checking on, link-state auto, trust off  
vf 1 MAC 00:00:00:00:00:00, spoof checking on, link-state auto, trust off  
vf 2 MAC 00:00:00:00:00:00, spoof checking on, link-state auto, trust off  
vf 3 MAC fa:16:3e:1e:0b:ee, spoof checking on, link-state auto, trust off  
vf 4 MAC fa:16:3e:0d:05:62, spoof checking on, link-state auto, trust off  
vf 5 MAC 5e:46:0d:79:de:f8, spoof checking on, link-state auto, trust off  
vf 6 MAC fa:16:3e:db:ea:b3, spoof checking on, link-state auto, trust off  
vf 7 MAC 00:00:00:00:00:00, spoof checking on, link-state auto, trust off
```

These steps complete the procedure for deploying a Citrix ADC VPX instance that uses SRIOV technology, on OpenStack.

Configure a Citrix ADC VPX instance on KVM to use OVS DPDK-based host interfaces

November 12, 2024

You can configure a Citrix ADC VPX instance running on KVM (Fedora and RHOS) to use Open vSwitch (OVS) with Data Plane Development Kit (DPDK) for better network performance. This document describes how to configure the Citrix ADC VPX instance to operate on the vhost-user ports exposed by OVS-DPDK on KVM host.

[OVS](#) is a multilayer virtual switch licensed under the open-source Apache 2.0 license. [DPDK](#) is a set of libraries and drivers for fast packet processing.

The following Fedora, RHOS, OVS, and DPDK versions are qualified for configuring a Citrix ADC VPX instance:

Fedora	RHOS
Fedora 25	RHOS 7.4
OVS 2.7.0	OVS 2.6.1
DPDK 16.11.12	DPDK 16.11.12

Prerequisites

Before you install DPDK, make sure the host has 1 GB hugepages.

For more information, see this [DPDK system requirements documentation](#).

Here is the summary of the steps required to configuring a Citrix ADC VPX Instance on KVM to use OVS DPDK-based host interfaces:

- Install DPDK.
- Build and Install OVS.
- Create an OVS bridge.
- Attach a physical interface to the OVS bridge.
- Attach vhost-user ports to the OVS data path.
- Provision a KVM-VPX with OVS-DPDK based vhost-user ports.

Install DPDK

To install DPDK, follow the instruction given at this [Open vSwitch with DPDK](#) document .

Build and install OVS

Download OVS from the OVS [download page](#). Next, build and install OVS by using a DPDK datapath. Follow the instructions given in the [Installing Open vSwitch](#) document.

For more detailed information, [DPDK Getting Started Guide for Linux](#).

Create an OVS bridge

Depending on your need, type the Fedora or RHOS command to create an OVS bridge:

Fedora command:

```
1 > $OVS_DIR/utilities/ovs-vsctl add-br ovs-br0 -- set bridge ovs-br0
   datapath_type=netdev
```

RHOS command:

```
1 ovs-vsctl add-br ovs-br0 -- set bridge ovs-br0 datapath_type=netdev
```

Attach physical interface to the OVS bridge

Bind the ports to DPDK and then attach them to the OVS bridge by typing the following Fedora or RHOS commands:

Fedora command:

```
1 > $OVS_DIR/utilities/ovs-vsctl add-port ovs-br0 dpdk0 -- set Interface
   dpdk0 type=dtpk options:dtpk-devargs=0000:03:00.0
2
3 > $OVS_DIR/utilities/ovs-vsctl add-port ovs-br0 dpdk1 -- set Interface
   dpdk1 type=dtpk options:dtpk-devargs=0000:03:00.1
```

RHOS command:

```
1 ovs-vsctl add-port ovs-br0 dpdk0 -- set Interface dpdk0 type=dtpk
   options:dtpk-devargs=0000:03:00.0
2
3
4 ovs-vsctl add-port ovs-br0 dpdk1 -- set Interface dpdk1 type=dtpk
   options:dtpk-devargs=0000:03:00.1
```

The `dpdk-devargs` shown as part of `options` specifies the PCI BDF of the respective physical NIC.

Attach vhost-user ports to the OVS data path

Type the following Fedora or RHOS commands to attach vhost-user ports to the OVS data path:

Fedora command:

```

1 > $OVS_DIR/utilities/ovs-vsctl add-port ovs-br0 vhost-user1 -- set
  Interface vhost-user1 type=dpdkvhostuser -- set Interface vhost-
  user1 mtu_request=9000
2
3 > $OVS_DIR/utilities/ovs-vsctl add-port ovs-br0 vhost-user2 -- set
  Interface vhost-user2 type=dpdkvhostuser -- set Interface vhost-
  user2 mtu_request=9000
4
5 chmod g+w /usr/local/var/run/openvswitch/vhost*

```

RHOS command:

```

1 ovs-vsctl add-port ovs-br0 vhost-user1 -- set Interface vhost-user1
  type=dpdkvhostuser -- set Interface vhost-user1 mtu_request=9000
2
3 ovs-vsctl add-port ovs-br0 vhost-user2 -- set Interface vhost-user2
  type=dpdkvhostuser -- set Interface vhost-user2 mtu_request=9000
4
5 chmod g+w /var/run/openvswitch/vhost*

```

Provision a KVM-VPX with OVS-DPDK-based vhost-user ports

You can provision a VPX instance on Fedora KVM with OVS-DPDK-based vhost-user ports only from CLI by using the following QEMU commands:

Fedora command:

```

1 qemu-system-x86_64 -name KVM-VPX -cpu host -enable-kvm -m 4096M \
2
3 -object memory-backend-file,id=mem,size=4096M,mem-path=/dev/hugepages,
  share=on -numa node,memdev=mem \
4
5 -mem-prealloc -smp sockets=1,cores=2 -drive file=<absolute-path-to-disc
  -image-file>,if=none,id=drive-ide0-0-0,format=<disc-image-format> \
6
7 -device ide-drive,bus=ide.0,unit=0,drive=drive-ide0-0-0,id=ide0-0-0,
  bootindex=1 \
8
9 -netdev type=tap,id=hostnet0,script=no,downscript=no,vhost=on \
10
11 -device virtio-net-pci,netdev=hostnet0,id=net0,mac=52:54:00:3c:d1:ae,
  bus=pci.0,addr=0x3 \
12
13 -chardev socket,id=char0,path=</usr/local/var/run/openvswitch/vhost-
  user1> \
14
15 -netdev type=vhost-user,id=mynet1,chardev=char0,vhostforce -device
  virtio-net-pci,mac=00:00:00:00:00:01,netdev=mynet1,mrg_rxbuf=on \
16

```

```
17 -chardev socket,id=char1,path=</usr/local/var/run/openvswitch/vhost-  
    user2> \  
18  
19 -netdev type=vhost-user,id=mynet2,chardev=char1,vhostforce -device  
    virtio-net  
20  
21 pci,mac=00:00:00:00:00:02,netdev=mynet2,mrg_rxbuf=on \  
22  
23 --nographic
```

For RHOS, use the following sample XML file to provision the Citrix ADC VPX instance, by using virsh.

```
<domain type='kvm'>  
  <name>dpdk-vpx1</name>  
  <uuid>aedb844b-f6bc-48e6-a4c6-36577f2d68d6</uuid>  
  <memory unit='KiB'>16777216</memory>  
  <currentMemory unit='KiB'>16777216</currentMemory>  
  <memoryBacking>  
    <hugepages>  
      <page size='1048576'unit='KiB' />  
    </hugepages>  
  </memoryBacking>  
  <vcpu placement='static'>6</vcpu>  
  <cputune>  
    <shares>4096</shares>  
    <vcpupin vcpu='0'cpuset='0' />  
    <vcpupin vcpu='1'cpuset='2' />  
    <vcpupin vcpu='2'cpuset='4' />  
    <vcpupin vcpu='3'cpuset='6' />  
    <emulatorpin cpuset='0,2,4,6' />  
  </cputune>  
  <numatune>  
    <memory mode='strict'nodeset='0' />  
  </numatune>  
  <resource>
```

```
<partition>/machine</partition>
</resource>
<os>
  <type arch='x86_64' machine='pc-i440fx-rhel7.0.0'>hvm</type>
  <boot dev='hd' />
</os>
<features>
  <acpi/>
  <apic/>
</features>
<cpu mode='custom' match='minimum' check='full'>
  <model fallback='allow'>Haswell-noTSX</model>
  <vendor>Intel</vendor>
  <topology sockets='1' cores='6' threads='1' />
  <feature policy='require' name='ss' />
  <feature policy='require' name='pcid' />
  <feature policy='require' name='hypervisor' />
  <feature policy='require' name='arat' />
<domain type='kvm'>
  <name>dpdk-vpx1</name>
  <uuid>aedb844b-f6bc-48e6-a4c6-36577f2d68d6</uuid>
  <memory unit='KiB'>16777216</memory>
  <currentMemory unit='KiB'>16777216</currentMemory>
  <memoryBacking>
    <hugepages>
      <page size='1048576' unit='KiB' />
    </hugepages>
  </memoryBacking>
  <vcpu placement='static'>6</vcpu>
```



```
<cputune>
  <shares>4096</shares>
  <vcupin vcpu='0' cpuset='0' />
  <vcupin vcpu='1' cpuset='2' />
  <vcupin vcpu='2' cpuset='4' />
  <vcupin vcpu='3' cpuset='6' />
  <emulatorpin cpuset='0,2,4,6' />
</cputune>
<numatune>
  <memory mode='strict' nodeset='0' />
</numatune>
<resource>
  <partition>/machine</partition>
</resource>
<os>
  <type arch='x86_64' machine='pc-i440fx-rhel7.0.0'>hvm</type>
  <boot dev='hd' />
</os>
<features>
  <acpi />
  <apic />
</features>
<cpu mode='custom' match='minimum' check='full'>
  <model fallback='allow'>Haswell-noTSX</model>
  <vendor>Intel</vendor>
  <topology sockets='1' cores='6' threads='1' />
  <feature policy='require' name='ss' />
  <feature policy='require' name='pcid' />
  <feature policy='require' name='hypervisor' />
</cpu>
```

```
<feature policy='require' name='arat' />
<feature policy='require' name='tsc_adjust' />
<feature policy='require' name='xsaveopt' />
<feature policy='require' name='pdpe1gb' />
<numa>
  <cell id='0' cpus='0-5' memory='16777216' unit='KiB' memAccess='shared' />
</numa>
</cpu>
<clock offset='utc' />
<on_poweroff>destroy</on_poweroff>
<on_reboot>restart</on_reboot>
<on_crash>destroy</on_crash>
<devices>
  <emulator>/usr/libexec/qemu-kvm</emulator>
  <disk type='file' device='disk'>
    <driver name='qemu' type='qcow2' cache='none' />
    <source file='/home/NSVPX-KVM-12.0-52.18_nc.qcow2' />
    <target dev='vda' bus='virtio' />
    <address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0' />
  </disk>
  <controller type='ide' index='0'>
    <address type='pci' domain='0x0000' bus='0x00' slot='0x01' function='0x1' />
  </controller>
  <controller type='usb' index='0' model='piix3-uhci'>
    <address type='pci' domain='0x0000' bus='0x00' slot='0x01' function='0x2' />
  </controller>
  <controller type='pci' index='0' model='pci-root' />
  <interface type='direct'>
    <mac address='52:54:00:bb:ac:05' />
  </interface>
</devices>
```

```
<source dev='enp129s0f0' mode='bridge' />
<model type='virtio' />
<address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0' />
</interface>
<interface type='vhostuser'>
  <mac address='52:54:00:55:55:56' />
  <source type='unix' path='/var/run/openvswitch/vhost-user1' mode='client' />
  <model type='virtio' />
  <address type='pci' domain='0x0000' bus='0x00' slot='0x04' function='0x0' />
</interface>
<interface type='vhostuser'>
  <mac address='52:54:00:2a:32:64' />
  <source type='unix' path='/var/run/openvswitch/vhost-user2' mode='client' />
  <model type='virtio' />
  <address type='pci' domain='0x0000' bus='0x00' slot='0x05' function='0x0' />
</interface>
<interface type='vhostuser'>
  <mac address='52:54:00:2a:32:74' />
  <source type='unix' path='/var/run/openvswitch/vhost-user3' mode='client' />
  <model type='virtio' />
  <address type='pci' domain='0x0000' bus='0x00' slot='0x06' function='0x0' />
</interface>
<interface type='vhostuser'>
  <mac address='52:54:00:2a:32:84' />
  <source type='unix' path='/var/run/openvswitch/vhost-user4' mode='client' />
  <model type='virtio' />
  <address type='pci' domain='0x0000' bus='0x00' slot='0x09' function='0x0' />
</interface>
<serial type='pty'>
```

```
<target port='0' />
</serial>
<console type='pty'>
  <target type='serial' port='0' />
</console>
<input type='mouse' bus='ps2' />
<input type='keyboard' bus='ps2' />
<graphics type='vnc' port='-1' autoport='yes'>
  <listen type='address' />
</graphics>
<video>
  <model type='cirrus' vram='16384' heads='1' primary='yes' />
  <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x0' />
</video>
<memballoon model='virtio'>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x08' function='0x0' />
</memballoon>
</devices>
</domain
```

Points to note

In the XML file, the hugepage size must be 1 GB, as shown in the sample file.

```
<memoryBacking>
  <hugepages>
    <page size='1048576' unit='KiB' />
  </hugepages>
```

Also, in the sample file vhost-user1 is the vhostuser port bound to ovs-br0.

```
<interface type='vhostuser'>
  <mac address='52:54:00:55:55:56' />
```

```
<source type='unix' path='/var/run/openvswitch/vhost-user1' mode='client' />
<model type='virtio' />
<address type='pci' domain='0x0000' bus='0x00' slot='0x04' function='0x0' />
</interface>
```

To bring up the Citrix ADC VPX instance, start using virsh command.

Deploy a Citrix ADC VPX instance on AWS

November 13, 2024

You can launch a Citrix ADC VPX instance on Amazon Web Services (AWS). The Citrix ADC VPX appliance is available as an Amazon Machine Image (AMI) in AWS marketplace. A Citrix ADC VPX instance on AWS enables customer like you to leverage AWS Cloud computing capabilities and use Citrix ADC load balancing and traffic management features for their business needs. The VPX instance supports all the traffic management features of a physical Citrix ADC appliance, and they can be deployed as standalone instances or in HA pairs.

This section includes the following topics:

- AWS terminology
- How a Citrix ADC VPX instance on AWS works
- Supported instance type, ENI, and IP addresses

AWS terminology

Here is a brief description of the terms used in this document. For more information, see [AWS Glossary](#).

Term	Defintion
Amazon Machine Image (AMI)	A machine image, which provides the information required to launch an instance, which is a virtual server in the cloud.
Elastic Block Store	Provides persistent block storage volumes for use with Amazon EC2 instances in the AWS Cloud.
Simple Storage Service (S3)	Storage for the Internet. It is designed to make web-scale computing easier for developers.

Term	Defintion
Elastic Compute Cloud (EC2)	A web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers.
Elastic Load Balancing (ELB)	Distributes incoming application traffic across multiple EC2 instances, in multiple Availability Zones. This increases the fault tolerance of your applications.
Elastic network interface (ENI)	A virtual network interface that you can attach to an instance in a VPC.
Elastic IP (EIP) address	A static, public IPv4 address that you have allocated in Amazon EC2 or Amazon VPC and then attached to an instance. Elastic IP addresses are associated with your account, not a specific instance. They are elastic because you can easily allocate, attach, detach, and free them as your needs change.
Instance type	Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instance types comprise varying combinations of CPU, memory, storage, and networking capacity and give you the flexibility to choose the appropriate mix of resources for your applications.
Identity and Access Management (IAM)	An AWS identity with permission policies that determine what the identity can and cannot do in AWS. You can use an IAM role to enable applications running on an EC2 instance to securely access your AWS resources.IAM role is required for deploying VPX instances in a high-availability setup.
Internet Gateway	Connects a network to the Internet. You can route traffic for IP addresses outside your VPC to the Internet gateway.
Key pair	A set of security credentials that you use to prove your identity electronically. A key pair consists of a private key and a public key.

Term	Defintion
Route tables	A set of routing rules that controls the traffic leaving any subnet that is associated with the route table. You can associate multiple subnets with a single route table, but a subnet can be associated with only one route table at a time.
Security groups	A named set of allowed inbound network connections for an instance.
Subnets	A segment of the IP address range of a VPC that EC2 instances can be attached to. You can create subnets to group instances according to security and operational needs.
Virtual Private Cloud (VPC)	A web service for provisioning a logically isolated section of the AWS cloud where you can launch AWS resources in a virtual network that you define.
Auto Scaling	A web service to launch or terminate Amazon EC2 instances automatically based on user-defined policies, schedules, and health checks.
CloudFormation	A service for writing or changing templates that create and delete related AWS resources together as a unit.

How a Citrix ADC VPX instance on AWS works

The Citrix ADC VPX instance is available as an AMI in AWS marketplace, and it can be launched as an EC2 instance within an AWS VPC. The Citrix ADC VPX AMI instance requires a minimum of 2 virtual CPUs and 2 GB of memory. An EC2 instance launched within an AWS VPC can also provide the multiple interfaces, multiple IP addresses per interface, and public and private IP addresses needed for VPX configuration. Each VPX instance requires at least three IP subnets:

- A management subnet
- A client-facing subnet (VIP)
- A back-end facing subnet (SNIP, MIP, etc.)

Citrix recommends three network interfaces for a standard VPX instance on AWS installation.

AWS currently makes multi-IP functionality available only to instances running within an AWS VPC. A VPX instance in a VPC can be used to load balance servers running in EC2 instances. An Amazon VPC

allows you to create and control a virtual networking environment, including your own IP address range, subnets, route tables, and network gateways.

Note: By default, you can create up to 5 VPC instances per AWS region for each AWS account. You can request higher VPC limits by submitting Amazon’s request form <http://aws.amazon.com/contact-us/vpc-request>.

Figure 1. A Sample Citrix ADC VPX Instance Deployment on AWS Architecture

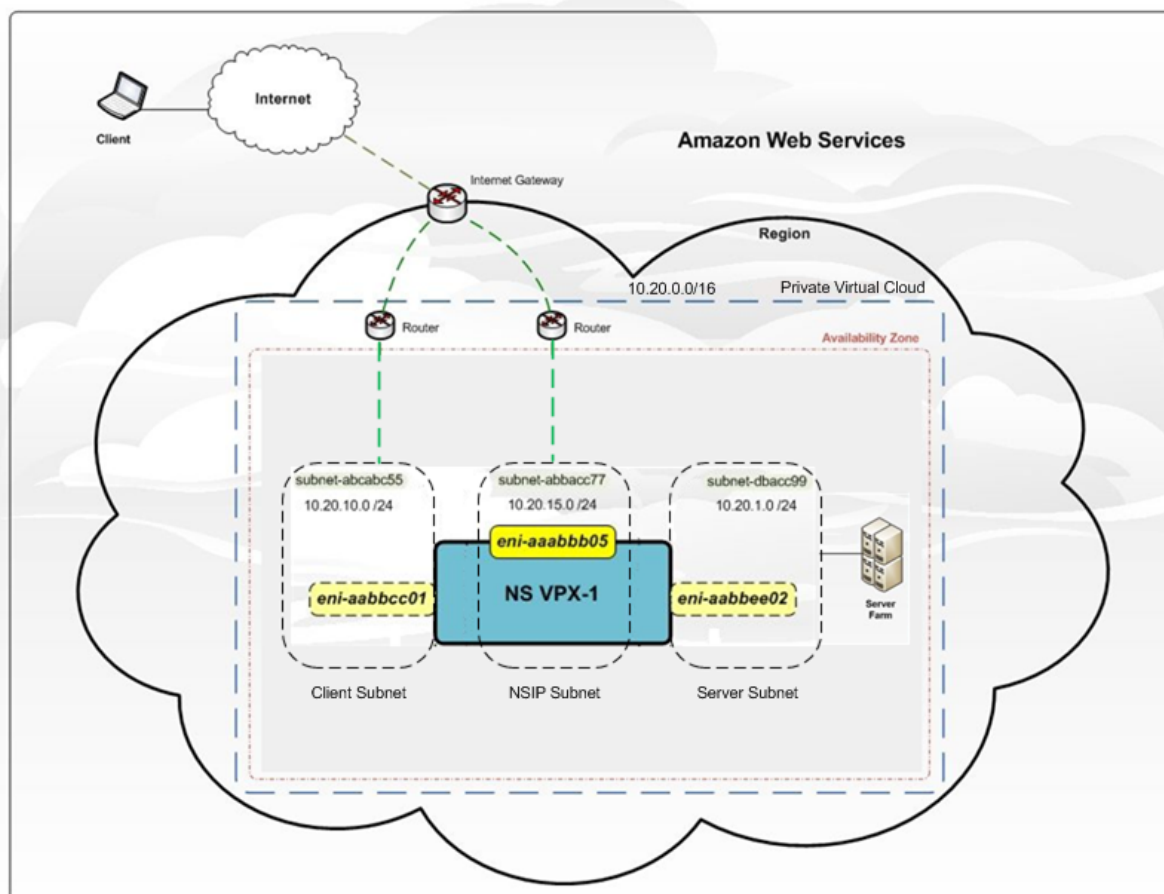


Figure 1 shows a simple topology of an AWS VPC with a Citrix ADC VPX deployment. The AWS VPC has:

1. A single Internet gateway to route traffic in and out of the VPC.
2. Network connectivity between the Internet gateway and the Internet.
3. Three subnets, one each for management, client, and server.
4. Network connectivity between the Internet gateway and the two subnets (management and client).
5. A standalone Citrix ADC VPX instance deployed within the VPC. The VPX instance has three ENIs, one attached to each subnet.

Supported instance type, ENI, and IP addresses

For more information about Amazon EC2 instances and IP addresses supported per NIC per instance type:

- [Instance types](#)
- [IP addresses per network interface per instance type](#)

For higher bandwidth, Citrix recommends the following instance types:

Instance type	Bandwidth	Enhanced networking (SR-IOV)
M4.10x large	3 Gbps and 5 Gbps	Yes
C4.8x large	3 Gbps and 5 Gbps	Yes

Limitations and usage guidelines

November 13, 2024

The following limitations and usage guidelines apply when deploying a Citrix ADC VPX instance on AWS:

- Before you start, read the AWS terminology section in [Deploy a Citrix ADC VPX instance on AWS](#).
- The clustering feature is not supported for VPX.
- For the high availability setup to work effectively, associate a dedicated NAT device to management Interface or associate EIP to NSIP. For more information on NAT, in the AWS documentation, see [NAT Instances](#).
- Data traffic and management traffic must be segregated with ENIs belonging to different subnets.
- Only the NSIP address must be present on the management ENI.
- If a NAT instance is used for security instead of assigning an EIP to the NSIP, appropriate VPC level routing changes are required. For instructions on making VPC level routing changes, in the AWS documentation, see [Scenario 2: VPC with Public and Private Subnets](#).
- A VPX instance can be moved from one EC2 instance type to another (for example, from m3.large to an m3.xlarge).
- For storage options for VPX on AWS, Citrix recommends EBS, because it is durable and the data is available even after it is detached from instance.

- Dynamic addition of ENIs to VPX is not supported. Restart the VPX instance to apply the update. Citrix recommends you to stop the standalone or HA instance, attach the new ENI, and then restart the instance.
- You can assign multiple IP addresses to an ENI. The maximum number of IP addresses per ENI is determined by the EC2 instance type, see the section “IP Addresses Per Network Interface Per Instance Type” in [Elastic Network Interfaces](#). You must allocate the IP addresses in AWS before you assign them to ENIs. For more information, see [Elastic Network Interfaces](#).
- Citrix recommends that you avoid using the enable and disable interface commands on Citrix ADC VPX interfaces.
- The Citrix ADC `set ha node \<NODE_ID\> -haStatus STAYPRIMARY` and `set ha node \<NODE_ID\> -haStatus STAYSECONDARY` commands are disabled by default.
- IPv6 is not supported for VPX.
- Due to AWS limitations, these features are not supported:
 - Gratuitous ARP (GARP)
 - L2 mode
 - Tagged VLAN
 - Dynamic Routing
 - Virtual MAC (VMAC)
- For RNAT to work, ensure **Source/Destination Check** is disabled. For more information, see “Changing the Source/Destination Checking” in [Elastic Network Interfaces](#).
- In a Citrix ADC VPX deployment on AWS, in some AWS regions, the AWS infrastructure might not be able to resolve AWS API calls. This happens if the API calls are issued through a nonmanagement interface on the Citrix ADC VPX instance.
As a workaround, restrict the API calls to the management interface only. To do that, create a NSVLAN on the VPX instance and bind the management interface to the NSVLAN by using the appropriate command.
For example:

```
set ns config -nsvlan <vlan id> -ifnum 1/1 -tagged NO
save config
```


Restart the VPX instance at the prompt. For more information about configuring NSVLAN, see [Configuring NSVLAN](#).
- In the AWS console, the vCPU usage shown for a VPX instance under the **Monitoring** tab might be high (up to 100 percent), even when the actual usage is much lower. To see the actual vCPU usage, use the VPX GUI or CLI.

Prerequisites

November 13, 2024

Before attempting to create a VPX instance in AWS, ensure you have the following:

- **An AWS account:** to launch a Citrix ADC VPX AMI in an Amazon Web Services (AWS) Virtual Private Cloud (VPC). You can create an AWS account for free at .
- **An AWS Identity and Access Management (IAM) user account:** to securely control access to AWS services and resources for your users. For more information about how to create an IAM user account, see the topic [Creating IAM Users \(Console\)](#).

An IAM role is mandatory for both standalone and high availability deployments. The IAM role must have the following privileges:

```
1 ec2:DescribeInstances
2 ec2:DescribeNetworkInterfaces
3 ec2:DetachNetworkInterface
4 ec2:AttachNetworkInterface
5 ec2:StartInstances
6 ec2:StopInstances
7 ec2:RebootInstances
8 ec2:DescribeAddresses
9 ec2:AssociateAddress
10 ec2:DisassociateAddress
11 ec2:AssignPrivateIpAddresses
12 ec2:UnassignPrivateIpAddresses
13 autoscaling:*
14 sns:CreateTopic
15 sns>DeleteTopic
16 sns:ListTopics
17 sns:Subscribe
18 sqs:CreateQueue
19 sqs:ListQueues
20 sqs>DeleteMessage
21 sqs:GetQueueAttributes
22 sqs:SetQueueAttributes
23 iam:SimulatePrincipalPolicy
24 iam:GetRole
```

If you use the Citrix CloudFormation template, the IAM role is automatically created. The template does not allow selecting an already created IAM role.

Note: When you log on the VPX instance through GUI, a prompt to configure the required privileges for IAM role appears. Ignore the prompt if you've already configured the privileges.

- **AWS CLI:** to use all the functionality provided by the AWS Management Console from your terminal program. For more information, see the [AWS CLI user guide](#). You also need the AWS CLI to change the network interface type to SR-IOV.

- **Elastic Network Adapter (ENA):** For ENA driver-enabled instance type, the firmware version must be 13.0 and above.

Deploy a Citrix ADC VPX standalone instance on AWS

November 14, 2024

You can deploy a Citrix ADC VPX standalone instance on AWS by using the following options:

- AWS web console
- Citrix-authored CloudFormation template
- AWS CLI

This topic describes the procedure for deploying a Citrix ADC VPX instance on AWS.

Before you start your deployment, read the following topics:

- [Prerequisites](#)
- [Limitation and usage guidelines](#)

Deploy a Citrix ADC VPX instance on AWS by using the AWS web console

You can deploy a Citrix ADC VPX instance on AWS through the AWS web console. The deployment process includes the following steps:

1. Create a Key Pair
2. Create a Virtual Private Cloud (VPC)
3. Add additional subnets
4. Create security groups and security rules
5. Add route tables
6. Create an internet gateway
7. Create a Citrix ADC VPX instance
8. Create and attach additional network interfaces
9. Attach elastic IPs to the management NIC
10. Connect to the VPX instance

Step 1: Create a key pair.

Amazon EC2 uses a key pair to encrypt and decrypt logon information. To log on to your instance, you must create a key pair, specify the name of the key pair when you launch the instance, and provide the private key when you connect to the instance.

When you review and launch an instance by using the AWS Launch Instance wizard, you are prompted to use an existing key pair or create a new key pair. For more information about how to create a key pair, see [Amazon EC2 Key Pairs](#).

Step 2: Create a VPC.

A Citrix ADC VPC instance is deployed inside an AWS VPC. A VPC allows you to define virtual network dedicated to your AWS account. For more information about AWS VPC, see [Getting Started With Amazon VPC](#).

While creating a VPC for your Citrix ADC VPX instance, keep the following points in mind.

- Use the VPC with a Single Public Subnet Only option to create a AWS VPC in an AWS availability zone.
- Citrix recommends that you create at least **three subnets**, of the following types:
 - One subnet for management traffic. You place the management IP(NSIP) on this subnet. By default elastic network interface (ENI) eth0 is used for management IP.
 - One or more subnets for client-access (user-to-Citrix ADC VPX) traffic, through which clients connect to one or more virtual IP (VIP) addresses assigned to Citrix ADC load balancing virtual servers.
 - One or more subnets for the server-access (VPX-to-server) traffic, through which your servers connect to VPX-owned subnet IP (SNIP) addresses. For more information about Citrix ADC load balancing and virtual servers, virtual IP addresses (VIPs), and subnet IP addresses (SNIPs), see:
 - All subnets must be in the same availability zone.

Step 3: Add subnets.

When you used the VPC wizard, only one subnet was created. Depending on your requirement, you might want to create additional subnets. For more information about how to create additional subnets, see [Adding a Subnet to Your VPC](#).

Step 4: Create security groups and security rules.

To control inbound and outbound traffic, create security groups and add rules to the groups. For more information how to create groups and add rules, see [Security Groups for Your VPC](#).

For Citrix ADC VPX instances, the EC2 wizard gives default security groups, which are generated by AWS Marketplace and is based on recommended settings by Citrix. However, you can create additional security groups based on your requirements.

Note:

Port 22, 80, 443 to be opened on the Security group for SSH, HTTP, and HTTPS access respectively.

Step 5: Add route tables.

Route table contains a set of rules, called routes, that are used to determine where network traffic is directed. Each subnet in your VPC must be associated with a route table. For more information about how to create a route table, see [Route Tables](#).

Step 6: Create an internet gateway.

An internet gateway serves two purposes: to provide a target in your VPC route tables for internet-routable traffic, and to perform network address translation (NAT) for instances that have been assigned public IPv4 addresses.

Create an internet gateway for internet traffic. For more information about how to create an Internet Gateway, see the section [Attaching an Internet Gateway](#).

Step 7: Create a Citrix ADC VPX instance by using the AWS EC2 service.

To create a Citrix ADC VPX instance by using the AWS EC2 service, complete the following steps.

1. From the AWS dashboard, go to **Compute > EC2 > Launch Instance > AWS Marketplace**.

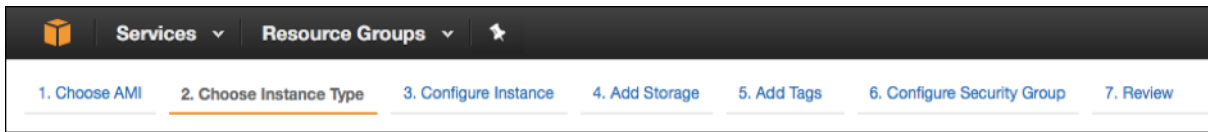
Before you click **Launch Instance**, ensure your region is correct by checking the note that appears under Launch Instance.



2. In the Search AWS Marketplace bar, search with the keyword Citrix ADC VPX.
3. Select the version you want to deploy and then click **Select**. For the Citrix ADC VPX version, you the following options:
 - A licensed version
 - Citrix ADC VPX Express appliance (This is a free virtual appliance, which is available from Citrix ADC 12.0 56.20.)
 - Bring your own device

The Launch Instance wizard starts. Follow the wizard to create an instance. The wizard prompts you to:

- 1 - Choose Instance Type
- 2 - Configure Instance
- 3 - Add Storage
- 4 - Add Tags
- 5 - Configure Security Group
- 6 - Review



Step 8: Create and attach additional network interfaces.

Create two additional network interfaces for VIP and SNIP. For more information about how to create additional network interfaces, see the [Creating a Network Interface](#) section.

After you've created the network interfaces, you must attach them to the VPX instance. Before attaching the interface, shut down the VPX instance, attach the interface, and power on the instance. For more information about how to attach network interfaces, see the [Attaching a Network Interface When Launching an Instance](#) section.

Step 9: Allocate and associate elastic IPs.

If you assign a public IP address to an EC2 instance, it remains assigned only until the instance is stopped. After that, the address is released back to the pool. When you restart the instance, a new public IP address is assigned.

In contrast, an elastic IP (EIP) address remains assigned until the address is disassociated from an instance.

Allocate and associate an elastic IP for the management NIC. For more information about how to allocate and associate elastic IP addresses, see these topics:

- [Allocating an Elastic IP Address](#)
- [Associating an Elastic IP Address with a Running Instance](#)

These steps complete the procedure to create a Citrix ADC VPX instance on AWS. It can take a few minutes for the instance to be ready. Check that your instance has passed its status checks. You can view this information in the **Status Checks** column on the Instances page.

Step 10: Connect to the VPX instance.

After you've created the VPX instance, you connect the instance by using the GUI and an SSH client.

- GUI

The following are the default administrator credentials to access a Citrix ADC VPX instance

User name: `nsroot`

Password: The default password for the `nsroot` account is set to the AWS instance-ID of the Citrix ADC VPX instance.

- SSH client

From the AWS management console, select the Citrix ADC VPX instance and click **Connect**. Follow the instructions given on the **Connect to Your Instance** page.

For more information about how to deploy a Citrix ADC VPX standalone instance on AWS by using the AWS web console, see:

- [Scenario: standalone instance](#)
- [How to configure a Citrix NetScaler VPX instance on AWS by using Citrix CloudFormation template](#)

Configure a Citrix ADC VPX instance by using the Citrix CloudFormation template

You can use the Citrix-provided CloudFormation template to automate VPX instance launch. The template provides functionality to launch a single Citrix ADC VPX instance, or to create a high availability environment with a pair of Citrix ADC VPX instances.

You can launch the template from AWS Marketplace or GitHub.

The CloudFormation template requires an existing VPC environment, and it launches a VPX instance with three elastic network interfaces (ENIs). Before you start the CloudFormation template, ensure that you complete the following requirements:

- An AWS virtual private cloud (VPC)
- Three subnets within the VPC: one for management, one for client traffic, and one for back-end servers
- An EC2 key pair to enable SSH access to the instance
- A security group with UDP 3003, TCP 3009–3010, HTTP, SSH ports open

See the “Deploy a Citrix ADC VPX Instance on AWS by Using the AWS Web Console” section or AWS documentation for more information about how to complete the prerequisites.

Watch this [video](#) to learn about how to configure and launch a Citrix ADC VPX standalone instance by using the Citrix CloudFormation template available in the AWS Marketplace.

Further, you configure and launch a Citrix ADC VPX Express standalone instance by using the Citrix CloudFormation template available in GitHub:

<https://github.com/citrix/citrix-adc-aws-cloudformation/tree/master/templates/standalone/1nic/express-single-nic>

An IAM role is not mandatory for a standalone deployment. However, Citrix recommends that you create and attach an IAM role with the required privileges to the instance, for future need. The IAM role ensures that the standalone instance is easily converted to a high availability node with SR-IOV, when required.

For more information about the required privileges, see [Configuring Citrix ADC VPX instances to use the SR-IOV Network Interface](#).

Configure a Citrix ADC VPX instance by using the AWS CLI

You can use the AWS CLI to launch instances. For more information, see the [AWS Command Line Interface Documentation](#).

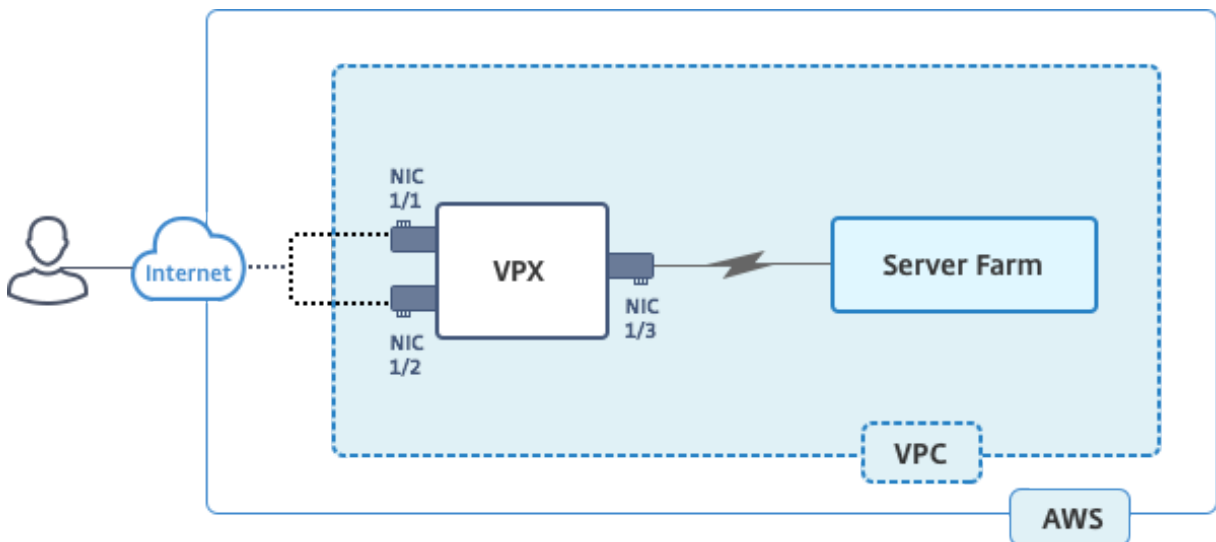
Scenario: standalone instance

November 13, 2024

May 24, 2018

This scenario illustrates how to deploy a Citrix ADC VPX standalone EC2 instance in AWS by using the AWS GUI. Create a standalone VPX instance with three NICs. The instance, which is configured as a load balancing virtual server, communicates with backend servers (the server farm). For this configuration, set up the required communication routes between the instance and the back-end servers, and between the instance and the external hosts on the public internet.

For more details about the procedure for deploying a VPX instance., see [Deploy a Citrix ADC VPX standalone instance on AWS](#).



Create three NICs. Each NIC can be configured with a pair of IP addresses (public and private). The NICs serve the following purposes.

NIC	Purpose	Associated with
eth0	Serves management traffic (NSIP)	A public IP address and a private IP address

NIC	Purpose	Associated with
eth1	Serves client-side traffic (VIP)	A public IP address and a private IP address
eth1	Communicates with backend servers (SNIP)	A public IP address (Private IP address not mandatory)

Step 1: Create a VPC.

1. Log on to the AWS web console and navigate to **Networking & Content Delivery > VPC**. Click **Start VPC Wizard**.
2. Select **VPC with a Single Public Subnet** and click **Select**.
3. Set the IP CIDR Block to 10.0.0.0/16, for this scenario.
4. Give a name for the VPC.
5. Set the public subnet to 10.0.0.0/24. (This is the management network).
6. Select an availability zone.
7. Give a give a name for the subnet.
8. Click Create **VPC**.

Step 2: VPC with a Single Public Subnet

IPv4 CIDR block*: 10.0.0.0/16 (65531 IP addresses available)

IPv6 CIDR block: No IPv6 CIDR Block
 Amazon provided IPv6 CIDR block

VPC name: NSDoc

Public subnet's IPv4 CIDR*: 10.0.0.0/24 (251 IP addresses available)

Availability Zone*: ap-south-1a

Subnet name: NSDoc-MGMT

You can add more subnets after AWS creates the VPC.

Service endpoints

Enable DNS hostnames*: Yes No

Hardware tenancy*: Default

Step 2: Create additional subnets.

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. In the navigation pane, choose Subnets, Create Subnet after you enter the following details.
 - Name tag: Provide a name for your subnet.
 - VPC: Choose the VPC for which you’re creating the subnet.
 - Availability Zone: Choose the availability zone in which you created the VPC in step 1.

- IPv4 CIDR block: Specify an IPv4 CIDR block for your subnet. For this scenario, choose 10.0.1.0/24.

Create Subnet
✕

Use the CIDR format to specify your subnet's IP address block (e.g., 10.0.0.0/24). Note that block sizes must be between a /16 netmask and /28 netmask. Also, note that a subnet can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag ⓘ

VPC ⓘ

VPC CIDRs

CIDR	Status	Status Reason
10.0.0.0/16	associated	

Availability Zone ⓘ

IPv4 CIDR block ⓘ

Cancel
Yes, Create

1. Repeat the steps to create one more subnet for back-end servers.

Create Subnet
✕

Use the CIDR format to specify your subnet's IP address block (e.g., 10.0.0.0/24). Note that block sizes must be between a /16 netmask and /28 netmask. Also, note that a subnet can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag ⓘ

VPC ⓘ

VPC CIDRs

CIDR	Status	Status Reason
10.0.0.0/16	associated	

Availability Zone ⓘ

IPv4 CIDR block ⓘ

Cancel
Yes, Create

Step 3: Create a route table.

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. In the navigation pane, choose **Route Tables > Create Route Table**.
3. In the Create Route Table window, add a name and select the VPC that you created in step 1.

4. Click **Yes, Create**.

Create Route Table ✕

A route table specifies how packets are forwarded between the subnets within your VPC, the Internet, and your VPN connection.

Name tag i

VPC i

Cancel
Yes, Create

The route table is assigned to all the subnets that you created for this VPC, so that routing of traffic from instance in one subnet can reach an instance in another subnet.

1. Click **Subnet Associations** and then click **Edit**.
2. Click the management and client subnet and click **Save**. This creates a route table for internet traffic only.

rtb-4329082a | NSDoc-internet-traffic

Summary
Routes
Subnet Associations
Route Propagation
Tags

Cancel
Save

Associate	Subnet	IPv4 CIDR	IPv6 CIDR	Current Route Table
<input checked="" type="checkbox"/>	subnet-c4ce9aad NSDoc-MGMT	10.0.0.0/24	-	rtb-735a7b1a
<input checked="" type="checkbox"/>	subnet-31ce9a58 NSDoc-client	10.0.1.0/24	-	Main
<input type="checkbox"/>	subnet-d0cd99b9 NSDoc-server	10.0.2.0/24	-	Main

1. Click **Routes > Edit** > Add another route.
2. In the **Destination** field add 0.0.0.0/0, and click the Target field to select igw-<xxx> the Internet Gateway that the VPC Wizard created automatically.
3. Click **Save**.

rtb-4329082a | NSDoc-internet-traffic

Summary Routes Subnet Associations Route Propagation Tags

Cancel Save

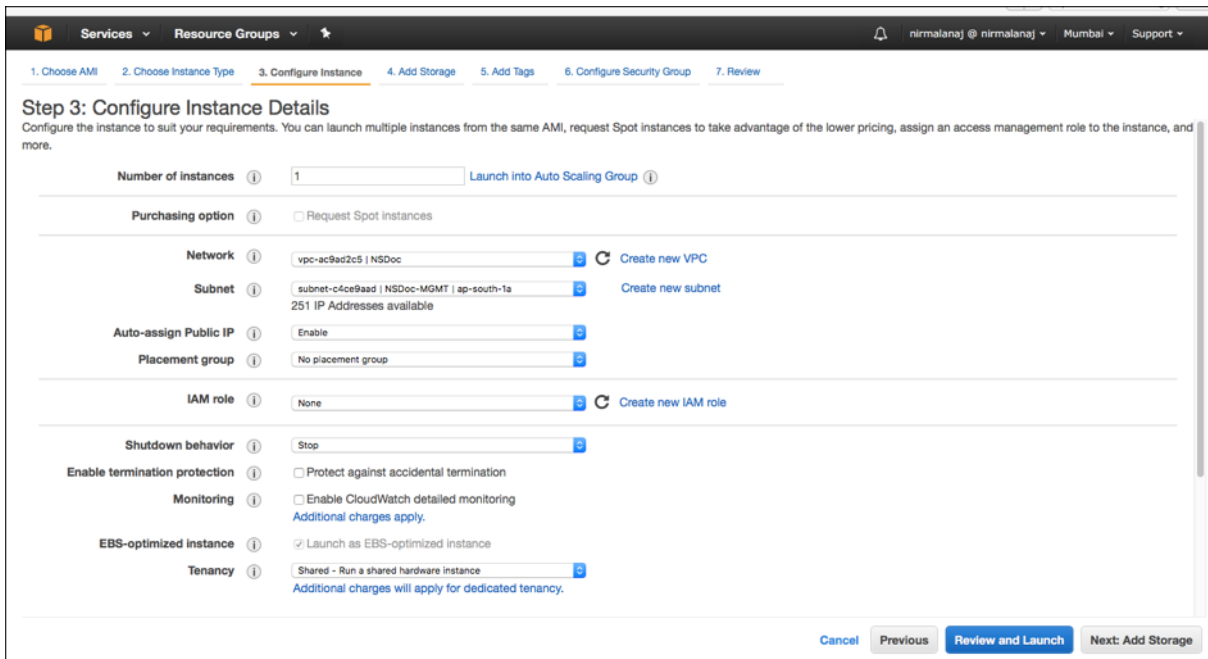
View: All rules

Destination	Target	Status	Propagated	Remove
10.0.0.0/16	local	Active	No	
<input type="text" value="0.0.0.0/0"/>	<input type="text" value="igw-9fbe2df6"/>		No	<input type="button" value="✕"/>

1. Follow the steps to create a route table for server-side traffic.

Step 4: Create a Citrix ADC VPX instance.

1. Log on the AWS management console and click **EC2** under **Compute**.
2. Click AWS Marketplace. In the Search AWS Marketplace bar, type Citrix ADC VPX and press Enter. The available Citrix ADC VPX editions are displayed.
3. Click **Select** to choose the desired Citrix ADC VPX edition. The EC2 instance wizard starts.
4. In the **Choose Instance Type** page, select **m4. Xlarge** (recommended) and click **Next: Configure Instance Details**.
5. In the Configure Instance Details page, select the following, and then click Next: Add Storage.
 - Number of instances: 1
 - Network: the VPC that created in Step 1
 - Subnet: the management subnet
 - Auto-assign Public IP: Enable



1. In the Add Storage page, select the default option, and click Next: Add Tags.
2. In the Add Tags page, add a name for the instance, and click Next: Configure Security Group.
3. In the Configure Security Group page, select the default option (which is generated by AWS Marketplace and is based on recommended settings by Citrix Systems) and then click Review and Launch > Launch.
4. You are prompted to select an existing key pair or create and new key pair. From the Select a key pair drop-down list, select the key pair that you created as a prerequisite (See the Prerequisite section.)
5. Check the box to acknowledge the key pair and click Launch Instances.

Select an existing key pair or create a new key pair ✕

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. [Learn more about removing existing key pairs from a public AMI.](#)

Select a key pair

I acknowledge that I have access to the selected private key file (NSDOCKeypair.pem), and that without this file, I won't be able to log into my instance.

Cancel
Launch Instances

Launch Instance Wizard displays the Launch Status, and the instance appears in the list of instances when it is fully launched.

The check instance, go the AWS console click EC2 > Running Instances. Select the instance and add a name. Make sure the Instance State is running and Status Checks is complete.

Step 5: Create and attach more network interfaces.

When you created the VPC, only one network interface associated with it. Now add two more network interfaces to the VPC, for the VIP and SNIP.

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. In the navigation pane, choose Network Interfaces.
3. Choose Create Network Interface.
4. For Description, enter a descriptive name.
5. For Subnet, select the subnet that you created previously for the VIP.
6. For Private IP, leave the default option.
7. For Security groups, select the group.
8. Click **Yes, Create**.

1. After the network interface is created, add a name to the interface.
2. Repeat the steps to create a network interface for server-side traffic.

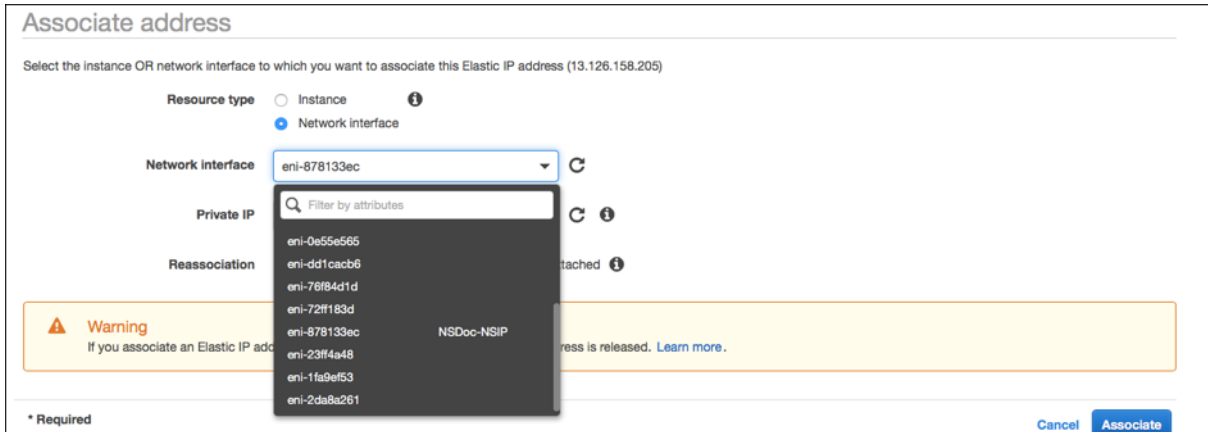
Attach the network interfaces:

1. In the navigation pane, choose Network Interfaces.
2. Select the network interface and choose Attach.
3. In the Attach Network Interface dialog box, select the instance and choose Attach.

Step 6: Attach elastic IP to the NSIP.

1. From the AWS management console, go to **NETWORK & SECURITY > Elastic IPs**.
2. Check for available free EIP to attach. If none, click **Allocate new address**.
3. Select the newly allocated IP address and choose **Actions > Associate address**.
4. Click the **Network interface** radio button.

5. From the Network interface drop-down list, select the management NIC.
6. From the **Private IP** drop-down menu, select the AWS-generated IP address.
7. Select the **Reassociation** check box.
8. Click **Associate**.



Access the VPX instance:

After you’ve configured a standalone Citrix ADC VPX instance with three NICs, log on to the VPX instance to complete the Citrix ADC-side configuration. Use of the following options:

- GUI: Type the public IP of the management NIC in the browser. Log on by using nsroot as the user name and the instance ID (i-0c1ffe1d987817522) as the password.
- SSH: Open an ssh client and type:

ssh -i <location of your private key> nsroot@<public DNS of the instance>

To find the public DNS, click the instance, and click Connect.

Related information:

- To configure the Citrix ADC-owned IP addresses (NSIP, VIP, and SNIP), see [Configuring Citrix ADC-Owned IP Addresses](#).
- You’ve configured a BYOL version of the Citrix ADC VPX appliance, for more information see the VPX Licensing Guide at <http://support.citrix.com/article/CTX122426>

Download a Citrix ADC VPX license

November 13, 2024

After the launch of Citrix ADC VPX-customer licensed instance from the AWS marketplace, a license is required. For more information on VPX licensing, see [Licensing overview](#).

You have to:

1. Use the licensing portal within Citrix website to generate a valid license.
2. Upload the license to the instance.

If this is a **paid** marketplace instance, then you do not need to install a license. The correct feature set and performance activate automatically.

If you use a Citrix ADC VPX instance with a model number higher than VPX 5000, the network throughput might not be the same as specified by the instance’s license. However, other features, such as SSL throughput and SSL transactions per second, might improve.

5 Gbps network bandwidth is observed in the `c4.8xlarge` instance type.

How to migrate the AWS subscription to BYOL

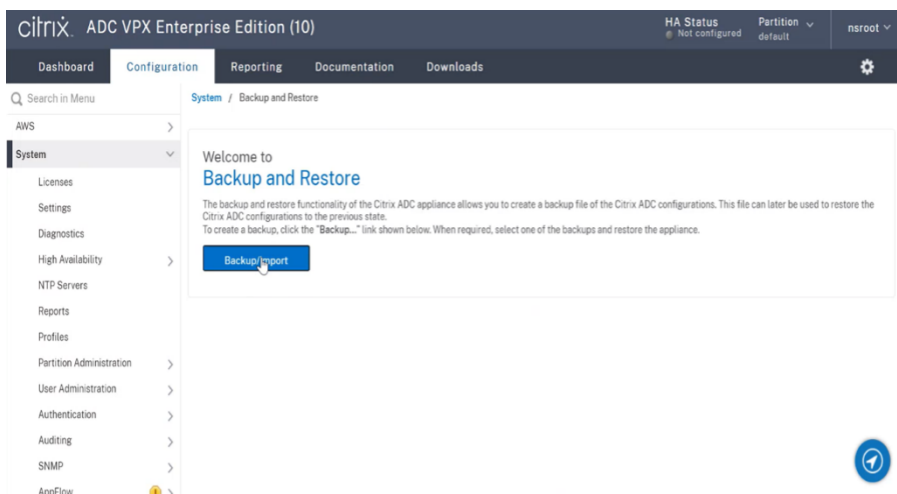
This section describes the procedure to migrate from AWS subscription to Bring your own license (BYOL), and conversely.

Do the following steps to migrate an AWS subscription to BYOL:

Note:

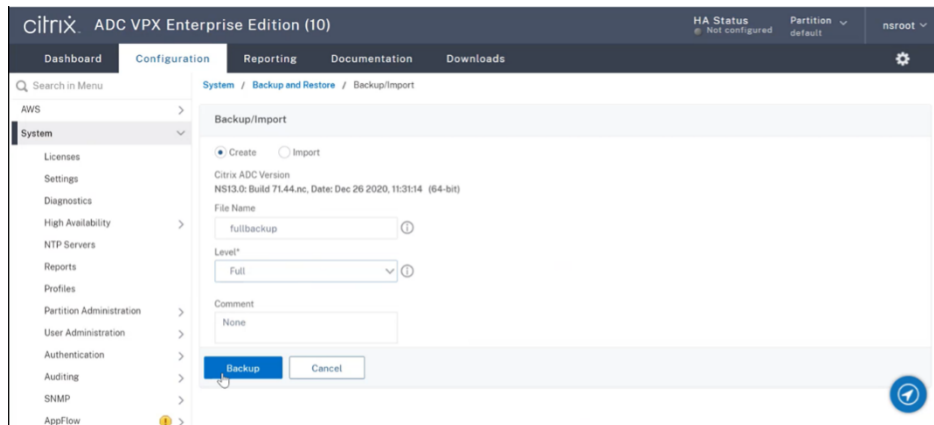
The **Step 2** and **Step 3** are done on the Citrix ADC VPX instance, and all other steps are done on the AWS portal.

1. Create a BYOL EC2 instance using [Citrix ADC VPX - Customer Licensed](#) in the same availability zone as the old EC2 instance that has the same security group, IAM role, and subnet. The new EC2 instance must have only one ENI interface.
2. To back up the data on the old EC2 instance using the Citrix ADC GUI, follow these steps.
 - a) Navigate to **System > Backup and Restore**.
 - b) In the **Welcome** page, click **Backup/Import** to start the process.

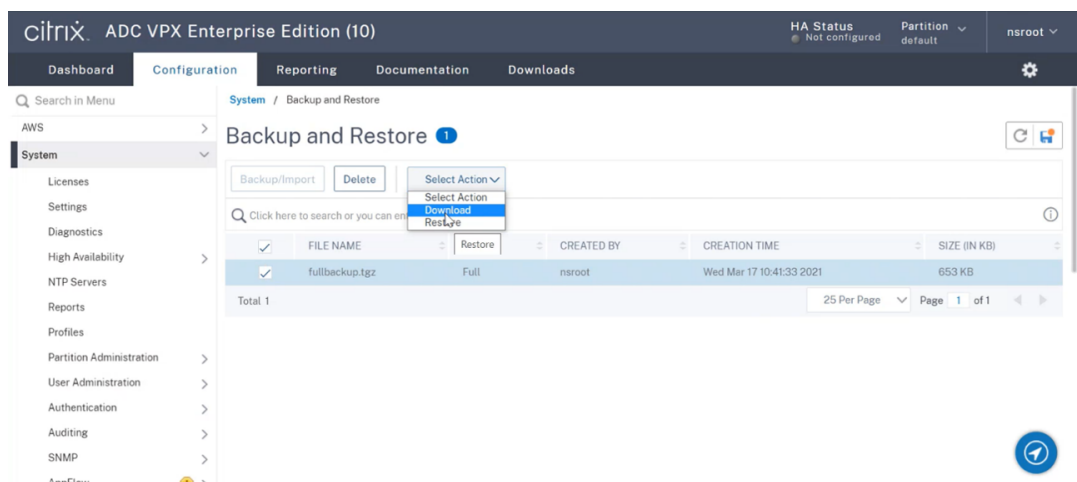


c) In the **Backup/Import** page, fill in the following details:

- **Name** –Name of the backup file.
- **Level** –Select the backup level as **Full**.
- **Comment** –Provide a brief description of the backup.

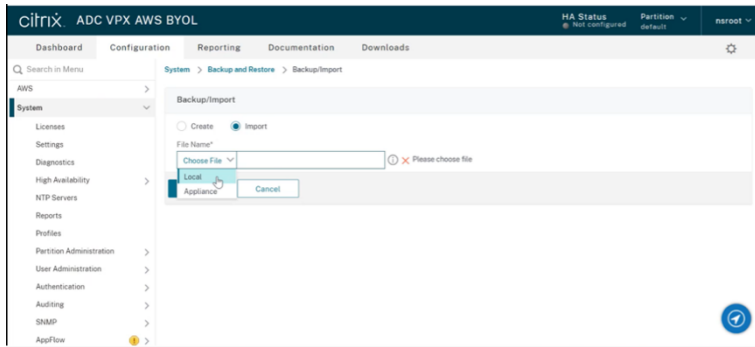


d) Click **Backup**. Once the backup is complete, you can select the file and download it to your local machine.



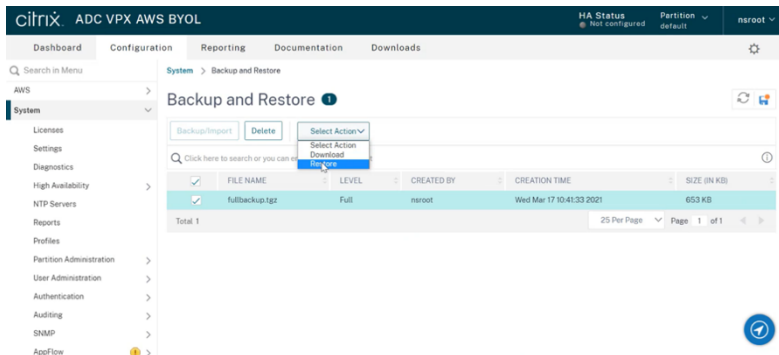
3. To restore the data on the new EC2 instance using the Citrix ADC GUI, follow these steps:

- Navigate to **System > Backup and Restore**.
- Click **Backup/Import** to start the process.
- Select the **Import** option and upload the backup file.

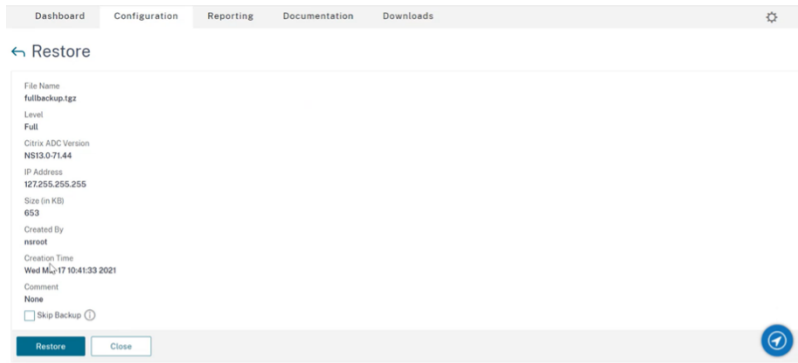


d) Select the file.

e) From the **Select Action** drop-down menu, select **Restore**.



f) On the **Restore** page, verify the file details, and click **Restore**.



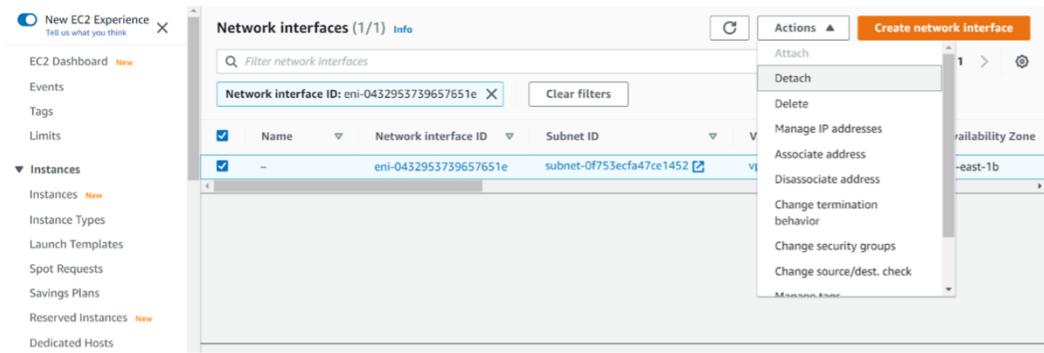
g) After the restore, reboot the EC2 instance.

4. Move all interfaces (except the management interface to which the NSIP address is bound) from the old EC2 instance to the new EC2 instance. To move a network interface from one EC2 instance to another, follow these steps:

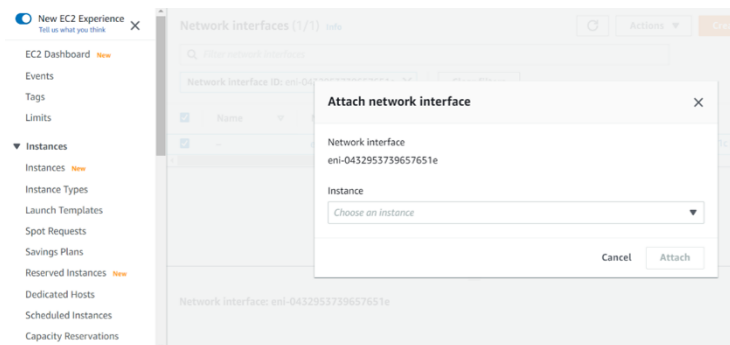
a) In the **AWS Portal**, stop both the old and new EC2 instances.

b) Navigate to **Network Interfaces**, and select the network interface attached to the old EC2 instance.

c) Detach the EC2 instance by clicking **Actions > Detach**.



d) Attach the network interface to the new EC2 instance by clicking **Actions > Attach**. Enter the EC2 instance name to which the network interface must be attached.

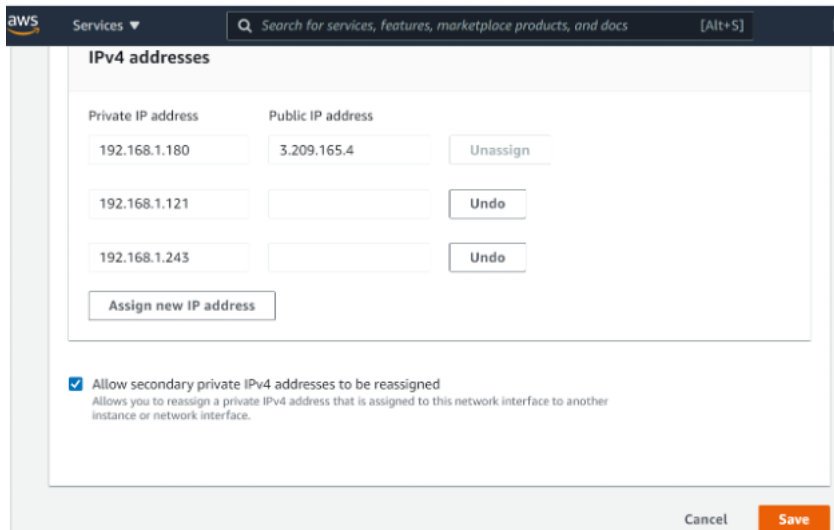


e) Do the **Step 1 to Step 4** for all other interfaces that are attached. Make sure to follow the sequence and maintain the interface order. That is, first detach interface 2 and attach it, and then detach interface 3 and attach it, and so on.

5. You can't detach the management interface from an old EC2 instance. So, move all the secondary IP addresses (if any) on the management interface (primary network interface) of the old EC2 instance to the new EC2 instance. To move an IP address from one interface to another, follow these steps:

- a) In the **AWS Portal**, make sure that both the old and new EC2 instances are in **Stop** state.
- b) Navigate to **Network Interfaces**, and select the management network interface attached to the old EC2 instance.
- c) Click **Actions > Manage IP Address**, and make note of all the secondary IP addresses assigned (if any).
- d) Navigate to the management network interface or primary interface of the new EC2 instance.
- e) Click **Actions > Manage IP Addresses**.
- f) Under **IPv4 Addresses**, click **Assign new IP address**.
- g) Enter the IP addresses, which are noted in the **Step 3**.

- h) Select **Allow secondary private IP addresses to be reassigned** check box.
- i) Click **Save**.



6. Start the new EC2 instance and verify the configuration. After all the configuration is moved, you can delete or keep the old EC2 instance as per your requirement.
7. If any EIP address is attached to the NSIP address of the old EC2 instance, move the old instance NSIP address to the new instance NSIP address.
8. If you want to revert to the old instance, then follow the same steps in the opposite way between the old and new instance.
9. After you move from subscription instance to BYOL instance, a license is required. To install a license follow these steps:
 - Use the licensing portal in the Citrix website to generate a valid license.
 - Upload the license to the instance. For more information, see [VPX ADC - Install a new license](#).

Note:

When you move BYOL instance to subscription instance (paid marketplace instance), you need not install the license. The correct feature set and performance is automatically activated.

Limitations

The management interface can't be moved to the new EC2 instance. So Citrix recommends you manually configure the management interface. For more information, see **Step 5** in the preceding procedure. A new EC2 instance is created with the exact replica of the old EC2 instance but only the NSIP address has a new IP address.

Load balancing servers in different availability zones

November 13, 2024

A VPX instance can be used to load balance servers running in the same availability zone, or in:

- A different availability zone (AZ) in the same AWS VPC
- A different AWS region
- AWS EC2 in a VPC

To enable a VPX instance to load balance servers running outside the AWS VPC that the VPX instance is in, configure the instance to use EIPs to route traffic through the Internet gateway, as follows:

1. Configure a SNIP on the Citrix ADC VPX instance by using the Citrix ADC CLI or the GUI.
2. Enable traffic to be routed out of the AZ, by creating a public facing subnet for the server-side traffic.
3. Add an Internet gateway route to the routing table, using the AWS GUI console.
4. Associate the routing table you just updated with the server-side subnet.
5. Associate an EIP with the server-side private IP address that is mapped to a Citrix ADC SNIP address.

Deploy a high availability pair on AWS

November 13, 2024

You can configure two Citrix ADC VPX instances on AWS as a high availability (HA) active-passive pair. When you configure one instance as the primary node and the other as the secondary node, the primary node accepts connections and manages servers. The secondary node monitors the primary. If for any reason, the primary node is unable to accept connections, the secondary node takes over.

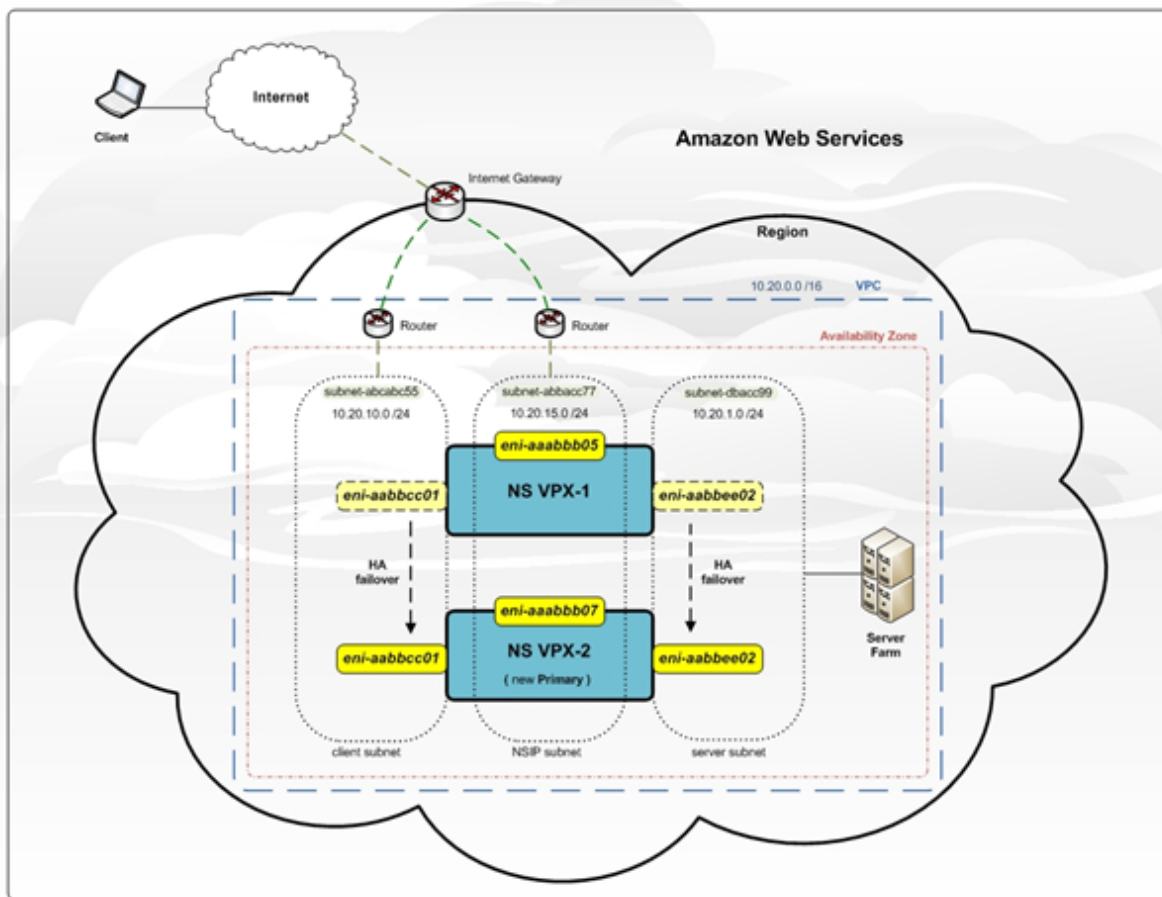
For more information on HA, see [High availability](#).

Before you start your deployment, read the following topics:

- [Prerequisites](#) for the required IAM role privileges.
- [Limitations and usage guidelines](#)

The following figure shows an example of the HA deployment architecture for Citrix ADC VPX instances on AWS.

Figure 1. A Citrix ADC VPX HA Pair on AWS



You can deploy two VPX instances on AWS as an HA pair by using one of the following options:

- Create the instances with IAM Role manually by using the AWS Management Console and then configure HA on them.
- Or automate the high availability deployment by using the Citrix CloudFormation template.

The CloudFormation template significantly decreases the number of steps involved for creating an HA pair, and it automatically creates an IAM Role. This section shows how to deploy a Citrix ADC VPX HA (active-passive) pair by using the Citrix CloudFormation template.

Keep the following points in mind while deploying two Citrix ADC VPX instances as an HA pair.

Points to note

- HA on AWS requires the primary node to have at least two ENIs (one for management and the other for data traffic), and the secondary node to have one management ENI. However, for security purposes, create three ENIs on the primary node, because this setup allows you to segregate private and public network (recommended).

- The secondary node always has one ENI interface (for management) and the primary node can have up to four ENIs.
- The NSIP addresses for each VPX instance in a high availability pair must be configured on the default ENI of the instance.
- Because Amazon does not allow any broadcast/multicast packets in AWS, HA is implemented by migrating data-plane ENIs from the primary to the secondary (new primary) VPX instance when the primary VPX instance fails.
- Because the default (management) ENI cannot be moved to another VPX instance, do not use the default ENI for client and server traffic (data-plane traffic).
- The message `AWSCONFIG_IOCTL_NSAPI_HOTPLUG_INTF` success output 0 in the `/var/log/ns.log` indicates that the two data ENIs have successfully attached to the secondary instance (the new primary).
- Failover might take up to 20 seconds due to the AWS detach/attach ENI mechanism.
- Upon failover, the failed instance always restarts.
- The heartbeat packets are received only on the management interface.
- The configuration file of the primary and secondary VPX instances is synchronized, including the `nsroot` password. The `nsroot` password of the secondary node is set to that of the primary node after the HA configuration synchronization.
- To have access to the AWS API servers, either the VPX instance must have a public IP address assigned or routing must be set up correctly at VPC subnet level pointing to internet gateway of the VPC
- Nameservers/DNS servers are configured at VPC level using DHCP options.
- The Citrix CloudFormation template does not create an HA setup between different availability zones.
- The Citrix CloudFormation template does not create an INC mode.
- The AWS debug messages are available in the log file, `/var/log/ns.log`, on the VPX instance.

Deploy a high availability pair by using the Citrix CloudFormation template

Before start the CloudFormation template, ensure that you complete the following requirements:

- A VPC
- Three subnets within the VPC
- A security group with UDP 3003, TCP 3009–3010, HTTP, SSH ports open
- A key pair
- Create an internet gateway
- Edit route tables for client and management networks to point to the internet gateway

Note:

The Citrix CloudFormation template automatically creates an IAM Role. Existing IAM Roles do not appear in the template.

To launch the Citrix CloudFormation template:

1. Log on to the [AWS marketplace](#) by using your AWS credentials.
2. In the search field, type **NetScaler ADC VPX** to search for the Citrix ADC AMI, and click **Go**.
3. On the search result page, click the desired Citrix ADC VPX offering.
4. Click the **Pricing** tab, to go to **Pricing Information**.
5. Select the region and **Fulfillment Option** as **Netscaler AWS-VPX Cluster**.
6. Click **Continue to Subscribe**.
7. Check the details in the **Subscribe** page and click **Continue to Subscribe**.
8. Select **Fulfillment Option** as **CloudFormation Template**.
9. Select **Software Version** and click **Continue to Launch**.
10. Under **Choose Action**, select **Launch CloudFormation**, and click **Launch**.
11. The **Select Template** page appears. Click **Next**.
12. The **Specify Details** appears. Enter the following details.
 1. Type a **Stack name**. The name must be within 25 characters.
 1. Under **High Availability Configuration**, select **Yes** from the menu for **Create HA pair?**
 1. Under **Virtual Private Network Configuration**, select the VPC that you've already created for **VPC ID**.

```
1 Type **Remote SSH CIDR IP**.  
2  
3 Type **Remote HTTP CIDR IP**.  
4  
5 Type **Remote HTTPS CIDR IP.**
```

Select the key pair that you've already created from the drop-down menu for **Key Pair**.

1. Under Network Interface Configuration

Select **Management Subnetwork**, **Client Subnetwork**, and **Server Subnetwork**. Ensure that you select the correct subnetworks you created within the VPC that you selected under VPC ID in step c.

Add **Primary Management IP**, **Secondary Management IP**, **Client IP**, and **Server IP**. The IP addresses must belong to the same subnets of the respective subnetworks. Alternatively, you can let the template assign the IP addresses automatically.

1. Under **Other Parameters**

Select **m4.large** for **Instant Type**.

Select **default** for **Tenancy Type**.

1. Click **Next**.

1. The **Options** page appears. (This is an optional page.) Click **Next**.

1. The **Review** page appears. Take a moment to review the settings and make necessary changes if necessary.

1. Select the **I acknowledge that AWS CloudFormation might create IAM resources** check box, and then click **Create**.

1. The **CREATE-IN-PROGRESS** status appears. Wait until the status is **CREATE-COMPLETE**. If the status does not change to “COMPLETE,” check the **Events** tab for the reason of failure and recreate the instance with proper configurations.

1. After an IAM resource is created, go to EC2 **Management Console > Instances**. You will find two VPX instances created with IAM role. The primary node is created with three private IP addresses and three network interfaces.

The secondary node is created with one private IP address and one network interface.

Note:

The secondary node is created with one interface by default in AWS. During failover, the interface from the primary node gets attached to the secondary node (the new primary node) and gets detached from the original primary node (the new secondary node).

1. Log on to the primary node with user name nsroot and the instance ID as the password. From the GUI, go to **System > High Availability**. The Citrix ADC VPX HA pair appears.

Next, configure the HA pairing on both the instances. Configure the instance with three ENIs before configuring HA on the instance with one ENI). Use the add HA node command, from within the VPX CLI, or from the GUI.

```
add HA node <private IP of the first instance>
```

```
add HA node <private IP of the second instance>
```

After you run the “add HA node” commands, the two nodes form an HA pair, and configuration information is synchronized between the two VPX instances.

Configuring SR-IOV on a high availability setup

Support for SR-IOV interfaces in a high availability setup is available from Citrix ADC release 12.0 57.19 onwards. For more information about how to configure SR-IOV, see [Configuring Citrix ADC VPX instances to Use SR-IOV Network Interface](#).

Related resources

[High availability across AWS availability zones](#)

High availability across AWS availability zones

November 13, 2024

You can configure two Citrix ADC VPX instances on two different subnets or two different AWS availability zones, as a high availability active-passive pair in Independent Network Configuration (INC) mode. If for any reason, the primary node is unable to accept connections, the secondary node takes over.

For more information about high availability, see [High availability](#). For more information about INC, see [Configuring high availability nodes in different subnets](#).

Points to note

- Read the following documents before you start your deployment:
 - [AWS terminology](#)
 - [Prerequisites](#)
 - [Limitations and usage guidelines](#)
- The VPX high availability pair can either reside in the same availability zone in a different subnet or in two different AWS availability zones.
- Citrix recommends that you use different subnets for management (NSIP), client traffic (VIP), and back-end server (SNIP).
- High availability must be set in Independent Network Configuration (INC) mode for a failover to work.
- The two instances also should have port 3003 open for UDP traffic as that is used for heartbeats.
- The management subnets of both the nodes should also have access to internet or to AWS API server through internal NAT so that the rest APIs are functional.
- IAM role should have E2 permission for the public IP or elastic IP (EIP) migration.

How high availability across AWS availability zones works

Upon failover, the EIP of the VIP of the primary instance migrates to the secondary, which takes over as the new primary. In the failover process, AWS API

1. Checks the virtual servers that have IP sets attached to them.
2. Finds the IP address that has an associated public IP, from the two IP addresses the virtual server is listening on. One that is directly attached to the virtual server, and one that is attached through the IP set.
3. Reassociates the public IP (EIP) to the private IP belonging to the new primary VIP.

Note:

To protect your network from attacks such as denial-of-service (DoS), when using an EIP, you can create security groups in AWS to restrict the IP access. For high availability, you can switch from EIP to a private IP movement solution as per your deployments.

How to deploy a VPX high availability pair across different AWS zones

The following is the summary of steps for deploying a VPX pair on two different subnets or two different AWS availability zones.

1. Create an Amazon virtual private cloud.
2. Deploy two VPX instances in two different availability zones or in the same zone but in different subnets.
3. Configure high availability
 - a) Set up high availability in INC mode in both the instances.
 - b) Add IP set in both the instances.
 - c) Bind IP set in both the instances to the VIP.
 - d) Add a virtual server in the primary instance.

For steps 1 and 2, use the AWS console. For steps 3, use the Citrix ADC VPX GUI or the CLI.

Step 1. Create an Amazon virtual private cloud (VPC).

Step 2. Deploy two VPX instance in two different availability zones or in the same zone but in different subnets. Attach an EIP to the VIP of the primary VPX.

For more information about how to create a VPC and deploy a VPX instance on AWS, see [Deploy a Citrix ADC VPX standalone instance on AWS](#) and [Scenario: standalone instance](#).

Step 3. Configure high availability. You can use the Citrix ADC VPX CLI or the GUI to set up high availability.

Configure high availability by using the CLI

1. Set up high availability in INC mode in both the instances.

On the primary node:

```
add ha node 1 <sec_ip> -inc ENABLED
```

On the secondary node:

```
add ha node 1 <prim_ip> -inc ENABLED
```

<sec_ip> refers to the private IP address of the management NIC of the secondary node

<prim_ip> refers to the private IP address of the management NIC of the primary node

2. Add IP set in both the instances.

Type the following command on both the instances.

```
add ipset <ipsetname>
```

1. Bind IP set to the VIP set on both the instances.

Type the following command on both the instances:

```
add ns ip <secondary vip> <subnet> -type VIP
```

```
bind ipset <ipsetname> <secondary VIP>
```

Note:

You can bind the IP set to the primary VIP or to the secondary VIP. However, if you bind the IP set to the primary VIP, use the secondary VIP to add to the virtual server, and conversely.

2. Add a virtual server on the primary instance.

Type the following command:

```
add <server_type> vserver <vserver_name> <protocol> <primary_vip> <port> -ipset <ipset_name>
```

Configure high availability by using the GUI

1. Set up high availability in INC mode on both the instances
2. Log on to the primary node with user name nsroot and instance ID as password.
3. From the GUI, go to **Configuration > System > High Availability**. Click **Add**.
4. At the **Remote Node IP address** field, add the private IP address of the management NIC of the secondary node.

5. Select **Turn on NIC (Independent Network Configuration)** mode on self node.
6. Under **Remote System Login Credential**, add the user name and password for the secondary node and click **Create**.
7. Repeat the steps in the secondary node.
8. Add IP set and bind IP set to the VIP set on both the instances.
9. From the GUI, navigate to **System > Network > IPs > Add**.
10. Add the required values for IP Address, Netmask, IP Type (virtual IP) and click **Create**.
11. Navigate to **System > Network > IP Sets > Add**. Add an IP set name and click **Insert**.
12. From the IPV4s page, select the virtual IP and click **Insert**. Click **Create** to create the IP set.
13. Add a virtual server in the primary instance

From the GUI, go to **Configuration > Traffic Management > Virtual Servers > Add**.

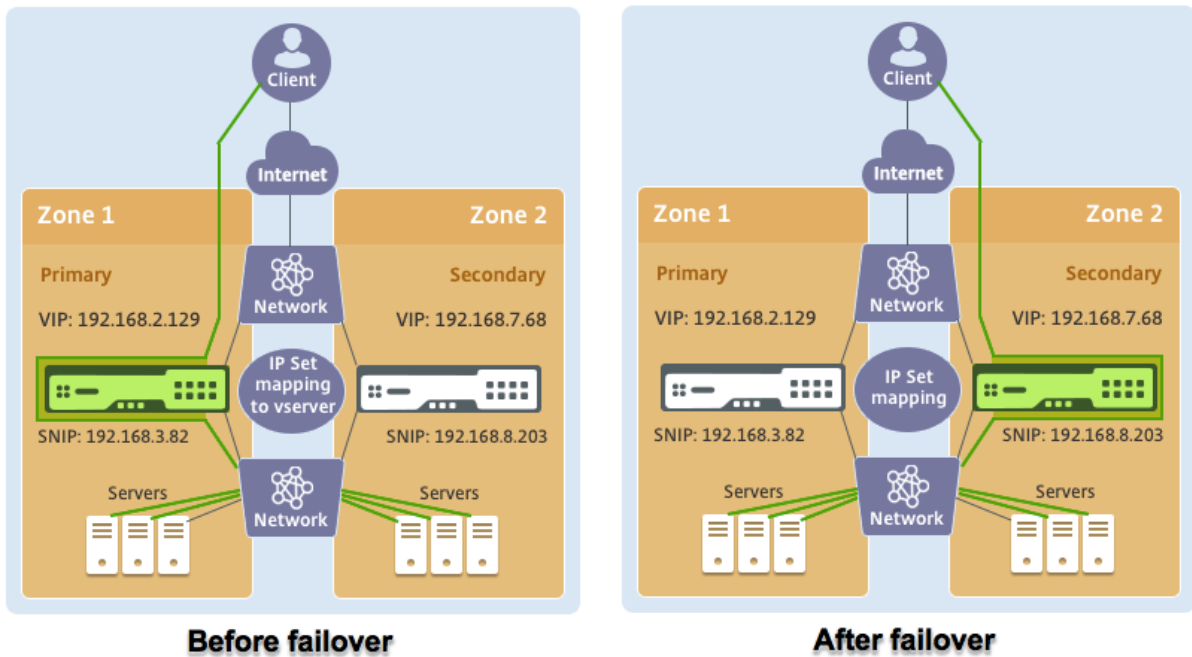
Load Balancing Virtual Server | [Export as a Template](#)

Basic Settings	
Name	vserver1
Protocol	HTTP
State	● DOWN
IP Address	192.168.2.129
Port	80
Traffic Domain	0
Listen Priority	-
Listen Policy Expression	NONE
Redirection Mode	IP
Range	1
IPset	ipset123
RHI State	PASSIVE
AppFlow Logging	ENABLED
Retain Connections on Cluster	NO

Scenario

In this scenario, a single VPC is created. In that VPC, two VPX instances are created in two availability zones. Each instance has three subnets - one for management, one for client, and one for back-end server. An EIP is attached to the VIP of the primary node.

Diagram: This diagram illustrates the Citrix ADC VPX high availability setup in INC mode, on AWS



For this scenario, use CLI to configure high availability.

1. Set up high availability in INC mode on both the instances.

Type the following commands on the primary and the secondary nodes.

On primary:

```
add ha node 1 192.168.6.82 -inc enabled
```

Here, 192.168.6.82 refers to the private IP address of the management NIC of the secondary node.

On secondary:

```
add ha node 1 192.168.1.108 -inc enabled
```

Here, 192.168.1.108 refers to the private IP address of the management NIC of the primary node.

2. Add an IP set and bind the IP set to the VIP on both the instances

On primary:

```
add ipset ipset123
```

```
add ns ip 192.168.7.68 255.255.255.0 -type VIP
```

```
bindipset ipset123 192.168.7.68
```

On secondary:

```
add ipset ipset123
```

```
add ns ip 192.168.7.68 255.255.255.0 -type VIP
```


bind ipset ipset123 192.168.7.68

3. Add a virtual server on the primary instance.

Type the following command:

add lbserver vserver1 http 192.168.2.129 80 -ipset ipset123

4. Save the configuration.

ID	IP Address	Host Name	Master State	Node State	INC	Synchronization State
0	192.168.1.108		Primary	UP	ENABLED	ENABLED
1	192.168.6.82		Secondary	UP	ENABLED	SUCCESS

5. After a forced failover, the secondary becomes the new primary.

ID	IP Address	Host Name	Master State	Node State	INC	Synchronization State
0	192.168.1.108		Secondary	UP	ENABLED	SUCCESS
1	192.168.6.82		Primary	UP	ENABLED	ENABLED

Add back-end AWS Autoscaling service

November 13, 2024

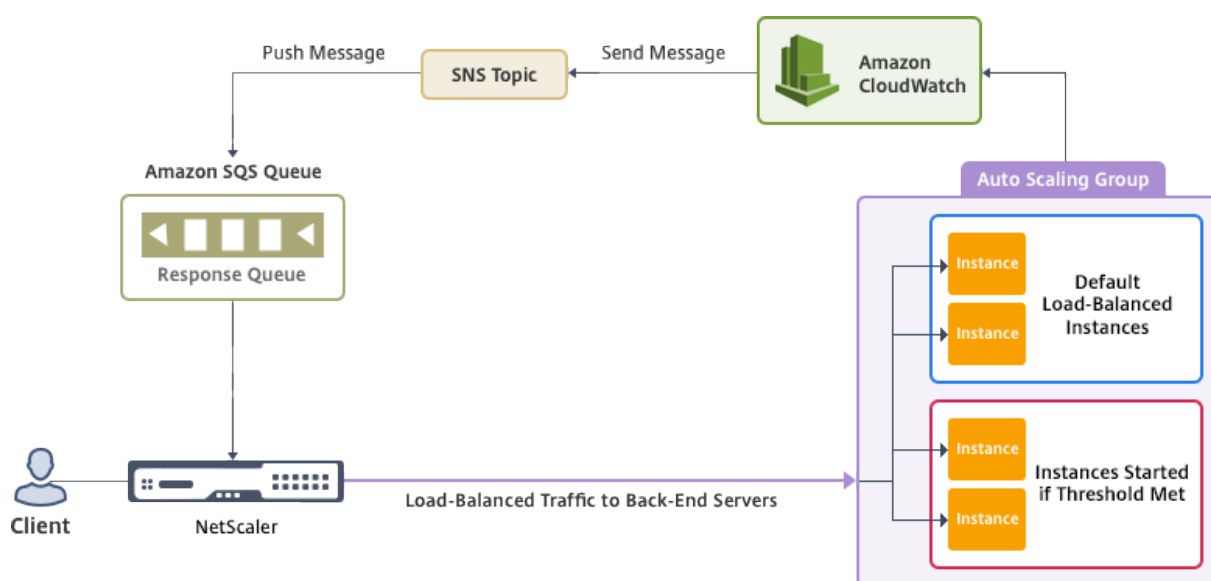
Efficient hosting of applications in a cloud involves easy and cost-effective management of resources depending on the application demand. To meet increasing demand, you have to scale network resources upward. Whether demand subsides, you need to scale down to avoid the unnecessary cost of idle resources. To minimize the cost of running the application by deploying only as many instances as are necessary during any given time, you constantly have to monitor traffic, memory and CPU use, and so on. However, monitoring traffic manually is cumbersome. For the application environment to scale up or down dynamically, you must automate the processes of monitoring traffic and of scaling resources up and down whenever necessary.

Integrated with AWS Auto Scaling service, the Citrix ADC VPX instance provides the following advantages:

- **Load balance and management:** Auto configures servers to scale up and scale down, depending on demand. The VPX instance auto detects Autoscale groups in the back-end subnet and allows a user to select the Autoscale groups to balance the load. All of this is done by auto configuring the virtual and subnet IP addresses on the VPX instance.

- **High availability:** Detects Autoscale groups that span multiple availability zones and load-balance servers.
- **Better network availability:** The VPX instance supports:
 - Back-end servers on different VPCs, by using VPC peering
 - Back-end servers on same placement groups
 - Back-end servers on different availability zones
- **Graceful connection termination:** Removes Autoscale servers gracefully, avoiding loss of client connections when scale-down activity occurs, by using the Graceful Timeout feature.

Diagram: AWS Autoscaling service with a Citrix ADC VPX Instance



This diagram illustrates how the AWS Autoscaling service is compatible with a Citrix ADC VPX instance (Load balancing virtual server). For more information, see the following AWS topics.

- [Autoscaling groups](#)
- [CloudWatch](#)
- [Simple Notification Service \(SNS\)](#)
- [Simple Queue Service \(Amazon SQS\)](#)

Before you begin

Before you start using Autoscaling with your Citrix ADC VPX instance, you must complete the following tasks.

1. Read the following topics:

- [Prerequisites](#)

- [Limitation and usage guidelines](#)
2. Create a Citrix ADC VPX instance on AWS according to your requirement.
 - For more information about how to create a Citrix ADC VPX standalone instance, see [Deploy a Citrix ADC VPX standalone instance on AWS](#) and [Scenario: standalone instance](#)
 - For more information about how to deploy VPX instances in HA mode, see [Deploy a high availability pair on AWS](#).

Note:

Citrix recommends the CloudFormation template for creating Citrix ADC VPX instances on AWS.

Citrix recommends you create three interfaces: one for management (NSIP), one for client-facing LB virtual server (VIP), and one for subnet IP (NSIP).

3. Create an AWS Autoscale group. If you don't have an existing Autoscaling configuration, you must:
 - a) Create a Launch Configuration
 - b) Create an Autoscaling Group
 - c) Verify the Autoscaling Group

For more information, see <http://docs.aws.amazon.com/autoscaling/latest/userguide/GettingStartedTutorial.html>.
4. In the AWS Autoscale group, you must specify at least one scale-down policy. The Citrix ADC VPX instance supports only the Step scaling policy. The Simple scaling policy and Target tracking scaling policy are not supported for Autoscale group.

Add the AWS Autoscaling service to a Citrix ADC VPX instance

You can add the Autoscaling service to a VPX instance with a single click by using the GUI. Complete these steps to add the Autoscaling service to the VPX instance:

1. Log on to the VPX instance by using your credentials for `nsroot`.
2. When you log on to the Citrix ADC VPX instance for the first time, you see the default Cloud Profile page. Select the AWS Autoscaling group from the drop-down menu and click **Create** to create a cloud profile. Click **Skip** if you want to create the cloud profile later.

Points to keep in mind while creating a Cloud Profile: By default the CloudFormation Template creates and attaches the below IAM Role.

```
{
```

```
“Version”: “2012-10-17”,
“Statement”: [
  {
    “Action”: [
      “ec2:DescribeInstances”,
      “ec2:DescribeNetworkInterfaces”,
      “ec2:DetachNetworkInterface”,
      “ec2:AttachNetworkInterface”,
      ”ec2:StartInstances”,
      ”ec2:StopInstances”,
      ”ec2:RebootInstances”,
      “autoscaling:*”,
      ”sns:*”,
      “sqs:*”
    ],
    “iam: SimulatePrincipalPolicy”
    “iam: GetRole”
  ],
  “Resource”: “*”,
  “Effect”: “Allow”
}
]
```

Ensure the IAM Role of instance has proper permissions.

- The virtual server IP address is autopopulated from the free IP address available to the VPX instance. <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/MultipleIP.html#ManageMultipleIP>
- Autoscale group is prepopulated from the Autoscale group configured on your AWS account. <http://docs.aws.amazon.com/autoscaling/latest/userguide/AutoScalingGroup.html>.
- While selecting the Autoscaling Group protocol and port, ensure your servers listen on those protocol and ports, and you bind the correct monitor in the service group. By default, TCP monitor is used.

- For SSL Protocol type Autoscaling, after you create the Cloud Profile the load balance virtual server or service group is down because of a missing certificate. You can bind the certificate to the virtual server or service group manually.
- Select the Graceful Timeout option to remove Autoscale servers gracefully. If this option is not selected the server is the Autoscale group is removed immediately after the load goes down, which might cause service interruption for the existing connected clients. Selecting Graceful and giving a timeout means in the event of scale down. The VPX instance does not remove the server immediately but marks one of the servers for graceful deletion. During this period, the instance does not allow new connections to this server. Existing connection are served until the timeout occurs, and after timeout the VPX instance removes the server.

Figure: Default Cloud Profile page

Name

Virtual Server IP Address*

Load Balancing Server Protocol*

Load Balancing Server Port*

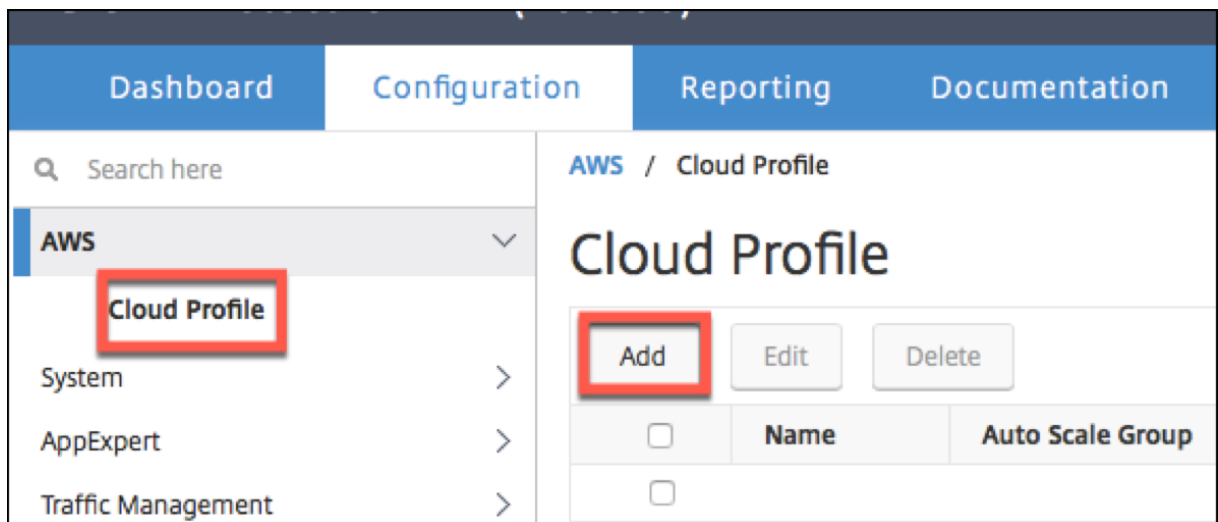
Auto Scale Group*

Auto Scale Group Protocol

Auto Scale Group Port*

Select this option to drain the connections gracefully. Else the connections will be dropped
 Graceful

3. After the first time logon if you want to create Cloud Profile, on the GUI go to **System > AWS > Cloud Profile** and click **Add**.



The **Create Cloud Profile** configuration page appears.

Citrix NetScaler VPX (3000)

Dashboard Configuration Reporting Documentation Downloads

← Create Cloud Profile

Name
SharePoint_CloudProfile

Virtual Server IP Address*
21.0.2.29

Load Balancing Server Protocol
HTTP

Load Balancing Server Port
80

Auto Scale Group*
SharePoint

Auto Scale Group Protocol
HTTP

Auto Scale Group Port
80

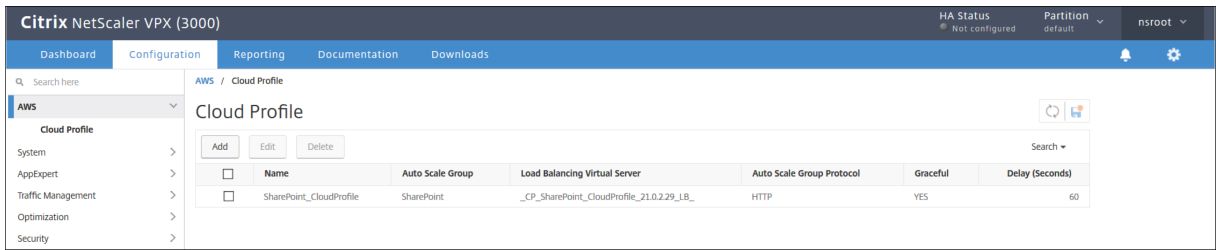
Select this option to drain the connections gracefully. Else the connections will be dropped in the event of scale down.

Graceful

Delay (Seconds)
60

Create Close

Cloud Profile creates a Citrix ADC load-balancing virtual server and a service group with members as the servers of the Autoscaling group. Your back-end servers must be reachable through the SNIP configured on the VPX instance.



Note:

To view Autoscale-related information in the AWS console, go to **EC2 > Dashboard > Auto Scaling > Auto Scaling Group**.

Configure a Citrix ADC VPX instance to use SR-IOV network interface

November 13, 2024

Note

Support for SR-IOV interfaces in a high availability setup is available from Citrix ADC release 12.0 57.19 onwards.

After you have created a Citrix ADC VPX instance on AWS, you can configure the virtual appliance to use SR-IOV network interfaces, by using AWS CLI.

In all Citrix ADC VPX models, except Citrix ADC VPX AWS Marketplace Editions of 3G and 5G, SR-IOV is not enabled in the default configuration of a network interface.

Before you start the configuration, read the following topics:

- [Prerequisites](#)
- [Limitations and Usage Guidelines](#)

This section includes the following topics:

- Change the Interface Type to SR-IOV
- Configure SR-IOV on a High Availability Setup

Change the interface type to SR-IOV

You can run the show interface summary command to check the default configuration of a network interface.

Example 1: The following CLI screen capture shows the configuration of a network interface where SR-IOV is enabled by default on Citrix ADC VPX AWS Marketplace Editions of 3G and 5G.

```
> show interface summary
-----
Interface  MTU      MAC              Suffix
-----
1  1/1      1500      0a:1e:2e:17:a2:37  Intel 82599 10G VF Interface
2  LO/1     1500      0a:1e:2e:17:a2:37  Netscaler Loopback interface
Done
```

Example 2: The following CLI screen capture shows the default configuration of a network interface where SR-IOV is not enabled.

```
Done
[> sh int s
-----
Interface  MTU      MAC              Suffix
-----
1  1/1      1500      12:fc:04:c5:d0:12  NetScaler Virtual Interface
2  LO/1     1500      12:fc:04:c5:d0:12  Netscaler Loopback interface
Done
>
```

For more information about changing the interface type to SR-IOV, see <http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/sriov-networking.html>

To change the interface type to SR-IOV:

1. Shut down the Citrix ADC VPX instance running on AWS.
2. To enable SR-IOV on the network interface, type the following command in the AWS CLI.
\$ aws ec2 modify-instance-attribute --instance-id <instance_id> --sriov-net-support simple
3. To check if SR-IOV has been enabled, type the following command in the AWS CLI.
\$ aws ec2 describe-instance-attribute --instance-id <instance_id> --attribute sriovNetSupport

Example 3: Network interface type changed to SR-IOV, by using AWS CLI.

```
aws ec2 modify-instance-attribute --instance-id i-008c1230aaf303bee --sriov-net-support simple
aws ec2 describe-instance-attribute --instance-id i-008c1230aaf303bee --attribute sriovNetSupport
{
  "InstanceId": "i-008c1230aaf303bee",
  "SriovNetSupport": {
    "Value": "simple"
  }
}
```

If SR-IOV is not enabled, value for SriovNetSupport is absent.

Example 4: In the following example, SR-IOV support is not enabled.


```
{
  "InstanceId": "i-0c3e84cfa65b04cc8",
  "SriovNetSupport": {}
}
```

3. Power on the VPX instance. To see the changed status of the network interface, type “show interface summary” in CLI.

Example 5: The following screen capture shows the network interfaces with SR-IOV enabled. The interfaces 10/1, 10/2, 10/3 are SR-IOV enabled.

```
> show interface summary
-----
Interface  MTU      MAC              Suffix
-----
1  10/1      1500            0a:1e:2e:17:a2:37  Intel 82599 10G VF Interface
2  10/2      1500            0a:df:17:0a:fe:83  Intel 82599 10G VF Interface
3  10/3      1500            0a:de:5d:31:bf:c3  Intel 82599 10G VF Interface
4  L0/1      1500            0a:1e:2e:17:a2:37  Netscaler Loopback interface
Done
```

These steps complete the procedure to configure VPX instances to use SR-IOV network interfaces.

Configure SR-IOV on a high availability setup

High availability is supported with SR-IOV interfaces from Citrix ADC release 12.0 build 57.19 onwards.

If the high availability setup was deployed manually or by using the Citrix CloudFormation template for Citrix ADC version 12.0 56.20 and lower, the IAM role attached to the high availability setup must have the following privileges:

- ec2:DescribeInstances
- ec2:DescribeNetworkInterfaces
- ec2:DetachNetworkInterface
- ec2:AttachNetworkInterface
- ec2:StartInstances
- ec2:StopInstances
- ec2:RebootInstances
- autoscaling:*
- sns:*
- sqs:*
- iam:SimulatePrincipalPolicy

- iam:GetRole

By default, the Citrix CloudFormation template for Citrix ADC version 12.0 57.19 automatically adds the required privileges to the IAM role.

Note:

A high availability setup with SR-IOV Interfaces take around 100 seconds of down time.

Related resources:

For more information about IAM roles, see [AWS documentation](#).

Upgrade a Citrix ADC VPX instance on AWS

November 13, 2024

You can upgrade the EC2 instance type, throughput, software edition, and the system software of a Citrix ADC VPX running on AWS. For certain types of upgrades, Citrix recommends using the High Availability Configuration method to minimize downtime.

Note:

- Citrix ADC software release 10.1.e-124.1308.e or later for a Citrix ADC VPX AMI (including both utility license and customer license) does not support the M1 and M2 instance families.
- Because of changes in VPX instance support, downgrading from 10.1.e-124 or a later release to 10.1.123.x or an earlier release is not supported.
- Most of the upgrades do not require the launch of a new AMI, and the upgrade can be done on the current Citrix ADC AMI instance. If you do want to upgrade to a new Citrix ADC AMI instance, use the high availability configuration method.

Change the EC2 instance type of a Citrix ADC VPX instance on AWS

If your Citrix ADC VPX instances are running release 10.1.e-124.1308.e or later, you can change the EC2 instance type from the AWS console as follows:

1. Stop the VPX instance.
2. Change the EC2 instance type from the AWS console.
3. Start the instance.

You can also use the above procedure to change the EC2 instance type for a release, earlier than 10.1.e-124.1308.e, unless you want to change the instance type to M3. In that case, you must first follow the

standard Citrix ADC upgrade procedure, at , to upgrade the Citrix ADC software to 10.1.e-124 or a later release, and then follow the above steps.

Upgrade the throughput or software edition of a Citrix ADC VPX instance on AWS

To upgrade the software edition (for example, to upgrade from Standard to Premium edition) or throughput (for example, to upgrade from 200 mbps to 1000mbps), the method depends on the instance's license.

Using a customer license (Bring-Your-Own-License)

If you are using a customer license, you can purchase and download the new license from the Citrix website, and then install the license on the VPX instance. For more information about downloading and installing a license from the Citrix website, see the VPX Licensing Guide.

Using a utility license (Utility license with hourly fee)

AWS does not support direct upgrades for fee-based instances. To upgrade the software edition or throughput of a fee based Citrix ADC VPX instance, launch a new AMI with the desired license and capacity and migrate the older instance configuration to the new instance. This can be achieved by using a Citrix ADC high availability configuration as described in Upgrade to a new Citrix ADC AMI instance by using a Citrix ADC high availability configuration subsection in this page.

Upgrade the system software of a Citrix ADC VPX instance on AWS

If you need to upgrade a VPX instance running 10.1.e-124.1308.e or a later release, follow the standard Citrix ADC upgrade procedure at [Upgrade and downgrade a Citrix ADC appliance](#).

If you need to upgrade a VPX instance running a release older than 10.1.e-124.1308.e to 10.1.e-124.1308.e or a later release, first upgrade the system software, and then change the instance type to M3 as follows:

1. Stop the VPX instance.
2. Change the EC2 instance type from the AWS console.
3. Start the instance.

Upgrade to a new Citrix ADC AMI instance by using a Citrix ADC high availability configuration

To use the high availability method of upgrading to a new Citrix ADC AMI instance, perform the following tasks:

- Create a new instance with the desired EC2 instance type, software edition, throughput, or software release from the AWS marketplace.
- Configure high availability between the old instance (to be upgraded) and the new instance. After high availability is configured between the old and the new instance, configuration from the old instance is synchronized to the new instance.
- Force an HA failover from the old instance to the new instance. As a result, the new instance becomes primary and starts receiving traffic.
- Stop, and reconfigure or remove the old instance from AWS.

Prerequisites and points to consider

- Ensure you understand how high availability works between two Citrix ADC VPX instances on AWS. For more information about high availability configuration between two Citrix ADC VPX instances on AWS, see [Deploy a high availability pair on AWS](#).
- You must create the new instance in the same availability zone as the old instance, having the exact same security group and subnet.
- High availability setup requires access and secret keys associated with the user's AWS Identity and Access Management (IAM) account for both instances. If the correct key information is not used when creating VPX instances, the HA setup fails. For more information about creating an IAM account for a VPX instance, see [Prerequisites](#).
 - You must use the EC2 console to create the new instance. You cannot use the AWS 1-click launch, because it does not accept the access and secret keys as the input.
 - The new instance should have only one ENI interface.

To upgrade a Citrix ADC VPX Instance by using a high availability configuration, follow these steps:

1. Configure high availability between the old and the new instance. To configure high availability between two Citrix ADC VPX instances, at the command prompt of each instance, type:
 - add ha node <nodeID> <IPaddress of the node to be added>
 - save config

Example:

At the command prompt of the old instance, type:

```
1 add ha node 30 192.0.2.30
2 Done
```

At the command prompt of the new instance, type:

```
1 add ha node 10 192.0.2.10
2 Done
```

Note the following:

- In the HA setup, the old instance is the primary node and the new instance is the secondary node.
- The NSIP IP address is not copied from the old instance to the new instance. Therefore, after the upgrade, your new instance has a different management IP address from the previous one.
- The nsroot account password of the new instance is set to that of the old instance after HA synchronization.

For more information about high availability configuration between two Citrix ADC VPX instances on AWS, see [Deploy a high availability pair on AWS](#).

2. Force an HA failover. To force a failover in a high availability configuration, at the command prompt of either of the instances, type:

```
1 force HA failover
```

As the result of forcing a failover, the ENIs of the old instance are migrated to the new instance and traffic flows through the new instance (the new primary node). The old instance (the new secondary node) restarts.

If the following warning message appears, type N to abort the operation:

```
1 [WARNING]:Force Failover may cause configuration loss, peer health
   not optimum. Reason(s):
2 HA version mismatch
3 HA heartbeats not seen on some interfaces
4 Please confirm whether you want force-failover (Y/N)?
```

The warning message appears because the system software of the two VPX instances is not HA compatible. As a result, the configuration of the old instance cannot be automatically synced to the new instance during a forced failover.

Following is the workaround for this issue:

1. At the Citrix ADC shell prompt of the old instance, type the following command to create a backup of the configuration file (ns.conf):
copy /nsconfig/ns.conf to /nsconfig/ns.conf.bkp

- Remove the following line from the backup configuration file (ns.conf.bkp):

```
set ns config -IPAddress \<IP\> -netmask \<MASK\>
```

For example, set ns config -IPAddress 192.0.2.10 -netmask 255.255.255.0

- Copy the old instance's backup configuration file (ns.conf.bkp) to the /nsconfig directory of the new instance.
- At the Citrix ADC shell prompt of the new instance, type the following command to load the old instance's configuration file (ns.conf.bkp) on the new instance:

```
batch -f /nsconfig/ns.conf.bkp
```

- Save the configuration on the new instance.

```
Save config
```

- At the command prompt of either of the nodes, type the following command to force a failover. Then type Y, for the warning message to confirm the force failover operation:

```
force ha failover
```

Example:

```
1 force ha failover
2
3 [WARNING]:Force Failover may cause configuration loss, peer health
  not optimum.
4 Reason(s):
5 HA version mismatch
6 HA heartbeats not seen on some interfaces
7 Please confirm whether you want force-failover (Y/N)? Y
```

- Remove the HA configuration, so that the two instances are no longer in an HA configuration. First remove the HA configuration from the secondary node and then remove the HA configuration from the primary node.

To remove an HA configuration between two Citrix ADC VPX instances, at the command prompt of each instance, type:

```
1 > remove ha node \<nodeID\>
2 > save config
```

For more information about high availability configuration between two VPX instances on AWS, see [Deploy a high availability pair on AWS](#).

Example:

At the command prompt of the old instance (new secondary node), type:

```
1 > remove ha node 30
2 Done
```

```
3 > save config
4 Done
```

At the command prompt of the new instance (new primary node), type:

```
1 > remove ha node 10
2 Done
3 > save config
4 Done
```

Troubleshoot a VPX instance on AWS

November 12, 2024

Amazon does not provide console access to a Citrix ADC VPX instance. To troubleshoot, you have to use the AWS GUI to view the activity log. You can debug only if the network is connected. To view an instance’s system log, right-click the instance and select system log.

Citrix provides support for fee based Citrix ADC VPX instances (utility license with hourly fee) on AWS. To file a support case, find your AWS account number and support PIN code, and call Citrix support. You will also be asked for your name and email address. To find the support PIN, log on to the VPX GUI and navigate to the System page.

Here is an example of a system page showing the support PIN.

The screenshot shows the NetScaler System Information page. On the left is a navigation menu with 'System' selected. The main content area is titled 'System' and has tabs for 'System Information', 'System Sessions', and 'System Network'. Below the tabs are buttons for 'System Upgrade', 'Reboot', 'Migration', 'Statistics', and 'Call Home'. The 'System Information' section displays various system parameters:

Citrix ADC IP Address	
Netmask	
Node	Standalone
Technical Support PIN	
Time Zone	Coordinated Universal Time
System Time	Wed, 18 Dec 2019 06:16:59 UTC
Last Config Changed Time	Wed, 18 Dec 2019 06:16:40 UTC
Last Config Saved Time	Wed, 18 Dec 2019 05:41:16 UTC

The 'Hardware Information' section is partially visible below:

Platform	NetScaler Virtual Appliance 450040
Manufactured on	2/17/2009
CPU	2305 MHZ
Host Id	
Serial no	
Encoded serial no	
Citrix ADC UUID	

AWS FAQs

November 13, 2024

- **Does a Citrix ADC VPX instance support the encrypted volumes in AWS?**

Encryption and decryption happen at the hypervisor level, and hence it works seamlessly with any instance. For more information about the encrypted volumes see the following AWS document:

<https://docs.aws.amazon.com/kms/latest/developerguide/services-ebs.html>

- **What is the best way to provision Citrix ADC VPX instance on AWS?**

You can provision a Citrix ADC VPX instance on AWS by any of the following ways:

- AWS CloudFormation Template (CFT) in AWS marketplace
- Citrix ADM
- AWS Quick Starts
- Citrix AWS CFTs in Git hub
- Citrix Terraform Scripts in Git hub
- Citrix Ansible Playbooks in Git hub
- AWS EC2 launch workflow

You can choose any of the listed options based on the automation tool that you use.

For more details about the options, see [Citrix ADC VPX on AWS](#).

Deploy a Citrix ADC VPX instance on Microsoft Azure

November 13, 2024

When you deploy a Citrix ADC VPX instance on Microsoft Azure Resource Manager (ARM), you can use the Azure cloud computing capabilities and use Citrix ADC load balancing and traffic management features for your business needs. You can deploy Citrix ADC VPX instances on Azure Resource Manager either as standalone instances or as high availability pairs in active-standby modes.

You can deploy a Citrix ADC VPX instance on the Microsoft Azure in two ways:

- Through Azure Marketplace. The Citrix ADC VPX virtual appliance is available as an image in the Microsoft Azure Marketplace.
- Using the Citrix ADC Azure Resource Manager (ARM) json template available on GitHub. For more information, see the [GitHub repository for Citrix ADC solution templates](#).

Prerequisite

You need some prerequisite knowledge before deploying a Citrix VPX instance on Azure.

- Familiarity with Azure terminology and network details. For information, see [Azure terminology](#).
- Knowledge of a Citrix ADC appliance. For detailed information the Citrix ADC appliance, see [Citrix ADC](#)
- Knowledge of Citrix ADC networking. See the [Networking](#) topic.

How a Citrix ADC VPX instance works on Azure

In an on-premises deployment, a Citrix ADC VPX instance requires at least three IP addresses:

- Management IP address, called NSIP address
- Subnet IP (SNIP) address for communicating with the server farm
- Virtual server IP (VIP) address for accepting client requests

For more information, see [Network architecture for Citrix ADC VPX instances on Microsoft Azure](#).

Note:

VPX virtual appliances can be deployed on any instance type that has two or more Intel VT-X cores and more than 2 GB memory. For more information on system requirements, see [Citrix ADC VPX data sheet](#). Currently, Citrix ADC VPX instance supports only the Intel processors.

In an Azure deployment, you can provision a Citrix ADC VPX instance on Azure in three ways:

- Multi-NIC multi-IP architecture
- Single NIC multi IP architecture
- Single NIC single IP

Depending on your need, you can use any of these supported architecture types.

Multi-NIC multi-IP architecture

In this deployment type, you can have more than one network interfaces (NICs) attached to a VPX instance. Any NIC can have one or more IP configurations - static or dynamic public and private IP addresses assigned to it.

For more information, see the following use cases:

- [Configure a high-availability setup with multiple IP addresses and NICs](#)

- [Configure a high-availability setup with multiple IP addresses and NICs by using PowerShell commands](#)

Note:

To avoid MAC moves and interface mutes on Azure environments, Citrix recommends you to create a VLAN per data interface (without tag) of ADC VPX instance and bind the primary IP of NIC in Azure. For more information, see [CTX224626](#) article.

Single NIC multi IP architecture

In this deployment type, one network interfaces (NIC) associated with multiple IP configurations - static or dynamic public and private IP addresses assigned to it.

For more information, see the following use cases:

- [Configure multiple IP addresses for a Citrix ADC VPX standalone instance](#)
- [Configure multiple IP addresses for a Citrix ADC VPX standalone instance by using PowerShell commands](#)

Single NIC single IP

In this deployment type, one network interfaces (NIC) associated with a single IP address, which is used to perform the functions of NSIP, SNIP, and VIP.

For more information, see the following use case:

- [Configure a Citrix ADC VPX standalone instance](#)

Note:

The single IP mode is available only in Azure deployments. This mode is not available for a Citrix ADC VPX instance on your premises, on AWS, or in other type of deployment.

Citrix ADC VPX licensing

A Citrix ADC VPX instance on Azure requires a license. The following licensing options are available for Citrix ADC VPX instances running on Azure.

- **Subscription-based licensing:** Citrix ADC VPX appliances are available as paid instances on Azure Marketplace. Subscription-based licensing is a pay-as-you-go option. Users are charged hourly. The following VPX models and license types are available on Azure Marketplace.

VPX model	License Type
VPX10	Standard, Advanced, Premium
VPX200	Standard, Advanced, Premium
VPX1000	Standard, Advanced, Premium
VPX3000	Standard, Advanced, Premium

Citrix provides technical support for subscription-based license instances. To file a support case, see [Support for Citrix ADC on Azure –Subscription license with hourly price](#).

- **Bring your own license (BYOL):** If you bring your own license (BYOL), see the VPX Licensing Guide at <http://support.citrix.com/article/CTX122426>. You have to:
 - Use the licensing portal within Citrix website to generate a valid license.
 - Upload the license to the instance.
- **Citrix ADC VPX Check-In/Check-Out licensing:** For more information, see [Citrix ADC VPX Check-In/Check-Out Licensing](#).

Starting with Citrix ADC release 12.0 56.20, VPX Express for on-premises and cloud deployments does not require a license file. For more information on Citrix ADC VPX Express see the “Citrix ADC VPX Express license” section in [Citrix ADC Licensing Overview](#).

Note:

Regardless of the subscription-based hourly license bought from Azure Marketplace, in rare cases, the Citrix ADC VPX instance deployed on Azure might come up with a default Citrix ADC license. This happens due to issues with Azure Instance Metadata Service (IMDS).

Do a warm restart before making any configuration change on the Citrix ADC VPX instance, to enable the correct Citrix ADC VPX license.

Limitations

Running the Citrix ADC VPX load balancing solution on ARM imposes the following limitations:

- The Azure architecture does not accommodate support for the following Citrix ADC features:
 - Clustering
 - IPv6
 - Gratuitous ARP (GARP)
 - L2 Mode

- Tagged VLAN
 - Dynamic Routing
 - Virtual MAC (VMAC)
 - USIP
 - Jumbo Frames
- If you expect that you might have to shut down and temporarily deallocate the Citrix ADC VPX virtual machine at any time, assign a static Internal IP address while creating the virtual machine. If you do not assign a static internal IP address, Azure might assign the virtual machine a different IP address each time it restarts, and the virtual machine might become inaccessible.
 - In an Azure deployment, only the following Citrix ADC VPX models are supported: VPX 10, VPX 200, VPX 1000, and VPX 3000. For information, see the Citrix ADC VPX Data Sheet.

If you use a Citrix ADC VPX instance with a model number higher than VPX 3000, the network throughput might not be the same as specified by the instance's license. However, other features, such as SSL throughput and SSL transactions per second, might improve.

- The “deployment ID” that is generated by Azure during virtual machine provisioning is not visible to the user in ARM. You cannot use the deployment ID to deploy Citrix ADC VPX appliance on ARM.
- The Citrix ADC VPX instance supports 20 Mb/s throughput and standard edition features when it's initialized.
- Citrix ADC VPX instances lower than D16sv3 that are configured for ICA Proxy can experience a high degree of latency. Hence, Citrix recommends you to use the instance size of D16sv3 or D32sv3.
- For Citrix Virtual Apps and Citrix Virtual Desktops deployment, a VPN virtual server on a VPX instance can be configured in the following modes:
 - Basic mode, where the ICAOnly VPN virtual server parameter is set to ON. The Basic mode works fully on an unlicensed Citrix ADC VPX instance.
 - SmartAccess mode, where the ICAOnly VPN virtual server parameter is set to OFF. The SmartAccess mode works for only 5 Citrix ADC AAA session users on an unlicensed Citrix ADC VPX instance.

Note:

To configure the SmartControl feature, you must apply a Premium license to the Citrix ADC VPX instance.

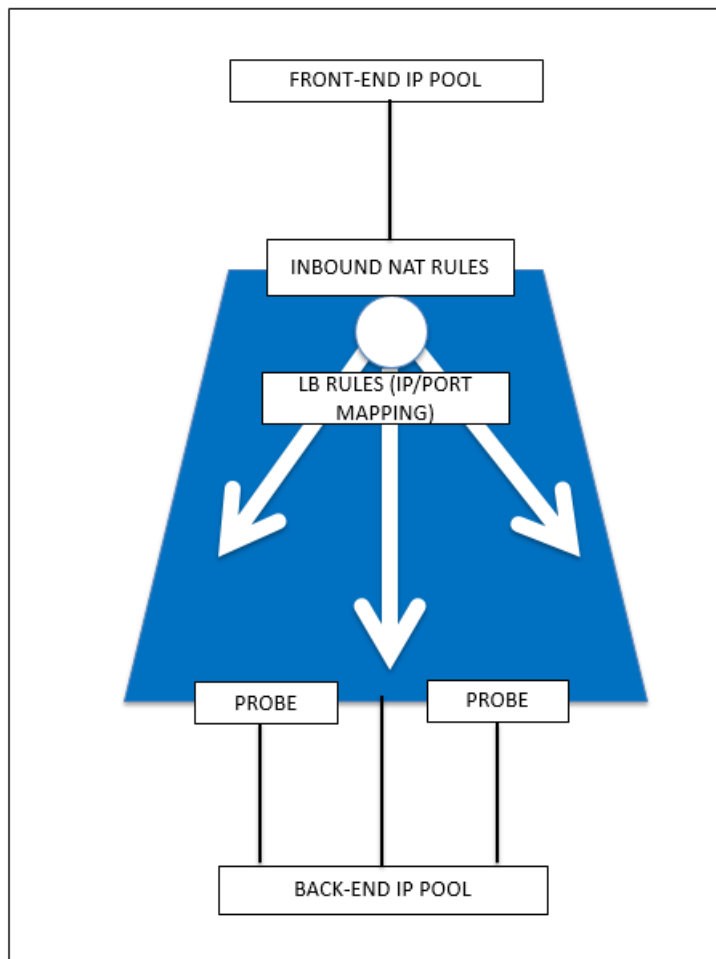
Azure terminology

November 12, 2024

Some of the Azure terms that are used in the Citrix ADC VPX Azure documentation are listed below.

1. Azure Load Balancer –Azure load balancer is a resource that distributes incoming traffic among computers in a network. Traffic is distributed among virtual machines defined in a load-balancer set. A load balancer can be external or internet-facing, or it can be internal.
2. Azure Resource Manager (ARM) –ARM is the new management framework for services in Azure. Azure Load Balancer is managed using ARM-based APIs and tools.
3. Back-End Address Pool –These are IP addresses associated with the virtual machine Network Interface Card (NIC) to which load will be distributed.
4. BLOB - Binary Large Object –Any binary object like a file or an image that can be stored in Azure storage.
5. Front-End IP Configuration –An Azure Load balancer can include one or more front-end IP addresses, also known as a virtual IPs (VIPs). These IP addresses serve as ingress for the traffic.
6. Instance Level Public IP (ILPIP) –An ILPIP is a public IP address that you can assign directly to your virtual machine or role instance, rather than to the cloud service that your virtual machine or role instance resides in. This does not take the place of the VIP (virtual IP) that is assigned to your cloud service. Rather, it's an additional IP address that you can use to connect directly to your virtual machine or role instance.

Note: In the past, an ILPIP was referred to as a PIP, which stands for public IP.
7. Inbound NAT Rules –This contains rules mapping a public port on the load balancer to a port for a specific virtual machine in the back end address pool.
8. IP-Config - It can be defined as an IP address pair (public IP and private IP) associated with an individual NIC. In an IP-Config, the public IP address can be NULL. Each NIC can have multiple IP-Config associated with it, which can be upto 255.
9. Load Balancing Rules –A rule property that maps a given front-end IP and port combination to a set of back-end IP addresses and port combination. With a single definition of a load balancer resource, you can define multiple load balancing rules, each rule reflecting a combination of a front end IP and port and back end IP and port associated with virtual machines.



10. Network Security Group (NSG) –NSG contains a list of Access Control List (ACL) rules that allow or deny network traffic to your virtual machine instances in a virtual network. NSGs can be associated with either subnets or individual virtual machine instances within that subnet. When a NSG is associated with a subnet, the ACL rules apply to all the virtual machine instances in that subnet. In addition, traffic to an individual virtual machine can be restricted further by associating a NSG directly to that virtual machine.

11. Private IP addresses –Used for communication within an Azure virtual network, and your on-premises network when you use a VPN gateway to extend your network to Azure. Private IP addresses allow Azure resources to communicate with other resources in a virtual network or an on-premises network through a VPN gateway or ExpressRoute circuit, without using an Internet-reachable IP address. In the Azure Resource Manager deployment model, a private IP address is associated with the following types of Azure resources –virtual machines, internal load balancers (ILBs), and application gateways.

12. Probes –This contains health probes used to check availability of virtual machines instances in the back end address pool. If a particular virtual machine does not respond to health probes for some time, then it is taken out of traffic serving. Probes enable you to keep track of the health of virtual

instances. If a health probe fails, the virtual instance will be taken out of rotation automatically.

13. **Public IP Addresses (PIP)** –PIP is used for communication with the Internet, including Azure public-facing services and is associated with virtual machines, Internet-facing load balancers, VPN gateways, and application gateways.

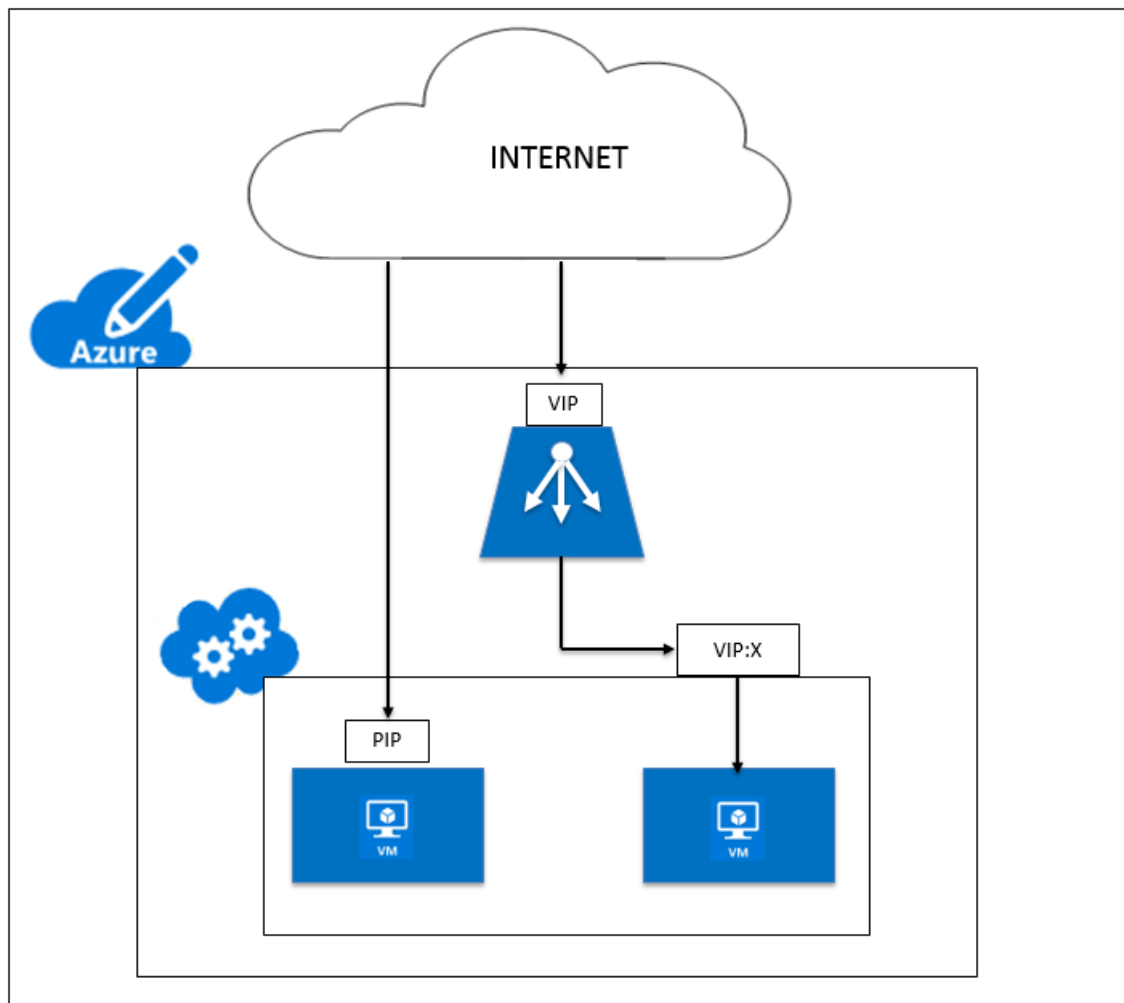
14. **Region** - An area within a geography that does not cross national borders and that contains one or more datacenters. Pricing, regional services, and offer types are exposed at the region level. A region is typically paired with another region, which can be up to several hundred miles away, to form a regional pair. Regional pairs can be used as a mechanism for disaster recovery and high availability scenarios. Also referred to generally as location.

15. **Resource Group** - A container in Resource Manager holds related resources for an application. The resource group can include all of the resources for an application, or only those resources that are logically grouped together

16. **Storage Account** –An Azure storage account gives you access to the Azure blob, queue, table, and file services in Azure Storage. Your storage account provides the unique namespace for your Azure storage data objects.

17. **Virtual Machine** –The software implementation of a physical computer that runs an operating system. Multiple virtual machines can run simultaneously on the same hardware. In Azure, virtual machines are available in a variety of sizes.

18. **Virtual Network** - An Azure virtual network is a representation of your own network in the cloud. It is a logical isolation of the Azure cloud dedicated to your subscription. You can fully control the IP address blocks, DNS settings, security policies, and route tables within this network. You can also further segment your VNet into subnets and launch Azure IaaS virtual machines and cloud services (PaaS role instances). Additionally, you can connect the virtual network to your on-premises network using one of the connectivity options available in Azure. In essence, you can expand your network to Azure, with complete control on IP address blocks with the benefit of enterprise scale Azure provides.



Network architecture for Citrix ADC VPX instances on Microsoft Azure

November 13, 2024

In Azure Resource Manager (ARM), a Citrix ADC VPX virtual machine (VM) resides in a virtual network. A single network interface can be created in a given subnet of the Virtual Network and can be attached to the VPX instance. You can filter network traffic to and from a VPX instance in an Azure virtual network with a network security group. A network security group contains security rules that allow or deny inbound network traffic to or outbound network traffic from a VPX instance. For more information, see [Security groups](#).

Network security group filters the requests to the Citrix ADC VPX instance, and the VPX instance sends them to the servers. The response from a server follows the same path in reverse. The Network security group can be configured to filter a single VPX VM, or, with subnets and virtual networks, can filter

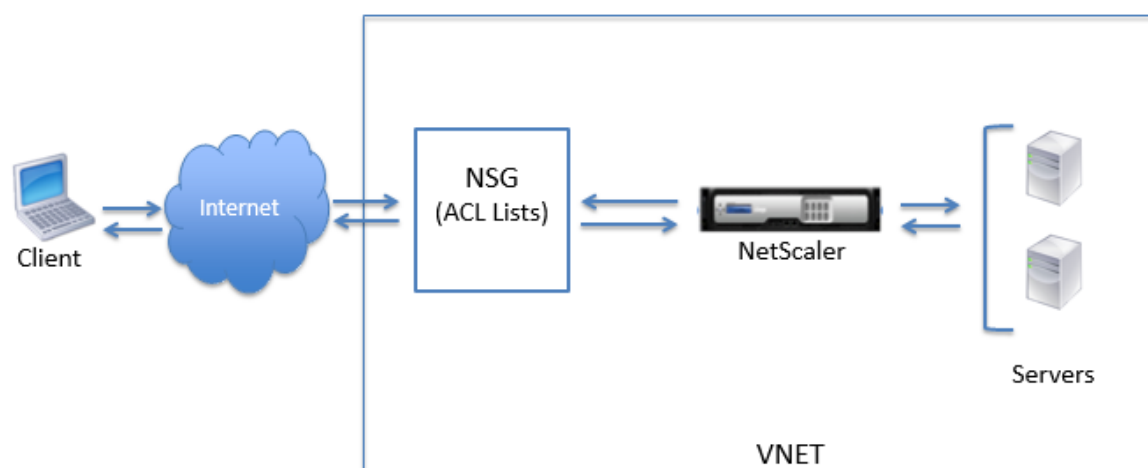
traffic in deployment of multiple VPX instances.

The NIC contains network configuration details such as the virtual network, subnets, internal IP address, and Public IP address.

While on ARM, it is good to know the following IP addresses that are used to access the VMs deployed with a single NIC and a single IP address:

- Public IP (PIP) address is the internet-facing IP address configured directly on the virtual NIC of the NetScaler VM. This allows you to directly access a VM from the external network.
- Citrix ADC IP (also known as NSIP) address is internal IP address configured on the VM. It is non-routable.
- Virtual IP address (VIP) is configured by using the NSIP and a port number. Clients access NetScaler services through the PIP address, and when the request reaches the NIC of the NetScaler VPX VM or the Azure load balancer, the VIP gets translated to internal IP (NSIP) and internal port number.
- Internal IP address is the private internal IP address of the VM from the virtual network's address space pool. This IP address cannot be reached from the external network. This IP address is by default dynamic unless you set it to static. Traffic from the internet is routed to this address according to the rules created on the NSG. The NSG works with the NIC to selectively send the right type of traffic to the right port on the NIC, which depends on the services configured on the VM.

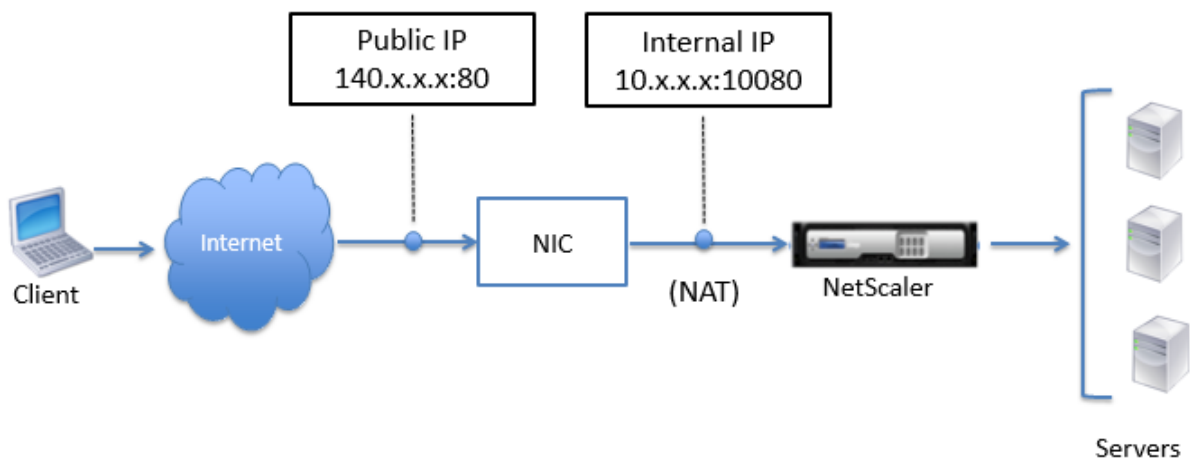
The following figure shows how traffic flows from a client to a server through a NetScaler VPX instance provisioned in ARM.



Traffic flow through network address translation

You can also request a public IP (PIP) address for your Citrix ADC VPX instance (instance level). If you use this direct PIP at the VM level, you need not define inbound and outbound rules to intercept the network traffic. The incoming request from the Internet is received on the VM directly. Azure performs network address translation (NAT) and forwards the traffic to the internal IP address of the VPX instance.

The following figure shows how Azure performs network address translation to map the NetScaler internal IP address.



In this example, the Public IP assigned to the NSG is 140.x.x.x and the internal IP address is 10.x.x.x. When the inbound and outbound rules are defined, public HTTP port 80 is defined as the port on which the client requests are received, and a corresponding private port, 10080, is defined as the port on which the Citrix ADC VPX instance listens. The client request is received on the Public IP address (140.x.x.x). Azure performs network address translation to map the PIP to the internal IP address 10.x.x.x on port 10080, and forwards the client request.

Note:

Citrix ADC VPX VMs in high availability are controlled by external or internal load balancers that have inbound rules defined on them to control the load balancing traffic. The external traffic is first intercepted by these load balancers and the traffic is diverted according to the load balancing rules configured, which has back-end pools, NAT rules, and health probes defined on the load balancers.

Port usage guidelines

You can configure more inbound and outbound rules in NSG while creating the Citrix ADC VPX instance or after the virtual machine is provisioned. Each inbound and outbound rule is associated with a

public port and a private port.

Before configuring NSG rules, note the following guidelines regarding the port numbers you can use:

1. The Citrix ADC VPX instance reserves the following ports. You cannot define these as private ports when using the Public IP address for requests from the internet.

Ports 21, 22, 80, 443, 8080, 67, 161, 179, 500, 520, 3003, 3008, 3009, 3010, 3011, 4001, 5061, 9000, 7000.

However, if you want internet-facing services such as the VIP to use a standard port (for example, port 443) you have to create port mapping by using the NSG. The standard port is then mapped to a different port that is configured on the NetScaler for this VIP service.

For example, a VIP service might be running on port 8443 on the VPX instance but be mapped to public port 443. So, when the user accesses port 443 through the Public IP, the request is directed to private port 8443.

2. Public IP address does not support protocols in which port mapping is opened dynamically, such as passive FTP or ALG.
3. High availability does not work for traffic that uses a public IP address (PIP) associated with a VPX instance, instead of a PIP configured on the Azure load balancer.

Note:

In Azure Resource Manager, a Citrix ADC VPX instance is associated with two IP addresses - a public IP address (PIP) and an internal IP address. While the external traffic connects to the PIP, the internal IP address or the NSIP is non-routable. To configure VIP in VPX, use the internal IP address and any of the free ports available. Do not use the PIP to configure VIP.

Configure a Citrix ADC VPX standalone instance

November 13, 2024

You can provision a single Citrix ADC VPX instance in Azure Resource Manager (ARM) portal in a standalone mode by creating the virtual machine and configuring other resources.

Before you begin

Ensure that you have the following:

- A Microsoft Azure user account

- Access to Microsoft Azure Resource Manager
- Microsoft Azure SDK
- Microsoft Azure PowerShell

On the [Microsoft Azure Portal](#) page, log on to the Azure Resource Manager portal by providing your user name and password.

Note:

In ARM portal, clicking an option in one pane opens a new pane to the right. Navigate from one pane to another to configure your device.

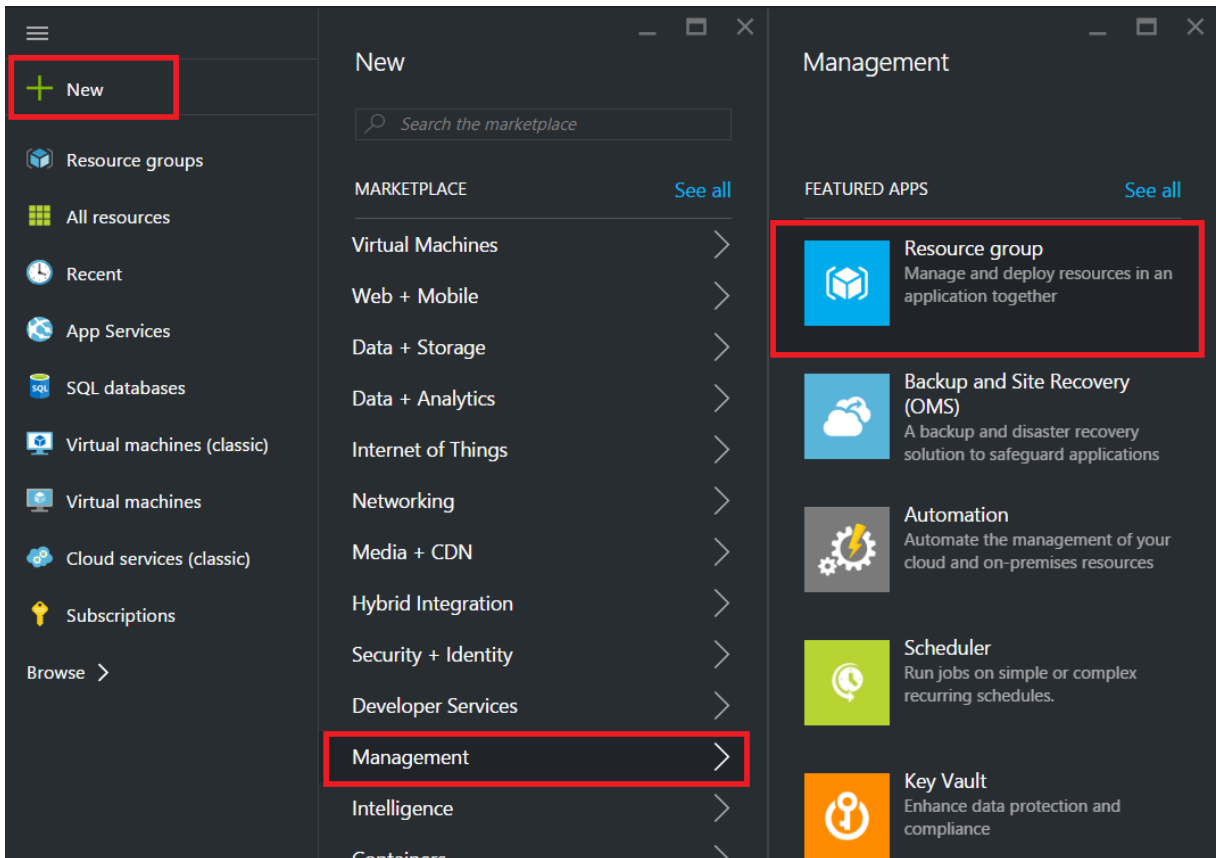
Summary of configuration steps

1. Configure a resource group
2. Configure a network security group
3. Configure virtual network and its subnets
4. Configure a storage account
5. Configure an availability set
6. Configure a Citrix ADC VPX instance.

Configure a resource group

Create a new resource group that is a container for all your resources. Use the resource group to deploy, manage, and monitor your resources as a group.

1. Click **New > Management > Resource group**.
2. In the **Resource group** pane, enter the following details:
 - Resource group name
 - Resource group location
3. Click **Create**.



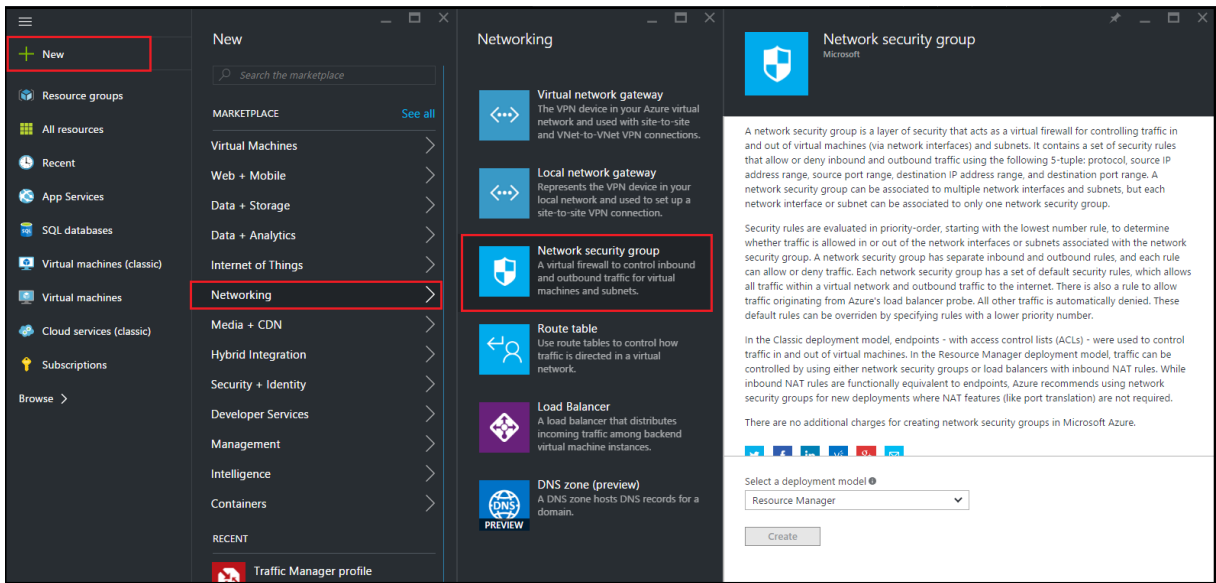
Configure a network security group

Create a network security group (NSG) to assign inbound and outbound rules to control the incoming and outgoing traffic within the virtual network. NSG allows you to define security rules for a single virtual machine and also to define security rules for a virtual network subnet.

1. Click **New > Networking > Network security group**.
2. In the **Create network security group** pane, enter the following details, and then click **Create**.
 - Name - type a name for the security group
 - Resource group - select the resource group from the drop-down list

Note:

Ensure that you have selected the correct location. The list of resources that appear in the drop-down list is different for different locations.

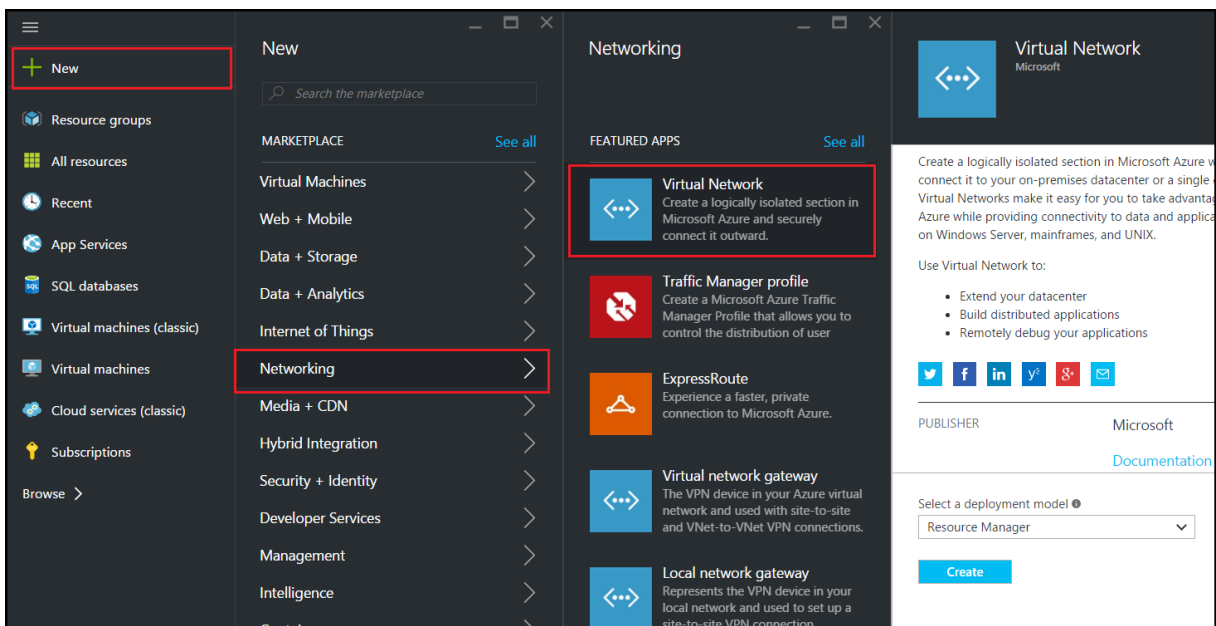


Configure a virtual network and subnets

Virtual networks in ARM provide a layer of security and isolation to your services. VMs and services that are part of the same virtual network can access each other.

For these steps to create a virtual network and subnets.

1. Click **New** > **Networking** > **Virtual Network**.
2. In the **Virtual Network** pane, ensure the deployment mode is **Resource Manager** and click **Create**.



3. In the **Create virtual network** pane, enter the following values, and then click **Create**.

- Name of the virtual network
- Address space - type the reserved IP address block for the virtual network
- Subnet - type the name of the first subnet (you will create the second subnet later in this step)
- Subnet address range - type the reserved IP address block of the subnet
- Resource group - select the resource group created earlier from the drop-down list

Create virtual network

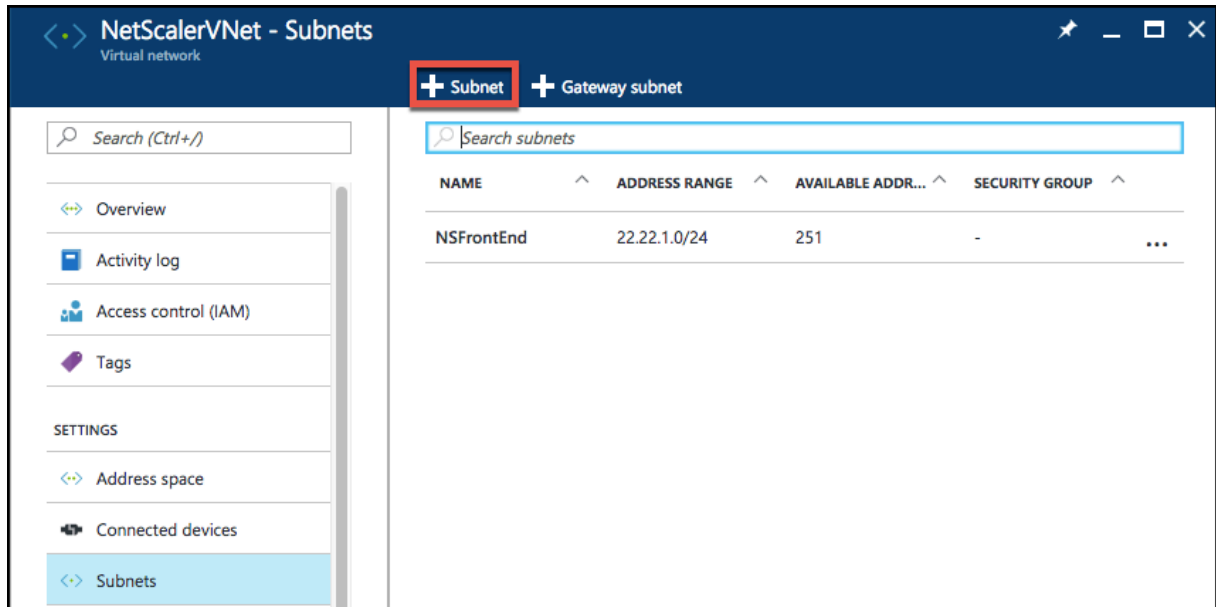
- * Name
NetScalerVNet ✓
- * Address space ⓘ
22.22.0.0/16 ✓
22.22.0.0 - 22.22.255.255 (65536 addresses)
- * Subnet name
NSFrontEnd ✓
- * Subnet address range ⓘ
22.22.1.0/24 ✓
22.22.1.0 - 22.22.1.255 (256 addresses)
- * Subscription
Microsoft Azure Enterprise ▼
- * Resource group ⓘ
 Create new Use existing
NSDocs ▼
- * Location
Southeast Asia ▼

Pin to dashboard

Create [Automation options](#)

Configure the second subnet

1. Select the newly created virtual network from **All resources** pane and in the **Settings** pane, click **Subnets**.



2. Click **+Subnet** and create the second subnet by entering the following details.
 - Name of the second subnet
 - Address range - type the reserved IP address block of the second subnet
 - Network security group - select the NSG from the drop-down list
3. Click **Create**.

Add subnet

NetScalerVNet

* Name
NSBackEnd ✓

* Address range (CIDR block) ⓘ
22.22.2.0/24 ✓
22.22.2.0 - 22.22.2.255 (256 addresses)

Network security group
None >

Route table
None >

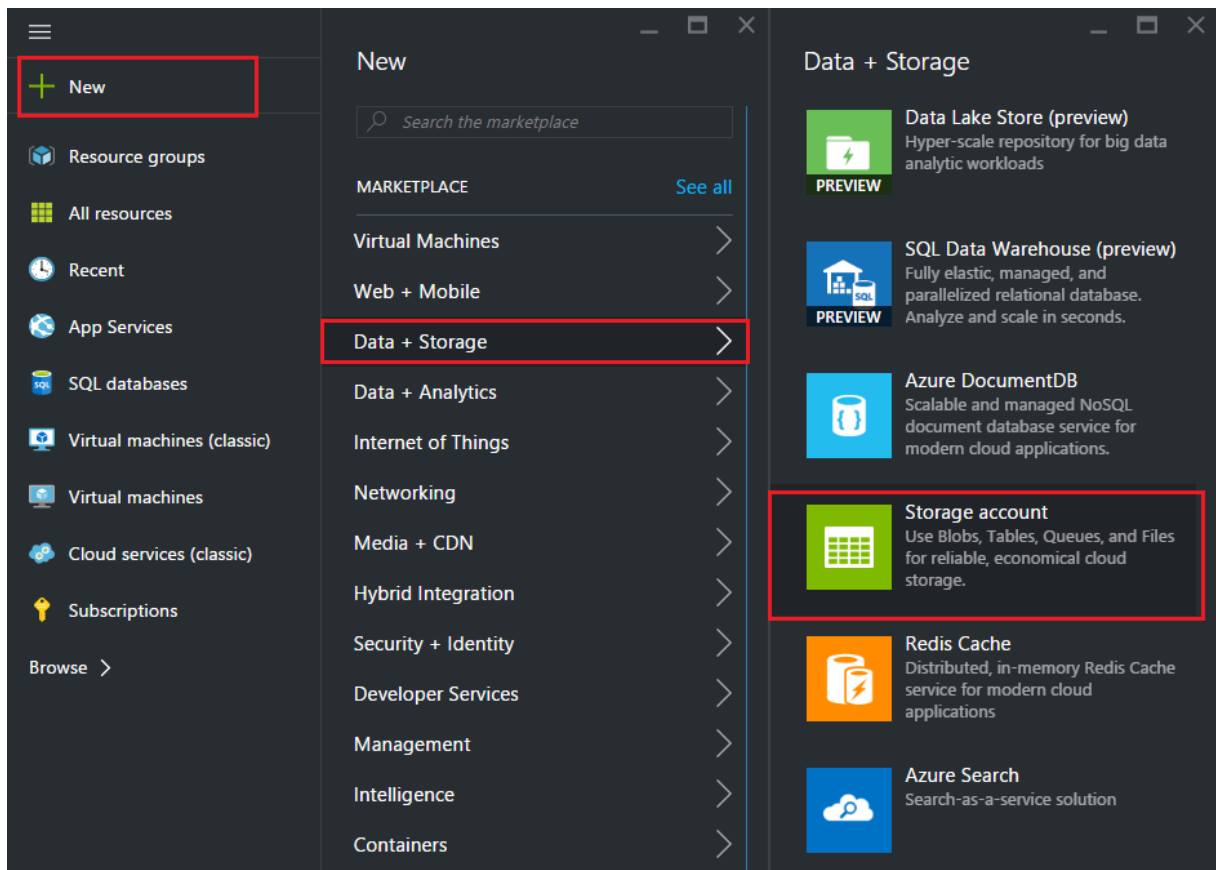
OK

Configure a storage account

The ARM IaaS infrastructure storage includes all services where we can store data in the form of blobs, tables, queues, and files. You can also create applications using these forms of storage data in ARM.

Create a storage account to store all your data.

1. Click **+New > Data + Storage > Storage account**.
2. In the **Create storage account** pane, enter the following details:
 - Name of the account
 - Deployment mode - make sure to select **Resource Manager**
 - Account kind - select **General purpose** from the drop-down list
 - Replication - select **Locally redundant storage** from the drop-down list
 - Resource group - select the newly created resource group from the drop-down list
3. Click **Create**.

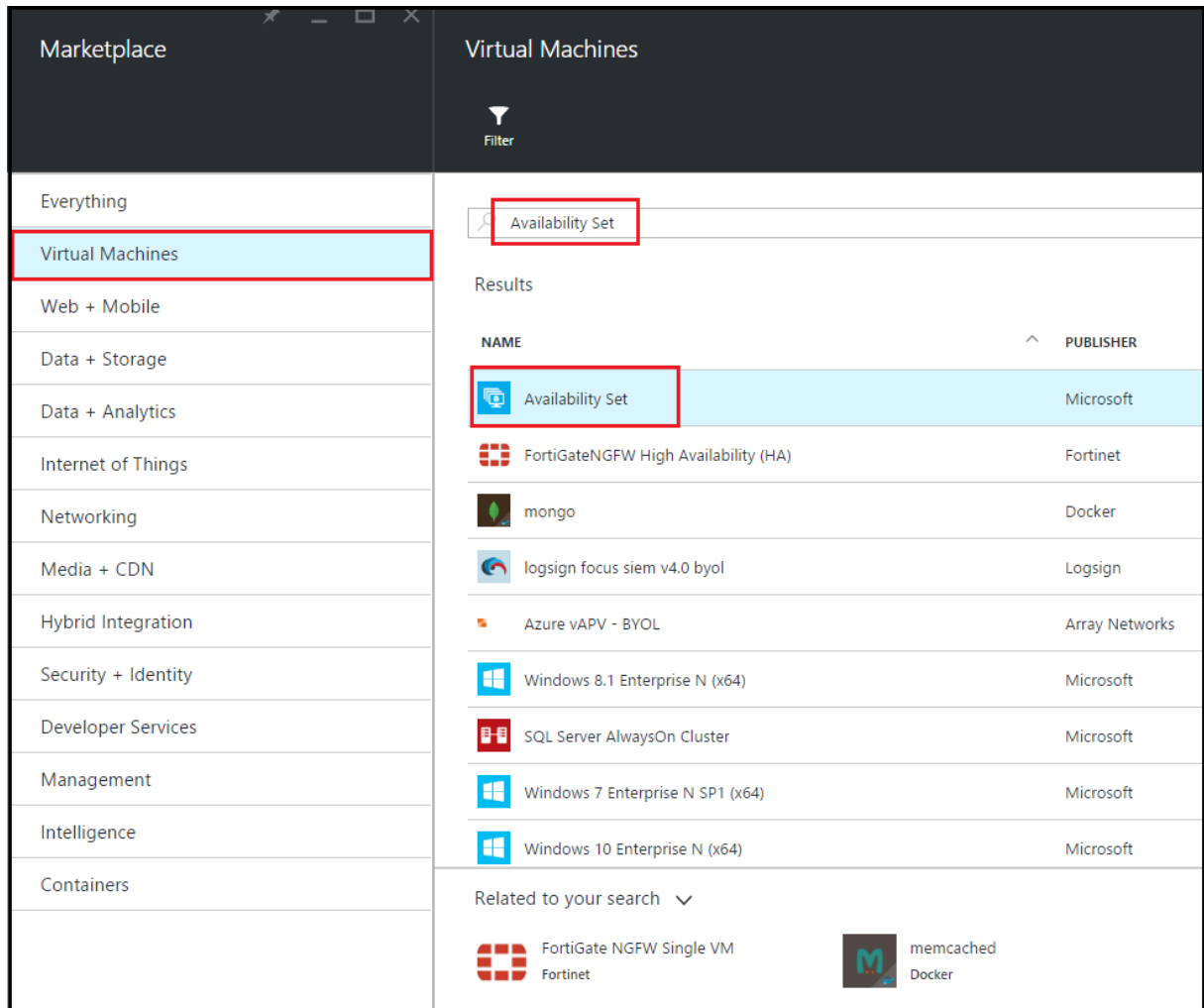


Configure an availability set

An availability set guarantees that at least one VM is kept up and running in case of planned or unplanned maintenance. Two or more VMs under the same 'availability set' are placed on different fault

domains to achieve redundant services.

1. Click **+New**.
2. Click **See all** in the MARKETPLACE pane and click **Virtual Machines**.
3. Search for availability set, and then select **Availability set** entity from the list displayed.



4. Click **Create**, and in the **Create availability set** pane, enter the following details:
 - Name of the set
 - Resource group - select the newly created resource group from the drop-down list
5. Click **Create**.

Create availability set

* Name
 ✓

Fault domains ⓘ
 3

Update domains ⓘ
 5

* Subscription
 ▼

* Resource group ⓘ
 Create new Use existing
 ▼

* Location
 ▼

Create

Configure a Citrix ADC VPX instance

Create an instance of Citrix ADC VPX in the virtual network. Obtain the Citrix ADC VPX image from the Azure marketplace, and then use the Azure Resource Manager portal to create a Citrix ADC VPX instance.

Before you begin creating the Citrix ADC VPX instance, make sure that you have created a virtual network with required subnets in which the instance will reside. You can create virtual networks during

VM provisioning, but without the flexibility to create different subnets. For information about creating virtual networks, see <http://azure.microsoft.com/en-us/documentation/articles/create-virtual-network/>.

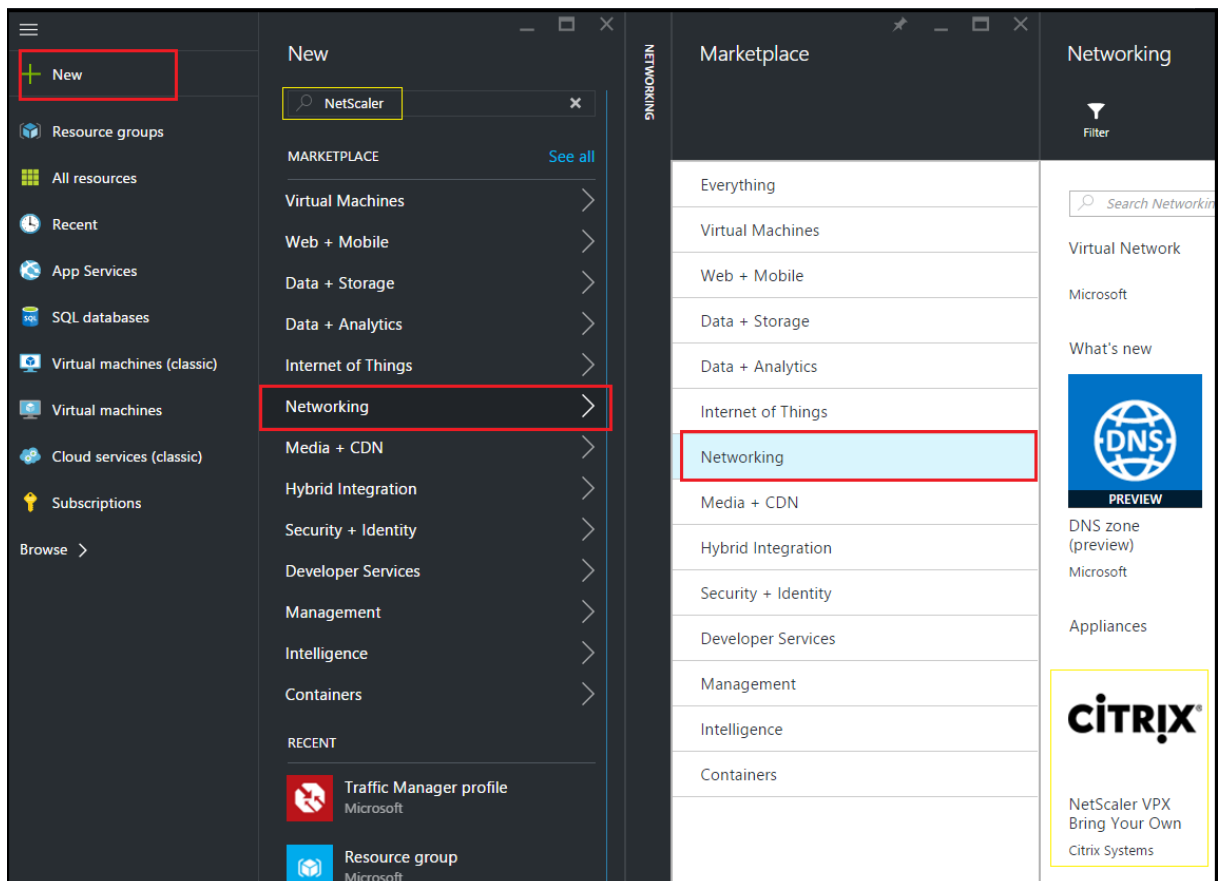
Optionally, configure DNS server and VPN connectivity that allows a virtual machine to access Internet resources.

Note:

Citrix recommends that you create resource group, network security group, virtual network, and other entities before you provision the Citrix ADC VPX VM, so that the network information is available during provisioning.

1. Click **+New > Networking**.
2. Click **See All** and in the Networking pane, click **Citrix ADC VPX Bring Your Own License**.

As a quick way to find any entity on ARM portal, you can also type the name of the entity in the Azure Marketplace search box and press <Enter>. Type NetScaler in the search box to find the Citrix NetScaler images.



Note

Ensure to select the latest image. Your Citrix NetScaler image might have the release number in the name.

3. On the **Citrix ADC VPX Bring Your Own License** page, from the drop-down list, select **Resource Manager** and click **Create**.

The screenshot displays the 'Create virtual machine' wizard in the 'Basics' step. The left-hand navigation pane shows five steps: 1. Basics (Configure basic settings), 2. Size (Choose virtual machine size), 3. Settings (Configure optional features), 4. Summary (NetScaler 11.1 VPX Bring Your ...), and 5. Buy. The right-hand pane contains the following configuration fields:

- Name:** Citrix-NetScaler-User (with a green checkmark)
- VM disk type:** SSD (dropdown menu)
- User name:** CitrixUser1 (with a green checkmark)
- Authentication type:** SSH public key (selected) and Password (button)
- Password:** [Redacted] (with a green checkmark)
- Confirm password:** [Redacted] (with a green checkmark)
- Subscription:** Microsoft Azure Enterprise (dropdown menu)
- Resource group:** Use existing (selected radio button) and NetScalerResGroup (dropdown menu)
- Location:** Southeast Asia (dropdown menu)

An **OK** button is located at the bottom of the configuration pane.

4. In the **Create virtual machine** pane, specify the required values in each section to create a virtual machine. Click **OK** in each section to save your configuration.

Basic:

- Name - specify a name for the Citrix ADC VPX instance
- VM disk type - select SSD (default value) or HDD from the drop-down menu
- User name and Password - specify a user name and password to access the resources in the resource group that you have created
- Authentication Type - select SSH Public Key or Password
- Resource group - select the resource group you have created from the drop-down list

You can create a resource group here, but Citrix recommends that you create a resource group from Resource groups in Azure Resource Manager and then select the group from the drop-down list.

Size:

Depending on the VM disk type, SSD or HDD, you selected in Basic settings, the disk sizes are displayed.

- Select a disk size according to your requirement and click **Select**.

Settings:

- Select the default (Standard) disk type
- Storage account - select the storage account
- Virtual network - select the virtual network
- Subnet - set the subnet address
- Public IP address - select the type of IP address assignment
- Network security group - select the security group that you have created. Ensure that inbound and outbound rules are configured in the security group.
- Availability Set - select the availability set from the drop-down box

Summary:

The configuration settings are validated and the Summary page displays the result of the validation. If the validation fails, the Summary page displays the reason of the failure. Go back to the particular section and make changes as required. If the validation passes, click **OK**.

Buy:

Review the offer details and legal terms on the Purchase page and click **Purchase**.

For high availability deployment, create two independent instances of Citrix ADC VPX in the same availability set and in the same resource group to deploy them in active-standby configuration.

Configure multiple IP addresses for a Citrix ADC VPX standalone instance

November 21, 2024

This section explains how to configure a standalone Citrix ADC VPX instance with multiple IP addresses, in Azure Resource Manager (ARM). The VPX instance can have one or more NIC attached to it, and each NIC can have one or more static or dynamic public and private IP addresses assigned to it. You can assign multiple IP addresses as NSIP, VIP, SNIP, and so on.

For more information, see the Azure documentation [Assign multiple IP addresses to virtual machines using the Azure portal](#).

If you want to use PowerShell commands, see [Configuring multiple IP addresses for a Citrix ADC VPX instance in standalone mode by using PowerShell commands](#).

Use case

In this use case, a standalone Citrix ADC VPX appliance is configured with a single NIC that is connected to a virtual network (VNET). The NIC is associated with three IP configurations (ipconfig), each servers a different purpose - as shown in the table.

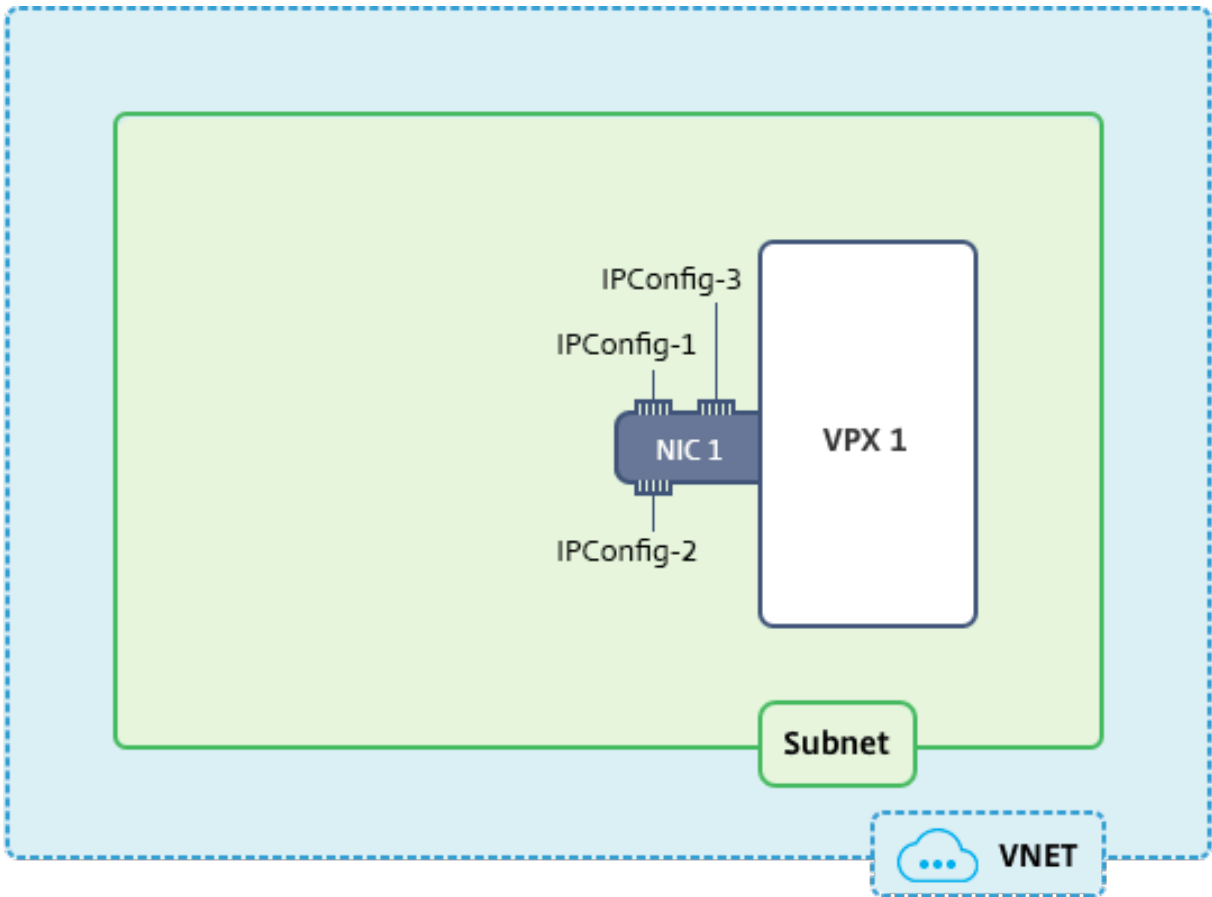
IPconfig	Associated with	Purpose
ipconfig1	Static public IP address; static private IP address	Serves management traffic
ipconfig2	Static public IP address; static private address	Serves client-side traffic
ipconfig3	Static private IP address	Communicates with back-end servers

Note:

IPConfig-3 is not associated with any public IP address.

Diagram: Topology

Here is the visual representation of the use case.

**Note:**

In a multi-NIC, multi-IP Azure Citrix ADC VPX deployment, the private IP associated with the primary (first) IPConfig of the primary (first) NIC is automatically added as the management NSIP of the appliance. The remaining private IP addresses associated with IPConfigs need to be added in the VPX instance as a VIP or SNIP by using the “add ns ip” command, according to your requirement.

Before you begin

Before you begin, create a VPX instance by following the steps given at this link:

[Configure a Citrix ADC VPX standalone instance](#)

For this use case, the NSDoc0330VM VPX instance is created.

Procedure to configure multiple IP addresses for a Citrix ADC VPX instance in standalone mode.

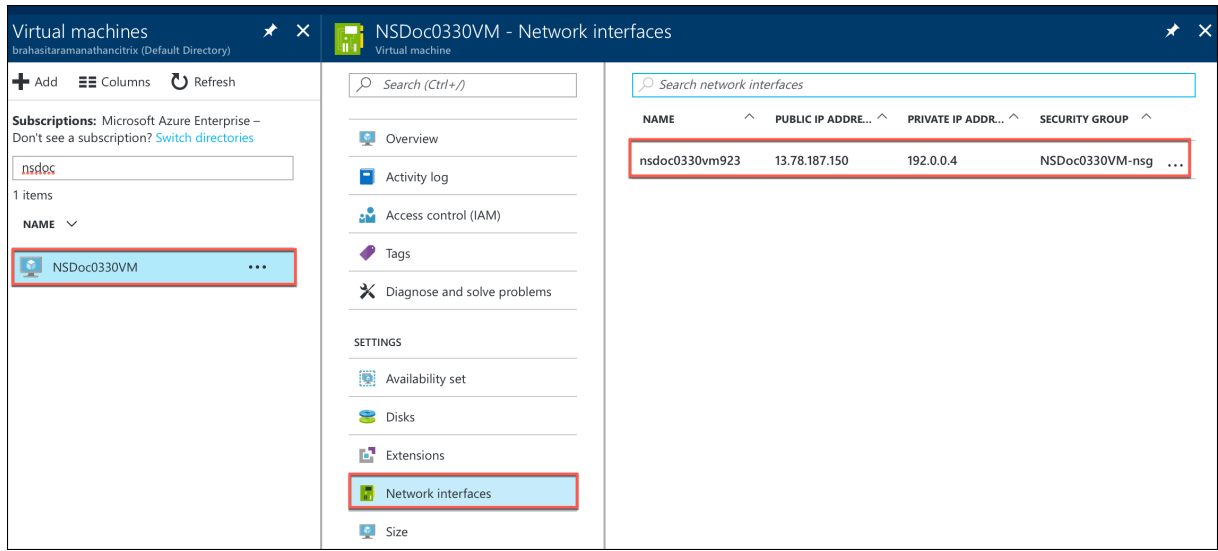
For configuring multiple IP addresses for a Citrix ADC VPX appliance in standalone mode:

1. Add IP addresses to the VM

2. Configure Citrix ADC -owned IP addresses

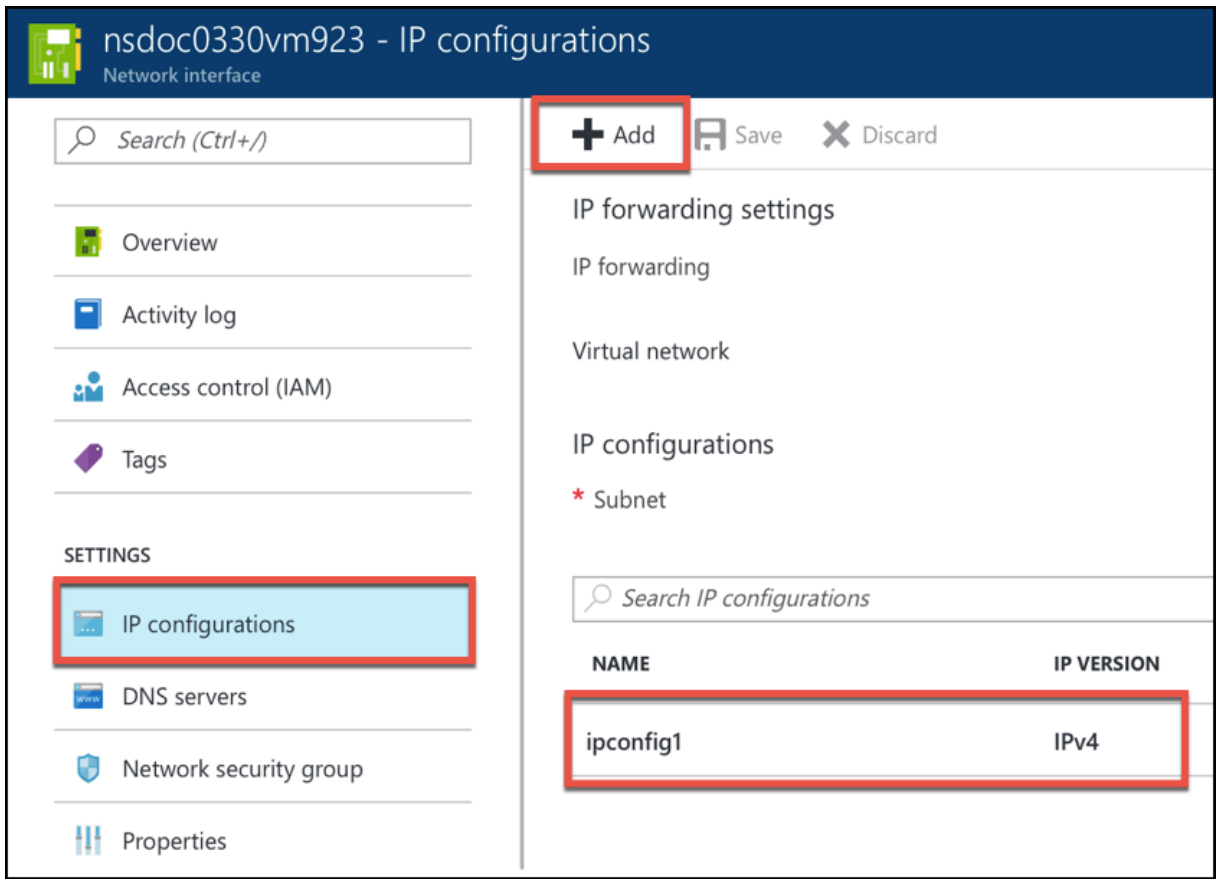
Step 1: Add IP addresses to the VM

1. In the portal, click **More services > type virtual machines** in the filter box, and then click **Virtual machines**.
2. In the **Virtual machines** blade, click the VM you want to add IP addresses to. Click **Network interfaces** in the virtual machine blade that appears, and then select the network interface.



In the blade that appears for the NIC you selected, click **IP configurations**. The existing IP configuration that was assigned when you created the VM, **ipconfig1**, is displayed. For this use case, make sure the IP addresses associated with ipconfig1 are static. Next, create two more IP configurations: ipconfig2 (VIP) and ipconfig3 (SNIP).

To create additional ipconfigs, create **Add**.



In the **Add IP configuration** window, enter a **Name**, specify allocation method as **Static**, enter an IP address (192.0.0.5 for this use case), and enable **Public IP address**.

Note:

Before adding a static private IP address, check for IP address availability and make sure the IP address belongs to the same subnet to which the NIC is attached.

The screenshot shows the 'Add IP configuration' window for a VM named 'nsdoc0330vm923'. The configuration is as follows:

- Name:** ipconfig2 (highlighted with a red box and a green checkmark)
- Type:** Secondary (selected)
- Message:** Primary IP configuration already exists
- Private IP address settings:**
 - Allocation:** Static (selected)
 - IP address:** 192.0.0.5 (highlighted with a red box and a green checkmark)
 - Public IP address:** Enabled (selected)
- Action:** A light blue button labeled 'IP address Configure required settings' with a right-pointing arrow is highlighted with a red box.

Next, click **Configure required settings** to create a static public IP address for ipconfig2.

By default, public IPs are dynamic. To make sure that the VM always uses the same public IP address, create a static Public IP.

In the Create public IP address blade, add a Name, under Assignment click **Static**. And then click **OK**.

Create public IP address

* Name
 ✓

Assignment
 Dynamic Static

Note:

Even when you set the allocation method to static, you cannot specify the actual IP address assigned to the public IP resource. Instead, it gets allocated from a pool of available IP addresses in the Azure location the resource is created in.

Follow the steps to add one more IP configuration for ipconfig3. Public IP is not mandatory.

Search IP configurations					
NAME	IP VERSION	TYPE	PRIVATE IP ADDRESS	PUBLIC IP ADDRESS	
ipconfig1	IPv4	Primary	192.0.0.4 (Static)	13.78.187.150 (NSDoc0330VM-ip)	
ipconfig2	IPv4	Secondary	192.0.0.5 (Static)	13.78.183.123 (ipconfig2_PIP2)	
ipconfig3	IPv4	Secondary	192.0.0.6 (Static)	-	

Step 2: Configure Citrix ADC-owned IP addresses

Configure the Citrix ADC-owned IP addresses by using the GUI or the command “add ns ip.” For more information, see [Configuring Citrix ADC-Owned IP Addresses](#).

You’ve now configured multiple IP addresses for a Citrix ADC VPX instance in standalone mode.

Configure a high-availability setup with multiple IP addresses and NICs

November 21, 2024

In a Microsoft Azure deployment, a high-availability configuration of two Citrix ADC VPX instances is achieved by using the Azure Load Balancer (ALB). This is achieved by configuring a health probe on ALB, which monitors each VPX instance by sending health probe at every 5 seconds to both primary and secondary instances.

In this setup, only the primary node responds to health probes and secondary does not. Once the primary sends the response to the health probe, the ALB starts sending the data traffic to the instance. If the primary instance misses two consecutive health probes, ALB does not redirect traffic to that instance. On failover, the new primary starts responding to health probes and the ALB redirects traffic to it. The standard VPX high availability failover time is three seconds. The total failover time that might take for traffic switching can be maximum of 13 seconds.

You can deploy a pair of Citrix ADC VPX instances with multiple NICs in an active-passive high availability (HA) setup on Azure. Each NIC can contain multiple IP addresses.

The following options are available for a multi-NIC high availability deployment:

- High availability using Azure availability set
- High availability using Azure availability zones

For more information about Azure Availability Set and Availability Zones, see the Azure documentation [Manage the availability of Linux virtual machines](#).

High availability using availability set

A high availability setup using availability set must meet the following requirements:

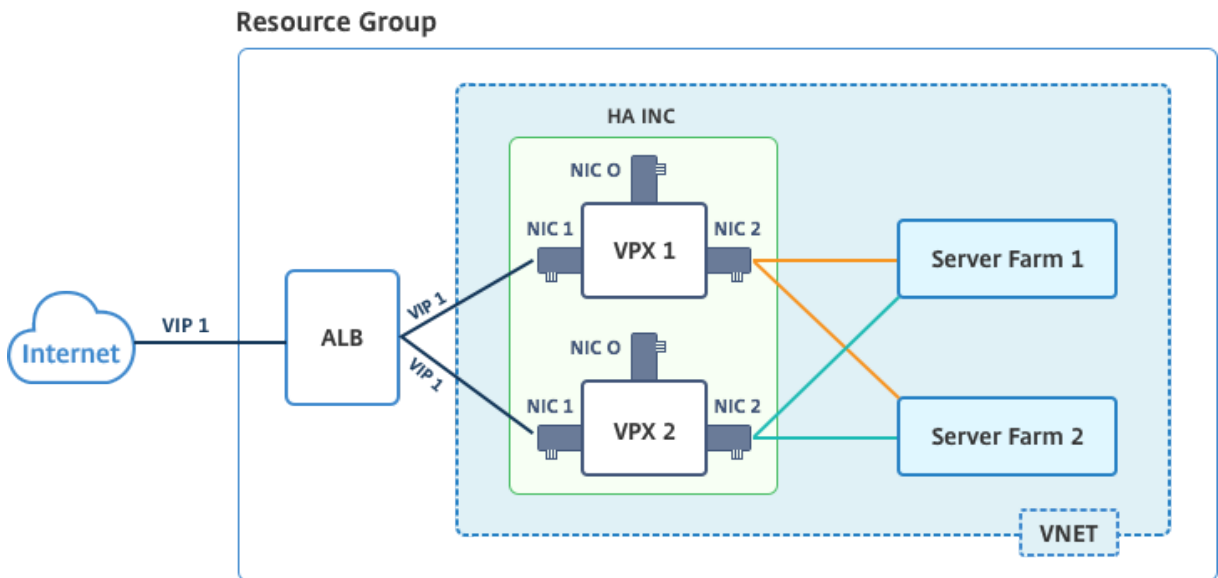
- An HA Independent Network Configuration (INC) configuration
- The Azure Load Balancer (ALB) in Direct Server Return (DSR) mode

All traffic goes through the primary node. The secondary node remains in standby mode until the primary node fails.

Note:

For a Citrix VPX high availability deployment on Azure cloud to work, you need a floating public IP (PIP) that can be moved between the two VPX nodes. The Azure Load Balancer (ALB) provides that floating PIP, which is moved to the second node automatically in the event of a failover.

Diagram: Example of a high availability deployment architecture, using Azure Availability Set



In an active-passive deployment, the ALB floating public IP (PIP) addresses are added as the VIP addresses in each VPX node. In HA-INC configuration, the VIP addresses are floating and SNIP addresses are instance specific.

You can deploy a VPX pair in active-passive high availability mode in two ways by using:

- **Citrix ADC VPX standard high availability template:** use this option to configure an HA pair with the default option of three subnets and six NICs.
- **Windows PowerShell commands:** use this option to configure an HA pair according to your subnet and NIC requirements.

This topic describes how to deploy a VPX pair in active-passive HA setup by using the Citrix template. If you want to use PowerShell commands, see [Configuring an HA Setup with Multiple IP Addresses and NICs by Using PowerShell Commands](#).

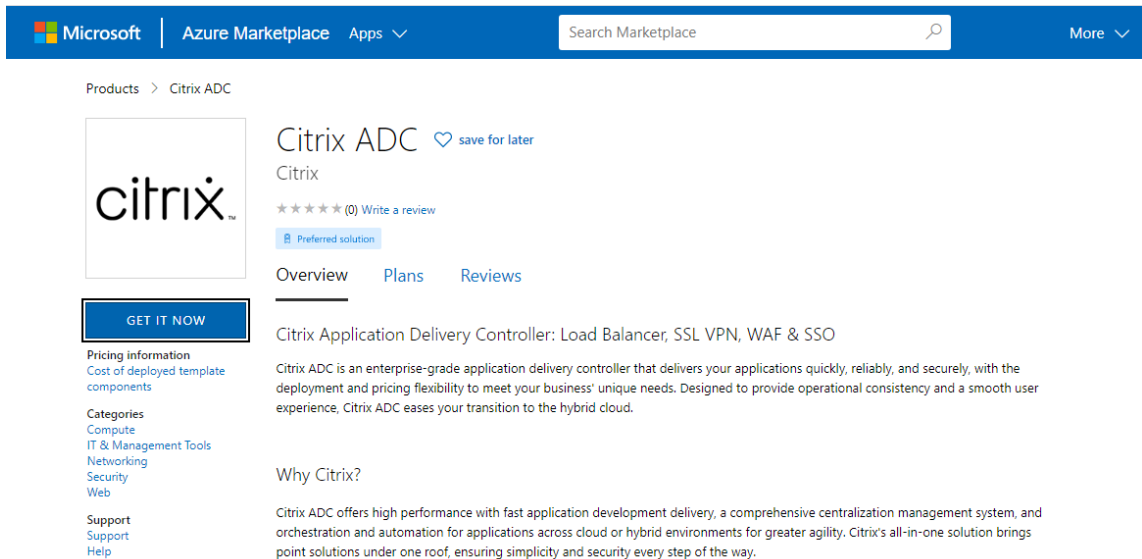
Configure HA-INC nodes by using the Citrix high availability template

You can quickly and efficiently deploy a pair of VPX instances in HA-INC mode by using the standard template. The template creates two nodes, with three subnets and six NICs. The subnets are for management, client, and server-side traffic, and each subnet has two NICs for both the VPX instances.

You can get the Citrix ADC HA Pair template at the [Azure Marketplace](#).

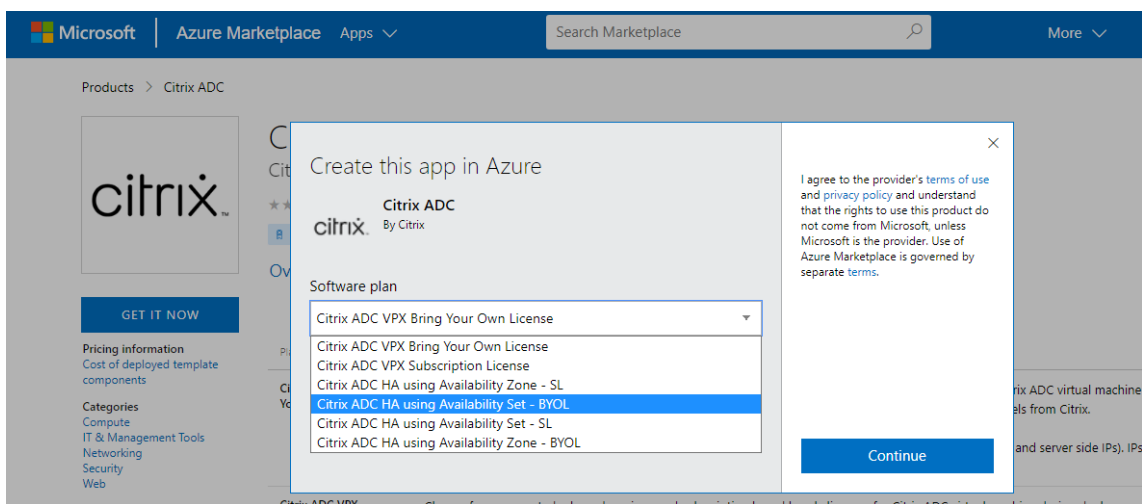
Complete the following steps to launch the template and deploy a high availability VPX pair, by using Azure availability sets.

1. From Azure Marketplace, search **Citrix ADC**.

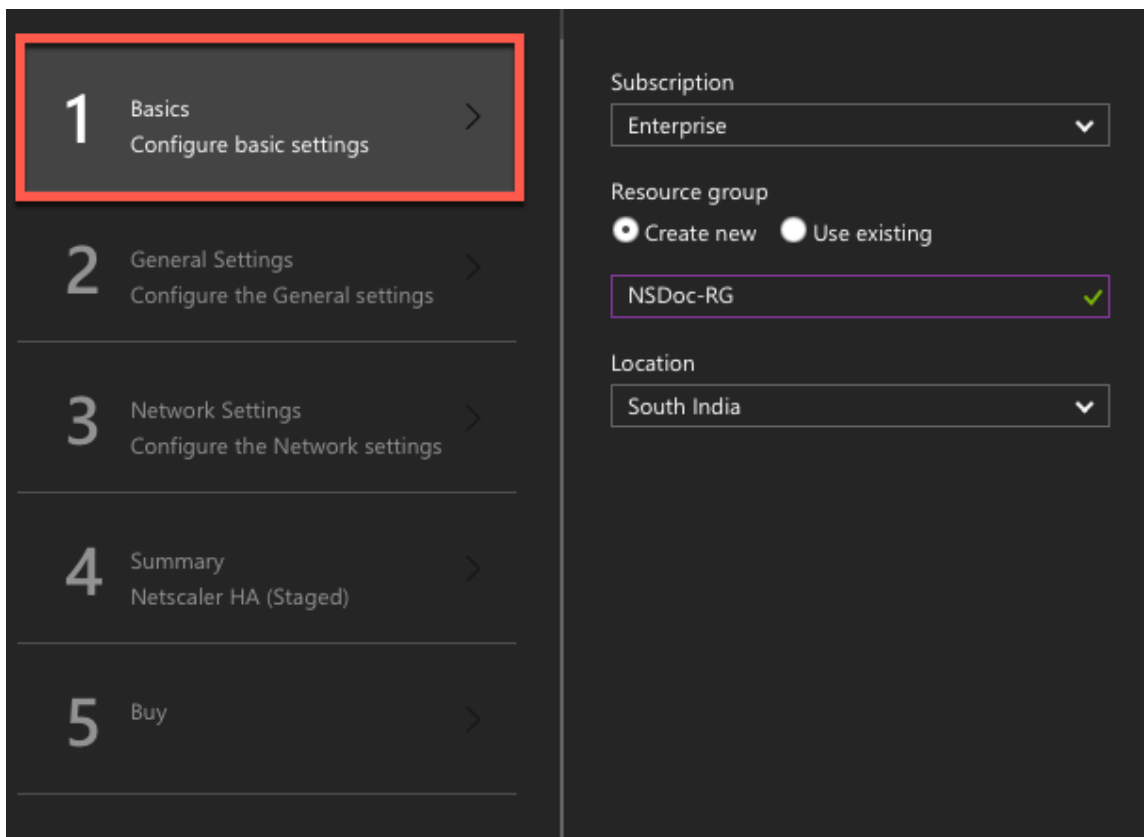


2. Click **GET IT NOW**.

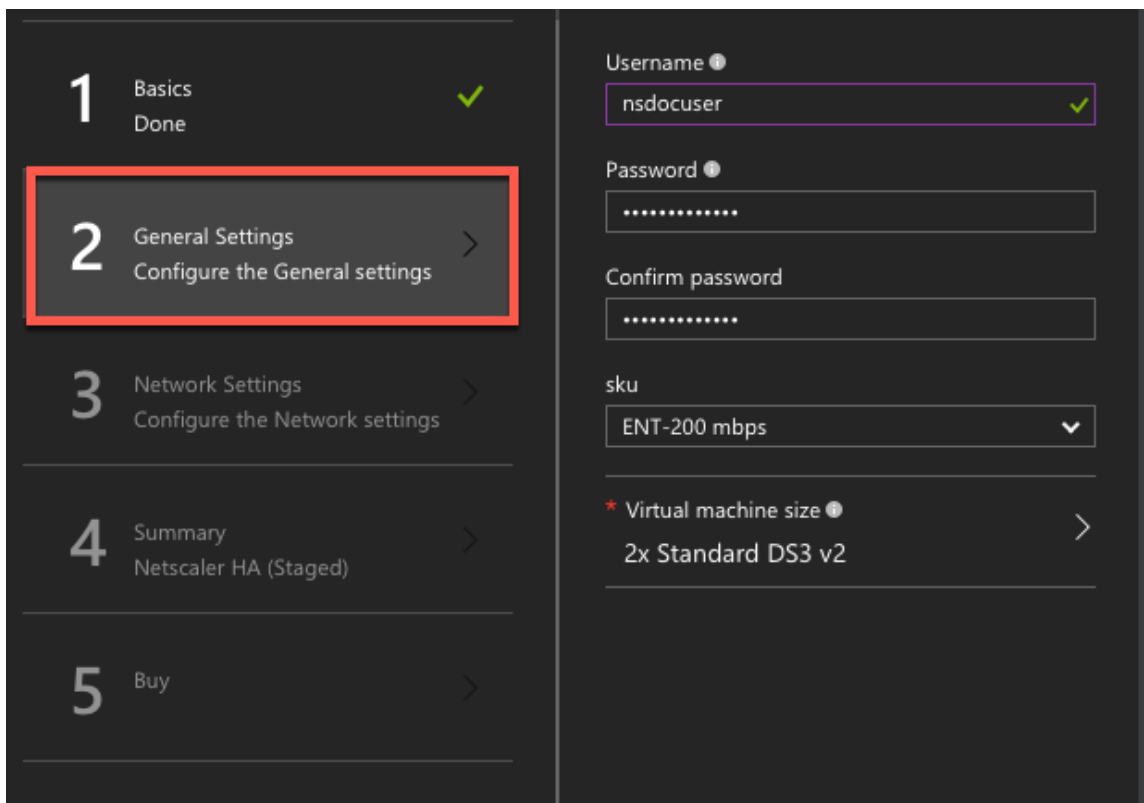
3. Select the required HA deployment along with license, and click **Continue**.



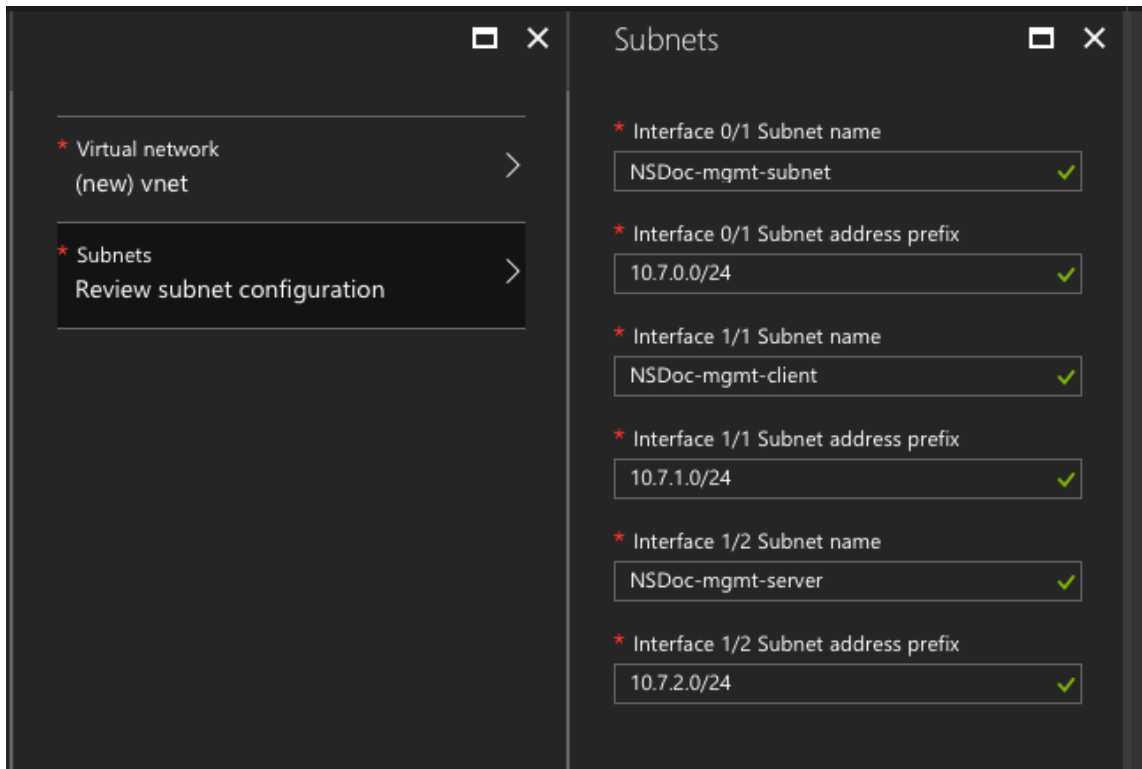
4. The **Basics** page appears. Create a Resource Group and select **OK**.



5. The **General Settings** page appears. Type the details and select **OK**.



- The **Network Setting** page appears. Check the vnet and subnet configurations, edit the required settings, and select **OK**.


























- The **Summary** page appears. Review the configuration and edit accordingly. Select **OK** to confirm.
- The **Buy** page appears. Select **Purchase** to complete the deployment.

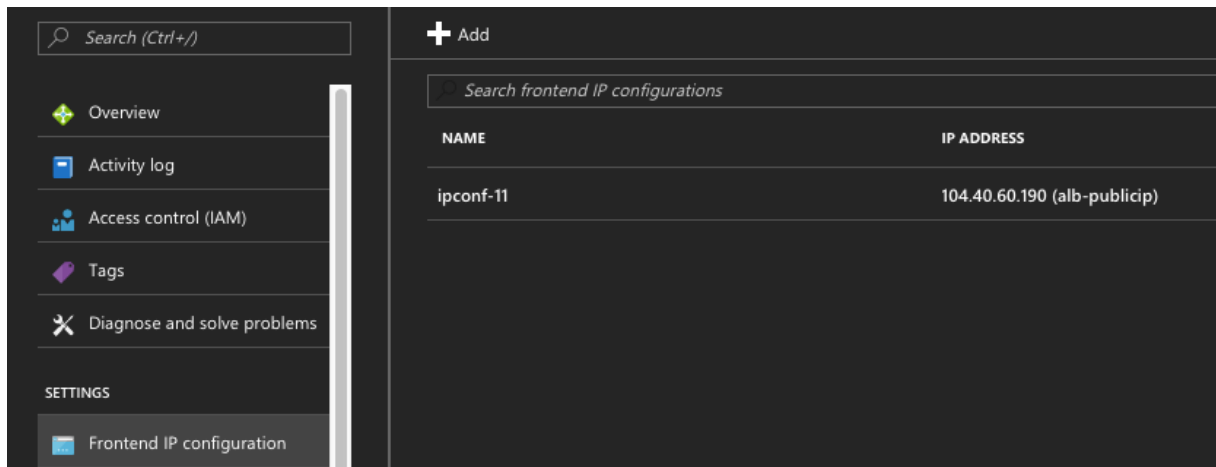
It might take a moment for the Azure Resource Group to be created with the required configurations. After completion, select the Resource Group in the Azure portal to see the configuration details, such as LB rules, back-end pools, health probes, and so on. The high availability pair appears as ns-vpx0 and ns-vpx1.

If further modifications are required for your HA setup, such as creating more security rules and ports, you can do that from the Azure portal.

23 items Show hidden types ⓘ

<input type="checkbox"/>	NAME ↑↓	TYPE ↑↓
<input type="checkbox"/>	 alb	Load balancer
<input type="checkbox"/>	 alb-publicip	Public IP address
<input type="checkbox"/>	 avl-set	Availability set
<input type="checkbox"/>	 ns-vp0	Disk
<input type="checkbox"/>	 ns-vp0	Virtual machine
<input type="checkbox"/>	 ns-vp0-mgmt-publicip	Public IP address
<input type="checkbox"/>	 ns-vp1	Disk
<input type="checkbox"/>	 ns-vp1	Virtual machine
<input type="checkbox"/>	 ns-vp1-mgmt-publicip	Public IP address
<input type="checkbox"/>	 ns-vp0-nic0-01	Network interface
<input type="checkbox"/>	 ns-vp0-nic0-11	Network interface
<input type="checkbox"/>	 ns-vp0-nic0-12	Network interface
<input type="checkbox"/>	 ns-vp0-nic1-01	Network interface
<input type="checkbox"/>	 ns-vp0-nic1-11	Network interface
<input type="checkbox"/>	 ns-vp0-nic1-12	Network interface
<input type="checkbox"/>	 ns-vp0-nic-nsg0-01	Network security group
<input type="checkbox"/>	 ns-vp0-nic-nsg0-11	Network security group
<input type="checkbox"/>	 ns-vp0-nic-nsg0-12	Network security group
<input type="checkbox"/>	 ns-vp0-nic-nsg1-01	Network security group
<input type="checkbox"/>	 ns-vp0-nic-nsg1-11	Network security group
<input type="checkbox"/>	 ns-vp0-nic-nsg1-12	Network security group
<input type="checkbox"/>	 vnet01	Virtual network
<input type="checkbox"/>	 vpxhamd7fi3wouvrk	Storage account

Next, you need to configure the load-balancing vserver with the ALB public IP (PIP) address, on each node. To find the ALB PIP, select ALB > **Frontend IP configuration**.



See the Resources section for more information about how to configure the load-balancing vserver.

Resources:

The following links provide additional information related to HA deployment and virtual server (vserver) configuration:

- [Configuring high availability nodes in different subnets](#)
- [Set up basic load balancing](#)

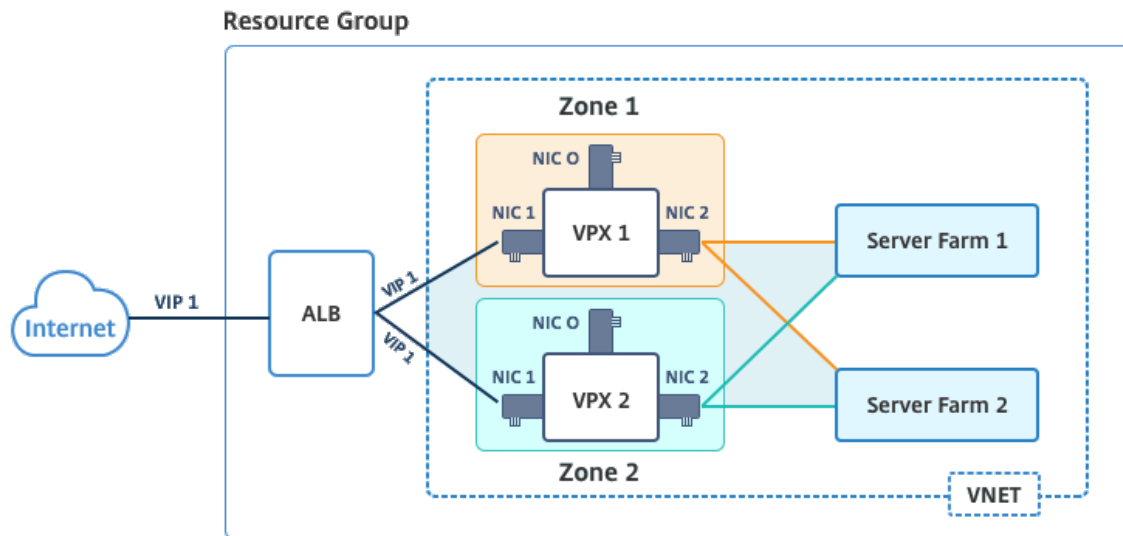
Related resources:

- [Configure a high-availability setup with multiple IP addresses and NICs by using PowerShell commands](#)
- [Configuring GSLB on Active-Standby HA Deployment on Azure](#)

High availability using availability zones

Azure Availability Zones are fault-isolated locations within an Azure region, providing redundant power, cooling, and networking and increasing resiliency. Only specific Azure regions support Availability Zones. For more information, see the Azure documentation [What are Availability Zones in Azure?](#).

Diagram: Example of a high availability deployment architecture, using Azure Availability Zones



You can deploy a VPX pair in high availability mode by using the template called “NetScaler 12.1 HA using Availability Zones,” available in Azure Marketplace.

Complete the following steps to launch the template and deploy a high availability VPX pair, by using Azure Availability Zones.

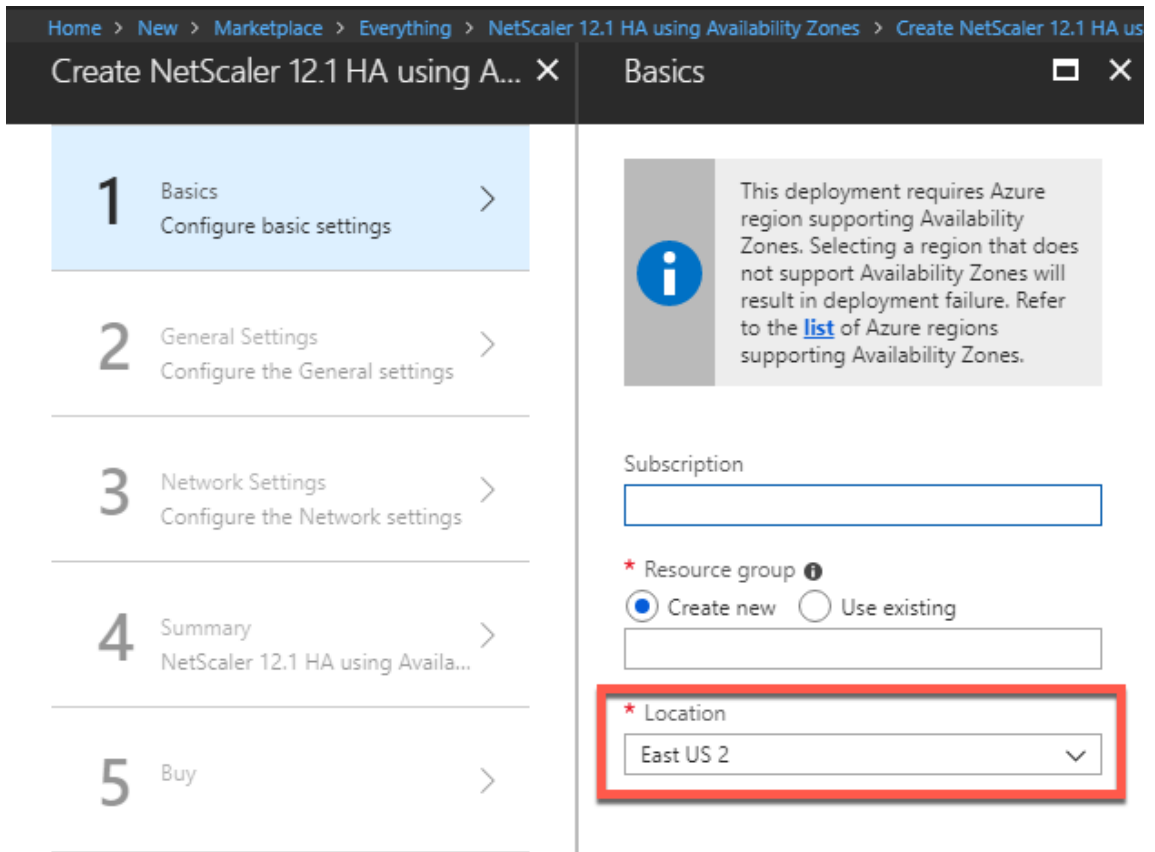
1. From Azure Marketplace, select and initiate the Citrix solution template.



2. Ensure deployment type is Resource Manager and select **Create**.
3. The **Basics** page appears. Enter the details and click **OK**.

Note:

Ensure that you select an Azure region that supports Availability Zones. For more information about regions that support Availability Zones, see Azure documentation [What are Availability Zones in Azure?](#)



4. The **General Settings** page appears. Type the details and select **OK**.
5. The **Network Setting** page appears. Check the vnet and subnet configurations, edit the required settings, and select **OK**.
6. The **Summary** page appears. Review the configuration and edit accordingly. Select **OK** to confirm.
7. The **Buy** page appears. Select **Purchase** to complete the deployment.

It might take a moment for the Azure Resource Group to be created with the required configurations. After completion, select the **Resource Group** to see the configuration details, such as LB rules, back-end pools, health probes, and so on, in the Azure portal. The high availability pair appears as ns-vp0 and ns-vp1. Also, you can see the location under the **Location** column.

Filter by name... All types All locations No grouping

22 items Show hidden types

NAME	TYPE	LOCATION
alb	Load balancer	East US 2
alb-publicip	Public IP address	East US 2
ns-vpx0	Virtual machine	East US 2
ns-vpx0_OsDisk_1_d7b757b8aa804bf1991a083f319e553a	Disk	East US 2
ns-vpx0-mgmt-publicip	Public IP address	East US 2
ns-vpx1	Virtual machine	East US 2
ns-vpx1_OsDisk_1_0c2364d43e2b47fa896bf14b02090ee0	Disk	East US 2
ns-vpx1-mgmt-publicip	Public IP address	East US 2
ns-vpx-nic0-01	Network interface	East US 2
ns-vpx-nic0-11	Network interface	East US 2
ns-vpx-nic0-12	Network interface	East US 2
ns-vpx-nic1-01	Network interface	East US 2
ns-vpx-nic1-11	Network interface	East US 2
ns-vpx-nic1-12	Network interface	East US 2
ns-vpx-nic-nsg0-01	Network security group	East US 2
ns-vpx-nic-nsg0-11	Network security group	East US 2
ns-vpx-nic-nsg0-12	Network security group	East US 2
ns-vpx-nic-nsg1-01	Network security group	East US 2
ns-vpx-nic-nsg1-11	Network security group	East US 2
ns-vpx-nic-nsg1-12	Network security group	East US 2
test1	Virtual network	East US 2
vpxhavadosvod3v5jeu	Storage account	East US 2

If further modifications are required for your HA setup, such as creating more security rules and ports, you can do that from the Azure portal.

Configure a high-availability setup with multiple IP addresses and NICs by using PowerShell commands

November 21, 2024

You can deploy a pair of Citrix ADC VPX instances with multiple NICs in an active-passive high availability (HA) setup on Azure. Each NIC can contain multiple IP addresses.

An active-passive deployment requires:

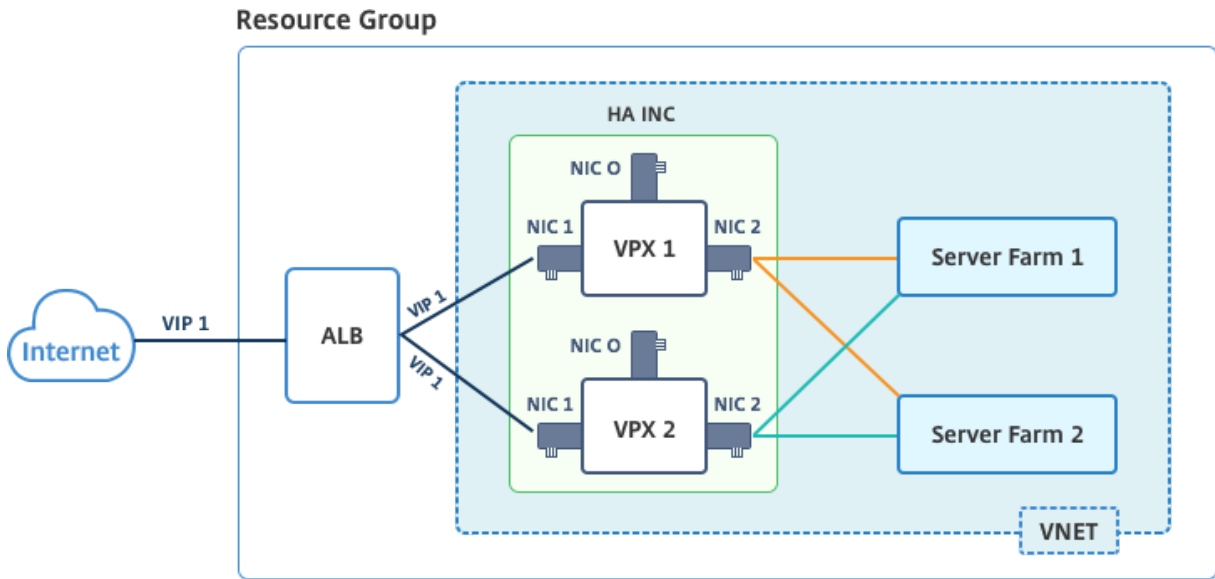
- An HA Independent Network Configuration (INC) configuration
- The Azure Load Balancer (ALB) in Direct Server Return (DSR) mode

All traffic goes through the primary node. The secondary node remains in standby mode until the primary node fails.

Note:

For a Citrix ADC VPX high availability deployment on Azure cloud to work, you need a floating public IP (PIP) that can be moved between the two high-availability nodes. The Azure Load Balancer (ALB) provides that floating PIP, which is moved to the second node automatically in the event of a failover.

Diagram: Example of an active-passive deployment architecture



In an active-passive deployment, the ALB floating public IP (PIP) addresses are added as the VIP addresses in each VPX node. In HA-INC configuration, the VIP addresses are floating and SNIP addresses are instance specific.

ALB monitors each VPX instances by sending health probe at every 5 seconds and redirects traffic to that instance only that sends health probes response on regular interval. So in an HA setup, the primary node responds to health probes and secondary does not. If the primary instances misses two consecutive health probes, ALB does not redirect traffic to that instance. On failover, the new primary starts responding to health probes and the ALB redirects traffic to it. The standard VPX high availability failover time is three seconds. The total failover time that might take for traffic switching can be maximum of 13 seconds.

You can deploy a VPX pair in active-passive HA setup in two ways by using:

- **Citrix ADC VPX Standard high availability template:** use this option to configure an HA pair with the default option of three subnets and six NICs.
- **Windows PowerShell commands:** use this option to configure an HA pair according to your subnet and NIC requirements.

This topic describes how to deploy a VPX pair in active-passive HA setup by using PowerShell commands. If you want to use the Citrix ADC VPX Standard HA template, see [Configuring an HA Setup with](#)

Multiple IP Addresses and NICs.

Configure HA-INC nodes by using PowerShell Ccmmands

Scenario: HA-INC PowerShell deployment

In this scenario, you deploy a Citrix ADC VPX pair by using the topology given in the table. Each VPX instance contains three NICs, with each NIC is deployed in a different subnet. Each NIC is assigned an IP configuration.

ALB	VPX1	VPX2
ALB is associated with public IP 3 (pip3)	Management IP is configured with IPConfig1, which includes one public IP (pip1) and one private IP (12.5.2.24); nic1; Mgmtsubnet=12.5.2.0/24	Management IP is configured with IPConfig5, which includes one public IP (pip3) and one private IP (12.5.2.26);nic4;Mgmtsubnet=12.5.2.0/24
LB rules and port configured are HTTP (80),SSL (443), health probe (9000)	Client-side IP is configured with IPConfig3, which includes one private IP(12.5.1.27);nic2; FrontEndsubnet=12.5.1.0/24	Client-side IP is configured with IPConfig7, which includes one private IP (12.5.1.28);nic5;FrontEndsubnet=12.5.1.0/24
-	Server-side IP is configured with IPConfig4, which includes one private IP(12.5.3.24); nic3;BackendSubnet=12.5.3.0/24	Server-side IP is configured with IPConfig8, which includes one private IP(12.5.3.28);nic6;BackendSubnet=12.5.3.0/24
-	Rules and ports for NSG areSSH (22),HTTP (80),HTTPS (443)	-

Parameter settings

The following parameter settings are used in this scenario:

```

1 $locName= "South east Asia"
2
3 $rgName = "MulitIP-MultiNIC-RG"
4
5 $nicName1= "VM1-NIC1"
6
7 $nicName2 = "VM1-NIC2"
8
9 $nicName3= "VM1-NIC3"
10

```

```
11 $nicName4 = "VM2-NIC1"
12
13 $nicName5= "VM2-NIC2"
14
15 $nicName6 = "VM2-NIC3"
16
17 $vNetName = "Azure-MultiIP-ALB-vnet"
18
19 $vNetAddressRange= "12.5.0.0/16"
20
21 $frontEndSubnetName= "frontEndSubnet"
22
23 $frontEndSubnetRange= "12.5.1.0/24"
24
25 $mgmtSubnetName= "mgmtSubnet"
26
27 $mgmtSubnetRange= "12.5.2.0/24"
28
29 $backEndSubnetName = "backEndSubnet"
30
31 $backEndSubnetRange = "12.5.3.0/24"
32
33 $prmStorageAccountName = "multiipmultinicbstorage"
34
35 $avSetName = "multiple-avSet"
36
37 $vmSize= "Standard\_DS4\_V2"
38
39 $publisher = "citrix"
40
41 $offer = "netscalervpx-120"
42
43 $sku = "netscalerbyol"
44
45 $version="latest"
46
47 $pubIPName1="VPX1MGMT"
48
49 $pubIPName2="VPX2MGMT"
50
51 $pubIPName3="ALBPIP"
52
53 $domName1="vpx1dns"
54
55 $domName2="vpx2dns"
56
57 $domName3="vpxalbdns"
58
59 $vmNamePrefix="VPXMultiIPALB"
60
61 $osDiskSuffix1="osmultiipalbdiskdb1"
62
63 $osDiskSuffix2="osmultiipalbdiskdb2"
```

```

64
65 $lbName= "MultiIPALB"
66
67 $frontEndConfigName1= "FrontEndIP"
68
69 $backendPoolName1= "BackendPoolHttp"
70
71 $lbRuleName1= "LBRuleHttp"
72
73 $healthProbeName= "HealthProbe"
74
75 $nsgName="NSG-MultiIP-ALB"
76
77 $rule1Name="Inbound-HTTP"
78
79 $rule2Name="Inbound-HTTPS"
80
81 $rule3Name="Inbound-SSH"

```

To complete the deployment, complete the following steps by using PowerShell commands:

1. Create a resource group, storage account, and availability set
2. Create a network security group and add rules
3. Create a virtual network and three subnets
4. Create public IP addresses
5. Create IP configurations for VPX1
6. Create IP configurations for VPX2
7. Create NICs for VPX1
8. Create NICs for VPX2
9. Create VPX1
10. Create VPX2
11. Create ALB

Create a resource group, storage account, and availability set.

```

1 New-AzureRmResourceGroup -Name $rgName -Location $locName
2
3
4 $prmStorageAccount=New-AzureRMStorageAccount -Name
   $prmStorageAccountName -ResourceGroupName $rgName -Type Standard_LRS
   -Location $locName
5
6
7 $avSet=New-AzureRMAvailabilitySet -Name $avSetName -ResourceGroupName
   $rgName -Location $locName

```

Create a network security group and add rules.

```

1 $rule1 = New-AzureRmNetworkSecurityRuleConfig -Name $rule1Name -
   Description "Allow HTTP" -Access Allow -Protocol Tcp -Direction

```

```

    Inbound -Priority 101
2
3
4 -SourceAddressPrefix Internet -SourcePortRange * -
    DestinationAddressPrefix * -DestinationPortRange 80
5
6
7 $rule2 = New-AzureRmNetworkSecurityRuleConfig -Name $rule2Name -
    Description "Allow HTTPS" -Access Allow -Protocol Tcp -Direction
    Inbound -Priority 110
8
9
10 -SourceAddressPrefix Internet -SourcePortRange * -
    DestinationAddressPrefix * -DestinationPortRange 443
11
12
13 $rule3 = New-AzureRmNetworkSecurityRuleConfig -Name $rule3Name -
    Description "Allow SSH" -Access Allow -Protocol Tcp -Direction
    Inbound -Priority 120
14
15
16 -SourceAddressPrefix Internet -SourcePortRange * -
    DestinationAddressPrefix * -DestinationPortRange 22
17
18
19 $nsg = New-AzureRmNetworkSecurityGroup -ResourceGroupName $rgName -
    Location $locName -Name $nsgName -SecurityRules $rule1,$rule2,$rule3

```

Create a virtual network and three subnets.

```

1 $frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
    $frontEndSubnetName -AddressPrefix $frontEndSubnetRange (this
    parameter value should be as per your requirement)
2
3
4 $mgmtSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name $mgmtSubnetName
    -AddressPrefix $mgmtSubnetRange
5
6
7 $backendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
    $backEndSubnetName -AddressPrefix $backEndSubnetRange
8
9
10 $vnet =New-AzureRmVirtualNetwork -Name $vNetName -ResourceGroupName
    $rgName -Location $locName -AddressPrefix $vNetAddressRange -Subnet
    $frontendSubnet,$backendSubnet, $mgmtSubnet
11
12
13 $subnetName ="frontEndSubnet"
14
15
16 \ $subnet1=\ $vnet.Subnets|?{
17 \ $_.Name -eq \ $subnetName }

```

```

18
19
20
21 $subnetName="backEndSubnet"
22
23
24 \ $subnet2=\$vnet.Subnets|?{
25   \ $\.Name -eq \$subnetName }
26
27
28
29 $subnetName="mgmtSubnet"
30
31
32 \ $subnet3=\$vnet.Subnets|?{
33   \ $\.Name -eq \$subnetName }

```

Create public IP addresses.

```

1 $pip1=New-AzureRmPublicIpAddress -Name $pubIPName1 -ResourceGroupName
   $rgName -DomainNameLabel $domName1 -Location $locName -
   AllocationMethod Dynamic
2
3 $pip2=New-AzureRmPublicIpAddress -Name $pubIPName2 -ResourceGroupName
   $rgName -DomainNameLabel $domName2 -Location $locName -
   AllocationMethod Dynamic
4
5 $pip3=New-AzureRmPublicIpAddress -Name $pubIPName3 -ResourceGroupName
   $rgName -DomainNameLabel $domName3 -Location $locName -
   AllocationMethod Dynamic

```

Create IP configurations for VPX1.

```

1 $IpConfigName1 = "IPConfig1"
2
3
4 $IPAddress = "12.5.2.24"
5
6
7 $IPConfig1=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName1 -
   Subnet $subnet3 -PrivateIpAddress $IPAddress -PublicIpAddress $pip1
   -Primary
8
9
10 $IPConfigName3="IPConfig-3"
11
12
13 $IPAddress="12.5.1.27"
14
15
16 $IPConfig3=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName3 -
   Subnet $subnet1 -PrivateIpAddress $IPAddress -Primary
17

```

```
18
19 $IPConfigName4 = "IPConfig-4"
20
21
22 $IPAddress = "12.5.3.24"
23
24
25 $IPConfig4 = New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName4 -
    Subnet $subnet2 -PrivateIpAddress $IPAddress -Primary
```

Create IP configurations for VPX2.

```
1 $IpConfigName5 = "IPConfig5"
2
3
4 $IPAddress="12.5.2.26"
5
6
7 $IPConfig5=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName5 -
    Subnet $subnet3 -PrivateIpAddress $IPAddress -PublicIpAddress $pip2
    -Primary
8
9
10 $IPConfigName7="IPConfig-7"
11
12
13 $IPAddress="12.5.1.28"
14
15
16 $IPConfig7=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName7 -
    Subnet $subnet1 -PrivateIpAddress $IPAddress -Primary
17
18
19 $IPConfigName8="IPConfig-8"
20
21
22 $IPAddress="12.5.3.28"
23
24
25 $IPConfig8=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName8 -
    Subnet $subnet2 -PrivateIpAddress $IPAddress -Primary
```

Create NICs for VPX1.

```
1 $nic1=New-AzureRmNetworkInterface -Name $nicName1 -ResourceGroupName
    $rgName -Location $locName -IpConfiguration $IpConfig1 -
    NetworkSecurityGroupId $nsg.Id
2
3
4 $nic2=New-AzureRmNetworkInterface -Name $nicName2 -ResourceGroupName
    $rgName -Location $locName -IpConfiguration $IpConfig3 -
    NetworkSecurityGroupId $nsg.Id
5
```

```

6
7 $nic3=New-AzureRmNetworkInterface -Name $nicName3 -ResourceGroupName
  $rgName -Location $locName -IpConfiguration $IpConfig4 -
  NetworkSecurityGroupId $nsg.Id

```

Create NICs for VPX2.

```

1 $nic4=New-AzureRmNetworkInterface -Name $nicName4 -ResourceGroupName
  $rgName -Location $locName -IpConfiguration $IpConfig5 -
  NetworkSecurityGroupId $nsg.Id
2
3
4 $nic5=New-AzureRmNetworkInterface -Name $nicName5 -ResourceGroupName
  $rgName -Location $locName -IpConfiguration $IpConfig7 -
  NetworkSecurityGroupId $nsg.Id
5
6
7 $nic6=New-AzureRmNetworkInterface -Name $nicName6 -ResourceGroupName
  $rgName -Location $locName -IpConfiguration $IpConfig8 -
  NetworkSecurityGroupId $nsg.Id

```

Create VPX1.

This step includes the following substeps:

- Create VM config object
- Set credentials, OS, and image
- Add NICs
- Specify OS disk and create VM

```

1 $suffixNumber = 1
2
3
4 $vmName=$vmNamePrefix + $suffixNumber
5
6
7 $vmConfig=New-AzureRMVMConfig -VMName $vmName -VMSize $vmSize -
  AvailabilitySetId $avSet.Id
8
9
10 $cred=Get-Credential -Message "Type the name and password for VPX login
  ."
11
12
13 $vmConfig=Set-AzureRMVMOperatingSystem -VM $vmConfig -Linux -
  ComputerName $vmName -Credential $cred
14
15
16 $vmConfig=Set-AzureRMVMSourceImage -VM $vmConfig -PublisherName
  $publisher -Offer $offer -Skus $sku -Version $version
17
18

```



```

19 $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic1.Id -
    Primary
20
21
22 $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic2.Id
23
24
25 $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic3.Id
26
27
28 $osDiskName=$vmName + "-" + $osDiskSuffix1
29
30
31 $osVhdUri=$prmStorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds/"
    + $osDiskName + ".vhd"
32
33
34 $vmConfig=Set-AzureRMVMOsDisk -VM $vmConfig -Name $osDiskName -VhdUri
    $osVhdUri -CreateOption fromImage
35
36
37 Set-AzureRmVMPlan -VM $vmConfig -Publisher $publisher -Product $offer -
    Name $sku
38
39
40 New-AzureRMVM -VM $vmConfig -ResourceGroupName $rgName -Location
    $locName

```

Create VPX2.

```

1 $suffixNumber=2
2
3
4 $vmName=$vmNamePrefix + $suffixNumber
5
6
7 $vmConfig=New-AzureRMVMConfig -VMName $vmName -VMSize $vmSize -
    AvailabilitySetId $avSet.Id
8
9
10 $cred=Get-Credential -Message "Type the name and password for VPX login
    ."
11
12
13 $vmConfig=Set-AzureRMVMOperatingSystem -VM $vmConfig -Linux -
    ComputerName $vmName -Credential $cred
14
15
16 $vmConfig=Set-AzureRMVMSourceImage -VM $vmConfig -PublisherName
    $publisher -Offer $offer -Skus $sku -Version $version
17
18
19 $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic4.Id -

```

```
Primary
20
21
22 $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic5.Id
23
24
25 $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic6.Id
26
27
28 $osDiskName=$vmName + "-" + $osDiskSuffix2
29
30
31 $osVhdUri=$prmStorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds/"
    + $osDiskName + ".vhd"
32
33
34 $vmConfig=Set-AzureRMVMOsdisk -VM $vmConfig -Name $osDiskName -VhdUri
    $osVhdUri -CreateOption fromImage
35
36
37 Set-AzureRmVMPlan -VM $vmConfig -Publisher $publisher -Product $offer -
    Name $sku
38
39
40 New-AzureRMVM -VM $vmConfig -ResourceGroupName $rgName -Location
    $locName
```

To view private and public IP addresses assigned to the NICs, type the following commands:

```
1 $nic1.IPConfig
2
3
4 $nic2.IPConfig
5
6
7 $nic3.IPConfig
8
9
10 $nic4.IPConfig
11
12
13 $nic5.IPConfig
14
15
16 $nic6.IPConfig
```

Create Azure load balance (ALB).

This step includes the following substeps:

- Create frontend IP config
- Create health probe
- Create backend address pool

- Create load-balancing rules (HTTP and SSL)
- Create ALB with frontend IP config, backend address pool, and LB rule
- Associate IP config with backend pools

```

1 $frontEndIP1=New-AzureRmLoadBalancerFrontendIpConfig -Name
   $frontEndConfigName1 -PublicIpAddress $pip3
2
3
4 $healthProbe=New-AzureRmLoadBalancerProbeConfig -Name $healthProbeName
   -Protocol Tcp -Port 9000 -IntervalInSeconds 5 -ProbeCount 2
5
6
7 $beAddressPool1=New-AzureRmLoadBalancerBackendAddressPoolConfig -Name
   $backendPoolName1
8
9
10 $lbRule1=New-AzureRmLoadBalancerRuleConfig -Name $lbRuleName1 -
    FrontendIpConfiguration $frontEndIP1 -BackendAddressPool
    $beAddressPool1 -Probe $healthProbe -Protocol Tcp -FrontendPort 80 -
    BackendPort 80 -EnableFloatingIP
11
12
13 $lb=New-AzureRmLoadBalancer -ResourceGroupName $rgName -Name $lbName -
    Location $locName -FrontendIpConfiguration $frontEndIP1 -
    LoadBalancingRule $lbRule1 -BackendAddressPool $beAddressPool1 -
    Probe $healthProbe
14
15
16 $nic2.IpConfigurations[0].LoadBalancerBackendAddressPools.Add($lb.
    BackendAddressPools[0])
17
18
19 $nic5.IpConfigurations[0].LoadBalancerBackendAddressPools.Add($lb.
    BackendAddressPools[0])
20
21
22 \ $lb=\ $lb | Set-AzureRmLoadBalancer
23
24
25 \ $nic2=\ $nic2 | Set-AzureRmNetworkInterface
26
27
28 \ $nic5=\ $nic5 | Set-AzureRmNetworkInterface

```

After you've successfully deployed the Citrix ADC VPX pair, log on to each VPX instance to configure HA-INC, and SNIP and VIP addresses.

1. Type the following command to add HA nodes.

```
1 add ha node 1 PeerNodeNSIP -inc Enabled
```

2. Add private IP addresses of client-side NICs as SNIPs for VPX1 (NIC2) and VPX2 (NIC5)

```

1 add nsip privateIPofNIC2 255.255.255.0 -type SNIP
2
3
4 add nsip privateIPofNIC5 255.255.255.0 -type SNIP
    
```

3. Add load-balancing vserver on the primary node with front-end IP address (public IP) of ALB.

```

1 add lb vserver v1 HTTP FrontEndIPofALB 80
    
```

Related resources:

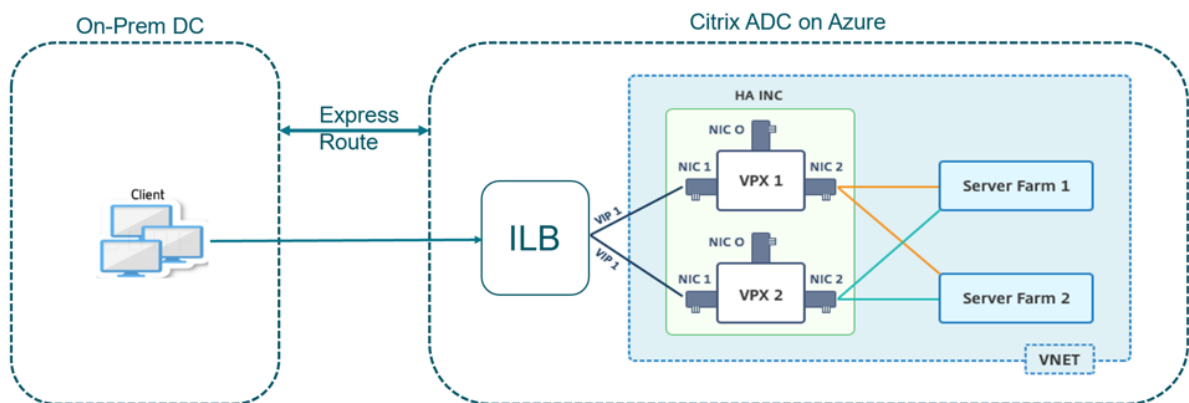
[Configuring GSLB on Active-Standby HA Deployment on Azure](#)

Configure HA-INC nodes by using the Citrix high availability template with Azure ILB

November 13, 2024

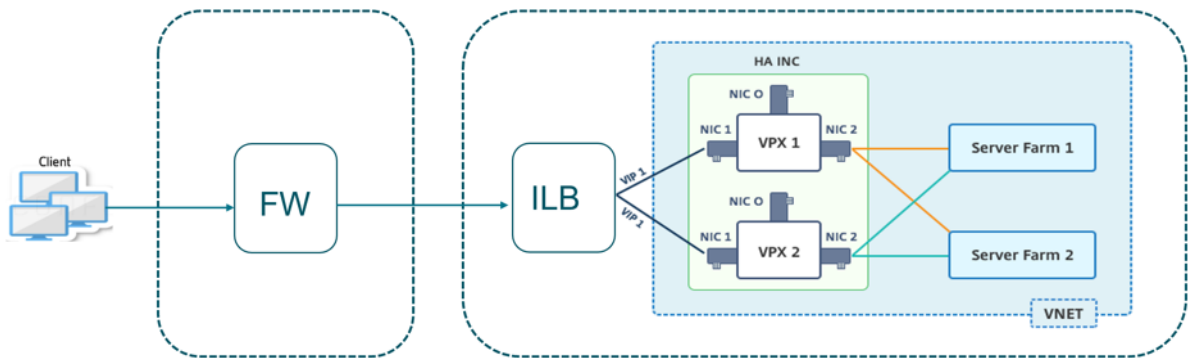
You can quickly and efficiently deploy a pair of VPX instances in HA-INC mode by using the standard template for intranet applications. The Azure internal load balancer (ILB) uses an internal or private IP address for the front end as shown in Figure 1. The template creates two nodes, with three subnets and six NICs. The subnets are for management, client, and server-side traffic with each subnet belonging to a different NIC on each device.

Figure 1: Citrix ADC HA pair for clients in an internal network



You can also use this deployment when the Citrix ADC HA pair is behind a firewall as shown in Figure 2. The public IP address belongs to the firewall and is NAT'd to the front-end IP address of the ILB.

Figure 2: Citrix ADC HA pair with firewall having public IP address



You can get the Citrix ADC HA pair template for intranet applications at the [Azure portal](#).

Complete the following steps to launch the template and deploy a high availability VPX pair by using Azure Availability Sets.

1. From the Azure portal, navigate to the **Custom deployment** page.
2. The **Basics** page appears. Create a Resource Group. Under the **Parameters** tab, enter details for the Region, Admin user name, Admin Password, license type (VM sku), and other fields.

Custom deployment
Deploy from a custom template

12 resources

[Edit template](#) [Edit parameters](#)

Deployment scope
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Resource group * ⓘ [Create new](#)

Parameters

Region * ⓘ

Admin Username ⓘ ✓

Admin Password * ⓘ ✓

Vm Size ⓘ

Vm Sku ⓘ

Vnet Name ⓘ

Vnet Resource Group ⓘ

Vnet New Or Existing

Subnet Name-01 ⓘ

Subnet Name-11 ⓘ

Subnet Name-12 ⓘ

Subnet Address Prefix-01 ⓘ

Subnet Address Prefix-11 ⓘ

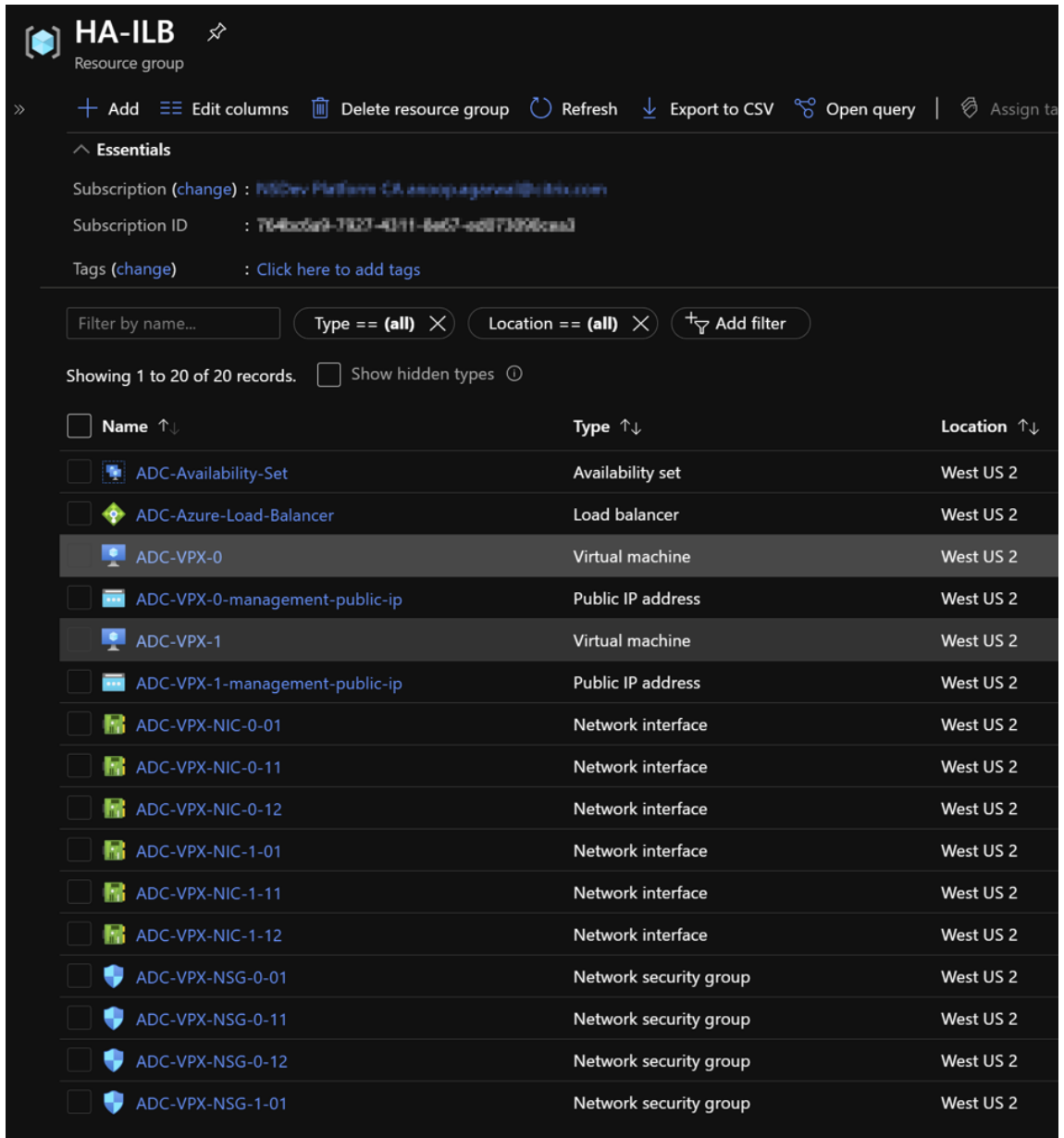
[Review + create](#) [< Previous](#) **[Next : Review + create >](#)**

3. Click **Next : Review + create >**.

It might take a moment for the Azure Resource Group to be created with the required configurations. After completion, select the Resource Group in the Azure portal to see the configuration details, such as LB rules, back-end pools, health probes. The high availability pair appears as ADC-VPX-0 and ADC-VPX-1.

If further modifications are required for your HA setup, such as creating more security rules and ports, you can do that from the Azure portal.

Once the required configuration is complete, the following resources are created.



4. You must log on to **ADC-VPX-0** and **ADC-VPX-1** nodes, and validate the following configuration:

- NSIP addresses for both nodes must be in the management subnet.
- On the primary (ADC-VPX-0) and secondary (ADC-VPX-1) nodes, you must see two SNIP addresses. One SNIP (client subnet) is used for responding to ILB probes and the other SNIP (server subnet) is used for back-end server communication.

Note:

In the HA-INC mode, the SNIP address of ADC-VPX-0 and ADC-VPX-1 VMs are different, un-

like with the classic on-premises ADC HA deployment.

On the Primary node (ADC-VPX-0)

```
> sh ip
-----
1) 10.11.0.5 0 NetScaler IP Active Enabled Enabled NA Enabled
2) 10.11.1.5 0 SNIP Active Enabled Enabled NA Enabled
3) 10.11.3.4 0 SNIP Active Enabled Enabled NA Enabled
Done
>
```

```
> sh ha node
1) Node ID: 0
   IP: 10.11.0.5 (ADC-VPX-0)
   Node State: UP
   Master State: Primary
   Fail-Safe Mode: OFF
   INC State: ENABLED
   Sync State: ENABLED
   Propagation: ENABLED
   Enabled Interfaces : 0/1 1/1 1/2
   Disabled Interfaces : None
   HA MON ON Interfaces : None
   HA HEARTBEAT OFF Interfaces : None
   Interfaces on which heartbeats are not seen : 1/1 1/2
   Interfaces causing Partial Failure: None
   SSL Card Status: NOT PRESENT
   Sync Status Strict Mode: DISABLED
   Hello Interval: 200 msec
   Dead Interval: 3 secs
   Node in this Master State for: 0:0:20:26 (days:hrs:min:sec)
2) Node ID: 1
   IP: 10.11.0.4
   Node State: UP
   Master State: Secondary
   Fail-Safe Mode: OFF
   INC State: ENABLED
   Sync State: SUCCESS
   Propagation: ENABLED
   Enabled Interfaces : 0/1 1/1 1/2
   Disabled Interfaces : None
   HA MON ON Interfaces : None
   HA HEARTBEAT OFF Interfaces : None
   Interfaces on which heartbeats are not seen : 1/1 1/2
   Interfaces causing Partial Failure: None
   SSL Card Status: NOT PRESENT
Done
>
```

On the secondary node (ADC-VPX-1)

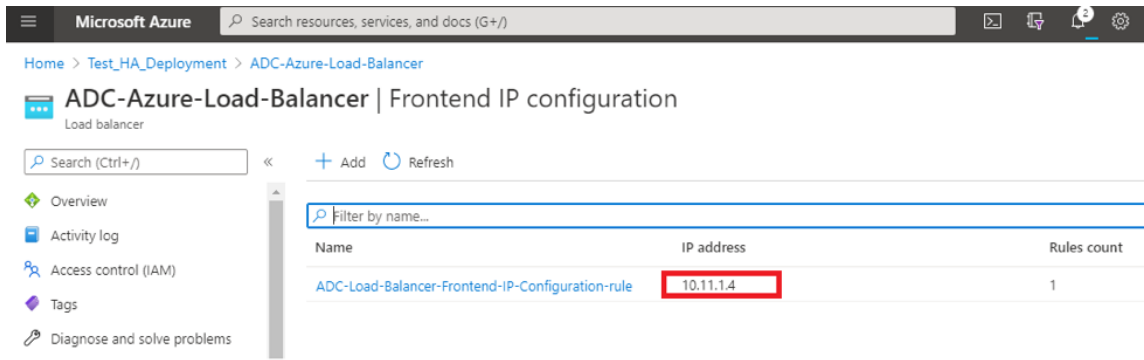
```
> sh ip
-----
1) 10.11.0.4 0 NetScaler IP Active Enabled Enabled NA Enabled
2) 10.11.1.6 0 SNIP Active Enabled Enabled NA Enabled
3) 10.11.3.5 0 SNIP Active Enabled Enabled NA Enabled
Done
>
```



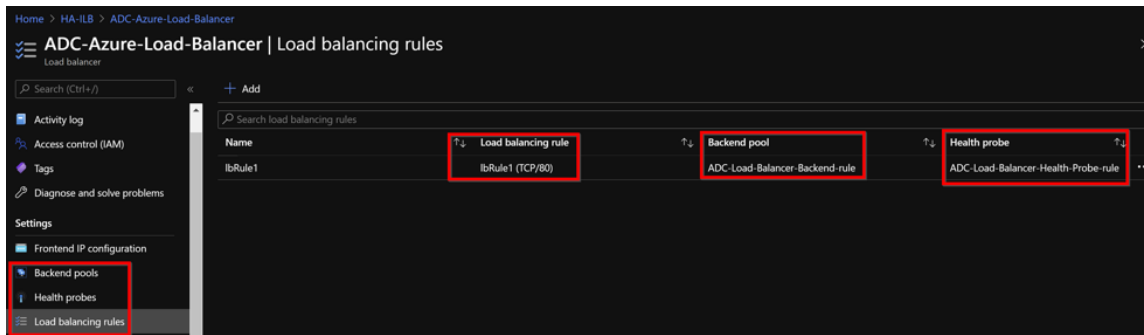
```
> sh ha node
1) Node ID: 0
   IP: 10.11.0.4 (ADC-VPX-1)
   Node State: UP
   Master State: Secondary
   Fail-Safe Mode: OFF
   INC State: ENABLED
   Sync State: SUCCESS
   Propagation: ENABLED
   Enabled Interfaces : 0/1 1/1 1/2
   Disabled Interfaces : None
   HA MON ON Interfaces : None
   HA HEARTBEAT OFF Interfaces : None
   Interfaces on which heartbeats are not seen : 1/1 1/2
   Interfaces causing Partial Failure: None
   SSL Card Status: NOT PRESENT
   Sync Status Strict Mode: DISABLED
   Hello Interval: 200 msec
   Dead Interval: 3 sec
   Node in this Master State for: 0:0:24:18 (days:hrs:min:sec)
2) Node ID: 1
   IP: 10.11.0.5
   Node State: UP
   Master State: Primary
   Fail-Safe Mode: OFF
   INC State: ENABLED
   Sync State: ENABLED
   Propagation: ENABLED
   Enabled Interfaces : 0/1 1/1 1/2
   Disabled Interfaces : None
   HA MON ON Interfaces : None
   HA HEARTBEAT OFF Interfaces : None
   Interfaces on which heartbeats are not seen : 1/1 1/2
   Interfaces causing Partial Failure: None
   SSL Card Status: NOT PRESENT

Done
> █
```

5. After the primary and secondary nodes are UP and the Synchronization status is **SUCCESS**, you must configure the load balancing virtual server or the gateway virtual server on the primary node (ADC-VPX-0) with the private floating IP (FIP) address of the ADC Azure load balancer. For more information, see the [Sample configuration](#) section.
6. To find the private IP address of ADC Azure load balancer, navigate to **Azure portal > ADC Azure Load Balancer > Frontend IP configuration**.



7. In the **Azure Load Balancer** configuration page, the ARM template deployment helps create the LB rule, back-end pools, and health probes.



- The LB Rule (LbRule1) uses port 80, by default.

lbRule1
ADC-Azure-Load-Balancer

Save Discard Delete

i A load balancing rule distributes incoming traffic that is sent to a selected IP address and port combination across a group of backend pool instances. Only backend instances that the health probe considers healthy receive new traffic.

Name *
lbRule1

IP Version *
 IPv4 IPv6

Frontend IP address * ⓘ
10.11.1.4 (ADC-Load-Balancer-Frontend-IP-Configuration-rule) ✓

Protocol
 TCP UDP

Port *
80

Backend port * ⓘ
80

- Edit the rule to use port 443, and save the changes.

Note

For enhanced security, Citrix recommends you to use SSL port 443 for LB virtual server or Gateway virtual server.

lbRule1
ADC-Azure-Load-Balancer

Save Discard Delete

i A load balancing rule distributes incoming traffic that is sent to a selected IP address and port combination across a group of backend pool instances. Only backend instances that the health probe considers healthy receive new traffic.

Name *
lbRule1

IP Version *
 IPv4 IPv6

Frontend IP address * ⓘ
10.11.1.4 (ADC-Load-Balancer-Frontend-IP-Configuration-rule) ▼

Protocol
 TCP UDP

Port *
443 ✓

Backend port * ⓘ
443

Backend pool ⓘ
ADC-Load-Balancer-Backend-rule (2 virtual machines) ▼

Health probe ⓘ
ADC-Load-Balancer-Health-Probe-rule (TCP:9000) ▼

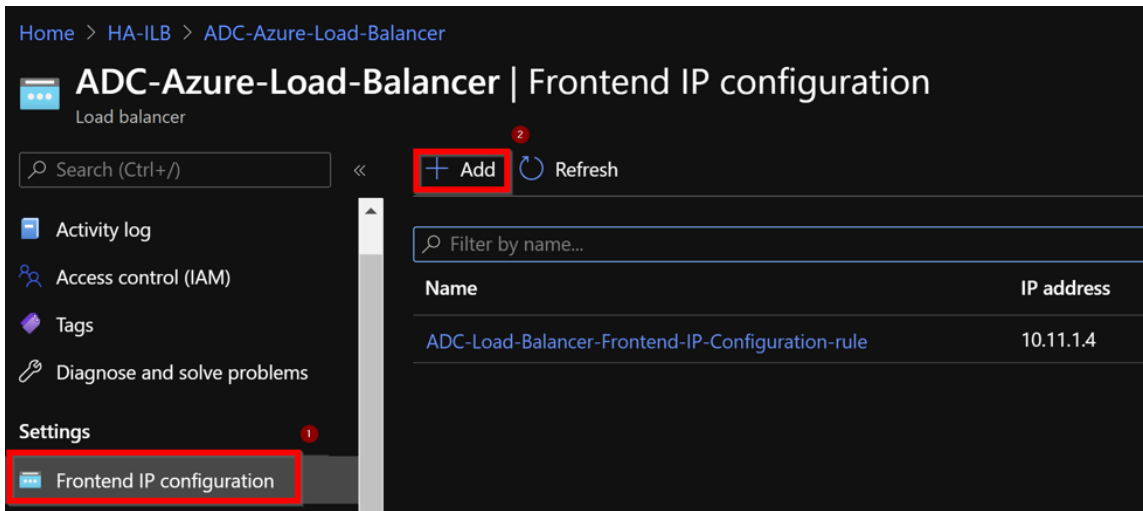
Session persistence ⓘ
None ▼

Idle timeout (minutes) ⓘ
4

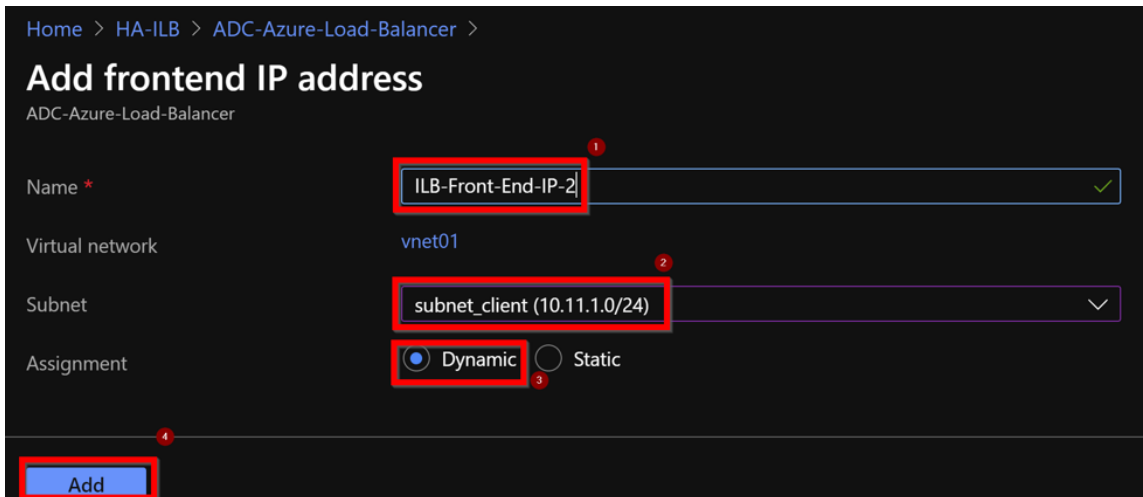
Floating IP ⓘ
Enabled

To add more VIP addresses on the ADC, perform the following steps:

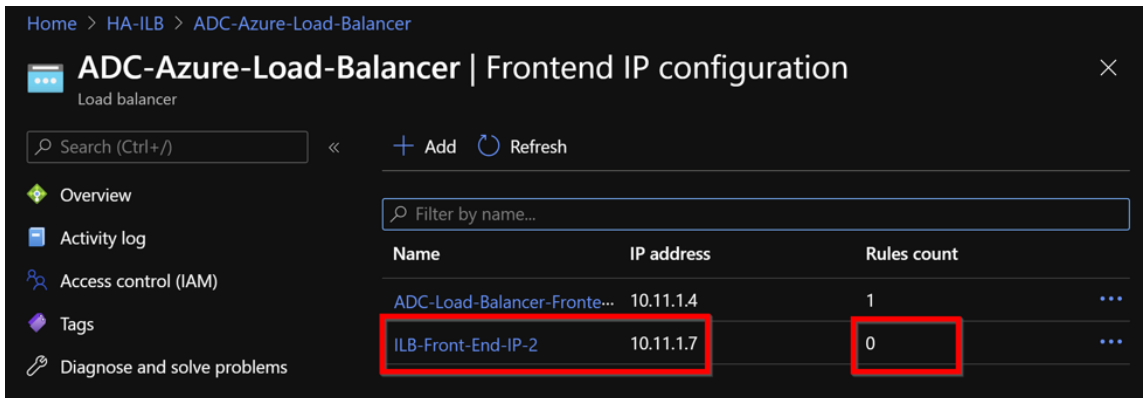
1. Navigate to **Azure Load Balancer > Frontend IP configuration**, and click **Add** to create a new internal load balancer IP address.



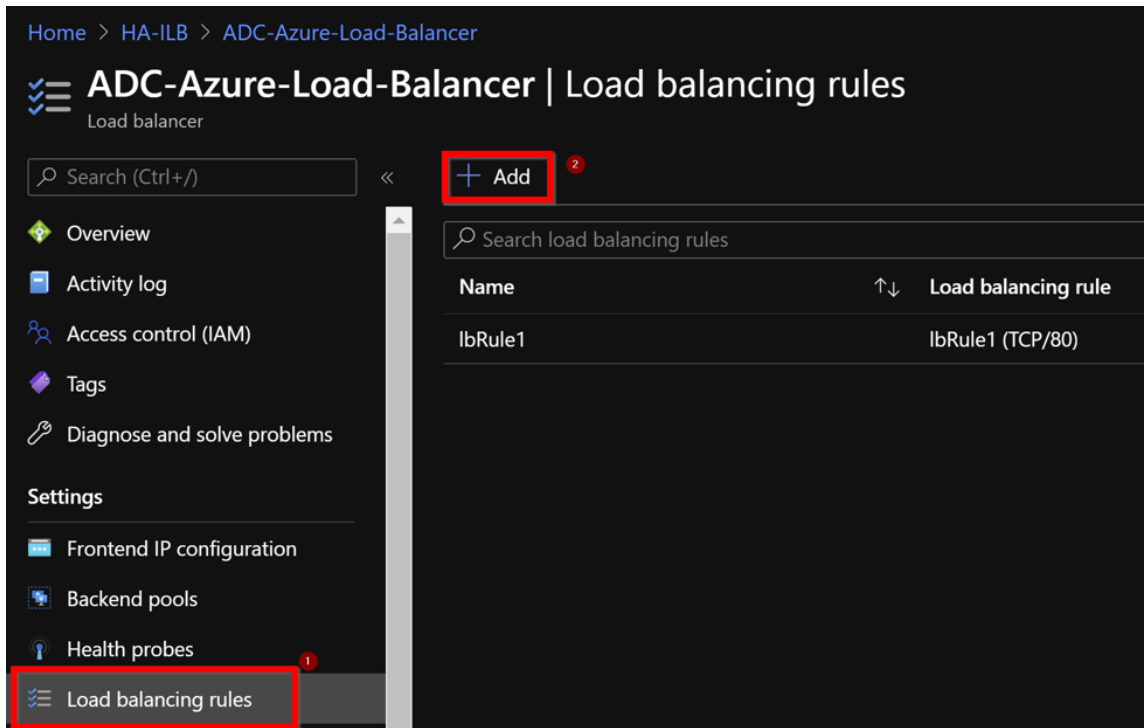
2. In the **Add frontend IP address** page, enter a name, choose the client subnet, assign either dynamic or static IP address, and click **Add**.



3. The front-end IP address is created but an LB Rule is not associated. Create a new load balancing rule, and associate it with the front-end IP address.



4. In the **Azure Load Balancer** page, select **Load balancing rules**, and then click **Add**.



5. Create a new LB Rule by choosing the new front-end IP address and the port. **Floating IP** field must be set to **Enabled**.

Home > HA-ILB > ADC-Azure-Load-Balancer >

Add load balancing rule

ADC-Azure-Load-Balancer

i A load balancing rule distributes incoming traffic that is sent to a selected IP address and port combination across a group of backend pool instances. Only backend instances that the health probe considers healthy receive new traffic.

1 Name *
lbrule2 ✓

IP Version *
 IPv4 IPv6

2 Frontend IP address * ⓘ
10.11.1.7 (ILB-Front-End-IP-2) ✓

Protocol
 TCP UDP

3 Port * **3**
443 ✓

4 Backend port * ⓘ **4**
443 ✓

5 Backend pool ⓘ **5**
ADC-Load-Balancer-Backend-rule (2 virtual machines) ✓

Health probe ⓘ
ADC-Load-Balancer-Health-Probe-rule (TCP:9000) ✓

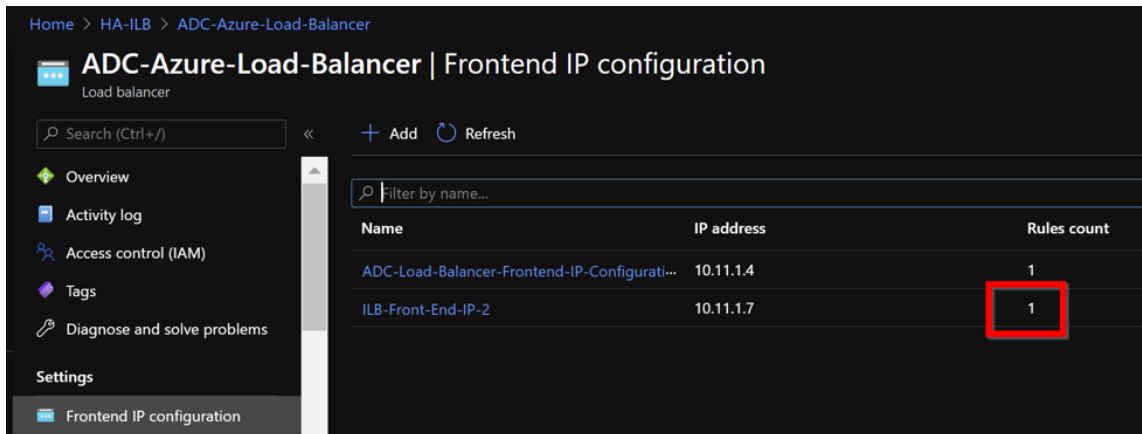
Session persistence ⓘ
None ✓

Idle timeout (minutes) ⓘ
 4

6 Floating IP ⓘ **6**
Disabled Enabled

7 OK **7**

6. Now the **Frontend IP configuration** shows the LB rule that is applied.



Sample configuration

To configure a gateway VPN virtual server and load balancing virtual server, run the following commands on the primary node (ADC-VPX-0). The configuration auto synchronizes to the secondary node (ADC-VPX-1).

Gateway sample configuration

```
1 Enable feature aaa LB SSL SSLVPN
2 add ip 10.11.1.4 255.255.255.0 -type VIP
3 add vpn vserver vpn_ssl SSL 10.11.1.4 443
4 add ssl certKey ckp -cert wild-cgwsanity.cer -key wild-cgwsanity.key
5 bind ssl vserver vpn_ssl -certkeyName ckp
```

Load balancing sample configuration

```
1 add ip 10.11.1.7 255.255.255.0 -type VIP
2 add lb vserver lb_vs1 SSL 10.11.1.7 443
3 bind ssl vserver lb_vs1 -certkeyName ckp
```

You can now access the load balancing or VPN virtual server using the fully qualified domain name (FQDN) associated with the internal IP address of ILB.

See the **Resources** section for more information about how to configure the load-balancing virtual server.

Resources:

The following links provide additional information related to HA deployment and virtual server configuration:

- [Configuring high availability nodes in different subnets](#)
- [Set up basic load balancing](#)

Related resources:

- [Configure a high-availability setup with multiple IP addresses and NICs by using PowerShell commands](#)
- [Configuring GSLB on Active-Standby HA Deployment on Azure](#)

Add Azure autoscale settings

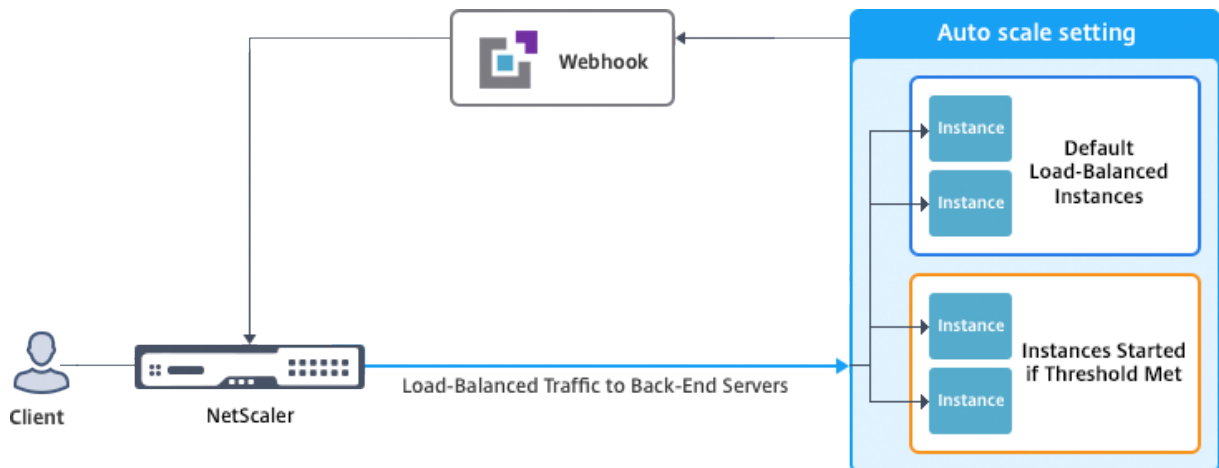
November 13, 2024

Efficient hosting of applications in a cloud involves easy and cost-effective management of resources depending on the application demand. To meet increasing demand, you have to scale network resources upward. Whether demand subsides, you must scale down to avoid the unnecessary cost of idle resources. To minimize the cost of running the application, you have to constantly monitor traffic, memory and CPU use, and so on. However, monitoring traffic manually is cumbersome. For the application environment to scale up or down dynamically, you must automate the processes of monitoring traffic and of scaling resources up and down whenever necessary.

You can use autoscale with Azure virtual machine scale sets (VMSS) for VPX multi-IP standalone and high availability deployment on Azure.

Integrated with Azure virtual machine scale sets (VMSS) and autoscale feature, the Citrix ADC VPX instance provides the following advantages:

- **Load balance and management:** Auto configures servers to scale up and scale down, depending on demand. The VPX instance auto detects VMSS autoscale setting in the back-end subnet in the same resource group as the VPX instance and allows user to select the VMSS autoscale setting to balance the load. All of this is done by auto configuring Citrix ADC virtual and subnet IP addresses on the VPX instance.
- **High availability:** Detects autoscale groups in the same resource group and load-balance servers.
- **Better network availability:** The VPX instance supports back-end servers on different virtual networks (VNets).



For more information, see the following Azure topic

- [Virtual Machine Scale Sets Documentation](#)
- [Overview of autoscale in Microsoft Azure Virtual Machines, Cloud Services, and Web Apps](#)

Before you begin

1. Read Azure-related usage guidelines. For more information, see [Deploy a Citrix ADC VPX instance on Microsoft Azure](#).
2. Create one or more Citrix ADC VPX instances with three network interfaces on Azure according to your requirement (standalone or high availability deployment).
3. Open the TCP 9001 port on the network security group of the 0/1 interface of the VPX instance. The VPX instance uses this port to receive the scale-out and scale-in notification.
4. Create an Azure virtual machine scale set (VMSS) in the same resource group. If you don't have an existing VMSS configuration, complete the following tasks:
 - a) Create a VMSS
 - b) Enable autoscale on VMSS
 - c) Create scale-in and scale-out policy in VMSS autoscale setting

For more information, see [Overview of autoscale with Azure virtual machine scale sets](#).

5. Create an Azure Active Directory (ADD) application and service principal that can access resources. Assign contributor role to the newly created AAD application. For more information, see [Use portal to create an Azure Active Directory application and service principal that can access resources](#).

Add VMSS to a Citrix ADC VPX instance

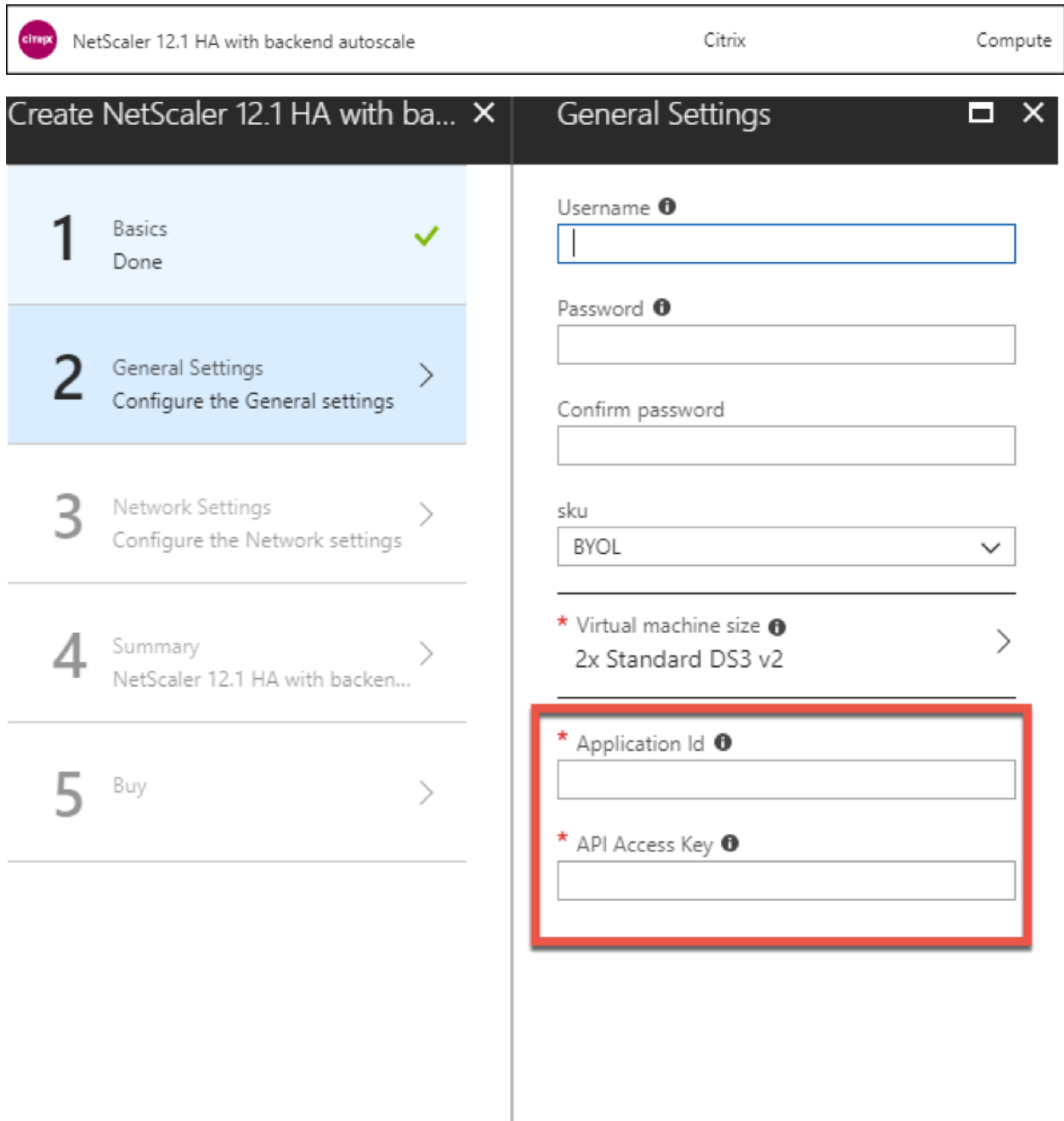
You can add the autoscale setting to a VPX instance with a single click by using the GUI. Complete these steps to add the autoscale setting to the VPX instance:

1. Log on to the VPX instance.
2. When you log on to the Citrix ADC VPX instance for the first time, you see the Set Credentials page. Add the required Azure credentials for the autoscale feature to work.

The screenshot shows the Citrix NetScaler VPX AZURE Configuration page. The main header is "Citrix NetScaler VPX AZURE". Below it are two tabs: "Dashboard" and "Configuration". The "Configuration" tab is active. The page title is "Set Credentials", indicated by a back arrow icon. There are three input fields: "Tenant ID", "Application ID", and "Application Secret". At the bottom, there are two buttons: "OK" and "Cancel".

The Set Credential page appears only when the application ID and API access key is not set or the correct application ID and API access keys (same as application secret) is not set in the Azure portal.

When you deploy the “NetScaler 12.1 HA with backend autoscale” offer from the Azure market place, the Azure portal prompts for Azure service principal credentials (application ID and API access key).



For information about how to create an application ID see [Adding an application](#) and to create an access key or application secret see [Configure a client application to access web APIs](#).

3. In the default cloud profile page, enter the details, as shown in the following example, and click Create.

Dashboard Configuration

Name
 ?

Virtual Server IP Address*

Load Balancing Server Protocol*

Load Balancing Server Port*

Auto Scale Setting*

Auto Scale Setting Protocol

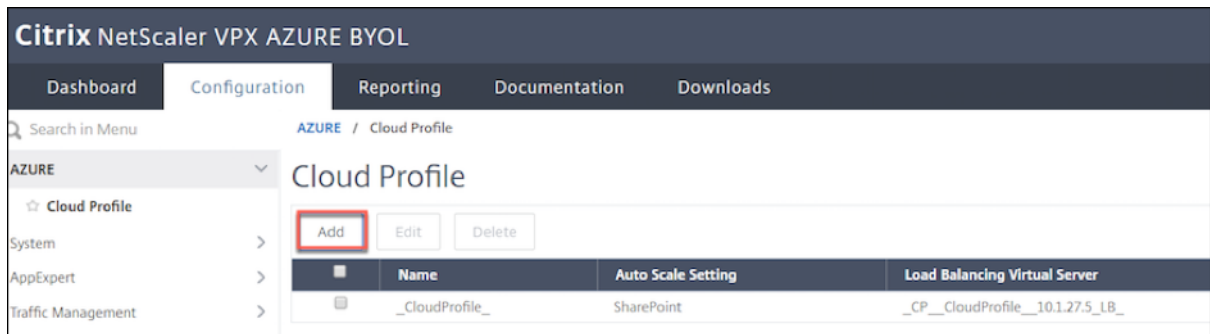
Auto Scale Setting Port*

Create Skip

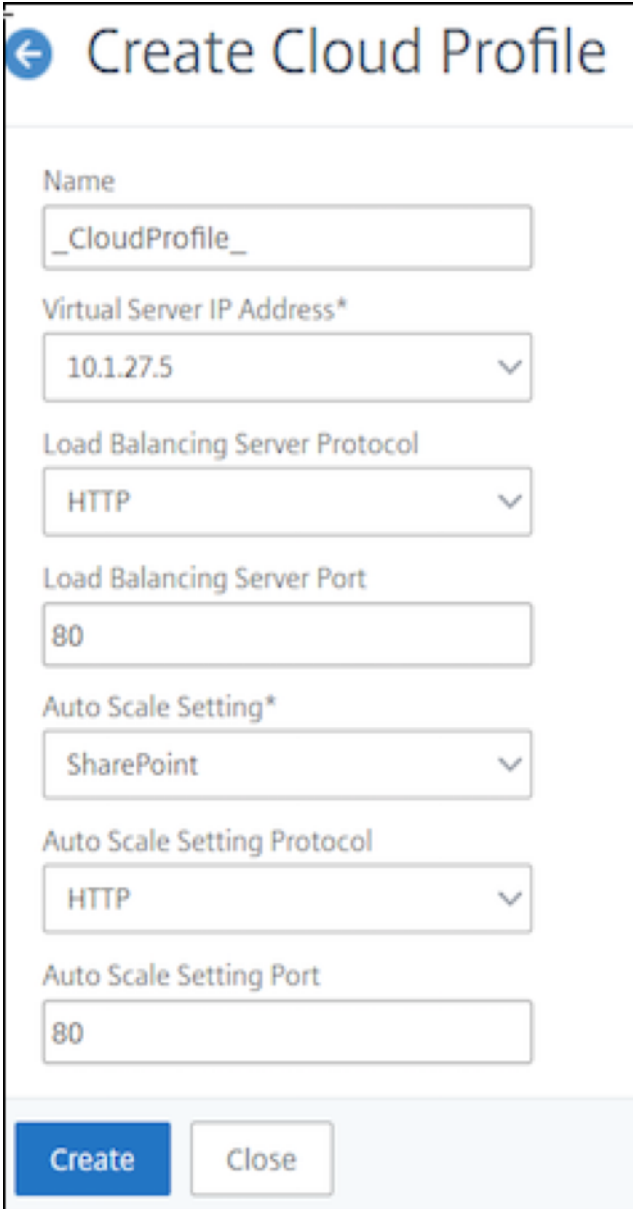
Points to keep in mind while creating a cloud profile

- The virtual server IP address is auto-populated from the free IP address available to the VPX instance. For more information, see [Assign multiple IP addresses to virtual machines using the Azure portal](#).
- Autoscale setting is prepopulated from the VMSS auto scale setting configured in current resource group on your Azure account. For more information, see [Overview of autoscale with Azure virtual machine scale sets](#).
- While selecting the Auto Scaling Group protocol and port, ensure your servers listen on those protocol and ports and you bind the correct monitor in the service group. By default, TCP monitor is used.
- For SSL Protocol type Autos Scaling, after you create the Cloud Profile the load balance virtual server or service group will be down because of a missing certificate. You can bind the certificate to the virtual server or service group manually.

After first time logon, if you want to create a cloud profile, on the GUI go to System > Azure > Cloud Profile and click Add.



The Create Cloud Profile configuration page appears.



Create Cloud Profile

Name
CloudProfile

Virtual Server IP Address*
10.1.27.5

Load Balancing Server Protocol
HTTP

Load Balancing Server Port
80

Auto Scale Setting*
SharePoint

Auto Scale Setting Protocol
HTTP

Auto Scale Setting Port
80

Create Close

Cloud Profile creates a Citrix ADC load-balancing (LB) virtual server (virtual server) and a service group with members (servers) as the servers of the Auto Scaling Group. Your back-end servers should be reachable through the SNIP configured on the VPX instance.

To view autoscale-related information in the Azure portal, go to All service > Virtual machine scale set > Select Virtual machine scale set > Scaling.

Configure GSLB on Citrix ADC VPX instances

November 21, 2024

Citrix ADC appliances configured for global server load balancing (GSLB) provide disaster recovery and continuous availability of applications by protecting against points of failure in a wide area network (WAN). GSLB can balance the load across data centers by directing client requests to the closest or best performing data center, or to surviving data centers in case of an outage.

This section describes how to enable GSLB on VPX instances on two sites in a Microsoft Azure environment, by using Windows PowerShell commands.

Note:

For more information about GSLB, see [Global Server Load Balancing](#).

You can configure GSLB on a Citrix ADC VPX instances on Azure, in two steps:

1. Create a VPX instance with multiple NICs and multiple IP addresses, on each site.
2. Enable GSLB on the VPX instances.

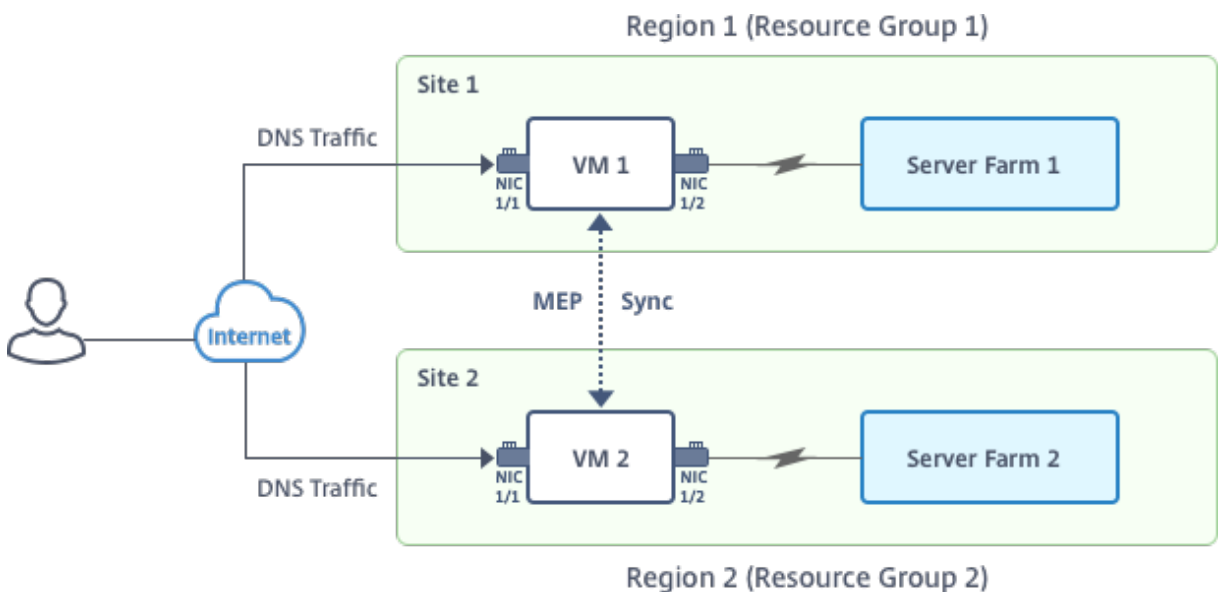
Note:

For more information about configuring multiple NICs and IP addresses see: [Configure multiple IP addresses for a Citrix ADC VPX instance in standalone mode by using PowerShell commands](#)

Scenario

This scenario includes two sites - Site 1 and Site 2. Each site has a VM (VM1 and VM2) configured with multiple NICs, multiple IP addresses, and GSLB.

Figure. GSLB setup implemented across two sites - Site 1 and Site 2.



In this scenario, each VM has three NICs - NIC 0/1, 1/1, and 1/2. Each NIC can have multiple private and public IP addresses. The NICs are configured for the following purposes.

- NIC 0/1: to serve management traffic
- NIC 1/1: to serve client-side traffic
- NIC 1/2: to communicate with back-end servers

For information about the IP addresses configured on each NIC in this scenario, see the IP configuration details section.

Parameters

Following are sample parameters settings for this scenario in this document.

```
1 $location="West Central US"
2
3 $vnetName="NSVPX-vnet"
4
5 $RGName="multiIP-RG"
6
7 $prmStorageAccountName="multiipstorageacctnt"
8
9 $avSetName="MultiIP-avset"
10
11 $vmSize="Standard\_DS3\_V2"
```

Note:

The minimum requirement for a VPX instance is 2 vCPUs and 2GB RAM.

```
1 $publisher="citrix"
2
3 $offer="netscalervpx111"
4
5 $sku="netscalerbyol"
6
7 $version="latest"
8
9 $vmNamePrefix="MultiIPVPX"
10
11 $nicNamePrefix="MultiipVPX"
12
13 $osDiskSuffix="osdiskdb"
14
15 $numberOfVMs=1
16
17 $ipAddressPrefix="10.0.0."
18
19 $ipAddressPrefix1="10.0.1."
20
```

```
21 $ipAddressPrefix2="10.0.2."  
22  
23 $pubIPName1="MultiIP-pip1"  
24  
25 $pubIPName2="MultiIP-pip2"  
26  
27 $IpConfigName1="IPConfig1"  
28  
29 $IPConfigName2="IPConfig-2"  
30  
31 $IPConfigName3="IPConfig-3"  
32  
33 $IPConfigName4="IPConfig-4"  
34  
35 $frontendSubnetName="default"  
36  
37 $backendSubnetName1="subnet\_1"  
38  
39 $backendSubnetName2="subnet\_2"  
40  
41 $suffixNumber=10
```

Create a VM

Follow steps 1-10 to create VM1 with multiple NICs and multiple IP addresses, by using PowerShell commands:

1. Create resource group
2. Create storage account
3. Create availability set
4. Create virtual network
5. Create public IP address
6. Create NICs
7. Create VM config object
8. Get credentials and set OS properties for the VM
9. Add NICs
10. Specify OS disk and create VM

After you complete all the steps and commands to create VM1, repeat these steps to create VM2 with parameters specific to it.

Create resource group

```
1 New-AzureRMResourceGroup -Name $RGName -Location $location
```

Create storage account

```
1 $prmStorageAccount=New-AzureRMStorageAccount -Name
  $prmStorageAccountName -ResourceGroupName $RGName -Type Standard_LRS
  -Location $location
```

Create availability set

```
1 $avSet=New-AzureRMAvailabilitySet -Name $avSetName -ResourceGroupName
  $RGName -Location $location
```

Create virtual network

1. Add subnets.

```
1 $subnet1=New-AzureRmVirtualNetworkSubnetConfig -Name
  $frontendSubnetName -AddressPrefix "10.0.0.0/24"
2 $subnet2=New-AzureRmVirtualNetworkSubnetConfig -Name
  $backendSubnetName1 -AddressPrefix "10.0.1.0/24"
3 $subnet3=New-AzureRmVirtualNetworkSubnetConfig -Name
  $backendSubnetName2 -AddressPrefix "10.0.2.0/24"
```

2. Add virtual network object.

```
1 $vnet=New-AzureRmVirtualNetwork -Name $vnetName -ResourceGroupName
  $RGName -Location $location -AddressPrefix 10.0.0.0/16 -Subnet
  $subnet1, $subnet2, $subnet3
```

3. Retrieve subnets.

```
1 $frontendSubnet=$vnet.Subnets|?{
2   $_.Name -eq $frontendSubnetName }
3
4 $backendSubnet1=$vnet.Subnets|?{
5   $_.Name -eq $backendSubnetName1 }
6
7 $backendSubnet2=$vnet.Subnets|?{
8   $_.Name -eq $backendSubnetName2 }
```

Create public IP address

```
1 $pip1=New-AzureRmPublicIpAddress -Name $pubIPName1 -ResourceGroupName
   $RGName -Location $location -AllocationMethod Dynamic
2 $pip2=New-AzureRmPublicIpAddress -Name $pubIPName2 -ResourceGroupName
   $RGName -Location $location -AllocationMethod Dynamic
```

Create NICs

Create NIC 0/1

```
1 $nic1Name=$nicNamePrefix + $suffixNumber + "-Mgmt"
2 $ipAddress1=$ipAddressPrefix + $suffixNumber
3 $IPConfig1=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName1 -
   SubnetId $frontendSubnet.Id -PublicIpAddress $pip1 -PrivateIpAddress
   $ipAddress1 -Primary
4 $nic1=New-AzureRMNetworkInterface -Name $nic1Name -ResourceGroupName
   $RGName -Location $location -IpConfiguration $IpConfig1
```

Create NIC 1/1

```
1 $nic2Name $nicNamePrefix + $suffixNumber + "-frontend"
2 $ipAddress2=$ipAddressPrefix1 + ($suffixNumber)
3 $ipAddress3=$ipAddressPrefix1 + ($suffixNumber + 1)
4 $IPConfig2=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName2 -
   PublicIpAddress $pip2 -SubnetId $backendSubnet1.Id -
   PrivateIpAddress $ipAddress2 -Primary
5 $IPConfig3=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName3 -
   SubnetId $backendSubnet1.Id -PrivateIpAddress $ipAddress3
6 nic2=New-AzureRMNetworkInterface -Name $nic2Name -ResourceGroupName
   $RGName -Location $location -IpConfiguration $IpConfig2, $IpConfig3
```

Create NIC 1/2

```
1 $nic3Name=$nicNamePrefix + $suffixNumber + "-backend"
2 $ipAddress4=$ipAddressPrefix2 + ($suffixNumber)
3 $IPConfig4=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName4 -
   SubnetId $backendSubnet2.Id -PrivateIpAddress $ipAddress4 -Primary
4 $nic3=New-AzureRMNetworkInterface -Name $nic3Name -ResourceGroupName
   $RGName -Location $location -IpConfiguration $IpConfig4
```

Create VM config object

```
1 $vmName=$vmNamePrefix
2 $vmConfig=New-AzureRMVMConfig -VMName $vmName -VMSize $vmSize -
   AvailabilitySetId $avSet.Id
```

Get credentials and set OS properties

```

1 $cred=Get-Credential -Message "Type the name and password for VPX login
  ."
2 $vmConfig=Set-AzureRMVMOperatingSystem -VM $vmConfig -Linux -
  ComputerName $vmName -Credential $cred
3 $vmConfig=Set-AzureRMVMSourceImage -VM $vmConfig -PublisherName
  $publisher -Offer $offer -Skus $sku -Version $version
  
```

Add NICs

```

1 $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic1.Id -
  Primary
2 $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic2.Id
3 $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic3.Id
  
```

Specify OS disk and create VM

```

1 $osDiskName=$vmName + "-" + $osDiskSuffix
2 $osVhdUri=$prmStorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds/"
  + $osDiskName + ".vhd"
3 $vmConfig=Set-AzureRMVMOSDisk -VM $vmConfig -Name $osDiskName -VhdUri
  $osVhdUri -CreateOption fromImage
4 Set-AzureRmVMPlan -VM $vmConfig -Publisher $publisher -Product $offer -
  Name $sku
5 New-AzureRMVM -VM $vmConfig -ResourceGroupName $RGName -Location
  $location
  
```

Note

Repeat steps 1-10 listed in “Create Multi-NIC VMs by Using PowerShell Commands” to create VM2 with parameters specific to VM2.

IP configuration details

The following IP addresses are used.

Table 1. IP addresses used in VM1

NIC	Private IP	Public IP (PIP)	Description
0/1	10.0.0.10	PIP1	Configured as NSIP (management IP)
1/1	10.0.1.10	PIP2	Configured as SNIP/GSLB Site IP

NIC	Private IP	Public IP (PIP)	Description
-	10.0.1.11	-	Configured as LB server IP. Public IP is not mandatory
1/2	10.0.2.10	-	Configured as SNIP for sending monitor probes to services; public IP is not mandatory

Table 2. IP addresses used in VM2

NIC	Internal IP	Public IP (PIP)	Description
0/1	20.0.0.10	PIP4	Configured as NSIP (management IP)
1/1	20.0.1.10	PIP5	Configured as SNIP/GSLB Site IP
-	20.0.1.11	-	Configured as LB server IP. Public IP is not mandatory
1/2	20.0.2.10	-	Configured as SNIP for sending monitor probes to services; public IP is not mandatory

Here are sample configurations for this scenario, showing the IP addresses and initial LB configurations as created through the Citrix ADC VPX CLI for VM1 and VM2.

Here's an example configuration on VM1.

```

1 add ns ip 10.0.1.10 255.255.255.0 -mgmtAccess ENABLED
2 Add nsip 10.0.2.10 255.255.255.0
3 add service svc1 10.0.1.10 ADNS 53
4 add lb vserver v1 HTTP 10.0.1.11 80
5 add service s1 10.0.2.120 http 80
6 Add service s2 10.0.2.121 http 80
7 Bind lb vs v1 s[1-2]
```

Here's an example configuration on VM2.

```
1 add ns ip 20.0.1.10 255.255.255.0 -mgmtAccess ENABLED
2 Add nsip 20.0.2.10 255.255.255.0
3 add service svc1 20.0.1.10 ADNS 53
4 add lb vserver v1 HTTP 20.0.1.11 80
5 Add service s1 20.0.2.90 http 80
6 Add service s2 20.0.2.91 http 80
7 Bind lb vs v1 s[1-2]
```

Configure GSLB sites and other settings

Perform the tasks described in the following topic to configure the two GSLB sites and other necessary settings:

Global Server Load Balancing

For more information, see this support article:

<https://support.citrix.com/article/CTX110348>

Here's an example GSLB configuration on VM1 and VM2.

```
1 enable ns feature LB GSLB
2 add gslb site site1 10.0.1.10 -publicIP PIP2
3 add gslb site site2 20.0.1.10 -publicIP PIP5
4 add gslb service site1_gslb_http_svc1 10.0.1.11 HTTP 80 -publicIP PIP3
  -publicPort 80 -siteName site1
5 add gslb service site2_gslb_http_svc1 20.0.1.11 HTTP 80 -publicIP PIP6
  -publicPort 80 -siteName site2
6 add gslb vserver gslb_http_vip1 HTTP
7 bind gslb vserver gslb_http_vip1 -serviceName site2_gslb_http_svc1
8 bind gslb vserver gslb_http_vip1 -serviceName site1_gslb_http_svc1
9 bind gslb vserver gslb_http_vip1 -domainName www.gslbindia.com -TTL 5
```

You've configured GSLB on Citrix ADC VPX instances running on Azure.

Configure GSLB on an active-standby high-availability setup

November 21, 2024

You can configure global server load balancing (GSLB) on active-standby HA deployment on Azure in three steps:

1. Create a VPX HA pair on each GSLB site. See [Configure a high-availability setup with multiple IP addresses and NICs](#) for information about how to create an HA pair.
2. Configure the Azure Load Balancer (ALB) with the front-end IP address and rules to allow GSLB and DNS traffic.

This step involves the following substeps. See the scenario in this section for the PowerShell commands used to complete these substeps.

```

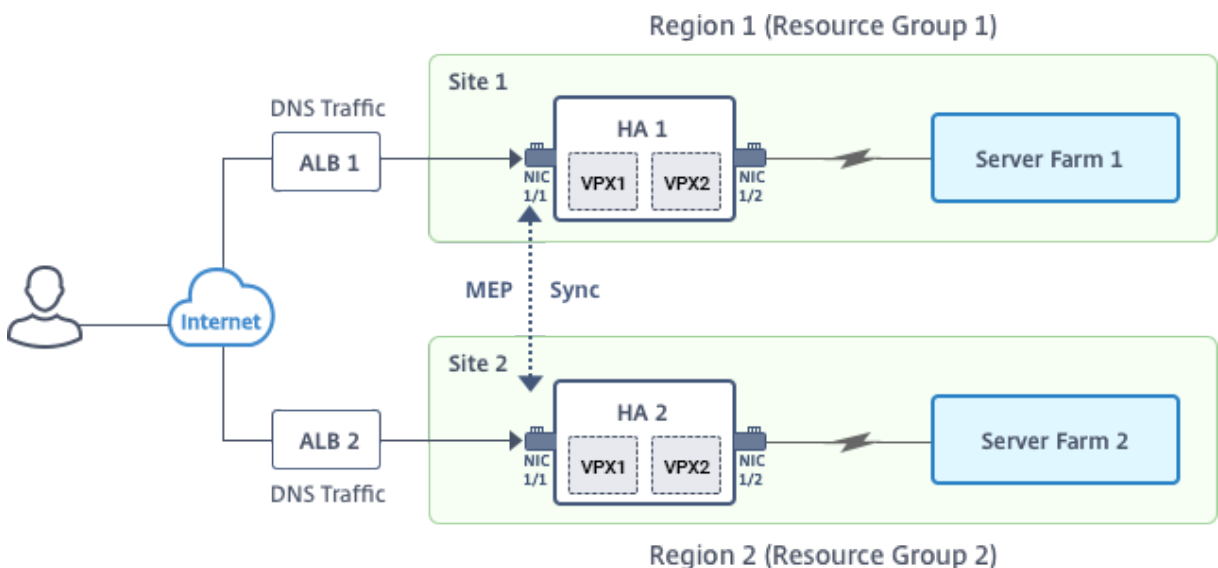
1 1. Create a front-end IPconfig for GSLB site.
2
3 1. Create back-end address pool with IP address of NIC 1/1 of nodes in
  HA.
4
5 1. Create load-balancing rules for following:
6
7     TCP/3011 - gslb communication
8     TCP/3010 - gslb communication
9     UDP/53 - DNS communication
10
11 1. Associate back-end address pool with the LB rules created in step c
    .
12
13 1. Update the network security group (NSG) of NIC 1/1 of nodes in both
    the HA pair to allow the traffic for TCP 3010, TCP 3011 and UDP 53
    ports.
    
```

1. Enable GSLB on each HA pair.

Scenario

This scenario includes two sites - Site 1 and Site 2. Each site has an HA pair (HA1 and HA2) configured with multiple NICs, multiple IP addresses, and GSLB.

Figure: GLSB on Active-Standy HA Deployment on Azure



In this scenario, each VM has three NICs - NIC 0/1, 1/1, and 1/2. The NICs are configured for the following purposes.

NIC 0/1: to serve management traffic

NIC 1/1: to serve client-side traffic

NIC 1/2: to communicate with back-end servers

Parameter Settings

Following are sample parameters settings for the ALB.

```
1 $locName="South east Asia"
2
3 $rgName="MultiIP-MultiNIC-RG"
4
5 $pubIPName4="PIPFORGSLB1"
6
7 $domName4="vpxgslbdns"
8
9 $lbName="MultiIPALB"
10
11 $frontEndConfigName2="FrontEndIP2"
12
13 $backendPoolName1="BackendPoolHttp"
14
15 $lbRuleName2="LBRuleGSLB1"
16
17 $lbRuleName3="LBRuleGSLB2"
18
19 $lbRuleName4="LBRuleDNS"
20
21 $healthProbeName="HealthProbe"
```

Configure ALB with the front-end IP address and rules to allow GSLB and DNS traffic

Step 1. Create a public IP for GSLB site IP

```
1 $pip4=New-AzureRmPublicIpAddress -Name $pubIPName4 -ResourceGroupName
   $rgName -DomainNameLabel $domName4 -Location $locName -
   AllocationMethod Dynamic
2
3
4 Get-AzureRmLoadBalancer -Name $lbName -ResourceGroupName $rgName |
   Add-AzureRmLoadBalancerFrontendIpConfig -Name $frontEndConfigName2
   -PublicIpAddress $pip4 | Set-AzureRmLoadBalancer
```

Step 2. Create LB rules and update the existing ALB.

```
1 $alb = get-AzureRmLoadBalancer -Name $lbName -ResourceGroupName $rgName
2
3
```

```

4 $frontendipconfig2=Get-AzureRmLoadBalancerFrontendIpConfig -
  LoadBalancer $alb -Name $frontEndConfigName2
5
6
7 $backendPool=Get-AzureRmLoadBalancerBackendAddressPoolConfig -
  LoadBalancer $alb -Name $backendPoolName1
8
9
10 $healthprobe=Get-AzureRmLoadBalancerProbeConfig -LoadBalancer $alb -
  Name $healthProbeName
11
12
13 \ $alb | Add-AzureRmLoadBalancerRuleConfig -Name \$lbRuleName2 -
  BackendAddressPool \$backendPool -FrontendIPConfiguration \
  $frontendipconfig2 -Protocol \"Tcp\" -FrontendPort 3011 -BackendPort
  3011 -Probe \$healthprobe -EnableFloatingIP | Set-
  AzureRmLoadBalancer
14
15
16 \ $alb | Add-AzureRmLoadBalancerRuleConfig -Name \$lbRuleName3 -
  BackendAddressPool \$backendPool -FrontendIPConfiguration \
  $frontendipconfig2 -Protocol \"Tcp\" -FrontendPort 3010 -BackendPort
  3010 -Probe \$healthprobe -EnableFloatingIP | Set-
  AzureRmLoadBalancer
17
18
19 \ $alb | Add-AzureRmLoadBalancerRuleConfig -Name \$lbRuleName4 -
  BackendAddressPool \$backendPool -FrontendIPConfiguration \
  $frontendipconfig2 -Protocol \"Udp\" -FrontendPort 53 -BackendPort
  53 -Probe \$healthprobe -EnableFloatingIP | Set-AzureRmLoadBalancer

```

Enable GSLB on each high availability pair

Now you've two front-end IP addresses for each ALB: ALB 1 and ALB 2. One IP address is for the LB virtual server and the other for the GSLB site IP.

HA 1 has the following front-end IP addresses:

- FrontEndIPofALB1 (for LB virtual server)
- PIPFORGSLB1 (GSLB IP)

HA 2 has the following front-end IP addresses:

- FrontEndIPofALB2 (for LB virtual server)
- PIPFORGSLB2 (GSLB IP)

The following commands are used for this scenario.

```

1 enable ns feature LB GSLB
2

```

```
3
4 add service dnssvc PIPFORGSLB1 ADNS 53
5
6
7 add gslb site site1 PIPFORGSLB1 -publicIP PIPFORGSLB1
8
9
10 add gslb site site2 PIPFORGSLB2 -publicIP PIPFORGSLB2
11
12
13 add gslb service site1_gslb_http_svc1 FrontEndIPofALB1 HTTP 80 -
    publicIP FrontEndIPofALB1 -publicPort 80 -siteName site1
14
15
16 add gslb service site2_gslb_http_svc1 FrontEndIPofALB2 HTTP 80 -
    publicIP FrontEndIPofALB2 -publicPort 80 -siteName site2
17
18
19 add gslb vserver gslb_http_vip1 HTTP
20
21
22 bind gslb vserver gslb_http_vip1 -serviceName site2_gslb_http_svc1
23
24
25 bind gslb vserver gslb_http_vip1 -serviceName site1_gslb_http_svc1
26
27
28 bind gslb vserver gslb_http_vip1 -domainName www.gslbindia.com -TTL 5
```

Related resources:

[Configure GSLB on Citrix ADC VPX instances](#)

[Global Server Load Balancing](#)

Configure address pools (IIP) for a Citrix Gateway appliance

November 21, 2024

In some situations, users who connect with the Citrix Gateway Plug-in need a unique IP address for a Citrix ADC Gateway appliance. When you enable address pools (also known as IP pooling) for a group, the Citrix Gateway appliance can assign a unique IP address alias to each user. You configure address pools by using intranet IP (IIP) addresses.

You can configure address pools on a Citrix Gateway appliance deployed on Azure by following this 2-step procedure:

- Registering the private IP addresses that will be used in the address pool, in Azure
- Configuring address pools in the Citrix Gateway appliance

Register a private IP address in the Azure portal

In Azure, you can deploy a Citrix ADC VPX instance with multiple IP addresses. You can add IP addresses to a VPX instance in two ways:

- While provisioning a VPX instance

For more information about how to add multiple IP addresses while provisioning a VPX instance, see [Configure multiple IP addresses for a Citrix ADC standalone instance](#). To add IP addresses by using PowerShell commands while provisioning a VPX instance, see [Configure multiple IP addresses for a Citrix ADC VPX instance in standalone mode by using PowerShell commands](#).

- After provisioning a VPX instance

After you've provisioned a VPX instance, follow these steps to register a private IP address in the Azure portal, which you configure as an address pool in the Citrix Gateway appliance.

1. From Azure Resource Manager (ARM), go to the already created Citrix ADC VPX instance > **Network interfaces**. Choose the network interface which is bound to a subnet to which the IIP that you want to register belongs.

```
1 ! [localized image] (/en-us/vpx/media/network-interface-iip.png)
```

2. Click **IP Configurations**, and then click **Add**.

```
1 ! [localized image] (/en-us/vpx/media/ip-configuration-iip.png)
```

3. Provide the required details as shown in the example below and click **OK**.

```
1 ! [localized image] (/en-us/vpx/media/details-iip.png)
```

Configure address pools in the Citrix Gateway appliance

For more information about how to configure address pools on the Citrix Gateway, see this [Configuring Address Pools](#).

Limitation:

You can not bind a range of IIP addresses to users. Every IIP address that is used in an address pool should be registered.

Configure multiple IP addresses for a Citrix ADC VPX standalone instance by using PowerShell commands

November 21, 2024

In an Azure environment, a Citrix ADC VPX virtual appliance can be deployed with multiple NICs. Each NIC can have multiple IP addresses. This section describes how to deploy a Citrix ADC VPX instance with a single NIC and multiple IP addresses, by using PowerShell commands. You can use the same script for multi-NIC and multi-IP deployment.

Note

In this document, IP-Config refers to a pair of IP addresses, public IP and private IP, that is associated with an individual NIC. For more information, see the [Azure terminology](#) section.

Use case

In this use case, a single NIC is connected to a virtual network (VNET). The NIC is associated with three IP configurations, as shown in the following table.

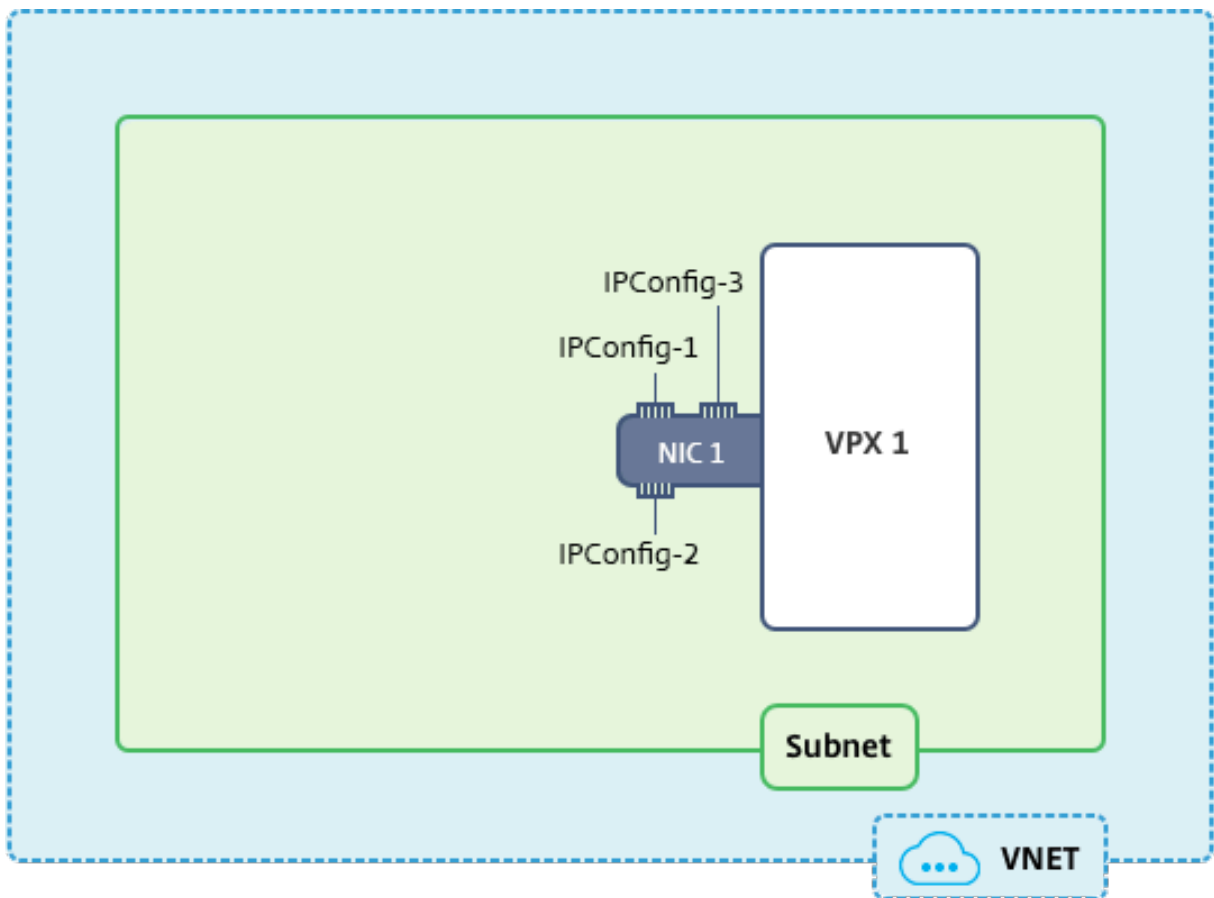
IPConfig	Associated with
IPConfig-1	Static public IP address; static private IP address
IPConfig-2	Static public IP address; static private address
IPConfig-3	Static private IP address

Note:

IPConfig-3 is not associated with any public IP address.

Diagram: Topology

Here is the visual representation of the use case.

**Note:**

In a multi-NIC, multi-IP Azure Citrix ADC VPX deployment, the private IP address associated with the primary (first) IPConfig of the primary (first) NIC is automatically added as the management NSIP address of the appliance. The remaining private IP addresses associated with IPConfigs must be added in the VPX instance as VIPs or SNIPs by using the “add ns ip” command, as determined by your requirements.

Here is the summary of the steps required for configuring multiple IP addresses for a Citrix ADC VPX virtual appliance in standalone mode:

1. Create Resource Group
2. Create Storage Account
3. Create Availability Set
4. Create NSG
5. Create Virtual Network
6. Create Public IP Address
7. Assign IP Configuration
8. Create NIC
9. Create Citrix ADC VPX Instance

10. Check NIC Configurations
11. Check VPX-side Configurations

Script

Parameters

Following are sample parameters settings for the use case in this document.

```
1 $locName="westcentralus"
2
3 $rgName="Azure-MultiIP"
4
5 $nicName1="VM1-NIC1"
6
7 $vNetName="Azure-MultiIP-vnet"
8
9 $vNetAddressRange="11.6.0.0/16"
10
11 $frontEndSubnetName="frontEndSubnet"
12
13 $frontEndSubnetRange="11.6.1.0/24"
14
15 $prmStorageAccountName="multiipstorage"
16
17 $avSetName="multiip-avSet"
18
19 $vmSize="Standard\_{DS4}\_V2" (This parameter creates a VM with upto four
    NICs.)
```

Note:

The minimum requirement for a VPX instance is 2 vCPUs and 2GB RAM.

```
1 $publisher="citrix"
2
3 $offer="netscalervpx110-6531" (You can use different offers.)
4
5 $sku="netscalerbyol" (According to your offer, the SKU can be different
    .)
6
7 $version="latest"
8
9 $pubIPName1="PIP1"
10
11 $pubIPName2="PIP2"
12
13 $domName1="multiipvpx1"
14
15 $domName2="multiipvpx2"
```



```

16
17 $vmNamePrefix="VPXMultiIP"
18
19 $osDiskSuffix="osmultiipalbdiskdb1"
20
21 **Network Security Group (NSG)-related information**:
22
23 $nsgName="NSG-MultiIP"
24
25 $rule1Name="Inbound-HTTP"
26
27 $rule2Name="Inbound-HTTPS"
28
29 $rule3Name="Inbound-SSH"
30
31 $IpConfigName1="IPConfig1"
32
33 $IPConfigName2="IPConfig-2"
34
35 $IPConfigName3="IPConfig-3"

```

1. Create Resource Group

```
New-AzureRmResourceGroup -Name $rgName -Location $locName
```

2. Create Storage Account

```
$prmStorageAccount = New-AzureRMStorageAccount -Name $prmStorageAccountName
                    -ResourceGroupName $rgName -Type Standard_LRS -Location $locName
```

3. Create Availability Set

```
$avSet = New-AzureRMAvailabilitySet -Name $avSetName -ResourceGroupName
                    $rgName -Location $locName
```

4. Create Network Security Group (NSG)

1. Add rules. You must add a rule to the NSG for any port that serves traffic.

```

1 $rule1=New-AzureRmNetworkSecurityRuleConfig -Name $rule1Name -
    Description "Allow HTTP" -Access Allow -Protocol Tcp -Direction
    Inbound -Priority 101 -SourceAddressPrefix Internet -
    SourcePortRange * -DestinationAddressPrefix * -DestinationPortRange
    80

```

```

2 $rule2=New-AzureRmNetworkSecurityRuleConfig -Name $rule2Name -
  Description "Allow HTTPS" -Access Allow -Protocol Tcp -Direction
  Inbound -Priority 110 -SourceAddressPrefix Internet -
  SourcePortRange * -DestinationAddressPrefix * -DestinationPortRange
  443
3 $rule3=New-AzureRmNetworkSecurityRuleConfig -Name $rule3Name -
  Description "Allow SSH" -Access Allow -Protocol Tcp -Direction
  Inbound -Priority 120 -SourceAddressPrefix Internet -SourcePortRange
  * -DestinationAddressPrefix * -DestinationPortRange 22

```

2. Create NSG object.

```

1 $nsg=New-AzureRmNetworkSecurityGroup -ResourceGroupName $rgName -
  Location $locName -Name $nsgName -SecurityRules $rule1,$rule2,$rule3

```

5. Create Virtual Network

1. Add subnets.

```

1 $frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
  $frontEndSubnetName -AddressPrefix $frontEndSubnetRange

```

2. Add virtual network object.

```

1 $vnet=New-AzureRmVirtualNetwork -Name $vNetName -ResourceGroupName
  $rgName -Location $locName -AddressPrefix $vNetAddressRange -Subnet
  $frontendSubnet

```

3. Retrieve subnets.

```

1 $subnetName="frontEndSubnet"
2 \ $subnet1=\ $vnet.Subnets|?{
3   \ $ \_.Name -eq \ $subnetName }

```

6. Create Public IP Address

```

1 $pip1=New-AzureRmPublicIpAddress -Name $pubIPName1 -ResourceGroupName
  $rgName -DomainNameLabel $domName1 -Location $locName -
  AllocationMethod Static
2 $pip2=New-AzureRmPublicIpAddress -Name $pubIPName2 -ResourceGroupName
  $rgName -DomainNameLabel $domName2 -Location $locName -
  AllocationMethod Static

```

Note:

Check availability of domain names before using.

Allocation method for IP addresses can be dynamic or static.

7. Assign IP Configuration

In this use case, consider the following points before assigning IP addresses:

- IPConfig-1 belongs to subnet1 of VPX1.
- IPConfig-2 belongs to subnet 1 of VPX1.
- IPConfig-3 belongs to subnet 1 of VPX1.

Note:

When you assign multiple IP configurations to a NIC, one configuration must be assigned as primary.

```

1 $IPAddress1="11.6.1.27"
2 $IPConfig1=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName1 -
    Subnet $subnet1 -PrivateIpAddress $IPAddress1 -PublicIpAddress $pip1
    - Primary
3 $IPAddress2="11.6.1.28"
4 $IPConfig2=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName2 -
    Subnet $subnet1 -PrivateIpAddress $IPAddress2 -PublicIpAddress $pip2
5 $IPAddress3="11.6.1.29"
6 $IPConfig3=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName3 -
    Subnet $subnet1 -PrivateIpAddress $IPAddress3 -Primary

```

Use a valid IP address that meets your subnet requirements and check its availability.

8. Create NIC

```

1 $nic1=New-AzureRmNetworkInterface -Name $nicName1 -ResourceGroupName
    $rgName -Location $locName -IpConfiguration $IpConfig1,$IpConfig2,
    $IPConfig3 -NetworkSecurityGroupId $nsg.Id

```

9. Create Citrix ADC VPX Instance

1. Initialize variables.

```

1 $suffixNumber = 1
2 $vmName = $vmNamePrefix + $suffixNumber

```

2. Create VM config object.

```

1 $vmConfig=New-AzureRMVMConfig -VMName $vmName -VMSize $vmSize -
    AvailabilitySetId $avSet.Id

```

3. Set credentials, OS, and image.

```

1 $cred=Get-Credential -Message "Type the name and password for VPX login
  ."
2 $vmConfig=Set-AzureRMVMOperatingSystem -VM $vmConfig -Linux -
  ComputerName $vmName -Credential $cred
3 $vmConfig=Set-AzureRMVMSourceImage -VM $vmConfig -PublisherName
  $publisher -Offer $offer -Skus $sku -Version $version

```

4. Add NIC.

```

1 $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic1.Id -
  Primary

```

Note:

In a multi-NIC VPX deployment, one NIC should be primary. So, “-Primary” needs to be appended while adding that NIC to the VPX instance.

5. Specify OS disk and create VM.

```

1 $osDiskName=$vmName + "-" + $osDiskSuffix1
2 $osVhdUri=$prmStorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds/"
  + $osDiskName + ".vhd"
3 $vmConfig=Set-AzureRMVMOsdisk -VM $vmConfig -Name $osDiskName -VhdUri
  $osVhdUri -CreateOption fromImage
4 Set-AzureRmVMPlan -VM $vmConfig -Publisher $publisher -Product $offer -
  Name $sku
5 New-AzureRMVM -VM $vmConfig -ResourceGroupName $rgName -Location
  $locName

```

10. Check NIC Configurations

After the VPX instance starts, you can check the IP addresses allocated to IPConfigs of the VPX NIC by using the following command.

```

1 $nic.IPConfig

```

11. Check VPX-side Configurations

When the Citrix ADC VPX instance starts, a private IP address associated with primary IPconfig of the primary NIC is added as the NSIP address. The remaining private IP addresses must be added as VIP or SNIP addresses, as determined by your requirements. Use the following command.

```

1 add nsip <Private IPAddress><netmask> -type VIP/SNIP

```

You’ve now configured multiple IP addresses for a Citrix ADC VPX instance in standalone mode.

Additional PowerShell scripts for Azure deployment

November 21, 2024

This section provides the PowerShell cmdlets with which you can perform the following configurations in Azure PowerShell:

- Provision a Citrix ADC VPX standalone instance
- Provision a Citrix ADC VPX pair in a high availability setup with an Azure external load balancer
- Provision a Citrix ADC VPX pair in a high availability setup with Azure internal load balancer

Also see the following topics for configurations that you can perform by using PowerShell commands:

- [Configure a high-availability setup with multiple IP addresses and NICs by using PowerShell commands](#)
- [Configure GSLB on Citrix ADC VPX instances](#)
- [Configure GSLB on a NetScaler active-standby high-availability setup](#)
- [Configure multiple IP addresses for a Citrix ADC VPX instance in standalone mode by using PowerShell commands](#)

Provision a Citrix ADC VPX standalone instance

1. Create a resource group

The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. The location specified here is the default location for resources in that resource group. Make sure all commands to create a load balancer use the same resource group.

```
1 $rgName="\<resource group name\>"
2
3 $locName="\<location name, such as West US\>"
4
5 New-AzureRmResourceGroup -Name $rgName -Location $locName
```

For example:

```
1 $rgName = "ARM-VPX"
2
3 $locName = "West US"
4
5 New-AzureRmResourceGroup -Name $rgName -Location $locName
```

2. Create a storage account

Choose a unique name for your storage account that contains only lowercase letters and numbers.

```

1  $saName="\<storage account name\>"
2
3  $saType="\<storage account type, specify one: Standard\_LRS,
4      Standard\_GRS, Standard\_RAGRS, or Premium\_LRS\>"
5  New-AzureRmStorageAccount -Name $saName -ResourceGroupName
    $rgName -Type $saType -Location $locName

```

For example:

```

1  $saName="vpxstorage"
2
3  $saType="Standard\_LRS"
4
5  New-AzureRmStorageAccount -Name $saName -ResourceGroupName
    $rgName -Type $saType -Location $locName

```

3. Create an availability set

Availability set helps to keep your virtual machines available during downtime, such as during maintenance. A load balancer configured with an availability set ensures that your application is always available.

```

1  $avName="\<availability set name\>"
2
3  New-AzureRmAvailabilitySet -Name $avName -ResourceGroupName
    $rgName -Location $locName

```

4. Create a virtual network

Add a new virtual network with at least one subnet, if the subnet was not created previously.

```

1  $FrontendAddressPrefix="10.0.1.0/24"
2
3  $BackendAddressPrefix="10.0.2.0/24"
4
5  $vnetAddressPrefix="10.0.0.0/16"
6
7  $frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
    frontendSubnet -AddressPrefix $FrontendAddressPrefix
8
9  $backendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
    backendSubnet -AddressPrefix $BackendAddressPrefix
10
11 New-AzureRmVirtualNetwork -Name TestNet -ResourceGroupName
    $rgName -Location $locName -AddressPrefix $vnetAddressPrefix
    -Subnet $frontendSubnet,$backendSubnet

```

For example:

```

1  $frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
    frontendSubnet -AddressPrefix $FrontendAddressPrefix
2
3  $backendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
    backendSubnet -AddressPrefix $BackendAddressPrefix
4
5  New-AzureRmVirtualNetwork -Name TestNet -ResourceGroupName
    $rgName -Location $locName -AddressPrefix $vnetAddressPrefix
    -Subnet $frontendSubnet,$backendSubnet

```

5. Create a NIC

Create a NIC and associate the NIC with the Citrix ADC VPX instance. The front end Subnet created in the above procedure is indexed at 0 and the back end Subnet is indexed at 1. Now create NIC in one of the three following ways:

- NIC with Public IP address

““

\$nicName="<name of the NIC of the VM>"

\$pip = New-AzureRmPublicIpAddress -Name \$nicName -ResourceGroupName \$rgName -Location \$locName -AllocationMethod Dynamic

\$nic = New-AzureRmNetworkInterface -Name \$nicName -ResourceGroupName \$rgName -Location \$locName -SubnetId \$vnet.Subnets[\$subnetIndex].Id -PublicIpAddressId \$pip.Id

- NIC with Public IP and DNS label

```

1  $nicName="\<name of the NIC of the VM\>"
2
3  $domName="\<domain name label\>"
4
5  $pip = New-AzureRmPublicIpAddress -Name $nicName -
    ResourceGroupName $rgName -DomainNameLabel $domName -
    Location $locName -AllocationMethod Dynamic

```

Before assigning \$domName, check it is available or not by using command:

```

1  Test-AzureRmDnsAvailability -DomainQualifiedName $domName -
    Location $locName
2
3  $nic = New-AzureRmNetworkInterface -Name $nicName -
    ResourceGroupName $rgName -Location $locName -SubnetId
    $vnet.Subnets\[ $subnetIndex\].Id -PublicIpAddressId $pip.
    Id

```

For example:

```

1  $nicName="frontendNIC"

```

```

2
3     $domName="vpxazure"
4
5     $pip = New-AzureRmPublicIpAddress -Name $nicName -
           ResourceGroupName $rgName -DomainNameLabel $domName -
           Location $locName -AllocationMethod Dynamic
6
7     $nic = New-AzureRmNetworkInterface -Name $nicName -
           ResourceGroupName $rgName -Location $locName -SubnetId
           $vnet.Subnets\[0\].Id -PublicIpAddressId $pip.Id

```

- NIC with Dynamic Public Address and Static Private IP address*

Make sure that the private (static) IP address you add to the VM should be the same range as that of the subnet specified.

```

1     $nicName="\<name of the NIC of the VM\>"
2
3     $staticIP="\<available static IP address on the subnet\>"
4
5     $pip = New-AzureRmPublicIpAddress -Name $nicName -
           ResourceGroupName $rgName -Location $locName -
           AllocationMethod Dynamic
6
7     $nic = New-AzureRmNetworkInterface -Name $nicName -
           ResourceGroupName $rgName -Location $locName -SubnetId
           $vnet.Subnets\[ $subnetIndex \].Id -PublicIpAddressId $pip.
           Id -PrivateIpAddress $staticIP

```

6. Create a virtual object

\$vmName="<VM name>"

\$vmSize="<VM size string>"

\$avSet=Get-AzureRmAvailabilitySet -Name \$avName -ResourceGroupName \$rgName

\$vm=New-AzureRmVMConfig -VMName \$vmName -VMSize \$vmSize -AvailabilitySetId \$avset.Id

1. Get the Citrix ADC VPX image

\$pubName="<Image publisher name>"

\$offerName="<Image offer name>"

\$skuName="<Image SKU name>"

\$cred=Get-Credential -Message "Type the name and password of the local administrator account."

Provide your credentials that is used to login into VPX

\$vm=Set-AzureRmVMOperatingSystem -VM \$vm -Linux -ComputerName \$vmName -Credential \$cred
-Verbose


```
$vm=Set-AzureRmVMSourceImage -VM $vm -PublisherName $pubName -Offer $offerName -Skus $skuName -Version "latest"
```

```
$vm=Add-AzureRmVMNetworkInterface -VM $vm -Id $nic.Id
```

For example:

```
1 $pubName="citrix"
```

The following command is used for displaying all offers from Citrix:

```
1 Get-AzureRMImageOffer -Location $locName -Publisher $pubName |
   Select Offer
2
3 $offerName="netscalervpx110-6531"
```

The following command is used to know sku offered by publisher for specific offer name:

```
Get-AzureRMImageSku -Location $locName      Select Skus
-Publisher $pubName -Offer $offerName
```

1. Create a virtual machine

\$diskName=""<name identifier for the disk in Azure storage, such as OSDisk>"

For example:

```
1 $diskName="dynamic"
2
3 $pubName="citrix"
4
5 $offerName="netscalervpx110-6531"
6
7 $skuName="netscalerbyol"
8
9 $storageAcc=Get-AzureRmStorageAccount -ResourceGroupName $rgName -Name
   $saName
10
11 $osDiskUri=$storageAcc.PrimaryEndpoints.Blob.ToString() + "vhds/" +
   $diskName + ".vhd"
12
13 $vm=Set-AzureRmVMOSDisk -VM $vm -Name $diskName -VhdUri $osDiskUri -
   CreateOption fromImage
```

When you create VM from Images present in marketplace, use the following command to specify the VM plan:

```
Set-AzureRmVMPlan -VM $vm -Publisher $pubName -Product $offerName -Name $skuName
```

```
New-AzureRmVM -ResourceGroupName $rgName -Location $locName -VM $vm
```

Provision a Citrix ADC VPX pair in a high availability setup with an Azure external load balancer

Log on to AzureRmAccount using your Azure user credentials.

1) Create a resource group

The location specified here is the default location for resources in that resource group. Make sure that all commands used to create a load balancer use the same resource group.

\$rgName="*<resource group name>*"

\$locName="*<location name, such as West US>*"

New-AzureRmResourceGroup -Name \$rgName -Location \$locName

For example:

```
1 $rgName = "ARM-LB-NS"
2
3 $locName = "West US"
4
5 New-AzureRmResourceGroup -Name $rgName -Location $locName
```

2) Create a storage account

Choose a unique name for your storage account that contains only lowercase letters and numbers.

\$saName="*<storage account name>*"

\$saType="*<storage account type, specify one: Standard_LRS, Standard_GRS, Standard_RAGRS, or Premium_LRS>*"

New-AzureRmStorageAccount -Name \$saName -ResourceGroupName \$rgName -Type \$saType -Location \$locName

For example:

```
1 $saName="vpxstorage"
2
3 $saType="Standard\_LRS"
4
5 New-AzureRmStorageAccount -Name $saName -ResourceGroupName $rgName -
  Type $saType -Location $locName
```

3) Create an availability set

A load balancer configured with an availability set ensures that your application is always available.

\$avName="*<availability set name>*"

New-AzureRmAvailabilitySet -Name \$avName -ResourceGroupName \$rgName -Location \$locName

4) Create a virtual network

Add a new virtual network with at least one subnet, if the subnet was not created previously.

```

1 $vnetName = "LBVnet"
2
3 $FrontendAddressPrefix="10.0.1.0/24"
4
5 $BackendAddressPrefix="10.0.2.0/24"
6
7 $vnetAddressPrefix="10.0.0.0/16"
8
9 $frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
   frontendSubnet -AddressPrefix $FrontendAddressPrefix
10
11 $backendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
   backendSubnet -AddressPrefix $BackendAddressPrefix
12
13 $vnet=New-AzureRmVirtualNetwork -Name $vnetName -ResourceGroupName
   $rgName -Location $locName -AddressPrefix $vnetAddressPrefix -Subnet
   $frontendSubnet,$backendSubnet

```

Note:

Choose the AddressPrefix parameter value as per your requirement.

Assign front end and back end subnet to the virtual network that you created earlier in this step.

If the front end subnet is the first element of array vnet, subnetId should be \$vnet.Subnets[0].Id.

If the front end subnet is the second element in the array, the subnetId should be \$vnet.Subnets[1].Id, and so on..

5) Configure front end IP address and create back end address pool

Configure a front end IP address for the incoming load balancer network traffic and create a back end address pool to receive the load balanced traffic.

```

1 $pubName="PublicIp1"
2
3 $publicIP1 = New-AzureRmPublicIpAddress -Name $pubName -
   ResourceGroupName $rgName -Location $locName -AllocationMethod
   Static -DomainNameLabel nsvpx

```

Note: Check for the availability of the value for DomainNameLabel.

```

1 $FIPName = "ELBFIP"
2
3 $frontendIP1 = New-AzureRmLoadBalancerFrontendIpConfig -Name $FIPName -
   PublicIpAddress $publicIP1
4
5 $BEPool = "LB-backend-Pool"
6

```

```
7 $beaddresspool1= New-AzureRmLoadBalancerBackendAddressPoolConfig -Name
   $BEPool
```

8) Create a health probe

Create a TCP health probe with port 9000 and interval 5 seconds.

```
1 $healthProbe = New-AzureRmLoadBalancerProbeConfig -Name HealthProbe -
   Protocol Tcp -Port 9000 -IntervalInSeconds 5 -ProbeCount 2
```

9) Create a load balancing rule

Create a LB rule for each service that you are load balancing.

For example:

You can use the following example to load balance http service.

```
1 $lbrule1 = New-AzureRmLoadBalancerRuleConfig -Name "HTTP-LB" -
   FrontendIpConfiguration $frontendIP1 -BackendAddressPool
   $beAddressPool1 -Probe $healthProbe -Protocol Tcp -FrontendPort 80 -
   BackendPort 80
```

10) Create inbound NAT rules

Create NAT rules for services that you are not load balancing.

For example, when creating a SSH access to a Citrix ADC VPX instance.

Note: Protocol-FrontEndPort-BackendPort triplet should not be the same for two NAT rules.

```
1 $inboundNATRule1= New-AzureRmLoadBalancerInboundNatRuleConfig -Name
   SSH1 -FrontendIpConfiguration $frontendIP1 -Protocol TCP -
   FrontendPort 22 -BackendPort 22
2
3 $inboundNATRule2= New-AzureRmLoadBalancerInboundNatRuleConfig -Name
   SSH2 -FrontendIpConfiguration $frontendIP1 -Protocol TCP -
   FrontendPort 10022 -BackendPort 22
```

11) Create a load balancer entity

Create the load balancer adding all objects (NAT rules, load balancer rules, probe configurations) together.

```
1 $lbName="ELB"
2
3 $NRPLB = New-AzureRmLoadBalancer -ResourceGroupName $rgName -Name
   $lbName -Location $locName -InboundNatRule $inboundNATRule1,
   $inboundNATRule2 -FrontendIpConfiguration $frontendIP1 -
   LoadBalancingRule $lbrule1 -BackendAddressPool $beAddressPool1 -
   Probe $healthProbe
```

12) Create a NIC

Create two NICs and associate each NIC with each VPX instance

a) NIC1 with VPX1

For example:

```

1 $nicName="NIC1"
2
3 $lbName="ELB"
4
5 $bePoolIndex=0
6
7 \* Rule indexes starts from 0.
8
9 $natRuleIndex=0
10
11 $subnetIndex=0
12
13 \* Frontend subnet index
14
15 $lb=Get-AzureRmLoadBalancer -Name $lbName -ResourceGroupName $rgName
16
17 $nic1=New-AzureRmNetworkInterface -Name $nicName -ResourceGroupName
    $rgName -Location $locName -Subnet $vnet.Subnets[$subnetIndex] -
    LoadBalancerBackendAddressPool $lb.BackendAddressPools[$bePoolIndex
    \] -LoadBalancerInboundNatRule $lb.InboundNatRules[$natRuleIndex\]
```

b) NIC2 with VPX2

For example:

```

1 $nicName="NIC2"
2
3 $lbName="ELB"
4
5 $bePoolIndex=0
6
7 $natRuleIndex=1
```

* Second Inbound NAT (SSH) rule we need to use

\$subnetIndex=0

* Frontend subnet index

```

1 $lb=Get-AzureRmLoadBalancer -Name $lbName -ResourceGroupName $rgName
2
3 $nic2=New-AzureRmNetworkInterface -Name $nicName -ResourceGroupName
    $rgName -Location $locName -Subnet $vnet.Subnets[$subnetIndex] -
    LoadBalancerBackendAddressPool $lb.BackendAddressPools[$bePoolIndex
    \] -LoadBalancerInboundNatRule $lb.InboundNatRules[$natRuleIndex\]
```

13) Create Citrix ADC VPX instances

Create two Citrix ADC VPX instances as part of the same resource group and availability set, and attach it to the external load balancer.

a) Citrix ADC VPX instance 1

For example:

```

1 $vmName="VPX1"
2
3 $vmSize="Standard\_A3"
4
5 $pubName="citrix"
6
7 $offerName="netscalervpx110-6531"
8
9 $skuName="netscalerbyol"
10
11 $avSet=Get-AzureRmAvailabilitySet -Name $avName -ResourceGroupName
    $rgName
12
13 $vm1=New-AzureRmVMConfig -VMName $vmName -VMSize $vmSize -
    AvailabilitySetId $avset.Id
14
15 $cred=Get-Credential -Message "Type Credentials which will be used to
    login to VPX instance"
16
17 $vm1=Set-AzureRmVMOperatingSystem -VM $vm1 -Linux -ComputerName $vmName
    -Credential $cred -Verbose
18
19 $vm1=Set-AzureRmVMSourceImage -VM $vm1 -PublisherName $pubName -Offer
    $offerName -Skus $skuName -Version "latest"
20
21 $vm1=Add-AzureRmVMNetworkInterface -VM $vm1 -Id $nic1.Id
22
23 $diskName="dynamic"
24
25 $storageAcc=Get-AzureRmStorageAccount -ResourceGroupName $rgName -Name
    $saName
26
27 $osDiskUri=$storageAcc.PrimaryEndpoints.Blob.ToString() + "vhds1/" +
    $diskName + ".vhd"
28
29 $vm1=Set-AzureRmVMOSDisk -VM $vm1 -Name $diskName -VhdUri $osDiskUri -
    CreateOption fromImage
30
31 Set-AzureRmVMPlan -VM $vm1 -Publisher $pubName -Product $offerName -
    Name $skuName
32
33 New-AzureRmVM -ResourceGroupName $rgName -Location $locName -VM $vm1

```

b) Citrix ADC VPX instance 2

For example:

```

1 $vmName="VPX2"
2
3 $vmSize="Standard\_A3"
4
5 $avSet=Get-AzureRmAvailabilitySet -Name $avName -ResourceGroupName
   $rgName
6
7 $vm2=New-AzureRmVMConfig -VMName $vmName -VMSize $vmSize -
   AvailabilitySetId $avset.Id
8
9 $cred=Get-Credential -Message " Type Credentials which will be used to
   login to VPX instance "
10
11 $vm2=Set-AzureRmVMOperatingSystem -VM $vm2 -Linux -ComputerName $vmName
   -Credential $cred -Verbose
12
13 $vm2=Set-AzureRmVMSourceImage -VM $vm2 -PublisherName $pubName -Offer
   $offerName -Skus $skuName -Version "latest"
14
15 $vm2=Add-AzureRmVMNetworkInterface -VM $vm2 -Id $nic2.Id
16
17 $diskName="dynamic"
18
19 $storageAcc=Get-AzureRmStorageAccount -ResourceGroupName $rgName -Name
   $saName
20
21 $osDiskUri=$storageAcc.PrimaryEndpoints.Blob.ToString() + "vhds2/" +
   $diskName + ".vhd"
22
23 $vm2=Set-AzureRmVMOSDisk -VM $vm2 -Name $diskName -VhdUri $osDiskUri -
   CreateOption fromImage
24
25 Set-AzureRmVMPlan -VM $vm2 -Publisher $pubName -Product $offerName -
   Name $skuName
26
27 New-AzureRmVM -ResourceGroupName $rgName -Location $locName -VM $vm2

```

14) Configure the virtual machines

When both the Citrix ADC VPX instances start, then connect to both Citrix ADC VPX instances using the SSH protocol to configure the virtual machines.

- a) Active-Active: Run the same set of configuration commands on the command line of both the Citrix ADC VPX instances.
- b) Active-Passive: Run this command on the command line of both the Citrix ADC VPX instances.

```
1 add ha node #nodeID <nsip of other Citrix ADC VPX>
```

In Active-Passive mode, run configuration commands on the primary node only.

Provision a Citrix ADC VPX pair in a high availability setup with Azure internal load balancer

Log on to AzureRmAccount using your Azure user credentials.

1) Create a resource group

The location specified here is the default location for resources in that resource group. Make sure all commands to create a load balancer use the same resource group.

\$rgName="**<resource group name>**"

\$locName="**<location name, such as West US>**"

New-AzureRmResourceGroup -Name \$rgName -Location \$locName

For example:

```
1 $rgName = "ARM-LB-NS"
2
3 $locName = "West US"
4
5 New-AzureRmResourceGroup -Name $rgName -Location $locName
```

2) Create a storage account

Choose a unique name for your storage account that contains only lowercase letters and numbers.

\$saName="**<storage account name>**"

\$saType="**<storage account type, specify one: Standard_LRS, Standard_GRS, Standard_RAGRS, or Premium_LRS>**"

New-AzureRmStorageAccount -Name \$saName -ResourceGroupName \$rgName -Type \$saType
-Location \$locName

For example:

```
1 $saName="vpxstorage"
2
3 $saType="Standard\_LRS"
4
5 New-AzureRmStorageAccount -Name $saName -ResourceGroupName $rgName -
  Type $saType -Location $locName
```

3) Create an availability set

A load balancer configured with an availability set ensures that your application is always available..

\$avName="**<availability set name>**"

New-AzureRmAvailabilitySet -Name \$avName -ResourceGroupName \$rgName -Location \$loc-
Name

4) Create a virtual network

Add a new virtual network with at least one subnet, if the subnet was not created previously.

```

1 $vnetName = "LBVnet"
2
3 $vnetAddressPrefix="10.0.0.0/16"
4
5 $FrontendAddressPrefix="10.0.1.0/24"
6
7 $BackendAddressPrefix="10.0.2.0/24"
8
9 $vnet=New-AzureRmVirtualNetwork -Name $vnetName -ResourceGroupName
    $rgName -Location $locName -AddressPrefix $vnetAddressPrefix -Subnet
    $frontendSubnet,$backendSubnet\`
10
11 $frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
    frontendSubnet -AddressPrefix $FrontendAddressPrefix
12
13 $backendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
    backendSubnet -AddressPrefix $BackendAddressPrefix

```

Note: Choose the AddressPrefix parameter value as per your requirement.

Assign front end and back end subnet to the virtual network that you created earlier in this step.

If the front end subnet is the first element of array vnet, subnetId should be \$vnet.Subnets[0].Id.

If the front end subnet is the second element in the array, the subnetId should be \$vnet.Subnets[1].Id, and so on..

5) Create an back end address pool

```

1 $beaddresspool= New-AzureRmLoadBalancerBackendAddressPoolConfig -Name "
    LB-backend"

```

6) Create NAT rules

Create NAT rules for services that you are not load balancing.

```

1 $inboundNATRule1= New-AzureRmLoadBalancerInboundNatRuleConfig -Name "
    Inboundnatrule1" -FrontendIpConfiguration $frontendIP -Protocol TCP
    -FrontendPort 3441 -BackendPort 3389
2
3 $inboundNATRule2= New-AzureRmLoadBalancerInboundNatRuleConfig -Name "
    RDP2" -FrontendIpConfiguration $frontendIP -Protocol TCP -
    FrontendPort 3442 -BackendPort 3389

```

Use front end and back end ports as per your requirement.

7) Create a health probe

Create a TCP health probe with port 9000 and interval 5 seconds.

```
1 $healthProbe = New-AzureRmLoadBalancerProbeConfig -Name "HealthProbe"
   " -Protocol tcp -Port 9000 -IntervalInSeconds 5 -ProbeCount 2
```

8) Create a load balancing rule

Create a LB rule for each service that you are load balancing.

For example:

You can use the following example to load balance http service.

```
1 $lbrule = New-AzureRmLoadBalancerRuleConfig -Name "lbrule1" -
   FrontendIpConfiguration $frontendIP -BackendAddressPool
   $beAddressPool -Probe $healthProbe -Protocol Tcp -FrontendPort 80 -
   BackendPort 80
```

Use front end and back end ports as per your requirement.

9) Create a load balancer entity

Create the load balancer adding all objects (NAT rules, load balancer rules, probe configurations) together.

```
1 $NRPLB = New-AzureRmLoadBalancer -ResourceGroupName $rgname -Name "
   InternalLB" -Location $locName -FrontendIpConfiguration $frontendIP
   -InboundNatRule $inboundNATRule1,$inboundNatRule2 -LoadBalancingRule
   $lbrule -BackendAddressPool $beAddressPool -Probe $healthProbe
```

10) Create a NIC

Create two NICs and associate each NIC with each Citrix ADC VPX instance

```
1 $backendnic1= New-AzureRmNetworkInterface -ResourceGroupName $rgName -
   Name lb-nic1-be -Location $locName -PrivateIpAddress 10.0.2.6 -
   Subnet $backendSubnet -LoadBalancerBackendAddressPool $nrplb.
   BackendAddressPools\[0\] -LoadBalancerInboundNatRule $nrplb.
   InboundNatRules\[0\]
```

This NIC is for Citrix ADC VPX 1. The Private IP should be in same subnet as that of subnet added.

```
1 $backendnic2= New-AzureRmNetworkInterface -ResourceGroupName $rgName -
   Name lb-nic2-be -Location $locName -PrivateIpAddress 10.0.2.7 -
   Subnet $backendSubnet -LoadBalancerBackendAddressPool $nrplb.
   BackendAddressPools\[0\] -LoadBalancerInboundNatRule $nrplb.
   InboundNatRules\[1\].
```

This NIC is for Citrix ADC VPX 2. The parameter Private IPAddress can have any private IP as per your requirement.

11) Create Citrix ADC VPX instances

Create two VPX instances part of same resource group and availability set and attach it to the internal load balancer.

a) Citrix ADC VPX instance 1

For example:

```
1 $vmName="VPX1"
2
3 $vmSize="Standard\_A3"
4
5 $avSet=Get-AzureRmAvailabilitySet -Name $avName -ResourceGroupName
  $rgName
6
7 $vm1=New-AzureRmVMConfig -VMName $vmName -VMSize $vmSize -
  AvailabilitySetId $avset.Id
8
9 $cred=Get-Credential -Message "Type Credentials which will be used to
  login to VPX instance"
10
11 $vm1=Set-AzureRmVMOperatingSystem -VM $vm1 -Linux -ComputerName $vmName
  -Credential $cred -Verbose
12
13 $vm1=Set-AzureRmVMSourceImage -VM $vm1 -PublisherName $pubName -Offer
  $offerName -Skus $skuName -Version "latest"
14
15 $vm1=Add-AzureRmVMNetworkInterface -VM $vm1 -Id $backendnic1.Id
16
17 $diskName="dynamic"
18
19 $storageAcc=Get-AzureRmStorageAccount -ResourceGroupName $rgName -Name
  $saName
20
21 $osDiskUri=$storageAcc.PrimaryEndpoints.Blob.ToString() + "vhds1/" +
  $diskName + ".vhd"
22
23 $vm1=Set-AzureRmVMOSDisk -VM $vm1 -Name $diskName -VhdUri $osDiskUri -
  CreateOption fromImage
24
25 Set-AzureRmVMPlan -VM $vm1 -Publisher $pubName -Product $offerName -
  Name $skuName
26
27 New-AzureRmVM -ResourceGroupName $rgName -Location $locName -VM $vm1
```

b) Citrix ADC VPX instance 2

For example:

```
1 $vmName="VPX2"
2
3 $vmSize="Standard\_A3"
4
5 $avSet=Get-AzureRmAvailabilitySet -Name $avName -ResourceGroupName
  $rgName
6
7 $vm2=New-AzureRmVMConfig -VMName $vmName -VMSize $vmSize -
  AvailabilitySetId $avset.Id
```

```
8
9 $cred=Get-Credential -Message " Type Credentials which will be used to
  login to VPX instance "
10
11 $vm2=Set-AzureRmVMOperatingSystem -VM $vm2 -Linux -ComputerName $vmName
  -Credential $cred -Verbose
12
13 $vm2=Set-AzureRmVMSourceImage -VM $vm2 -PublisherName $pubName -Offer
  $offerName -Skus $skuName -Version "latest"
14
15 $vm2=Add-AzureRmVMNetworkInterface -VM $vm2 -Id $backendnic2.Id
16
17 $diskName="dynamic"
18
19 $storageAcc=Get-AzureRmStorageAccount -ResourceGroupName $rgName -Name
  $saName
20
21 $osDiskUri=$storageAcc.PrimaryEndpoints.Blob.ToString() + "vhds/" +
  $diskName + ".vhd"
22
23 $vm2=Set-AzureRmVMOSDisk -VM $vm2 -Name $diskName -VhdUri $osDiskUri -
  CreateOption fromImage
24
25 Set-AzureRmVMPlan -VM $vm2 -Publisher $pubName -Product $offerName -
  Name $skuName
26
27 New-AzureRmVM -ResourceGroupName $rgName -Location $locName -VM $vm2
```

12) Configure the virtual machines

When both the Citrix ADC VPX instances start, then connect to both Citrix ADC VPX instances using the SSH protocol to configure the virtual machines.

a) Active-Active: Run the same set of configuration commands on the command line of both the Citrix ADC VPX instances.

b) Active-Passive: Run this command on the command line of both the Citrix ADC VPX instances.

add ha node #nodeID <nsip of other Citrix ADC VPX>

In Active-Passive mode, run configuration commands on the primary node only.

Azure FAQs

November 12, 2024

- **Is the upgrade procedure of Citrix ADC VPX instance installed from Azure Marketplace different from the on-premises upgrade procedure?**

No. You can upgrade your Citrix ADC VPX instance in the Microsoft Azure cloud to Citrix ADC VPX release 11.1 or later, using standard Citrix ADC VPX upgrade procedures. You can upgrade either using GUI or CLI procedures. For any new installations, use the Citrix ADC VPX image for Microsoft Azure cloud.

To download the Citrix ADC VPX upgrade builds, go to **Citrix Downloads** > [Citrix ADC Firmware](#).

- **How to correct MAC moves and interface mutes observed on Citrix ADC VPX instances hosted on Azure?**

In Azure Multi-NIC environment, by default, all data interfaces might show MAC moves and interface mutes. To avoid MAC moves and interface mutes on Azure environments, Citrix recommends you to create a VLAN per data interface (without tag) of the ADC VPX instance and bind the primary IP of the NIC in Azure.

For more information, see [CTX224626](#) article.

Deploy a Citrix ADC VPX instance on Google Cloud Platform

November 13, 2024

You can deploy a Citrix ADC VPX instance on Google Cloud Platform (GCP). A VPX instance in GCP enables you to take advantage of GCP cloud computing capabilities and use Citrix load balancing and traffic management features for your business needs. You can deploy VPX instances in GCP as stand-alone instances. Both single NIC and multi NIC configurations are supported.

Note:

VPX high availability deployment is not yet supported on GCP.

Supported features

A VPX instance running in GCP supports the following features:

- Load Balancing
- ICA Proxy
- Content Switching
- AAA
- Rewrite
- Responder
- RDP Proxy
- nFactor

- LDAP
- VPN (CVPN/Full)
- GSLB

Limitation

- IPv6 is not supported

Hardware requirements

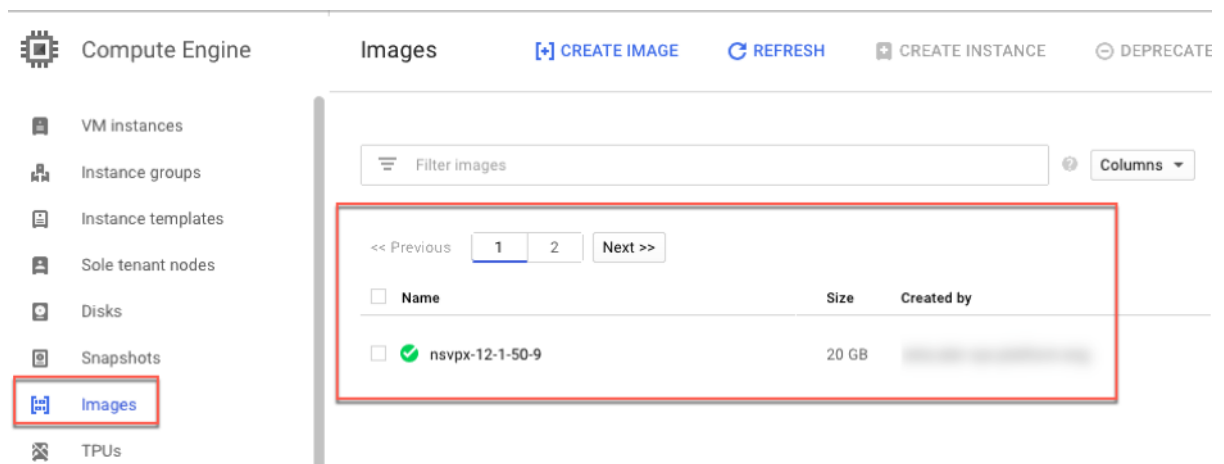
VPX instance in GCP must have minimum of 2 vCPUs and 4 GB RAM.

Prerequisites

1. Install the “gcloud” utility on your device. You can find the utility at this link: <https://cloud.google.com/sdk/install>
2. Download the NSVPX-GCP image from the Citrix download site.
3. Upload the file(for example, NSVPX-GCP-12.1-50.9_nc_64.tar.gz) to a storage bucket on Google by following the steps given at <https://cloud.google.com/storage/docs/uploading-objects>.
4. Run the following command on the gcloud utility to create an image.

```
1 gcloud compute images create <IMAGE_NAME> --source-uri=gs://<
  STORAGE_BUCKET_NAME>/<FILE_NAME>.tar.gz --guest-os-features=
  MULTI_IP_SUBNET
```

It might take a moment for the image to be created. After the image is created, it appears under **Compute > Compute Engine** in the GCP console.



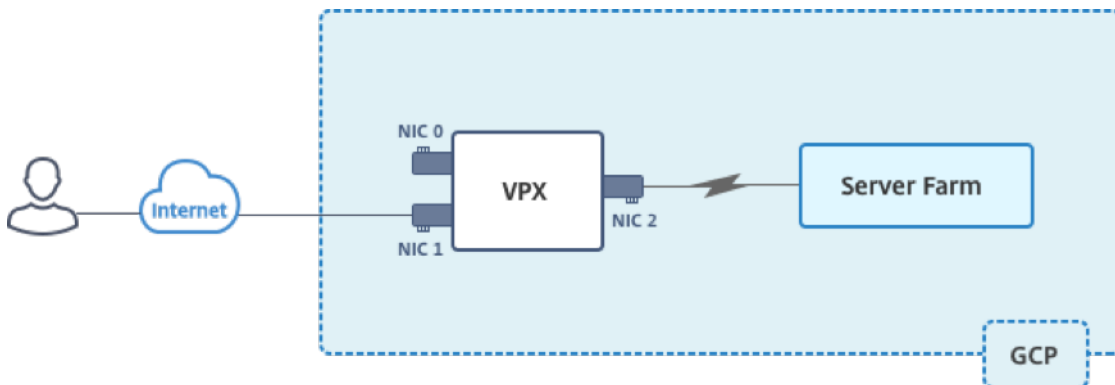
Points to note

Consider the following GCP-specific points before you begin your deployment.

- After creating the instance, you cannot add or remove any network interfaces.
- For a multi-NIC deployment, create separate VPC networks for each NIC. One NIC can be associated with only one network.
- For a single-NIC instance the GCP console creates a network by default.
- Minimum 4 VPCUs are required for an instance with more than two network interfaces.
- If IP forwarding is required, you must enable IP forwarding while creating the instance and configuring the NIC.

Scenario: deploy a multi-NIC, multi-IP standalone VPX instance

This scenario illustrates how to deploy a Citrix VPX standalone instance in GCP. In this scenario, you create a standalone VPX instance with multiple NICs. The instance communicates with back-end servers (the server farm).



Create three NICs to serve the following purposes.

NIC	Purpose	Associated with VPC network
NIC 0	Serves management traffic (Citrix ADC IP)	Management network
NIC 1	Serves client-side traffic (VIP)	Client network
NIC 2	Communicates with back-end servers (SNIP)	Back-end server network

Also, set up the required communication routes between the instance and the back-end servers, and between the instance and the external hosts on the public internet.

Summary of deployment steps

1. Create three VPC networks for three different NICs.
2. Create firewall rules for ports 22, 80, and 443
3. Create an instance with three NICs

Note:

Create instance in the same region where you've created the VPC networks.

Create VPC Networks

Create three VPC networks that will be associated with management NIC, client NIC, and server NIC. To create a VPC network, log on the Google console > **Networking** > **VPC network** > **Create VPC Network**. Complete the required fields, as shown in the screen capture, and click **Create**.

netscaler-vpx-platform-eng

← Create a VPC network

Name ?
vpxmgmt

Description (Optional)
management vpc

Subnets

Subnets let you create your own private cloud topology within Google Cloud. Click Automatic to create a subnet in each region, or click Custom to manually define the subnets. [Learn more](#)

Subnet creation mode
 Custom Automatic

New subnet

Name ?
vpxmgmtsubnet

[Add a description](#)

Region ?
asia-east1

IP address range ?
192.168.30.0/24

[Create secondary IP range](#)

Private Google access ?
 On
 Off

Flow logs
 On
 Off

Dynamic routing mode ?
 Regional
Cloud Routers will learn routes only in the region in which they were created

Global
Global routing lets you dynamically learn routes to and from all regions with a single VPN or interconnect and Cloud Router

Similarly, create VPC networks for client and server-side NICs.

Note:

All three VPC networks should be in the same region, which is asia-east1 in this scenario.

Create firewall rules for ports 22, 80, and 443

Create rules for SSH (port 22), HTTP (port 80), and HTTPS (port 443) for each VPC networks. For more information about firewall rules, see [Firewall Rules Overview](#).

netscaler-vpx-platform-eng

←

Create a firewall rule

Firewall rules control incoming or outgoing traffic to an instance. By default, incoming traffic from outside your network is blocked. [Learn more](#)

Name ?

Description (Optional)

Logs
Turning on firewall logs can generate a large number of logs which can increase costs in Stackdriver. [Learn more](#)

On
 Off

Network ?

Priority ?
Priority can be 0 - 65535 [Check priority of other firewall rules](#)

Direction of traffic ?

Ingress
 Egress

Action on match ?

Allow
 Deny

Targets ?

Source filter ?

Source IP ranges ?

Second source filter ?

Protocols and ports ?

Allow all
 Specified protocols and ports

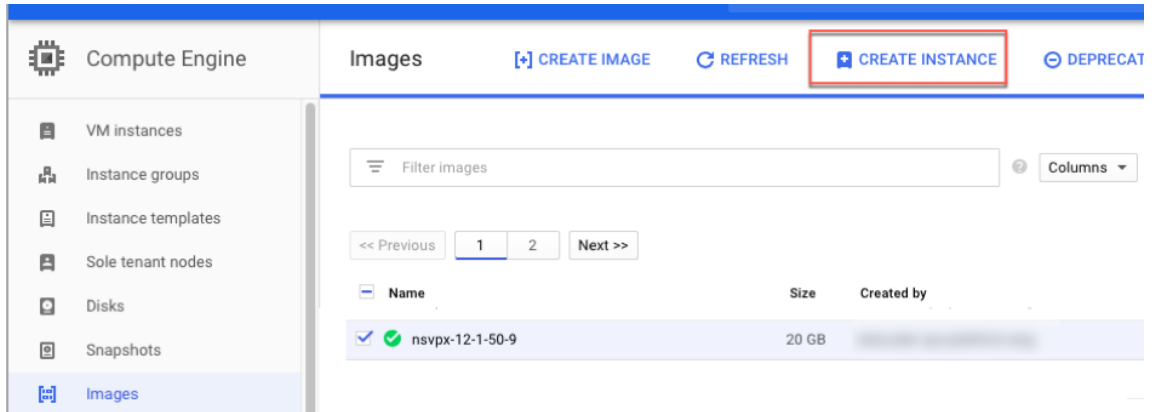
tcp :
 udp :
 Other protocols

⌵ [Disable rule](#)

Create
Cancel

Create the VPX instance

1. Log on to the GCP console.
2. Under **Compute**, hover over Compute Engine, and select **Images**.
3. Select the image, and click **Create Instance**.



4. Select an instance with 4 VPCUs, to support multiple NICs.
5. Click the networking option from Management, security, disks, networking, sole tenancy to add the additional NICs.



Note:

Container image is not supported on VPX instances on GCP.

6. Under **Networking interfaces**, click the edit icon to edit the default NIC. This NIC is the management NIC.
7. In the **Network interfaces** window, under **Network**, select the VPC network you created for management NIC.
8. For the management NIC, create a static external IP address. Under the External IP list, click **Create IP address**.
9. In the **Reserve a new static IP address** window, add a name and description and click **Reserve**.
10. Click **Add network interface** to create NICs for client and server-side traffic.

Network interfaces ?

default default (10.140.0.0/20) 

Network interface  

Network ?

vpxmgmt 

Subnetwork ?

vpxmgmtsubnet () 

Primary internal IP ?

Ephemeral (Automatic) 

 [Show alias IP ranges](#)

External IP ?

vpxpublic () 

Network Service Tier ?

Premium

 [Add network interface](#)

After you've created all the NICs, click **Create** to create the VPX instance.


i You have a draft that wasn't submitted, click Restore to keep working on it Restore

Name ?
vpctest1

Region ? **Zone** ?
asia-east1 (Taiwan) asia-east1-b

Machine type
Customize to select cores, memory and GPUs.
4 vCPUs 15 GB memory Customize

Container ?
 Deploy a container image to this VM instance. [Learn more](#)

Boot disk ?
 New 20 GB standard persistent disk
Image
nsvpx-12-1-50-9 Change

Identity and API access ?
Service account ?
Compute Engine default service account

Access scopes ?
 Allow default access
 Allow full access to all Cloud APIs
 Set access for each API




Firewall ?
Add tags and firewall rules to allow specific network traffic from the Internet
 Allow HTTP traffic
 Allow HTTPS traffic

! Firewalls setup is not available for multiple network interfaces

Management Security Disks Networking Sole Tenancy

Network tags ? (Optional)

Network interfaces ?

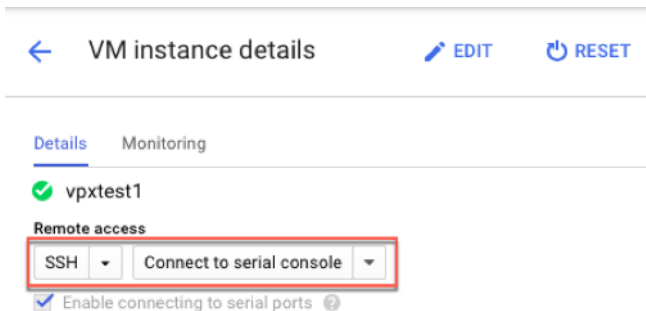
vpxmgmt vpxmgmtsubnet ()	
vpxclient vpxclientsubnet ()	
vpxbackend vpxbackendsubnet ()	

+ Add network interface

The instance appears under **VM instances**.



Use the GCP SSH or the serial console to configure and manage the VPX instance.



Points to note after you've deployed the VPX instance on GCP.

- Log on to the VPX with user name `nsroot` and instance ID as password.
- After first logon, change the default password.
- For collecting tech support bundle, run the command `shell /netscaler/showtech_cloud .pl` instead of the customary `show techsupport`.

GDM templates to deploy a Citrix ADC VPX instance

You can use a Citrix ADC VPX Google Deployment Manager (GDM) template to deploy a VPX instance on GCP. For details, see [Citrix ADC GDM Templates](#).

Resources

- [Creating Instances with Multiple Network Interfaces](#)
- [Creating and Starting a VM Instance](#)

Jumbo frames on Citrix ADC VPX instances

November 21, 2024

Citrix ADC VPX appliances support receiving and transmitting jumbo frames containing up to 9216 bytes of IP data. Jumbo frames can transfer large files more efficiently than it is possible with the standard IP MTU size of 1500 bytes.

A Citrix ADC appliance can use jumbo frames in the following deployment scenarios:

- Jumbo to Jumbo: The appliance receives data as jumbo frames and sends it as jumbo frames.
- Non-Jumbo to Jumbo: The appliance receives data as regular frames and sends it as jumbo frames.
- Jumbo to Non-Jumbo: The appliance receives data as jumbo frames and sends it as regular frames.

For more information, see [Configuring Jumbo Frames Support on a Citrix ADC Appliance](#).

Jumbo Frames support is available on Citrix ADC VPX appliances running on the following virtualization platforms:

- VMware ESX
- Linux-KVM Platform
- Citrix XenServer
- Amazon Web Services (AWS)

Jumbo frames on VPX appliances works similar to Jumbo frames on MPX appliances. For more information on Jumbo Frames and its use cases, see [Configuring Jumbo Frames on MPX appliances](#). The use cases of Jumbo Frames on MPX appliances also apply to VPX appliances.

Configure jumbo frames for a VPX instance running on VMware ESX

Perform the following tasks for configuring Jumbo Frames on a Citrix ADC VPX appliance running on VMware ESX server:

1. Set the MTU of the interface or channel of the VPX appliance to a value in the range 1501-9000. Use the CLI or GUI to set the MTU size. Note that Citrix ADC VPX appliances running on VMware ESX support receiving and transmitting jumbo frames containing up to only 9000 bytes of IP data.
2. Set the same MTU size on the corresponding physical interfaces of the VMware ESX server by using its management applications. For more information about setting the MTU size on the physical interfaces of VMware ESX, see <http://vmware.com/>.

Configure jumbo frames for a VPX instance running on Linux-KVM server

Perform the following tasks for configuring Jumbo Frames on a Citrix ADC VPX appliance running on a Linux-KVM Server:

1. Set the MTU of the interface or channel of the VPX appliance to a value in the range 1501-9216. Use the Citrix ADC VPX CLI or GUI to set the MTU size.

2. Set the same MTU size on the corresponding physical interfaces of a Linux-KVM Server by using its management applications. For more information about setting the MTU size on the physical interfaces of Linux-KVM, see <http://www.linux-kvm.org/>.

Configure jumbo frames for a VPX instance running on Citrix XenServer

Perform the following tasks to configure jumbo frames on a Citrix ADC VPX appliance running on Citrix XenServer:

1. Connect to the XenServer using XenCenter.
2. Shut down all the VPX instances that use the Networks for which the MTU must be changed.
3. On the **Networking** tab, select the network - network 0/1/2.
4. Select **Properties** and edit MTU.

After configuring jumbo frames on the XenServer, you can configure the jumbo frames on the ADC appliance. For more information, see [Configuring Jumbo Frames Support on a Citrix ADC Appliance](#).

Configure jumbo frames for a VPX instance running on AWS

Host-level configuration is not required for VPX on Azure. To configure Jumbo Frames on VPX, follow the steps given in [Configuring Jumbo Frames Support on a Citrix ADC Appliance](#).



© 2024 Cloud Software Group, Inc. All rights reserved. This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc. This and other products of Cloud Software Group may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at <https://www.cloud.com/legal>. Citrix, the Citrix logo, NetScaler, and the NetScaler logo and other marks appearing herein are either registered trademarks or trademarks of Cloud Software Group, Inc. and/or its subsidiaries in the United States and/or other countries. Other marks are the property of their respective owner(s) and are mentioned for identification purposes only. Please refer to Cloud SG's Trademark Guidelines and Third Party Trademark Notices (<https://www.cloud.com/legal>) for more information.