NetScaler VPX 14.1

Machine translated content

Disclaimer

Contents

NetScaler VPX サポートマトリックス **6**

VMware ESX、**Linux KVM**、および **Citrix Hypervisor** で **NetScaler ADC VPX** のパフォーマンスを最適化する **13**

NetScaler VPX 構成をクラウドで **NetScaler** アプライアンスの最初の起動時に適用する **29**

パブリッククラウドプラットフォームでの **SSL-TPS** パフォーマンスを向上させる **64**

パブリッククラウド上の **NetScaler VPX** 同時マルチスレッドを構成する **65**

NetScaler VPX インスタンスをベアメタルサーバーにインストールする **69**

Citrix Hypervisor に **NetScaler ADC VPX** インスタンスをインストールする **70**

シングルルート **I/O** 仮想化 **(SR-IOV)** ネットワークインターフェイスを使用するように **VPX** インスタンスを構成する **73**

VMware ESX に **Citrix ADC VPX** インスタンスをインストールする **78**

VMXNET3 ネットワークインターフェイスを使用するように **NetScaler VPX** インスタンスを構成する **83**

SR-IOV ネットワークインターフェイスの使用を **NetScaler ADC VPX** インスタンスで構成する **95**

SR-IOV モードでの **SSL** アクセラレーションにインテル **QAT** を使用するように **ESX** ハイパーバイザー上の **NetScaler VPX** を構成する **113**

E1000 から **SR-IOV** または **VMXNET3** ネットワークインターフェイスへの **NetScaler VPX** の移行 **117**

PCI パススルーネットワークインターフェイスを使用するように **NetScaler VPX** インスタンスを構成する **117**

VMware ESX ハイパーバイザーでの **NetScaler ADC** アプライアンスの初回起動時に **NetScaler ADC VPX** の構成を適用する **121**

AWS の **VMware** クラウドに **Citrix ADC VPX** インスタンスをインストールする **131**

Microsoft Hyper-V サーバーに **NetScaler VPX** インスタンスをインストールします **133**

Linux-KVM プラットフォームへの **Citrix ADC VPX** インスタンスのインストール **138**

Linux-KVM プラットフォームに **Citrix ADC VPX** インスタンスをインストールするための前提条件 **139**

OpenStack を使用して **Citrix ADC VPX** インスタンスをプロビジョニングする **144**

仮想マシンマネージャーを使用して **NetScaler VPX** インスタンスをプロビジョニングします **153**

SR-IOV ネットワークインターフェースを使用するように **NetScaler VPX** インスタンスを構成する **168**

SR-IOV モードでの **SSL** アクセラレーションに **Intel QAT** を使用するように **KVM** ハイパーバイザー上の **NetScaler VPX** を構成します **178**

PCI パススルーネットワークインターフェイスを使用するように **NetScaler VPX** インスタンスを構成する **182**

virsh プログラムを使用して **NetScaler ADC VPX** インスタンスをプロビジョニングする **187**

NetScaler VPX ゲスト仮想マシンの管理 **191**

OpenStack 上で **SR-IOV** を使用して **NetScaler VPX** インスタンスをプロビジョニングします **194**

KVM 上の **NetScaler VPX** インスタンスが **OVS DPDK** ベースのホストインターフェイスを使用するように構成する **200**

KVM ハイパーバイザーでの **NetScaler ADC** アプライアンスの初回起動時に **NetScaler ADC VPX** の構成を適用する **210**

AWS での **NetScaler VPX** **212**

AWS 用語 **215**

AWS-VPX サポートマトリックス **217**

制限事項と使用ガイドライン **220**

前提条件 **222**

NetScaler VPX インスタンスで **AWS IAM** ロールを設定します **225**

AWS 上の **NetScaler VPX** インスタンスの仕組み **235**

NetScaler VPX スタンドアロンインスタンスを **AWS** にデプロイする **237**

シナリオ**:** スタンドアロンインスタンス **242**

NetScaler VPX ライセンスをダウンロードする **251**

異なる可用性ゾーンでの負荷分散サーバー **257**

AWS での高可用性の機能 **257**

同じ **AWS** 可用性ゾーンに **VPX HA** ペアを展開する **260**

異なる **AWS** アベイラビリティーゾーンでの高可用 **271**

異なる **AWS** ゾーンに **Elastic IP** アドレスを使用した **VPX** 高可用性ペアをデプロイする **272**

異なる **AWS** ゾーンにプライベート **IP** アドレスを使用して **VPX** 高可用性ペアを展開する **277**

AWS Outpost で **NetScaler VPX** インスタンスを展開する **290**

NetScaler Web App Firewall を使用して **AWS API** ゲートウェイを保護 **294**

バックエンドの **AWS Autoscaling** サービスを追加する **297**

NetScaler GSLB を **AWS** に展開 **302**

SR-IOV ネットワークインターフェイスの使用を **NetScaler ADC VPX** インスタンスで構成する **317**

AWS ENA での拡張ネットワークの使用を **Citrix ADC VPX** インスタンスで構成する **320**

AWS 上の **NetScaler VPX** インスタンスのアップグレード **321**

AWS での **VPX** インスタンスのトラブルシューティング **326**

AWS に関するよくある質問 **326**

Microsoft Azure で **NetScaler VPX** インスタンスを展開する **329**

Azure 用語集 **335**

Microsoft Azure 上の **NetScaler ADC VPX** インスタンスのネットワークアーキテクチャ **339**

NetScaler VPX スタンドアロンインスタンスを構成する **342**

NetScaler VPX スタンドアロンインスタンスに複数の **IP** アドレスを構成する **356**

複数の **IP** アドレスと **NIC** を使用して高可用性設定を構成する **362**

PowerShell コマンドを使用して複数の **IP** アドレスと **NIC** を使用して高可用性セットアップを構成する **372**

フローティング **IP** 無効モードの **ALB** を使用して **Azure** に **NetScaler** 高可用性ペアをデプロイする **384**

Azure DNS プライベートゾーン用 **NetScaler** デプロイ **405**

Azure アクセラレーションネットワークを使用するように **NetScaler VPX** インスタンスを構成する **425**

Azure ILB で **NetScaler** 高可用性テンプレートを使用して **HA-INC** ノードを構成する **440**

インターネット向けアプリケーション用の **NetScaler** 高可用性テンプレートを使用して **HA-INC** ノードを構成する **453**

Azure 外部および内部ロードバランサーで同時に高可用性セットアップを構成する **464**

Azure VMware ソリューションに **NetScaler VPX** インスタンスをインストールする **468**

Azure VMware ソリューションでスタンドアロンの **NetScaler ADC VPX** インスタンスを構成する **485**

Azure VMware ソリューションで **Citrix ADC VPX** の高可用性セットアップを構成する **487**

NetScaler VPX HA ペアで **Azure** ルートサーバーを構成する **489**

バックエンドの **Azure** 自動スケーリングサービスを追加 **493**

NetScaler VPX 展開用の **Azure** タグ **500**

NetScaler VPX インスタンスで **GSLB** を構成する **506**

アクティブ/スタンバイの高可用性セットアップで **GSLB** を構成する **515**

Azure に **NetScaler GSLB** を展開 **518**

NetScaler Gateway アプライアンスのアドレスプールのイントラネット **IP** を構成する **534**

PowerShell コマンドを使用して、**NetScaler VPX** スタンドアロンインスタンスに複数の **IP** アドレスを構成する **536**

Azure 展開の追加の **PowerShell** スクリプト **543**

Create a support ticket for the VPX instance on Azure **559**

Azure に関するよくある質問 **561**

Google Cloud Platform への **NetScaler ADC VPX** インスタンスのデプロイ **562**

VPX の高可用性ペアを **Google Cloud Platform** に展開する **578**

Google Cloud Platform に外部の静的 **IP** アドレスを指定した **VPX** 高可用性ペアをデプロイする **579**

Google Cloud Platform にプライベート **IP** アドレスを指定した **1** つの **NIC VPX** 高可用性ペアをデプロイします **589**

プライベート **IP** アドレスを持つ **VPX** 高可用性ペアを **Google Cloud Platform** にデプロイする **598**

Google Cloud VMware Engine に **NetScaler VPX** インスタンスをインストールする **607**

バックエンドの **GCP Auto Scaling** サービスを追加する **626**

GCP 上の **NetScaler VPX** インスタンスの **VIP** スケーリングサポート **631**

GCP での **VPX** インスタンスのトラブルシューティング **638**

NetScaler VPX インスタンスのジャンボフレーム **639**

NetScaler の導入と構成を自動化する **641**

よくある質問 **644**

NetScaler VPX サポートマトリックス

October 17, 2024

このドキュメントでは、NetScaler VPX インスタンスでサポートされているさまざまなハイパーバイザーと機能について説明します。また、使用上のガイドラインと既知の制限事項についても説明します。

VMware ESX ハイパーバイザー上の **VPX** インスタンス

| ESXi バージョン | ESXi のリリース日 (年/月/日) | ESXi ビルド番号 | NetScaler VPX バージョン | パフォーマンス範囲 |
|————|——
- | ——- | ————–| ———–|
ESXi 8.0 アップデート 3	2024/06/25	24022510	14.1-17.x とそれ以降のビルド	10Mbps～100Gbps
ESXi 8.0 アップデート 2b		ハイパーバイザー →|XenServer 上の VPX||VMware ESX 上の VPX||||Microsoft Hyper-V 上の VPX| 汎用的 KVM 上の VPX| |^^Features ↓|^^||^^||||^^|^^||| |–|–|–|–|–|–|–|–|–|–|–|–| |** インターフェース →**|**PV**|**SR-IOV**|**PV**|**SR-IOV**|** エミュレート **|**PCI パススルー **|**PV**|**PV**|**SR-IOV**|**PCI パススルー **| |**Multi-PE のサポート **| はい	はい	はい	はい	はい	はい	はい	はい	はい	はい	|** クラスター化のサポート **| はい	はい [1]| はい	はい [1]| はい	はい	はい	はい	はい [1]| はい	|**VLAN タグ付け **| はい	はい	はい	はい	はい	はい	はい (2012R2 でのみ利用可能)| はい	はい	はい	|** リンクイベントの検出/Hamon**| いいえ [2]| はい [3]| いいえ [2]| はい [3]| いいえ [2]| はい [3]| いいえ [2]| いいえ [2]| はい [3]| はい [3]| |** インターフェイスパラメーター構成 **| いいえ	いいえ	いいえ	いいえ	いいえ	はい	いいえ	いいえ	いいえ	はい	|** 静的 LA**| はい [2]| はい [3]| はい [2]| いいえ	はい [2]| はい [3]| はい [2]| はい [2]| はい [3]| はい [3]| |**LACP**| いいえ	はい [3]| はい [2]| いいえ	はい [2]| はい [3]| いいえ	はい [2]| はい [3]| はい [3]| |** 静的 CLAG**| いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	|**LACP CLAG**| いいえ	いいえ	はい [2]| いいえ	はい [2]| はい [3]| いいえ	はい [2]| はい [3]| はい [3]| |** ホットプラグ **| いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ		23825572	14.1-17.x とそれ以降のビルド	^^
ESXi 8.0 アップデート 2	2024/02/29	23305546	14.1-4.x およびそれ以降のビルド	^^
ESXi 8.0 アップデート 2	2023/09/21	22380479	14.1-4.x およびそれ以降のビルド	^^
ESXi 8.0 アップデート 1	2023/04/18	21495797	14.1-4.x およびそれ以降のビルド	^^
ESXi 8.0c	2023/03/30	21493926	14.1-4.x およびそれ以降のビルド	^^
ESXi 8.0	2022/10/11	20513097	14.1-4.x およびそれ以降のビルド	^^
ESXi 7.0 アップデート 3q		ハイパーバイザー →	XenServer 上の VPX		VMware ESX 上の VPX				Microsoft Hyper-V 上の VPX	汎用的 KVM 上の VPX		^^Features ↓	^^		^^				^^	^^				–	–	–	–	–	–	–	–	–	–	–	–		** インターフェース →**	**PV**	**SR-IOV**	**PV**	**SR-IOV**	** エミュレート **	**PCI パススルー **	**PV**	**PV**	**SR-IOV**	**PCI パススルー **		**Multi-PE のサポート **	はい	はい	はい	はい	はい	はい	はい	はい	はい	はい		** クラスター化のサポート **	はい	はい [1]	はい	はい [1]	はい	はい	はい	はい	はい [1]	はい		**VLAN タグ付け **	はい	はい	はい	はい	はい	はい	はい (2012R2 でのみ利用可能)	はい	はい	はい		** リンクイベントの検出/Hamon**	いいえ [2]	はい [3]	いいえ [2]	はい [3]	いいえ [2]	はい [3]	いいえ [2]	いいえ [2]

はい [3]| はい [3]| |** インターフェイスパラメーター構成 **| いいえ | いいえ | いいえ | いいえ | いいえ | はい | いいえ | いいえ | いいえ | はい | |** 静的 LA**| はい [2]| はい [3]| はい [2]| いいえ | はい [2]| はい [3]| はい [2]| はい [2]| はい [3]| はい [3]| |**LACP**| いいえ | はい [3]| はい [2]| いいえ | はい [2]| はい [3]| いいえ | はい [2]| はい [3]| はい [3]| |** 静的 CLAG**| いいえ | いいえ | いいえ | いいえ | いいえ | いいえ | いいえ | いいえ | いいえ | いいえ | |**LACP CLAG**| いいえ | いいえ | はい [2]| いいえ | はい [2]| はい [3]| いいえ | はい [2]| はい [3]| はい [3]| |** ホットプラグ **| いいえ | いいえ | いいえ | いいえ | いいえ | いいえ | いいえ | いいえ | いいえ | いいえ | | 23794027 | 14.1-17.x とそれ以降のビルド | ^^ |
ESXi 7.0 update 3p	レール 7.6、レール 8.0、レール 9.3	23307199	14.1-4.x およびそれ以降のビルド	^^
ESXi 7.0 update 3o	2023/09/28	22348816	14.1-4.x およびそれ以降のビルド	^^
ESXi 7.0 update 3n	2023/07/06	21930508	14.1-8.x とそれ以降のビルド	^^
ESXi 7.0 update 3m	2023/05/03	21686933	14.1-4.x およびそれ以降のビルド	^^

> 注
>
> 各 ESXi パッチサポートは、前の表で指定されている NetScaler VPX バージョンで検証されており、NetScaler VPX 14.1 バージョンのすべての上位ビルドに適用されます。

使用ガイドラインの詳細については、「VMware ESXi ハイパーバイザーの使用ガイドライン」を参照してください。

XenServer または **Citrix Hypervisor** 上の **VPX** インスタンス

XenServer または Citrix Hypervisor のバージョン	SysID	パフォーマンス範囲
8.4、NetScaler VPX バージョン 14.1 ビルド 17.x 以降からサポート 8.2、NetScaler VPX バージョン 13.0 ビルド 64.x 以降でサポート 8.0, 7.6, 7.1	450000	10Mbps～40Gbps

Microsoft Hyper-V 上の **VPX** インスタンス

Hyper-V 版	SysID	パフォーマンス範囲
2016, 2019	450020	10Mbps～3Gbps

Nutanix AHV の **VPX** インスタンス

NetScaler VPX は、Citrix Ready パートナーシップを通じて Nutanix AHV でサポートされています。Citrix Ready

は、ソフトウェアおよびハードウェアのベンダーが自社製品を開発し、デジタルワークスペース、ネットワーキング、および分析用の NetScaler テクノロジーと統合するのを支援するテクノロジーパートナープログラムです。

NetScaler VPX インスタンスを Nutanix AHV にデプロイする段階的な方法の詳細については、「Nutanix AHV への NetScaler VPX デプロイ」を参照してください。

サードパーティサポート**:**

NetScaler 環境での特定のサードパーティ（Nutanix AHV）の統合で問題が発生した場合は、サードパーティパートナー（Nutanix）に直接サポートインシデントをオープンしてください。

パートナーが問題が NetScaler にあると判断した場合、パートナーは NetScaler サポートに連絡してさらにサポートを受けることができます。問題が解決するまで、パートナーの専任技術者が NetScaler サポートチームと協力します。

汎用 **KVM** 上の **VPX** インスタンス

汎用 KVM バージョン	SysID	パフォーマンス範囲
RHEL 7.6、RHEL 8.0、RHEL 9.3 Ubuntu 16.04、Ubuntu 18.04、Ubuntu 22.04	450070	10Mbps～100Gbps

注意事項:

KVM ハイパーバイザーを使用するときは、次の点を考慮してください。

- VPX インスタンスは、表 1-4 に記載されている Hypervisor リリースバージョンに対して認定されており、バージョン内のパッチリリースには適していません。ただし、VPX インスタンスは、サポートされているバージョンのパッチリリースとシームレスに動作することが期待されます。そうでない場合は、トラブルシューティングとデバッグのためのサポートケースを記録します。
- RHEL 7.6 を使用する前に、KVM ホストで以下のステップを完了します。
  1. /etc/default/grub を編集して`"kvm_intel.preemption_timer=0"`を`GRUB_CMDLINE_LINUX`変数に追加します。
  2. コマンド`"# grub2-mkconfig -o /boot/grub2/grub.cfg"`で grub.cfg を再生成します。
  3. ホストマシンを再起動します。
- Ubuntu 18.04 を使用する前に、KVM ホストで以下のステップを完了してください。
  1. /etc/default/grub を編集して`"kvm_intel.preemption_timer=0"`を`GRUB_CMDLINE_LINUX`変数に追加します。

2. コマンド"# grub-mkconfig -o /boot/grub/grub.cfg "で grub.cfg を再生成します。
3. ホストマシンを再起動します。

パブリッククラウド上の **VPX** インスタンス

パブリッククラウド	SysID	パフォーマンス範囲
AWS	450040	10Mbps～30Gbps
Azure	450020	10 Mbps ～10 Gbps
GCP	450070	10 Mbps ～10 Gbps

ハイパーバイザーでサポートされる **VPX** 機能

ハイパーバイザー →	XenServer 上の VPX				VMware ESX 上の VPX					
^^ 特徴 ↓	^^		^^				^^	^^		
インターフェイス →	**PV**	**SR-IOV**	**PV**	**SR-IOV**	エミュレート	**PCI** パススルー	**PV**	**PV**	**SR-IOV**	**PCI** パススルー
マルチ **PE** サポート	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
クラスタリングのサポート	Yes	はい [1]	Yes	はい [1]	Yes	Yes	Yes	Yes	はい [1]	Yes
VLAN タグ付け	Yes	Yes	Yes	Yes	Yes	Yes	はい (2012R2 のみ)	Yes	Yes	Yes

^^ 特徴 ↓	^^		^^				^^	^^		
リンクイベントの検出/**HAMon**	いいえ [2]	はい [3]	いいえ [2]	はい [3]	いいえ [2]	はい [3]	いいえ [2]	いいえ [2]	はい [3]	はい [3]
インターフェースパラメータの設定	いいえ	いいえ	いいえ	いいえ	いいえ	Yes	いいえ	いいえ	いいえ	Yes
静的 **LA**	はい [2]	はい [3]	はい [2]	いいえ	はい [2]	はい [3]	はい [2]	はい [2]	はい [3]	はい [3]
LACP	いいえ	はい [3]	はい [2]	いいえ	はい [2]	はい [3]	いいえ	はい [2]	はい [3]	はい [3]
静的 **CLAG**	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ
LACP クラグ	いいえ	いいえ	はい [2]	いいえ	はい [2]	はい [3]	いいえ	はい [2]	はい [3]	はい [3]
ホットプラグ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ

パブリッククラウドでサポートされる **VPX** 機能

パブリック クラウド →	AWS 上の VPX	Azure 上の VPX	GCP 上の VPX
^^ 特徴 ↓	^^	^^	^^
マルチ **PE** サポート	Yes	Yes	Yes
クラスタリングのサポート	いいえ	いいえ	いいえ
VLAN タグ付け	いいえ	いいえ	いいえ
リンクイベントの検出/**HAMon**	いいえ [2]	いいえ [2]	いいえ [2]

^^ 特徴 ↓	^^	^^	^^
インターフェースパラメータの設定	いいえ	いいえ	いいえ
静的 **LA**	いいえ	いいえ	いいえ
LACP	いいえ	いいえ	いいえ
静的 **CLAG**	いいえ	いいえ	いいえ
LACP クラグ	いいえ	いいえ	いいえ
ホットプラグ	Yes	いいえ	いいえ

前述の 2 つの表で使われている上付き文字 (1、2、3) は、以下の点とそれぞれの番号を示しています:

1. SRIOV では、バックプレーンではなく、クライアント側およびサーバ側インターフェイス用のクラスタリングサポートを利用できます。
2. インターフェイスダウンイベントは NetScaler VPX インスタンスには記録されません。
3. スタティック LA の場合、物理ステータスが DOWN のインターフェイスでトラフィックが送信される場合もあります。

次の点は、前述の 2 つの表で取り込まれたそれぞれの機能に適用されます:

- LACP の場合、ピアデバイスは LACP タイムアウトメカニズムに基づいてインターフェイス DOWN イベントを認識します。
  - 短いタイムアウト: 3 秒
  - 長いタイムアウト: 90 秒
- LACP では、VM 間でインターフェイスを共有しないでください。
- 動的ルーティングの場合、リンクイベントは検出されないため、コンバージェンス時間はルーティングプロトコルによって異なります。
- モニタ対象スタティックルート機能は、ルータの状態が VLAN ステータスに依存するため、モニタをスタティックルートにバインドしないと失敗します。VLAN ステータスは、リンクステータスによって異なります。
- リンク障害がある場合、高可用性では部分的な障害検出は行われません。リンク障害があると、高可用性の分割脳の状態が発生する可能性があります。
  - VPX インスタンスからリンクイベント（無効/有効化、リセット）が生成された場合、リンクの物理ステータスは変わりません。静的 LA の場合、ピアによって開始されたトラフィックはすべてインスタンスでドロップされます。
  - VLAN タグ付け機能を動作させるには、VMware ESX で、VMware ESX サーバーの vSwitch 上のポート グループの VLAN ID を 1 ～4095 に設定します。

- ホットプラグは、ENA インターフェイスを備えた VPX インスタンスではサポートされていないため、ホットプラグを試みるとインスタンスの動作が予測できない場合があります。ホットアドは、AWS 上の NetScaler を使用する PV および SRIOV インターフェイスでのみサポートされます。
- AWS ウェブコンソールまたは AWS CLI インターフェイスを介したホット削除は、NetScaler の PV、SRIOV、および ENA インターフェイスではサポートされていません。ホット削除を試みると、インスタンスの動作が予測できなくなる可能性があります。

サポートされているブラウザー

オペレーティングシステム	ブラウザとバージョン
Windows 7	Internet Explorer-8, 9, 10, 11; Mozilla Firefox 3.6.25 以降; Google Chrome-15 以降
Windows 64 ビット	Internet Explorer-8、9; Google Chrome-15 以降
MAC	Mozilla Firefox-12 以降; Safari-5.1.3; Google Chrome-15 以降

VPX インスタンスの AMD プロセッササポート

NetScaler リリース 13.1 以降、VPX インスタンスは Intel プロセッサーと AMD プロセッサーの両方をサポートしています。VPX 仮想アプライアンスは、2 つ以上の仮想コアと 2 GB を超えるメモリを備えた任意のインスタンスタイプにデプロイできます。システム要件の詳細については、NetScaler VPX のデータシートを参照してください。

VPX プラットフォーム vs. NIC マトリックス テーブル

次の表は、VPX プラットフォームまたはクラウドでサポートされている NIC の一覧です。

NIC →	Mellanox CX-3	Mellanox CX-4	Mellanox CX-5	Intel 82599 SRIOV VF	Intel X710/X722/XL710 SRIOV VF	Intel X710/XL710/XXV710 PCI パススルー モード
^^ プラットフォーム ↓	^^	^^	^^	^^	^^	^^
VPX (ESXi)	いいえ	Yes	いいえ	Yes	いいえ	Yes

^^ プラットフォーム↓	^^	^^	^^	^^	^^	^^
VPX (Citrix Hypervisor)	-	-	-	Yes	Yes	いいえ
VPX (KVM)	いいえ	Yes	Yes	Yes	Yes	いいえ
VPX (Hyper-V)	-	-	-	いいえ	いいえ	いいえ
VPX (AWS)	-	-	-	Yes	-	-
VPX (Azure)	Yes	Yes	Yes	-	-	-
VPX (GCP)	-	-	-	-	-	-

その他の参考文献

- Citrix Ready 製品については、Citrix Ready Marketplaceにアクセスしてください
- Citrix Ready 製品サポートについては、よくある質問ページを参照してください。
- VMware ESX ハードウェアバージョンについては、VMware Tools のアップグレードを参照してください。

VMware ESX、Linux KVM、および Citrix Hypervisor で NetScaler ADC VPX のパフォーマンスを最適化する

October 17, 2024

NetScaler VPX のパフォーマンスは、ハイパーバイザー、割り当てられたシステムリソース、およびホスト構成によって大きく異なります。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.

VMware ESX ハイパーバイザー上の NetScaler ADC VPX インスタンス

このセクションでは、構成可能なオプションと設定、および VMware ESX ハイパーバイザー上の NetScaler ADC VPX インスタンスの最適なパフォーマンスを実現するのに役立つその他の推奨事項について説明します。

- Recommended configuration on ESX hosts

- E1000 ネットワークインターフェイスを備えた NetScaler ADC VPX
- VMXNET3 ネットワークインターフェイスを備えた NetScaler ADC VPX
- SR-IOV および PCI パススルーネットワークインターフェイスを備えた NetScaler ADC VPX

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:

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

  - To set NUMA and vCPU affinity for a VM, see VMware documentation.

E1000 ネットワークインターフェイスを備えた NetScaler ADC VPX

Perform the following settings on the VMware ESX host:

- On the VMware ESX host, create two vNICs from one pNIC vSwitch. 複数の vNIC は、ESX ホストに複数の受信 (Rx) スレッドを作成します。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:

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

  - For ESX version 6.0 onwards:

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

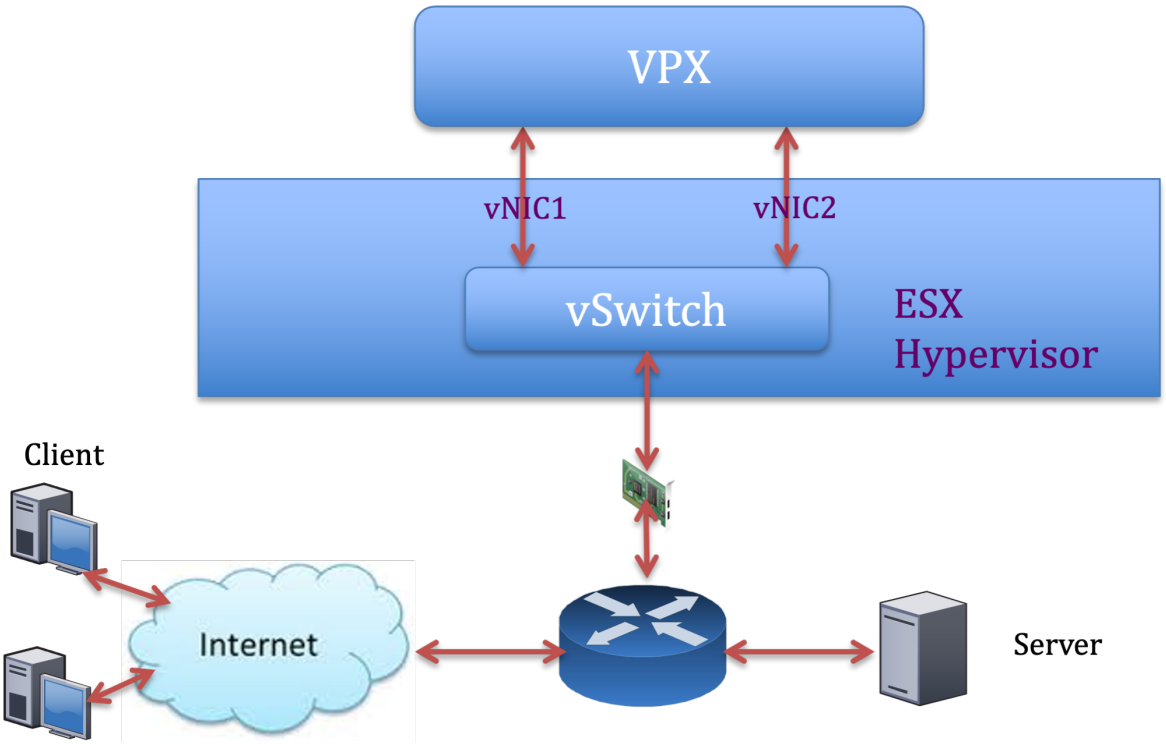
- To further increase the vNIC Tx throughput, use a separate Tx completion thread and Rx threads per device (NIC) queue. Use the following ESX command:


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

> 注
>
> 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.



NetScaler VPX 構成例:

前のサンプルトポロジに示した展開を実現するには、NetScaler VPX インスタンスで次の構成を実行します。

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

```
bind vlan 2 -ifnum 1/1  - tagged
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.

```
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
```

> 注
>
> 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

VMXNET3 ネットワークインターフェイスを備えた **NetScaler ADC VPX**

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. 複数の vNIC により、ESX ホストに複数の Rx スレッドが作成されます。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. 次のコマンドを使用します:


```
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:

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

- 仮想マシンの構成に次の設定を追加して、vNIC あたり最大 8 つの送信スレッドを使用するように仮想マシンを構成します。

```
ethernetX.ctxPerDev = "3"
```

> 注
>
> vNIC あたりの送信スレッド数を増やすと、ESX ホストでより多くの CPU リソース (最大 8 つ) が必要になります。前述の設定を行う前に、十分な CPU リソースが使用可能であることを確認してください。

> 注
>
> 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. 詳細については、「物理 NIC 展開ごとに 2 つの vNIC」を参照してください。

VMware ESX で **VMXNET3** デバイス用のマルチキューと **RSS** サポートを設定します　デフォルトでは、VMXNET3 デバイスは 8 つの Rx キューと Tx キューのみをサポートします。VPX の vCPU の数が 8 を超えると、VMXNET3 インターフェイスに設定されている Rx キューと Tx キューの数は、デフォルトで 1 に切り替わります。ESX の特定の構成を変更することで、VMXNET3 デバイス用に最大 19 個の Rx キューと Tx キューを設定できます。このオプションにより、パフォーマンスが向上し、VPX インスタンスの vCPU 間でパケットが均一に分散されます。

> 注
>
> NetScaler リリース 13.1 ビルド 48.x 以降、NetScaler VPX は VMXNET3 デバイスの ESX 上で最大 19 個の Rx キューと Tx キューをサポートします。

前提条件:

ESX で VMXNET3 デバイス用に最大 19 個の Rx キューと Tx キューを構成するには、次の前提条件が満たされていることを確認してください。

- NetScaler VPX バージョンは 13.1 ビルド 48.X 以降です。
- NetScaler VPX は、VMware ESX 7.0 以降でサポートされているハードウェアバージョン 17 以降の仮想マシンで構成されます。

8 つ以上の **Rx** キューと **Tx** キューをサポートするように **VMXNET3** インターフェイスを設定します。

1. 仮想マシンの構成ファイル (.vmx) ファイルを開きます。
2. `ethernetX.maxTxQueues`および`ethernetX.maxRxQueues`の値を設定して Rx キューと TX キューの数を指定します（X は設定する仮想 NIC の数）。設定するキューの最大数は、仮想マシンの vCPU 数を超えてはいけません。

   > 注
   >
   > キューの数を増やすと、ESX ホストのプロセッサオーバーヘッドも増加します。したがって、キューを増やす前に、ESX ホストに十分な CPU リソースがあることを確認してください。キューの数がパフォーマンスのボトルネックになっている場合は、サポートされるキューの最大数を増やすことができます。このような場合は、キューの数を徐々に増やすことをお勧めします。たとえば、8 から 12、次に 16 へ、そして 20 へ、というようになります。最大値まで直接上げるのではなく、各設定でパフォーマンスを評価してください。

SR-IOV および **PCI** パススルーネットワークインターフェイスを備えた **NetScaler ADC VPX**

SR-IOV および PCI パススルー ネットワーク インターフェイスを使用して NetScaler VPX の高パフォーマンスを実現するには、「ESX ホストでの推奨構成」を参照してください。

VMware ESXi ハイパーバイザーの使用ガイドライン

- NetScaler VPX インスタンスをサーバーのローカル ディスクまたは SAN ベースのストレージ ボリュームに展開することをお勧めします。

  『**VMware vSphere 6.5** のパフォーマンスのベストプラクティス』ドキュメントの「**VMware ESXi CPU** に関する考慮事項」セクションを参照してください。ここに抽出があります:

- CPU またはメモリのデマンドが高い仮想マシンを、オーバーコミットされたホストまたはクラスタにデプロイすることは推奨されません。
- ほとんどの環境では、ESXi は、仮想マシンのパフォーマンスに影響を与えることなく、かなりのレベルの CPU オーバーコミットメントを許可します。ホストでは、そのホスト内の物理プロセッサコアの総数よりも多くの vCPU を実行できます。
- ESXi ホストが CPU 飽和状態になった場合、つまり、仮想マシンおよびホスト上のその他の負荷がホストにあるすべての CPU リソースを要求すると、レイテンシの影響を受けやすいワークロードがうまく動作しない可能性があります。この場合、たとえば、一部の仮想マシンをパワーオフするか、別のホストに移行する（または DRS に自動的に移行させる）ことで、CPU 負荷を軽減します。
- NetScaler では、仮想マシンで ESXi ハイパーバイザーの最新の機能セットを利用するには、最新のハードウェア互換性バージョンを使用することをお勧めします。ハードウェアと ESXi のバージョンの互換性に関する詳細については、VMwareのドキュメントを参照してください。

- NetScaler VPX は、レイテンシーに敏感で高性能な仮想アプライアンスです。期待どおりのパフォーマンスを実現するには、アプライアンスに vCPU の予約、メモリの予約、および vCPU のホストへのピンニングが必要です。また、ホスト上でハイパースレッディングを無効にする必要があります。ホストがこれらの要件を満たしていない場合、次の問題が発生する可能性があります:
  - 高可用性フェイルオーバー
  - VPX インスタンス内の CPU スパイク
  - VPX CLI へのアクセスが遅い
  - ピットボスデーモンクラッシュ
  - パケットドロップ
  - 低スループット
- Hypervisor は、次の 2 つの条件のいずれかが満たされると、過剰プロビジョニングと見なされます:
  - ホストにプロビジョニングされた仮想コア (vCPU) の総数が、物理コア (pCPU) の総数を超えています。
  - プロビジョニングされた仮想マシンの合計数は、pCPU の合計数よりも多くの vCPU を消費します。

    インスタンスが過剰プロビジョニングされている場合、ハイパーバイザーのスケジューリングオーバーヘッド、バグ、またはハイパーバイザーの制限により、ハイパーバイザーがインスタンスのリザーブドリソース（CPU、メモリなど）を保証しない場合があります。この動作により、NetScaler の CPU リソースが不足し、「使用上のガイドライン」の最初のポイントで説明した問題が発生する可能性があります。管理者は、ホストにプロビジョニングされた vCPU の総数が PCU の総数以下になるように、ホストのテナンシーを減らすことをお勧めします。

    例

    ESX ハイパーバイザーの場合、`esxtop`コマンド出力で VPX vCPU の%RDY%パラメーターが 0 より大きい場合、ESX ホストにはスケジューリングオーバーヘッドがあると言われ、VPX インスタンスにレイテンシー関連の問題が発生する可能性があります。

    このような状況では、%RDY%が常に 0 に戻るように、ホストのテナンシーを減らします。または、ハイパーバイザーベンダーに連絡して、リソース予約が受け付けられない理由を優先順位付けしてください。

パケットエンジンの **CPU** 使用率を制御するコマンド

ハイパーバイザーおよびクラウド環境における VPX インスタンスのパケットエンジン（非管理）CPU 使用率の動作を制御するには、2 つのコマンド（`set ns vpxparam`および`show ns vpxparam`）を使用できます:

- `set ns vpxparam [-cpuyield (YES | NO | DEFAULT)] [-masterclockcpu1 (YES | NO)]`

  各 VM が、別の VM に割り当てられているが、使用されていない CPU リソースを使用できるようにします。

  `Set ns vpxparam` パラメータ:

  -cpuyield: 割り当てられているが未使用の CPU リソースを解放または解放しません。

- はい: 割り当てられているが未使用の CPU リソースを別の VM で使用できるようにします。
- いいえ: 割り当てられた VM のすべての CPU リソースを予約します。このオプションは、ハイパーバイザーおよびクラウド環境で VPX CPU 使用率が高いことを示します。
- デフォルト: いいえ。

> 注
>
> すべての NetScaler VPX プラットフォームで、ホストシステム上の vCPU 使用率は 100% です。`set ns vpxparam -cpuyield YES` コマンドを使用してこの使用方法を無効にしてください。

クラスタノードを「yield」に設定する場合は、CCO で次の追加設定を実行する必要があります:

- クラスターが形成されると、すべてのノードが「yield=DEFAULT」に設定されます。
- すでに「yield=Yes」に設定されたノードを使用してクラスターが形成されている場合、ノードは「DEFAULT」のイールドを使用してクラスターに追加されます。

> 注
>
> クラスタノードを「yield=YES」に設定する場合は、クラスタの形成後にのみ構成でき、クラスタが形成される前には設定できません。

-masterclockcpu1: メインクロックソースを CPU0 (管理 CPU) から CPU1 に移動できます。このパラメータには、次のオプションがあります。

- はい: 仮想マシンがメインクロックソースを CPU0 から CPU1 に移動できるようにします。
- いいえ:VM はメインクロックソースに CPU0 を使用します。デフォルトでは、CPU0 がメインクロックソースです。

- `show ns vpxparam`

  このコマンドは、現在の`vpxparam`設定を表示します。

Linux-KVM プラットフォーム上の NetScaler ADC VPX インスタンス

このセクションでは、構成可能なオプションと設定、および Linux-KVM プラットフォーム上の NetScaler ADC VPX インスタンスの最適なパフォーマンスを達成するのに役立つその他の推奨事項について説明します。

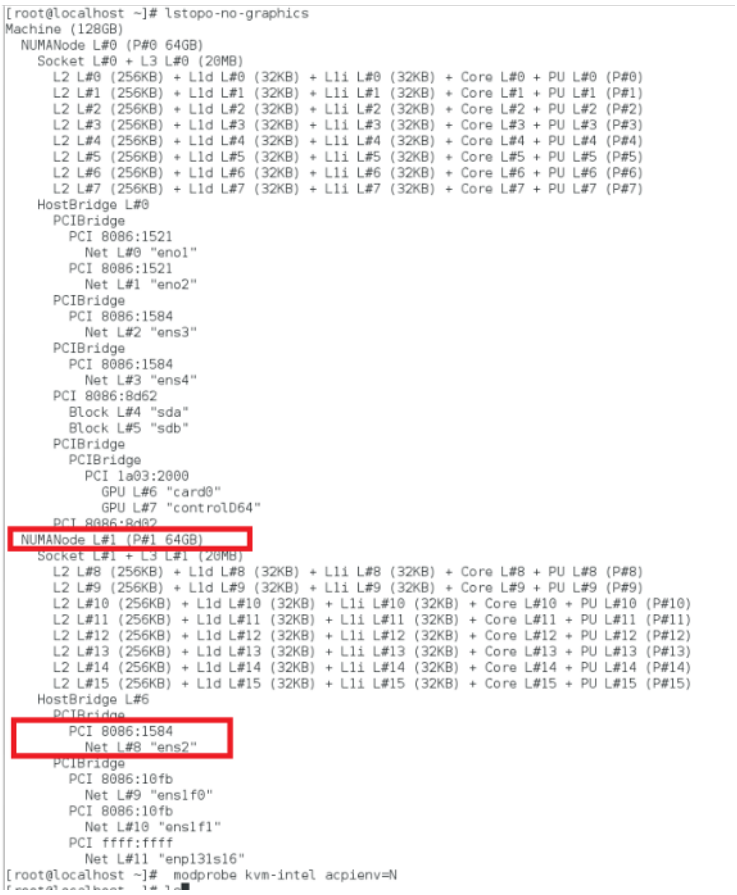
- KVM のパフォーマンス設定
- PV ネットワークインターフェイスを備えた NetScaler ADC VPX
- SR-IOV およびフォートビルの PCIe パススルーネットワークインターフェイスを備えた NetScaler ADC VPX

KVM のパフォーマンス設定

KVM ホストで次の設定を行います。

lstopoコマンドを使用して、**NIC** の **NUMA** ドメインを検索します。

Make sure that memory for the VPX and the CPU is pinned to the same location. VPX と CPU のメモリが同じ場所に固定されていることを確認します。次の出力では、10G NIC「ens2」は NUMA ドメイン #1 に関連付けられています。



NUMA ドメインから **VPX** メモリを割り当てます。

numactlコマンドは、メモリの割り当て元の NUMA ドメインを示します。次の出力では、NUMA ノード #0 から約 10 GB の RAM が割り当てられています。


```
[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 ~]#
```

NUMA ノードマッピングを変更するには、次の手順に従います。

1. ホスト上の VPX の.xml を編集します。

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

2. 次のタグを追加します。

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

3. VPX をシャットダウンします。
4. 次のコマンドを実行します:

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

このコマンドは、NUMA ノードマッピングを使用して VM の構成情報を更新します。

5. VPX の電源をオンにします。次に、ホスト上の`numactl -hardware`コマンド出力を確認して、VPX の更新されたメモリ割り当てを確認します。

```
[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 ~]#
```

VPX の **vCPU** を物理コアにピン留めします。

- VPX の vCPU から pCPU へのマッピングを表示するには、次のコマンドを入力します。

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

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

vCPU 0 ～4 は物理コア 8 ～11 にマッピングされます。

- 現在の pCPU 使用率を表示するには、次のコマンドを入力します。

```
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
```

この出力では、8 は管理 CPU、9 ～11 はパケットエンジンです。

- vCPU を pCPU 固定に変更するには、2 つのオプションがあります。
  - 次のコマンドを使用して、VPX の起動後に実行時に変更します。

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

  - VPX に静的な変更を加えるには、前と同じように次のタグを付けて `.xml` ファイルを編集します。
    1. ホスト上の VPX の.xml ファイルを編集します。

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

    2. 次のタグを追加します。


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

3. VPX をシャットダウンします。
4. 次のコマンドを使用して、NUMA ノードマッピングを使用して VM の設定情報を更新します。

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

5. VPX の電源をオンにします。次に、ホスト上の`virsh vcpupin <VPX name>`コマンド出力をチェックして、更新された CPU ピン接続を確認します。

ホスト割り込みオーバーヘッドを排除します。

- `kvm_stat`コマンドを使用して VM_EXITS を検出します。

  ハイパーバイザーレベルでは、ホスト割り込みは、VPX の仮想 CPU が固定されているのと同じ pCPU にマッピングされます。これにより、VPX 上の vCPU が定期的に追い出される可能性があります。

  ホストを実行している仮想マシンによって実行された VM の終了を確認するには、`kvm_stat`コマンドを使用します。

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

  1+M の順の値が大きいほど、問題があることを示します。

  単一の VM が存在する場合、予想される値は 30～100 K です。それ以上の場合は、同じ pCPU にマップされたホスト割り込みベクターが 1 つ以上あることを示している可能性があります。

- ホスト割り込みを検出し、ホスト割り込みを移行します。

  「/proc/interrupts」ファイルの`concatenate`コマンドを実行すると、すべてのホスト割り込みマッピングが表示されます。1 つ以上のアクティブな IRQ が同じ pCPU にマップされている場合、対応するカウンタが増分します。

  NetScaler VPX の pCPU と重複する割り込みを未使用の pCPU に移動します。

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

- IRQ バランスを無効にします。

  IRQ バランスデーモンを無効にして、その場で再スケジュールが実行されないようにします。


```
service irqbalance stop
service irqbalance show - To check the status
service irqbalance start - Enable if needed
```

必ずkvm_statコマンドを実行して、カウンタの数が多くないことを確認します。

PV ネットワークインターフェイスを備えた NetScaler ADC VPX

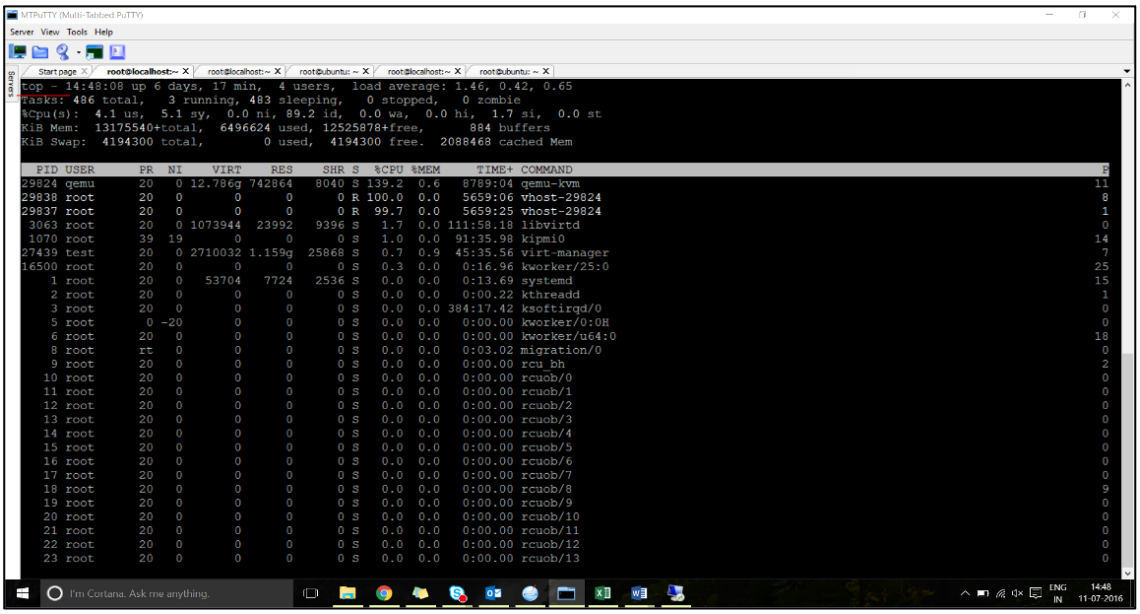
準仮想化（PV）、SR-IOV、および PCIe パススルーネットワークインターフェイスは、物理 NIC ごとに **2** つの **vNIC** 展開として設定できます。詳細については、「物理 NIC 展開ごとに 2 つの vNIC」を参照してください。

PV (virtio) インターフェイスの最適なパフォーマンスを得るには、次の手順に従います。

- PCIe スロット/NIC が属する NUMA ドメインを特定します。
- VPX のメモリと vCPU は、同じ NUMA ドメインにピン接続する必要があります。
- 仮想ホストスレッドは、同じ NUMA ドメイン内の CPU にバインドする必要があります。

仮想ホストスレッドを対応する **CPU** にバインドします。

1. トラフィックが開始されたら、ホストでtopコマンドを実行します。



2. 仮想ホストプロセス (vhost-<pid-of-qemu>という名前) アフィニティを識別します。
3. 次のコマンドを使用して、前に特定した NUMA ドメインの物理コアに vHost プロセスをバインドします。

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

例

```
taskset -pc 12 29838
```


4. NUMA ドメインに対応するプロセッサコアは、次のコマンドで識別できます。

```
[root@localhost ~]# virsh capabilities | grep cpu
<cpu>
    </cpu>
        <cpus num='8'>
            <cpu id='0' socket_id='0' core_id='0' siblings='0'/>
            <cpu id='1' socket_id='0' core_id='1' siblings='1'/>
            <cpu id='2' socket_id='0' core_id='2' siblings='2'/>
            <cpu id='3' socket_id='0' core_id='3' siblings='3'/>
            <cpu id='4' socket_id='0' core_id='4' siblings='4'/>
            <cpu id='5' socket_id='0' core_id='5' siblings='5'/>
            <cpu id='6' socket_id='0' core_id='6' siblings='6'/>
            <cpu id='7' socket_id='0' core_id='7' siblings='7'/>
        </cpus>

        <cpus num='8'>
        <cpu id='8' socket_id='1' core_id='0' siblings='8'/>
        <cpu id='9' socket_id='1' core_id='1' siblings='9'/>
        <cpu id='10' socket_id='1' core_id='2' siblings='10'/>
        <cpu id='11' socket_id='1' core_id='3' siblings='11'/>
        <cpu id='12' socket_id='1' core_id='4' siblings='12'/>
        <cpu id='13' socket_id='1' core_id='5' siblings='13'/>
        <cpu id='14' socket_id='1' core_id='6' siblings='14'/>
        <cpu id='15' socket_id='1' core_id='7' siblings='15'/>
        </cpus>

    <cpuselection/>
    <cpuselection/>
```

QEMU プロセスを対応する物理コアにバインドします。

1. QEMU プロセスが実行されている物理コアを特定します。詳細については、前述の出力を参照してください。
2. 次のコマンドを使用して、vCPU をバインドするのと同じ物理コアに QEMU プロセスをバインドします。

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

SR-IOV およびフォートビルの **PCIe** パススルーネットワークインターフェイスを備えた **NetScaler ADC VPX**

SR-IOV および Fortville PCIe パススルーネットワークインターフェイスのパフォーマンスを最適化するには、次の手順を実行します。

- PCIe スロット/NIC が属する NUMA ドメインを特定します。
- NetScaler VPX のメモリと vCPU は、同じ NUMA ドメインに固定する必要があります。

Linux KVM の **vCPU** およびメモリピンニング用のサンプル **VPX XML** ファイル:

```
    <domain type='kvm'>
        <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            <vcpupin vcpu='0' cpuset='8'/>
10           <vcpupin vcpu='1' cpuset='9'/>
11           <vcpupin vcpu='2' cpuset='10'/>
12           <vcpupin vcpu='3' cpuset='11'/>
13       </cputune>
14
15       <numatune>
16       <memory mode='strict' nodeset='1'/>
17       </numatune>
18
19       </domain>
```

Citrix Hypervisor 上の NetScaler ADC VPX インスタンス

このセクションでは、構成可能なオプションと設定、および Citrix Hypervisors 上の NetScaler ADC VPX インスタンスの最適なパフォーマンスを達成するのに役立つその他の推奨事項について説明します。

- Citrix Hypervisor のパフォーマンス設定
- SR-IOV ネットワークインターフェイスを備えた NetScaler ADC VPX
- 準仮想化インターフェイスを備えた NetScaler ADC VPX

Citrix Hypervisor のパフォーマンス設定

「**xl**」コマンドを使用して **NIC** の **NUMA** ドメインを見つけます。

```
1    xl info -n
```

VPX の **vCPU** を物理コアにピン留めします。

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

vCPU のバインドをチェックします。

```
1    xl vcpu-list
```

8 個を超える仮想 **CPU** を **NetScaler ADC** 仮想マシンに割り当てます。

8 個を超える仮想 CPU を構成するには、Citrix Hypervisor コンソールから次のコマンドを実行します。

```
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
```


SR-IOV ネットワークインターフェイスを備えた **NetScaler ADC VPX**

SR-IOV ネットワークインターフェイスの最適なパフォーマンスを得るには、次の手順を実行します。

- PCIe スロットまたは NIC が接続されている NUMA ドメインを特定します。
- VPX のメモリと vCPU を同じ NUMA ドメインに固定します。
- ドメイン 0 vCPU を残りの CPU にバインドします。

準仮想化インターフェイスを備えた **NetScaler ADC VPX**

最適なパフォーマンスを得るには、他の PV 環境と同様に、pNIC ごとに 2 つの vNIC、および pNIC 構成ごとに 1 つの vNIC を推奨します。

準仮想化 (netfront) インターフェイスの最適なパフォーマンスを実現するには、次の手順を実行します。

- PCIe スロットまたは NIC が属する NUMA ドメインを特定します。
- VPX のメモリと vCPU を同じ NUMA ドメインに固定します。
- ドメイン 0 vCPU を同じ NUMA ドメインの残りの CPU にバインドします。
- 仮想 NIC のホスト Rx/Tx スレッドをドメイン 0 vCPU に固定します。

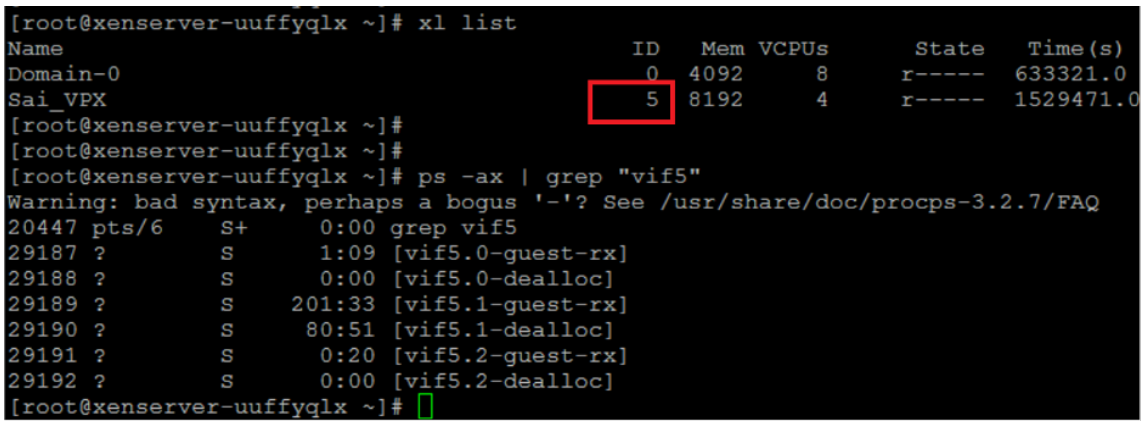
ホストスレッドをドメイン **0 vCPU** にピン留めします。

1. Citrix Hypervisor ホスト シェルで `xl list` コマンドを使用して、NetScaler VPX の Xen-ID を見つけます。
2. 次のコマンドを使用して、ホストスレッドを識別します。

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

   次の例では、これらの値は次のことを示しています。

   - **vif5.0** -XenCenter で VPX に割り当てられた最初のインターフェイス（管理インターフェイス）のスレッド。
   - **vif5.1** -VPX に割り当てられた 2 番目のインターフェースのスレッドなど。



3. 次のコマンドを使用して、スレッドをドメイン 0 vCPU に固定します。

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

例

```
taskset -pc 1 29189
```

NetScaler VPX 構成をクラウドで NetScaler アプライアンスの最初の起動時に適用する

October 17, 2024

NetScaler VPX 構成は、クラウド環境での NetScaler アプライアンスの最初の起動時に適用できます。このステージは、このドキュメントでプレブートステージとして取り上げられています。したがって、ADC プールライセンスなどの特定のケースでは、特定の VPX インスタンスがはるかに短時間で起動されます。この機能は、Microsoft Azure、Google Cloud Platform、および AWS クラウドで使用できます。

ユーザーデータとは何ですか

クラウド環境で VPX インスタンスをプロビジョニングする場合、ユーザーデータをインスタンスに渡すオプションがあります。ユーザーデータを使用すると、一般的な自動設定タスクの実行、インスタンスの起動動作のカスタマイズ、インスタンスの起動後にスクリプトを実行できます。最初の起動時に、NetScaler VPX インスタンスは次のタスクを実行します。

- ユーザーデータを読み取ります。
- ユーザーデータで提供される構成を解釈します。
- 新しく追加された構成をブート時に適用します。

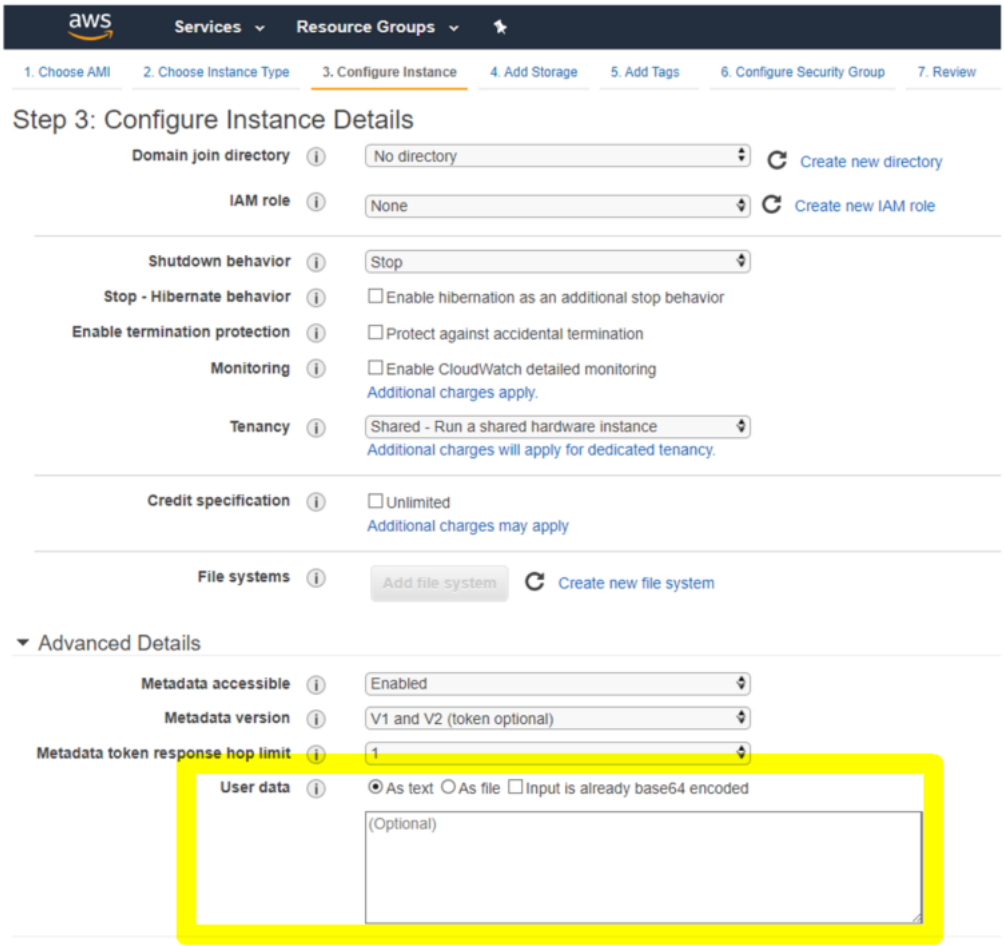
クラウドインスタンスでプレブートユーザーデータを提供する方法

プレブートユーザーデータを XML 形式でクラウドインスタンスに提供できます。クラウドによって、ユーザーデータを提供するためのインターフェースが異なります。

AWS コンソールを使用してプレブートユーザーデータを提供する

AWS コンソールを使用して NetScaler VPX インスタンスをプロビジョニングする場合は、[インスタンスの詳細の構成] **>** [詳細の詳細] に移動し、[ユーザーデータ] フィールドにプレブートユーザーデータ構成を指定します。

各手順の詳細な手順については、「AWS Web コンソールを使用して AWS に NetScaler VPX インスタンスをデプロイする」を参照してください。詳細については、AWS ドキュメントの「インスタンスの起動」を参照してください。



> 注
>
> プリブートユーザーデータ機能の AWS IMDSv2 専用モードは、NetScaler VPX リリース 13.1.48.x 以降のリリースでサポートされています。

AWS CLI を使用してプレブートユーザーデータを提供する

AWS CLI で次のコマンドを入力します。

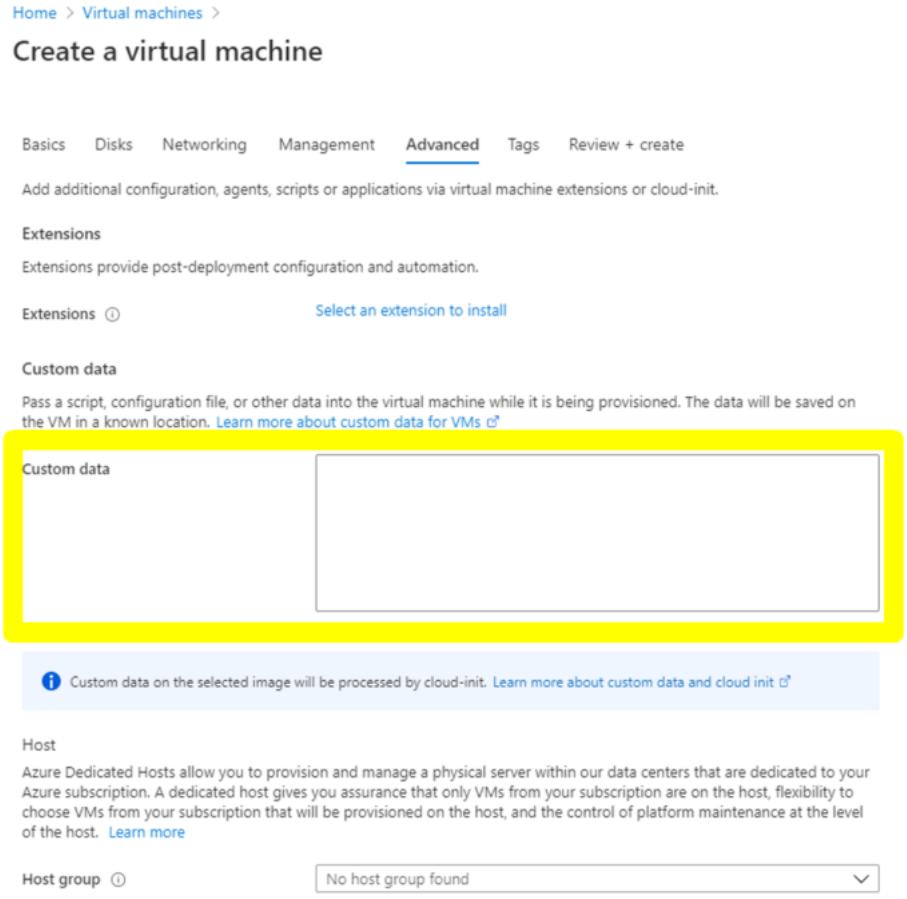
```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type t2.micro \
    --count 1 \
    --subnet-id subnet-08fc749671b2d077c \
    --key-name MyKeyPair \
    --security-group-ids sg-0b0384b66d7d692f9 \
    --user-data file://my_script.txt
```


詳細については、インスタンスの実行に関するAWS ドキュメントを参照してください。

詳細については、インスタンスユーザーデータの使用に関するAWS ドキュメントを参照してください。

Azure コンソールを使用してプリブートユーザーデータを提供する

Azure コンソールを使用して NetScaler VPX インスタンスをプロビジョニングする場合は、［仮想マシンの作成］**>** ［詳細設定］タブに移動します。[カスタムデータ] フィールドに、プリブートユーザーデータの構成を指定します。



Azure CLI を使用してプリブートユーザーデータを提供する

Azure CLI で次のコマンドを入力します。

```
az vm create \
  --resource-group myResourceGroup \
  --name MyVm \
  --image debian \
  --custom-data MyCloudInitScript.txt \
```

例


```
az vm create --resource-group MyResourceGroup -name MyVm --image
    debian --custom-data MyCloudInitScript.txt
```

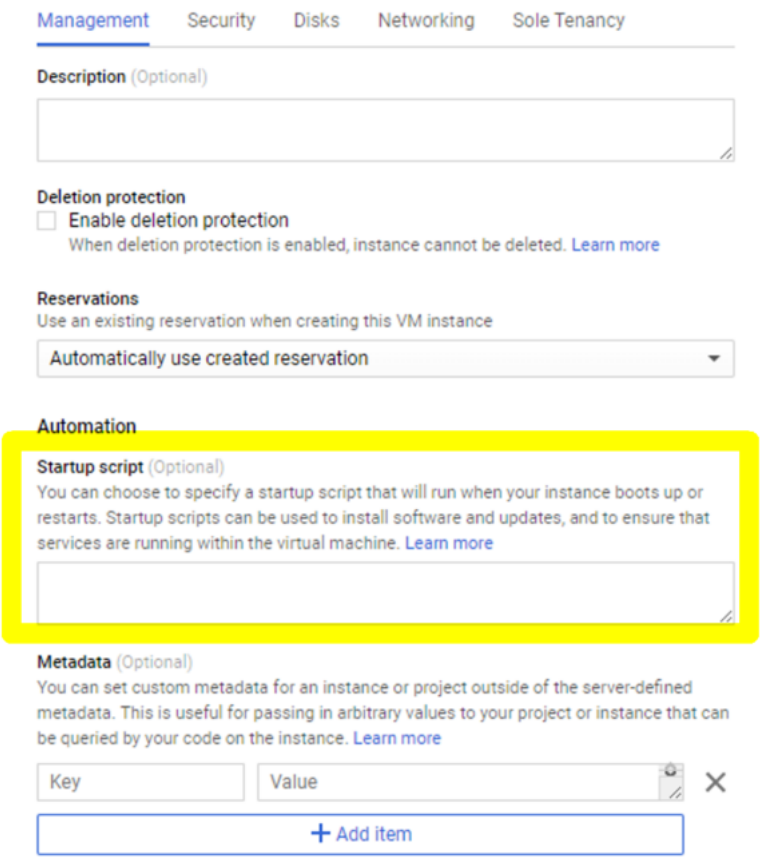
カスタムデータまたはプリブート設定をファイルとして「—custom-data」パラメータに渡すことができます。この例では、ファイル名は **MyCloudInitScript.txt** です。

詳細については、Azure CLI のドキュメントを参照してください。

GCP コンソールを使用してプレブートユーザーデータを提供する

GCP コンソールを使用して NetScaler VPX インスタンスをプロビジョニングする場合は、インスタンスのプロパティを入力します。管理、セキュリティ、ディスク、ネットワーキング、単独テナンシを展開します。[管理] タブに移動します。[自動化] セクションで、[スタートアップスクリプト] フィールドにプリブートユーザーデータ設定を指定します。

GCP を使用して VPX インスタンスを作成する方法の詳細については、「Google Cloud Platform に NetScaler VPX インスタンスをデプロイする」を参照してください。



gcloud CLI を使用してプレブートユーザーデータを提供する

GCP CLI で次のコマンドを入力します。

```
gcloud compute instances create INSTANCE_NAMES --metadata-from-file=
   startup-script=LOCAL_FILE_PATH
```

metadata-from-file -に格納されているファイルから値またはユーザーデータを読み取ります。.

詳細については、gcloud CLI ドキュメントを参照してください。

プレブートユーザーデータ形式

プレブートユーザーデータは XML 形式でクラウドインスタンスに提供する必要があります。起動時にクラウドインフラストラクチャを介して提供される NetScaler プレブートユーザーデータは、次の 4 つのセクションで構成されます。

- NetScaler 構成は<NS-CONFIG>タグで表されます。
- <NS-BOOTSTRAP>タグで表される NetScaler をカスタムブートストラップします。
- <NS-SCRIPTS>タグで表される NetScaler にユーザースクリプトを保存する。
- <NS-LICENSE-CONFIG>タグで表されるプールライセンス構成。

前の 4 つのセクションは、ADC のプレブート構成内で任意の順序で提供できます。プリブートユーザーデータを提供しながら、次のセクションに示す書式に厳密に従うようにしてください。プリブート ユーザー データを提供する際は、次のセクションに示すフォーマットに厳密に従ってください。

> 注
>
> 次の例に示すように、プレブートユーザーデータ構成全体を<NS-PRE-BOOT-CONFIG>タグで囲む必要があります。

例 **1**:

```
<NS-PRE-BOOT-CONFIG>
     <NS-CONFIG>           </NS-CONFIG>
     <NS-BOOTSTRAP>        </NS-BOOTSTRAP>
     <NS-SCRIPTS>          </NS-SCRIPTS>
     <NS-LICENSE-CONFIG>   </NS-LICENSE-CONFIG>
</NS-PRE-BOOT-CONFIG>
```

例 **2**:

```
<NS-PRE-BOOT-CONFIG>
    <NS-LICENSE-CONFIG> </NS-LICENSE-CONFIG>
    <NS-SCRIPTS>        </NS-SCRIPTS>
    <NS-BOOTSTRAP>      </NS-BOOTSTRAP>
    <NS-CONFIG>         </NS-CONFIG>
</NS-PRE-BOOT-CONFIG>
```


<NS-CONFIG>タグを使用して、プレブート段階で VPX インスタンスに適用する必要がある特定の NetScaler VPX 構成を指定します。

> 注
>
> <NS-CONFIG>セクションには、有効な ADC CLI コマンドが必要です。CLI は、構文エラーまたは形式について検証されません。

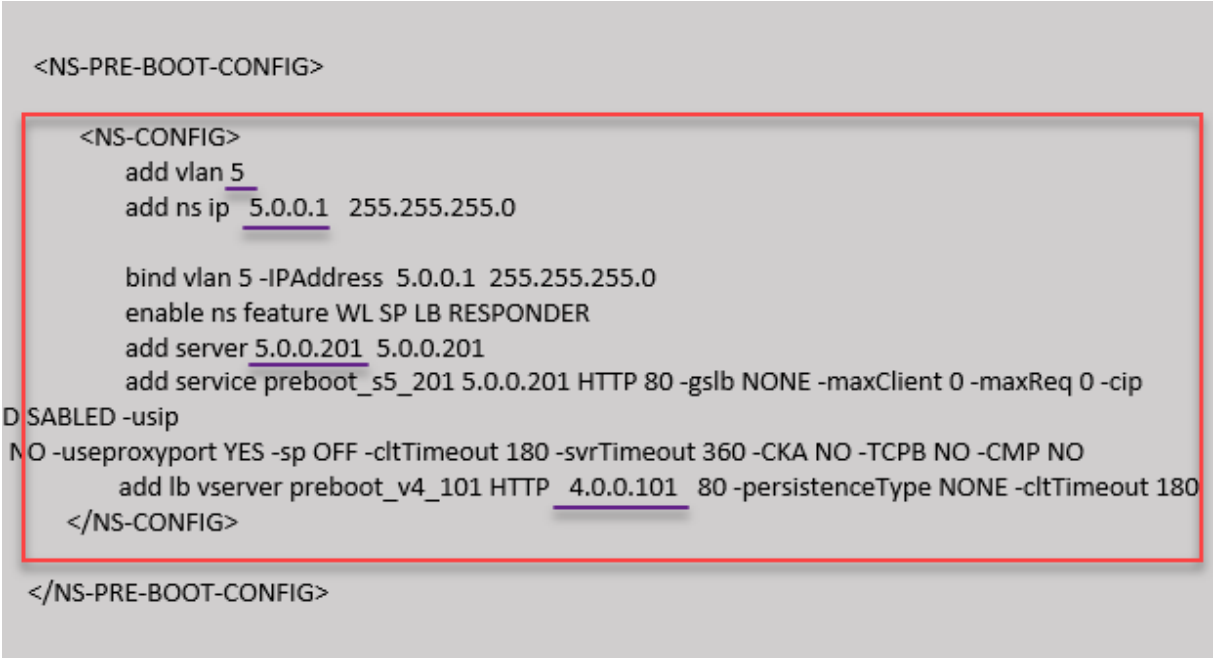
NetScaler 構成

<NS-CONFIG>タグを使用して、プレブート段階で VPX インスタンスに適用する必要がある特定の NetScaler VPX 構成を指定します。

> 注
>
> <NS-CONFIG>セクションには、有効な ADC CLI コマンドが必要です。CLI は、構文エラーまたは形式について検証されません。

例

次の例では、<NS-CONFIG>セクションに設定の詳細を示します。ID「5」の VLAN が設定され、SNIP（5.0.0.1）にバインドされます。負荷分散仮想サーバー (4.0.0.101) も構成されています。



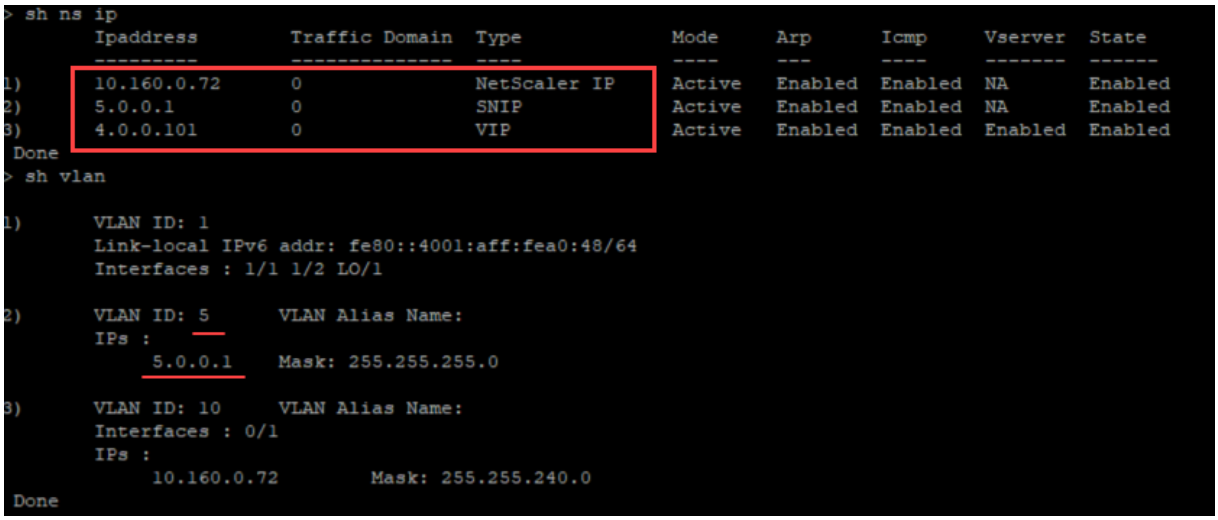
前のスクリーンショットに示した設定をここからコピーできます。

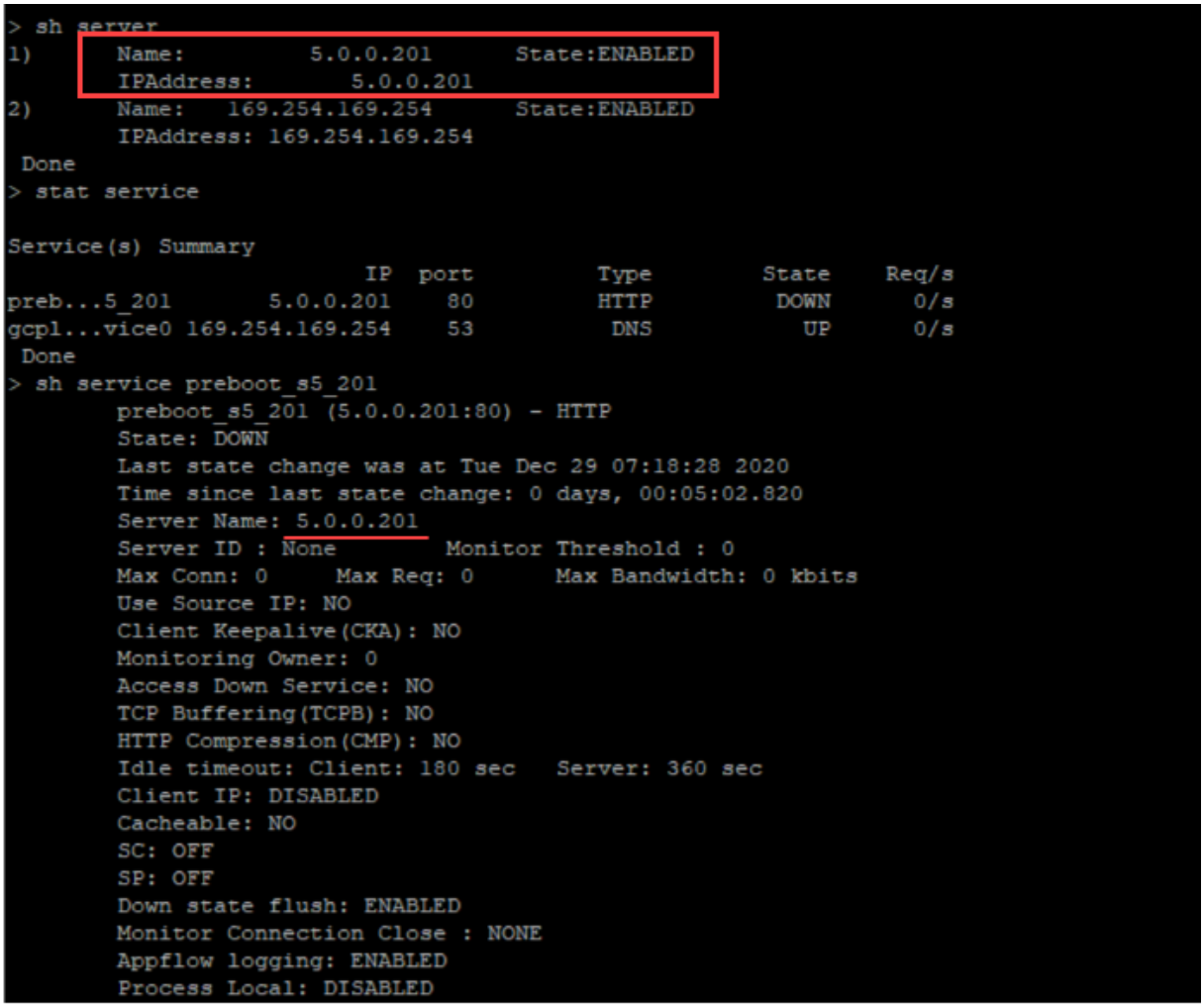
```
<NS-PRE-BOOT-CONFIG>
    <NS-CONFIG>
        add vlan 5
        add ns ip 5.0.0.1 255.255.255.0
```



```
5              bind vlan 5 -IPAddress 5.0.0.1 255.255.255.0
6              enable ns feature WL SP LB RESPONDER
7              add server 5.0.0.201 5.0.0.201
8              add service preboot_s5_201 5.0.0.201 HTTP 80 -gslb NONE -
                  maxClient 0 -maxReq 0 -cip DISABLED -usip
9       NO -useproxyport YES -sp OFF -cltTimeout 180 -svrTimeout 360 -CKA NO
            -TCPB NO -CMP NO
10             add lb vserver preboot_v4_101 HTTP 4.0.0.101 80 -
                  persistenceType NONE -cltTimeout 180
11          </NS-CONFIG>
12      </NS-PRE-BOOT-CONFIG>
```

NetScaler VPX インスタンスは、次の図に示すように、<NS-CONFIG>セクションに適用された構成を表示します。





ユーザースクリプト

<NS-SCRIPTS>タグを使用して、NetScaler VPX インスタンスに保存して実行する必要があるスクリプトを指定します。

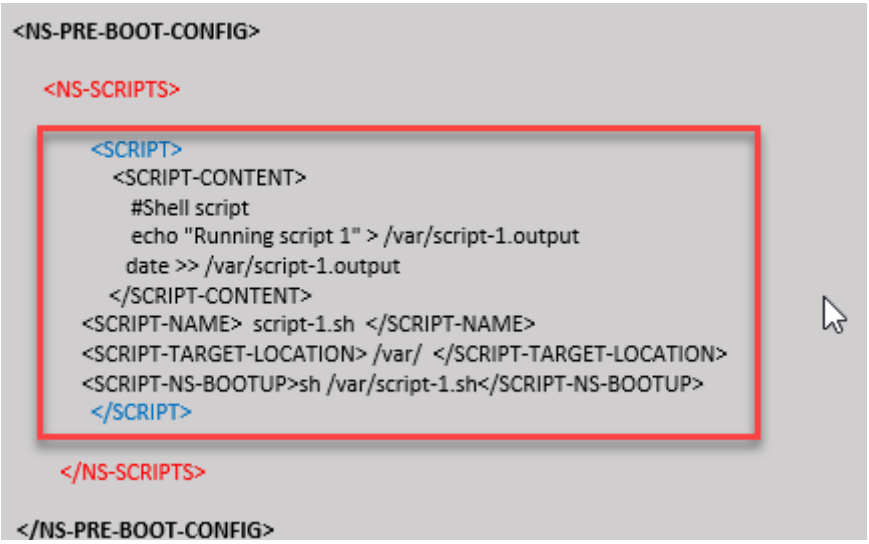
<NS-SCRIPTS>タグには多数のスクリプトを含めることができます。各スクリプトは<SCRIPT>タグ内に含める必要があります。各<SCRIPT>セクションは 1 つのスクリプトに対応し、次のサブタグを使用してスクリプトの詳細をすべて含みます。各 <SCRIPT> セクションは 1 つのスクリプトに対応し、次のサブタグを使用してスクリプトの詳細がすべて含まれています。

- **<SCRIPT-NAME>:**保存する必要のあるスクリプトファイルの名前を示します。
- **<SCRIPT-CONTENT>:**保存する必要のあるファイルの内容を示します。
- **<SCRIPT-TARGET-LOCATION>:**このファイルを保存する必要がある指定されたターゲットの場所を示します。ターゲットの場所が指定されていない場合、デフォルトでは、ファイルまたはスクリプトは「/nsconfig」ディレクトリに保存されます。
- **<SCRIPT-NS-BOOTUP>:**スクリプトの実行に使用するコマンドを指定します。

- <SCRIPT-NS-BOOTUP>セクションを使用する場合、セクションで提供されるコマンドは「/nsconfig/nsafter.sh」に保存され、コマンドは「nsafter.sh」実行の一部としてパケットエンジンが起動した後に実行されます。
- <SCRIPT-NS-BOOTUP>セクションを使用しない場合、スクリプトファイルは指定したターゲットの場所に保存されます。

例 **1**:

この例では、<NS-SCRIPTS>タグには script-1.sh というスクリプトの詳細が 1 つだけ含まれています。「script-1.sh」スクリプトは「/var」ディレクトリに保存されます。スクリプトは指定された内容で読み込まれ、パケットエンジンの起動後に「sh /var/script-1.sh」コマンドで実行されます。



前のスクリーンショットに示した設定をここからコピーできます。

```
<NS-PRE-BOOT-CONFIG>
    <NS-SCRIPTS>
    <SCRIPT>
            <SCRIPT-CONTENT>
                #Shell script
                echo "Running script 1" > /var/script-1.output
                date >> /var/script-1.output
            </SCRIPT-CONTENT>

                <SCRIPT-NAME> script-1.sh </SCRIPT-NAME>
                <SCRIPT-TARGET-LOCATION> /var/ </SCRIPT-TARGET-
                    LOCATION>
                <SCRIPT-NS-BOOTUP>sh /var/script-1.sh</SCRIPT-NS-
                    BOOTUP>
        </SCRIPT>
    </NS-SCRIPTS>
</NS-PRE-BOOT-CONFIG>
```

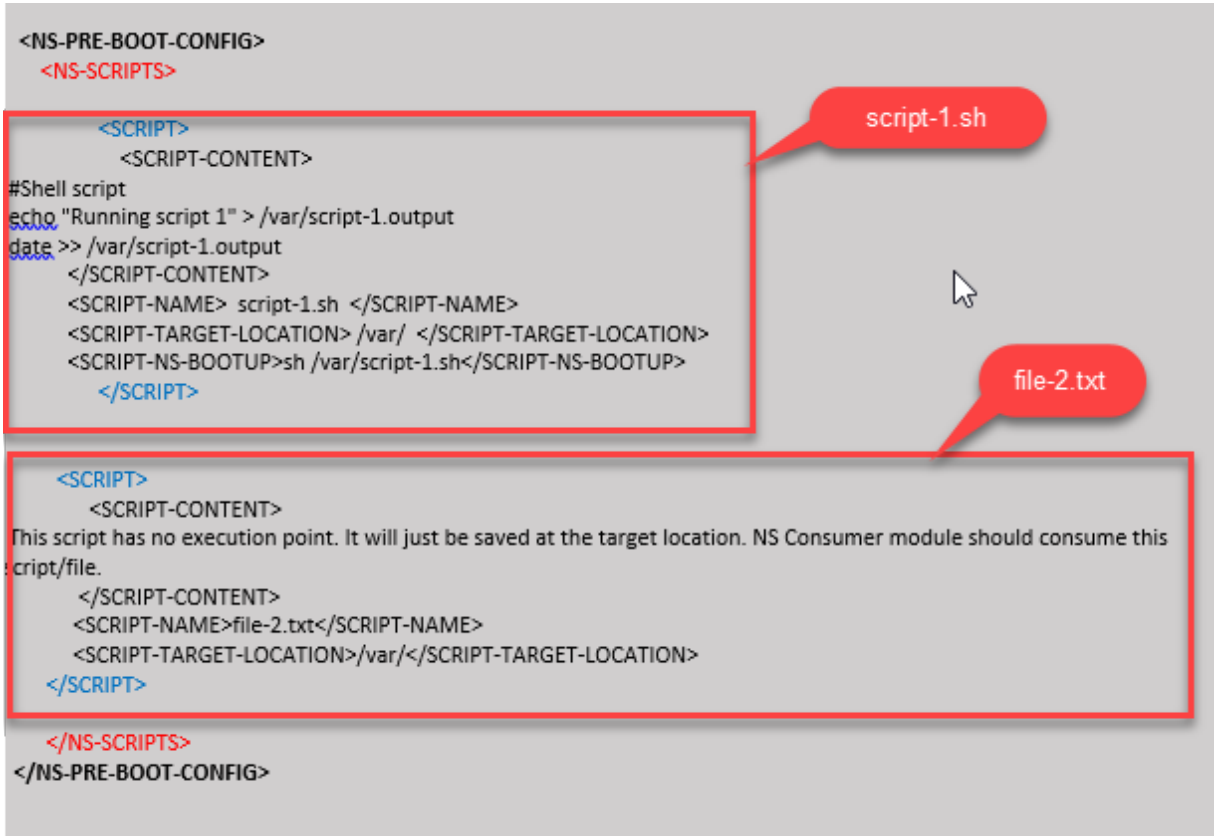
次のスナップショットでは、「script-1.sh」スクリプトが「/var/」ディレクトリに保存されていることを確認できます。「Script-1.sh」スクリプトが実行され、出力ファイルが適切に作成されます。


```
root@ns#
root@ns# ls /var/
.monit.id               core                    gui                     nsinstall               pubkey
.monit.state            crash                   install                 nslog                   python
.snap                   cron                    krb                     nsproflog               run
AAA                     db                      learnt_data             nssynclog               safenet
app_catalog             dev                     log                     nstemplates             script-1.output
cloudhadaemon           download                mastools                nstmp                   script-1.sh
cloudhadaemon.tgz       empty                   netscaler               nstrace                 tmp
clusterd                file-2.txt              ns_gui                  opt                     vpn
configdb                gcfl                    ns_sys_backup           osr_compliance          vpns
root@ns#
root@ns# cat /var/script-1.sh
#Shell script
echo "Running script 1" > /var/script-1.output
date >> /var/script-1.output
root@ns#
root@ns# cat /var/script-1.output
Running script 1
Wed Jan  6 05:25:33 UTC 2021
root@ns#
root@ns#
```

例 **2**:

次の例では、<NS-SCRIPTS>タグに 2 つのスクリプトの詳細が含まれています。

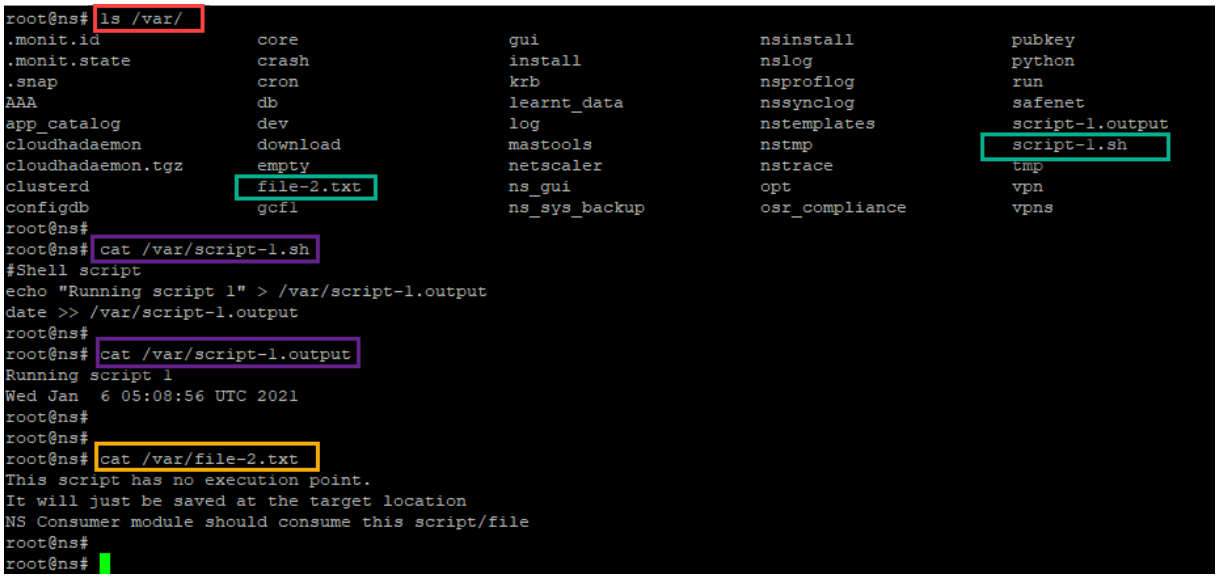
- 最初のスクリプトは「script-1.sh」として「/var」ディレクトリに保存されます。スクリプトは指定された内容で読み込まれ、パケットエンジンの起動後にコマンド「sh /var/script-1.sh」で実行されます。
- 2 番目のスクリプトは「file-2.txt」として「/var」ディレクトリに保存されます。このファイルには、指定されたコンテンツが入力されます。しかし、ブートアップ実行コマンド<SCRIPT-NS-BOOTUP>が提供されていないため、実行されません。



前のスクリーンショットに示した設定をここからコピーできます。

```
<NS-PRE-BOOT-CONFIG>
    <NS-SCRIPTS>
        <SCRIPT>
            <SCRIPT-CONTENT>
               #Shell script
               echo "Running script 1" > /var/script-1.output
               date >> /var/script-1.output
            </SCRIPT-CONTENT>

            <SCRIPT-NAME>  script-1.sh  </SCRIPT-NAME>
            <SCRIPT-TARGET-LOCATION> /var/  </SCRIPT-TARGET-LOCATION>
            <SCRIPT-NS-BOOTUP>sh /var/script-1.sh</SCRIPT-NS-BOOTUP>
            </SCRIPT>

        <SCRIPT>
            <SCRIPT-CONTENT>
                This script has no execution point.
                It will just be saved at the target location
                NS Consumer module should consume this script/file
            </SCRIPT-CONTENT>
            <SCRIPT-NAME>file-2.txt</SCRIPT-NAME>
            <SCRIPT-TARGET-LOCATION>/var/</SCRIPT-TARGET-LOCATION>
        </SCRIPT>
    </NS-SCRIPTS>
</NS-PRE-BOOT-CONFIG>
```

次のスナップショットでは、script-1.sh と file-2.txt が「/var/」ディレクトリに作成されていることを確認できます。Script-1.sh が実行され、出力ファイルが適切に作成されます。



ライセンス

VPX インスタンスの起動中に NetScaler プールライセンスを適用するには、<NS-LICENSE-CONFIG>タグを使用します。<NS-LICENSE-CONFIG>セクション内の<LICENSE-COMMANDS>タグを使用して、プールされたライセンスコマンドを指定します。これらのコマンドは構文的に有効である必要があります。

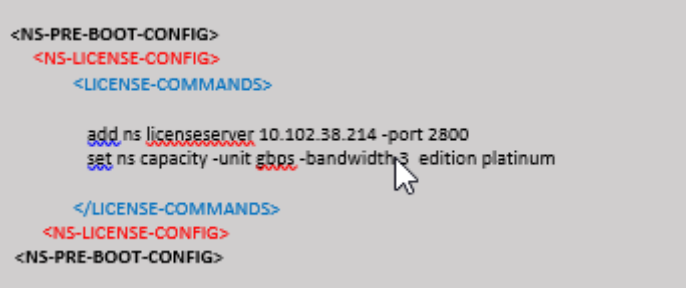
標準のプールライセンスコマンドを使用して、<LICENSE-COMMANDS>セクションで、ライセンスタイプ、容量、ライセンスサーバーなどのプールされたライセンスの詳細を指定できます。詳細については、「NetScaler プール容量ライセンスの構成」を参照してください。

<NS-LICENSE-CONFIG>を適用した後、VPX は起動時に要求されたエディションを起動し、VPX はライセンスサーバから構成されたライセンスをチェックアウトしようとします。

- ライセンスのチェックアウトが成功すると、構成された帯域幅が VPX に適用されます。
- ライセンスのチェックアウトに失敗した場合、約 10 ～12 分以内にライセンスはライセンスサーバから取得されません。その結果、システムがリブートし、ライセンスなしの状態になります。

例

次の例では、<NS-LICENSE-CONFIG>を適用した後、VPX は起動時にプレミアムエディションを起動し、VPX はライセンスサーバ（10.102.38.214）から構成されたライセンスをチェックアウトしようとします。



前のスクリーンショットに示した設定をここからコピーできます。

```
<NS-PRE-BOOT-CONFIG>
   <NS-LICENSE-CONFIG>
        <LICENSE-COMMANDS>
            add ns licenseserver 10.102.38.214 -port 2800
            set ns capacity -unit gbps -bandwidth 3  edition platinum
        </LICENSE-COMMANDS>
    </NS-LICENSE-CONFIG>
</NS-PRE-BOOT-CONFIG>
```

次の図に示すように、「ライセンスサーバーの表示」コマンドを実行し、ライセンスサーバー（10.102.38.214）が VPX に追加されていることを確認します。

ブートストラッピング

<NS-BOOTSTRAP>タグを使用して、カスタムブートストラップ情報を指定します。<NS-BOOTSTRAP>セクション内では、<SKIP-DEFAULT-BOOTSTRAP>タグと<NEW-BOOTSTRAP-SEQUENCE>タグを使用できます。このセクションでは、デフォルトのブートストラップを回避するかどうかを NetScaler アプライアンスに通知します。デフォルトのブートストラップが回避される場合、このセクションでは、新しいブートストラップシーケンスを提供するオプションを提供します。

デフォルトのブートストラップ構成

NetScaler アプライアンスのデフォルトのブートストラップ構成は、次のインターフェイスの割り当てに従います。

- **Eth0** -特定の NSIP アドレスを持つ管理インターフェイス。
- **Eth1** -特定の VIP アドレスを持つクライアント向けインターフェイス。
- **Eth2** -特定の SNIP アドレスを持つサーバー側インターフェイス。

ブートストラップ構成をカスタマイズする

デフォルトのブートストラップシーケンスをスキップして、NetScaler VPX インスタンスに新しいブートストラップシーケンスを指定することができます。<NS-BOOTSTRAP>タグを使用して、カスタムブートストラップ情報を指定します。たとえば、管理インターフェイス（NSIP）、クライアント側インターフェイス（VIP）、およびサーバー側インターフェイス（SNIP）が常に特定の順序で提供されるデフォルトのブートストラップを変更できます。

次の表に、<SKIP-DEFAULT-BOOTSTRAP>および<NEW-BOOTSTRAP-SEQUENCE>タグで許可されるさまざまな値を使用したブートストラップ動作を示します。

SKIP-DEFAULT-BOOTSTRAP	NEW-BOOTSTRAP-SEQUENCE	ブートストラップ動作
はい	はい	デフォルトのブートストラップ動作はスキップされ、<NS-BOOTSTRAP>セクションで提供される新しいカスタムブートストラップシーケンスが実行されます。
はい	いいえ	デフォルトのブートストラップ動作はスキップされます。「<NS-CONFIG>」セクションに記載されているブートストラップコマンドが実行されます。

ブートストラップ構成は、次の 3 つの方法でカスタマイズできます。

- インターフェイスの詳細のみを入力します。
- IP アドレスとサブネットマスクとともにインターフェイスの詳細を指定します。
- <NS-CONFIG>セクションにブートストラップ関連のコマンドを入力します。

方法 **1:** インターフェイスの詳細のみを指定してカスタムブートストラップ

管理インターフェイス、クライアント向けインターフェイス、およびサーバ側インターフェイスは指定しますが、その IP アドレスとサブネットマスクは指定しません。IP アドレスとサブネットマスクは、クラウドインフラストラクチャのクエリによって設定されます。

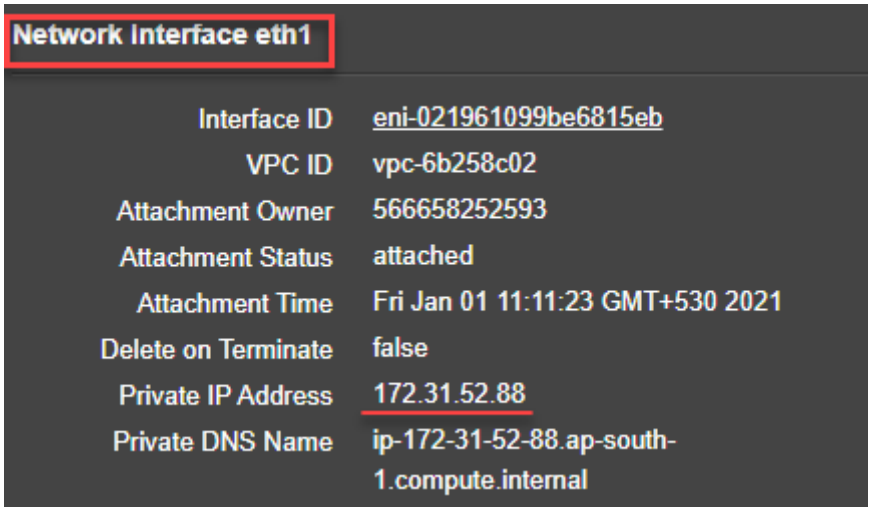
AWS のカスタムブートストラップの例

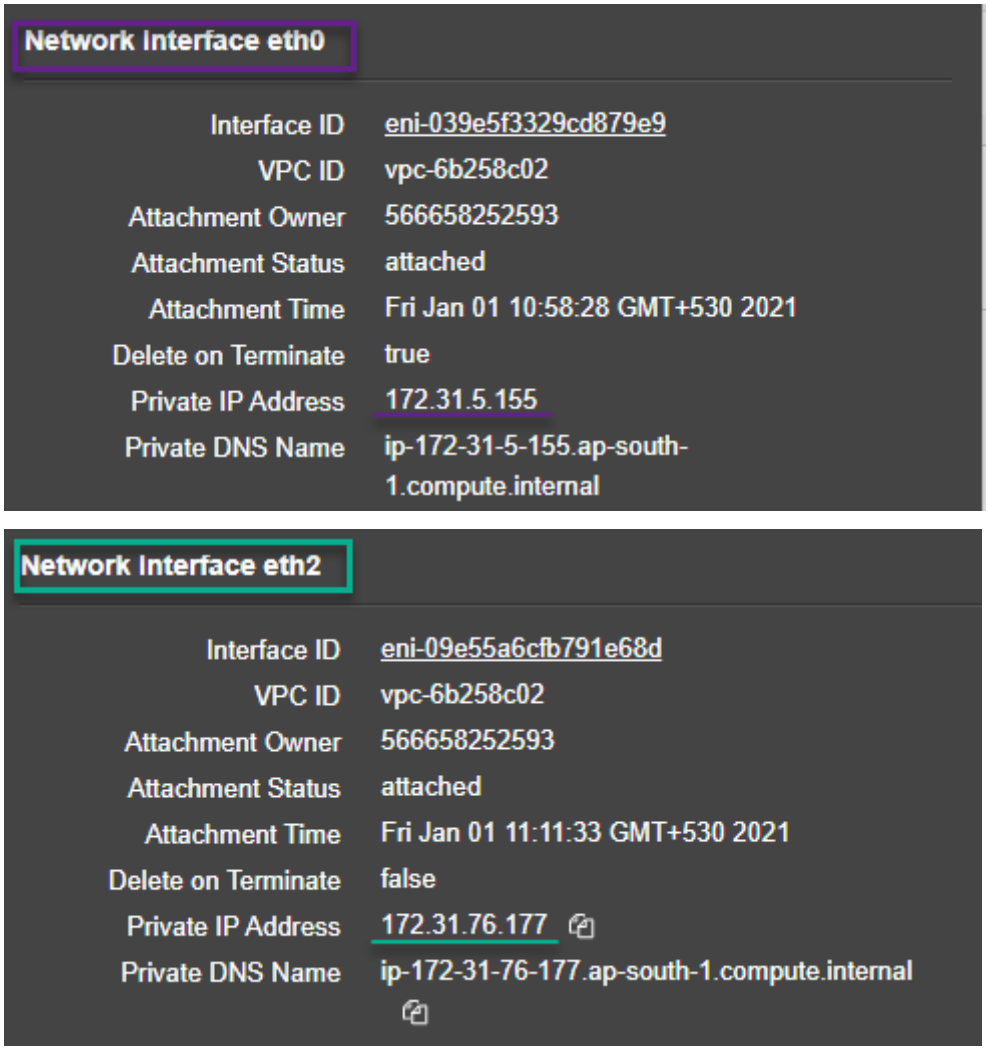
次の例に示すように、カスタムブートストラップシーケンスを指定します。詳細については、「クラウドインスタンスでプレブートユーザーデータを提供する方法」を参照してください。Eth2 インターフェイスは、管理インターフェイス（NSIP）として、Eth1 インターフェイスをクライアントインターフェイス（VIP）として、Eth0 インターフェイスをサーバインターフェイス（SNIP）として割り当てます。<NS-BOOTSTRAP>セクションには、インターフェイスの詳細のみが含まれ、IP アドレスとサブネットマスクの詳細は含まれません。


```
<NS-PRE-BOOT-CONFIG>
    <NS-BOOTSTRAP>
        <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
        <NEW-BOOTSTRAP-SEQUENCE>YES</NEW-BOOTSTRAP-SEQUENCE>

        <MGMT-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth1 </INTERFACE-NUM>
        </MGMT-INTERFACE-CONFIG>

        <CLIENT-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth0 </INTERFACE-NUM>
        </CLIENT-INTERFACE-CONFIG>

        <SERVER-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth2 </INTERFACE-NUM>
        </SERVER-INTERFACE-CONFIG>

    </NS-BOOT STRAP>
</NS-PRE-BOOT-CONFIG>
```

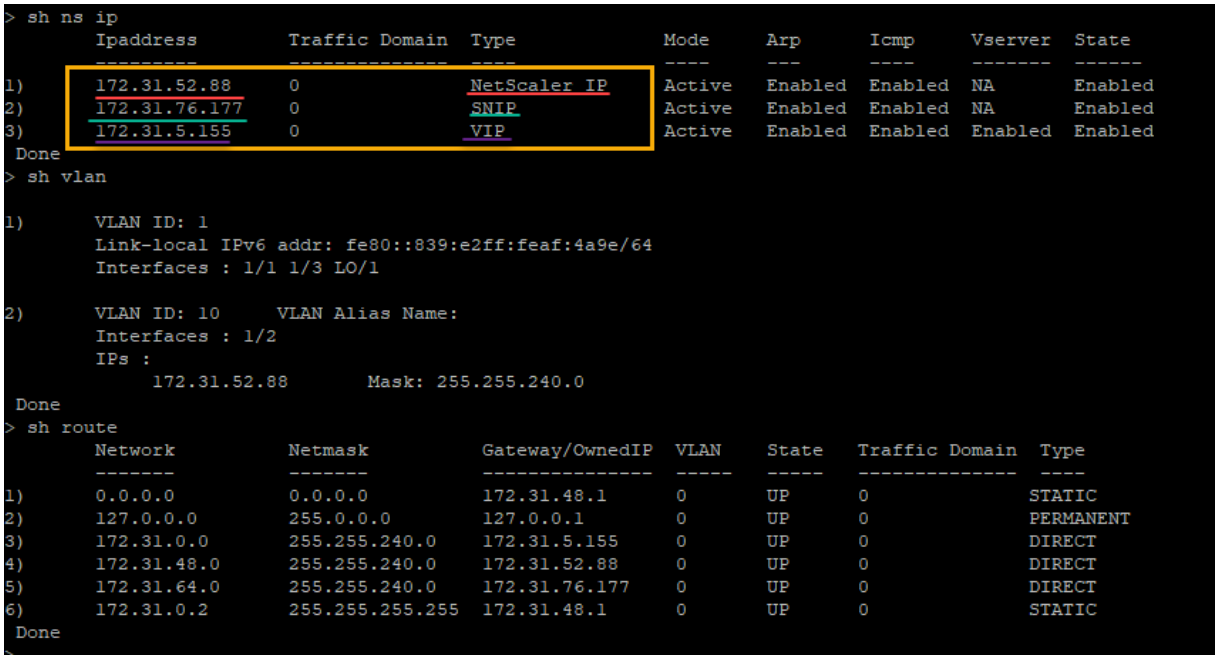
VM インスタンスの作成後、AWS ポータルで、ネットワークインターフェイスのプロパティを次のように確認できます。

1. [**AWS Portal] > [EC2** インスタンス] に移動し、カスタムブートストラップ情報を指定して作成したインスタンスを選択します。
2. [説明] タブでは、次の図に示すように、各ネットワークインタフェースのプロパティを確認できます。





ADC CLI で`show nsip`コマンドを実行し、ADC アプライアンスの初回起動時に NetScalerVPX インスタンスに適用されるネットワークインターフェイスを確認できます。



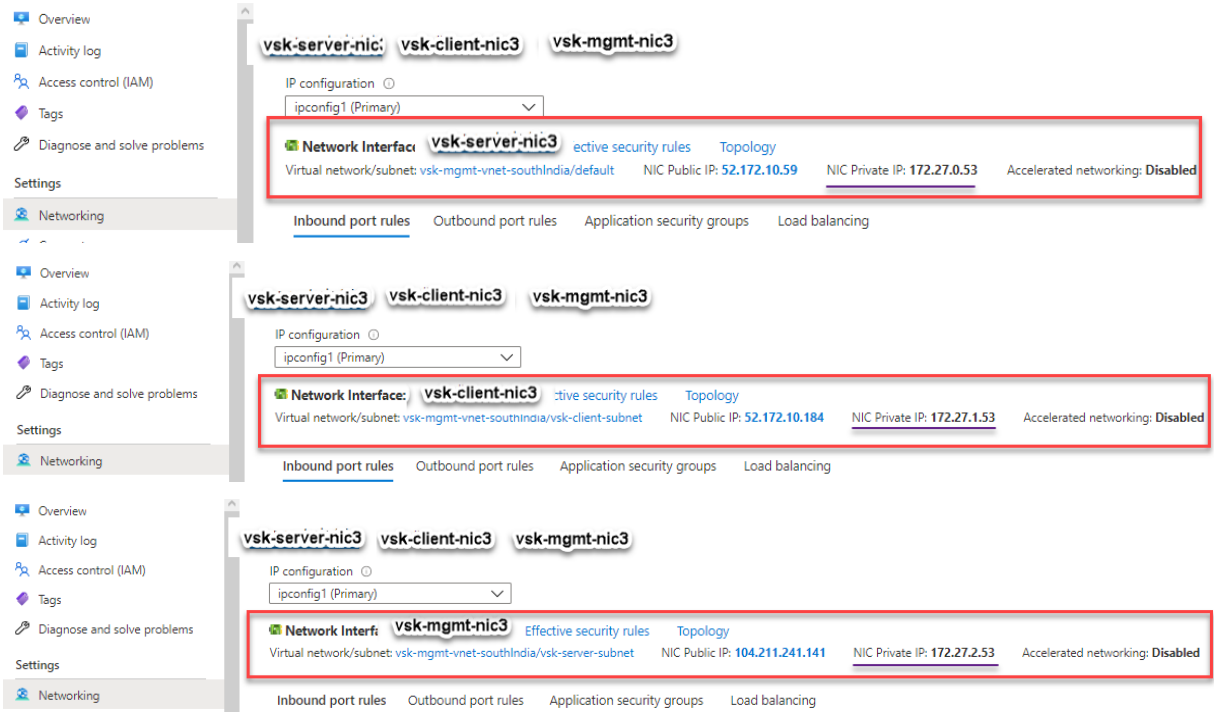
```
> sh ns ip
        Ipaddress        Traffic Domain  Type             Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----             ----     ---      ----     -------  ------
1)      172.31.52.88     0               NetScaler IP     Active   Enabled  Enabled  NA       Enabled
2)      172.31.76.177    0               SNIP             Active   Enabled  Enabled  NA       Enabled
3)      172.31.5.155     0               VIP              Active   Enabled  Enabled  Enabled  Enabled
 Done
> sh vlan

1)      VLAN ID: 1
        Link-local IPv6 addr: fe80::839:e2ff:feaf:4a9e/64
        Interfaces : 1/1 1/3 LO/1

2)      VLAN ID: 10     VLAN Alias Name:
        Interfaces : 1/2
        IPs :
             172.31.52.88       Mask: 255.255.240.0
 Done
> sh route
        Network          Netmask          Gateway/OwnedIP  VLAN    State   Traffic Domain  Type
        -------          -------          ---------------  -----   -----   --------------  ----
1)      0.0.0.0          0.0.0.0          172.31.48.1      0       UP      0               STATIC
2)      127.0.0.0        255.0.0.0        127.0.0.1        0       UP      0               PERMANENT
3)      172.31.0.0       255.255.240.0    172.31.5.155     0       UP      0               DIRECT
4)      172.31.48.0      255.255.240.0    172.31.52.88     0       UP      0               DIRECT
5)      172.31.64.0      255.255.240.0    172.31.76.177    0       UP      0               DIRECT
6)      172.31.0.2       255.255.255.255  172.31.48.1      0       UP      0               STATIC
 Done
```

Azure のカスタムブートストラップの例

次の例に示すように、カスタムブートストラップシーケンスを指定します。詳細については、「クラウドインスタンスでプレブートユーザーデータを提供する方法」を参照してください。Eth1 インターフェイスは管理インターフェイス（NSIP）、クライアントインターフェイス（VIP）として Eth0 インターフェイス、サーバインターフェイス（SNIP）として Eth2 インターフェイスが割り当てられます。<NS-BOOTSTRAP>セクションには、インターフェイスの詳細のみが含まれ、IP アドレスとサブネットマスクの詳細は含まれません。


```
<NS-PRE-BOOT-CONFIG>

    <NS-BOOTSTRAP>
        <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
        <NEW-BOOTSTRAP-SEQUENCE>YES</NEW-BOOTSTRAP-SEQUENCE>

        <MGMT-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth2 </INTERFACE-NUM>
        </MGMT-INTERFACE-CONFIG>

        <CLIENT-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth1 </INTERFACE-NUM>
        </CLIENT-INTERFACE-CONFIG>

        <SERVER-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth0 </INTERFACE-NUM>
        </SERVER-INTERFACE-CONFIG>

    </NS-BOOTSTRAP>

</NS-PRE-BOOT-CONFIG>
```

NetScaler VPX インスタンスが 3 つのネットワークインターフェイスで作成されていることがわかります。**Azure Portal > VM** インスタンス **>** ネットワークに移動し、次の図に示すように 3 つの NIC のネットワークプロパティを確認します。



ADC CLI で`show nsip`コマンドを実行し、`<NS-BOOTSTRAP>`セクションで指定された新しいブー

トストラップシーケンスが適用されていることを確認できます。「show route」コマンドを実行して、サブネットマスクを確認できます。

```
> sh ns ip
        Ipaddress        Traffic Domain  Type             Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----             ----     ---      ----     -------  ------
1)      172.27.2.53      0               NetScaler IP     Active   Enabled  Enabled  NA       Enabled
2)      172.27.0.53      0               SNIP             Active   Enabled  Enabled  NA       Enabled
3)      172.27.1.53      0               VIP              Active   Enabled  Enabled  Enabled  Enabled
 Done
> sh vlan

1)      VLAN ID: 1
        Link-local IPv6 addr: fe80::20d:3aff:fec9:c26c/64
        Interfaces : 0/1 1/1 LO/1

2)      VLAN ID: 10     VLAN Alias Name:
        Interfaces : 1/2
        IPs :
             172.27.2.53        Mask: 255.255.255.0
 Done
> sh route
        Network          Netmask          Gateway/OwnedIP  VLAN    State   Traffic Domain  Type
        -------          -------          ---------------  -----   -----   --------------  ----
1)      0.0.0.0          0.0.0.0          172.27.2.1       0       UP      0               STATIC
2)      127.0.0.0        255.0.0.0        127.0.0.1        0       UP      0               PERMANENT
3)      172.27.0.0       255.255.255.0    172.27.0.53      0       UP      0               DIRECT
4)      172.27.1.0       255.255.255.0    172.27.1.53      0       UP      0               DIRECT
5)      172.27.2.0       255.255.255.0    172.27.2.53      0       UP      0               DIRECT
6)      169.254.0.0      255.255.0.0      172.27.0.1       0       UP      0               STATIC
7)      168.63.129.16    255.255.255.255  172.27.0.1       0       UP      0               STATIC
8)      169.254.169.254  255.255.255.255  172.27.0.1       0       UP      0               STATIC
 Done
>
```

GCP のカスタムブートストラップの例

次の例に示すように、カスタムブートストラップシーケンスを指定します。詳細については、「クラウドインスタンスでプレブートユーザーデータを提供する方法」を参照してください。Eth2 インターフェイスは、管理インターフェイス（NSIP）として、Eth1 インターフェイスをクライアントインターフェイス（VIP）として、Eth0 インターフェイスをサーバインターフェイス（SNIP）として割り当てます。<NS-BOOTSTRAP>セクションには、インターフェイスの詳細のみが含まれ、IP アドレスとサブネットマスクの詳細は含まれません。


```
<NS-PRE-BOOT-CONFIG>
    <NS-BOOTSTRAP>
        <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
        <NEW-BOOTSTRAP-SEQUENCE>YES</NEW-BOOTSTRAP-SEQUENCE>

        <MGMT-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth1 </INTERFACE-NUM>
        </MGMT-INTERFACE-CONFIG>

        <CLIENT-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth0 </INTERFACE-NUM>
        </CLIENT-INTERFACE-CONFIG>

        <SERVER-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth2 </INTERFACE-NUM>
        </SERVER-INTERFACE-CONFIG>

    </NS-BOOT STRAP>
</NS-PRE-BOOT-CONFIG>
```

GCP ポータルで VM インスタンスを作成した後、ネットワークインターフェイスのプロパティを次のように確認できます。

1. カスタムブートストラップ情報を指定して、作成したインスタンスを選択します。
2. ネットワークインターフェイスのプロパティに移動し、NIC の詳細を次のように確認します。

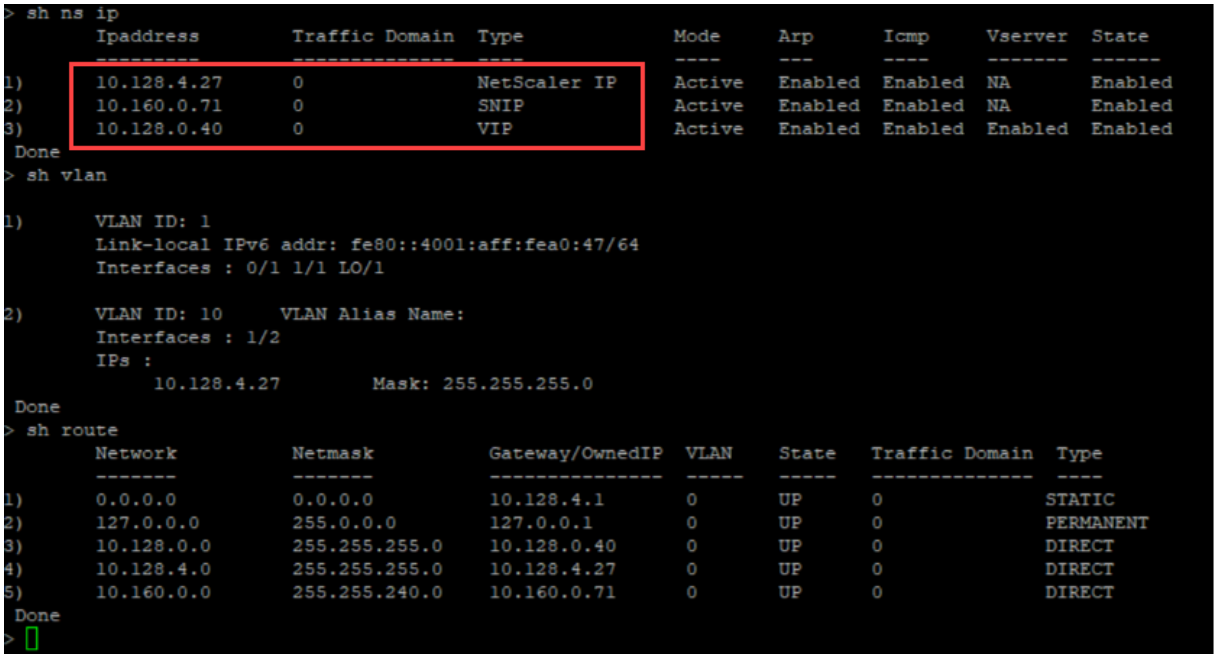
Network interfaces

Name	Network	Subnetwork	Primary internal IP	Alias IP ranges	External IP	Network Tier	IP forwarding	Network details
nic0	default	default	10.160.0.71	–	35.244.56.180 (ephemeral)	Premium	Off	View details
nic1	vsk-vpc-network-1	asia-south1-subnet-1	10.128.0.40	–	35.244.40.113 (ephemeral)	Premium		View details
nic2	vsk-vpc-network-2	asia-south1-subnet-5	10.128.4.27	–	34.93.241.147 (ephemeral)	Premium		View details

Public DNS PTR Record

None

ADC CLI で`show nsip`コマンドを実行し、ADC アプライアンスの初回起動時に NetScalerVPX インスタンスに適用されるネットワークインターフェイスを確認できます。



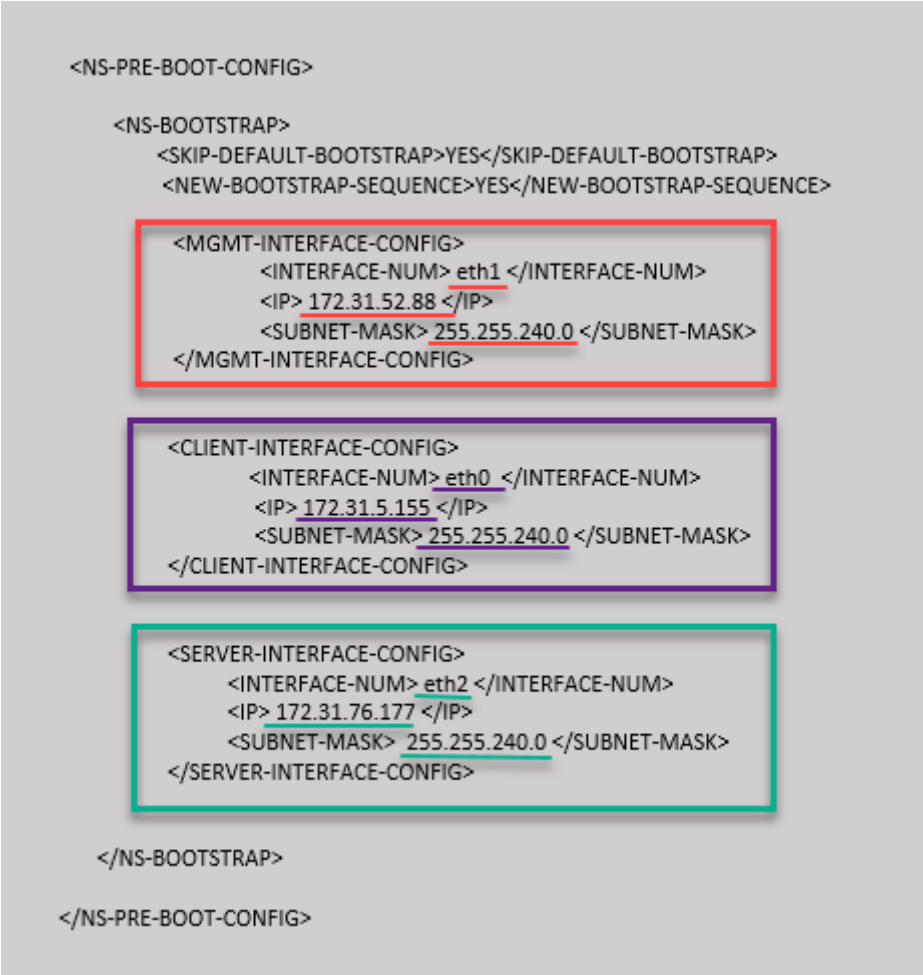
方法 **2:** インターフェイス、**IP** アドレス、およびサブネットマスクを指定してカスタムブートストラップ

管理インターフェイス、クライアント向けインターフェイス、およびサーバ向けインターフェイスと IP アドレスとサブネットマスクを指定します。

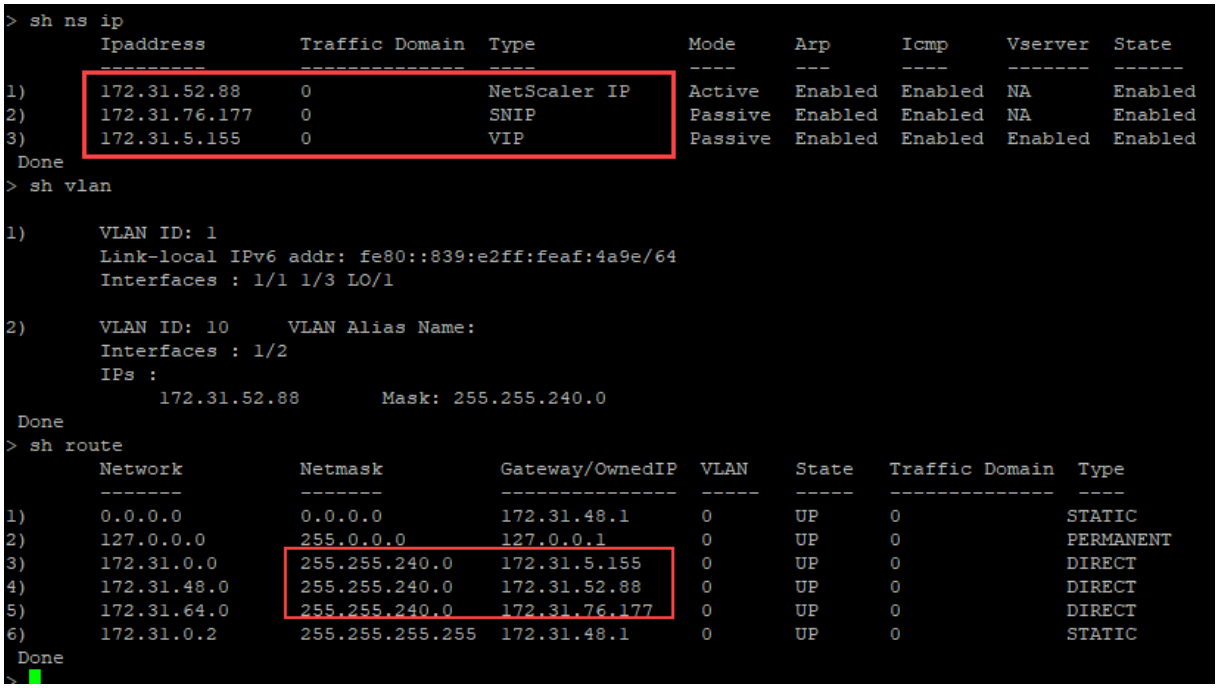
AWS のカスタムブートストラップの例

次の例では、デフォルトのブートストラップをスキップして、NetScaler アプライアンスの新しいブートストラップシーケンスを実行します。新しいブートストラップシーケンスでは、次の詳細を指定します。

- 管理インターフェイス：インターフェイス-Eth1、NSIP-172.31.52.88、およびサブネットマスク-255.255.240.0
- クライアント側インターフェイス：インターフェイス-Eth0、VIP-172.31.5.155、およびサブネットマスク-255.255.240.0。
- サーバー側インターフェイス：インターフェイス-Eth2、SNIP-172.31.76.177、サブネットマスク-255.255.240.0。



ADC CLI で`show nsip`コマンドを実行し、`<NS-BOOTSTRAP>`セクションで指定された新しいブートストラップシーケンスが適用されていることを確認できます。「show route」コマンドを実行して、サブネットマスクを確認できます。



```
> sh ns ip
        Ipaddress        Traffic Domain  Type             Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----             ----     ---      ----     -------  -----
1)      172.31.52.88     0               NetScaler IP     Active   Enabled  Enabled  NA       Enabled
2)      172.31.76.177    0               SNIP             Passive  Enabled  Enabled  NA       Enabled
3)      172.31.5.155     0               VIP              Passive  Enabled  Enabled  Enabled  Enabled
 Done
> sh vlan

1)      VLAN ID: 1
        Link-local IPv6 addr: fe80::839:e2ff:feaf:4a9e/64
        Interfaces : 1/1 1/3 LO/1

2)      VLAN ID: 10     VLAN Alias Name:
        Interfaces : 1/2
        IPs :
             172.31.52.88       Mask: 255.255.240.0
 Done
> sh route
        Network          Netmask          Gateway/OwnedIP  VLAN    State   Traffic Domain  Type
        -------          -------          ---------------  -----   -----   --------------  ----
1)      0.0.0.0          0.0.0.0          172.31.48.1      0       UP      0               STATIC
2)      127.0.0.0        255.0.0.0        127.0.0.1        0       UP      0               PERMANENT
3)      172.31.0.0       255.255.240.0    172.31.5.155     0       UP      0               DIRECT
4)      172.31.48.0      255.255.240.0    172.31.52.88     0       UP      0               DIRECT
5)      172.31.64.0      255.255.240.0    172.31.76.177    0       UP      0               DIRECT
6)      172.31.0.2       255.255.255.255  172.31.48.1      0       UP      0               STATIC
 Done
>
```

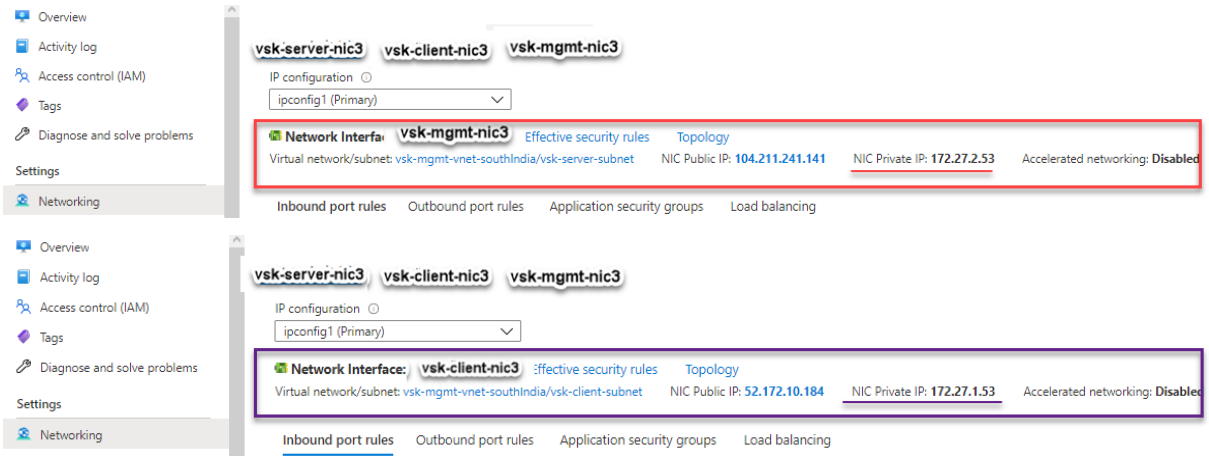
Azure のカスタムブートストラップの例

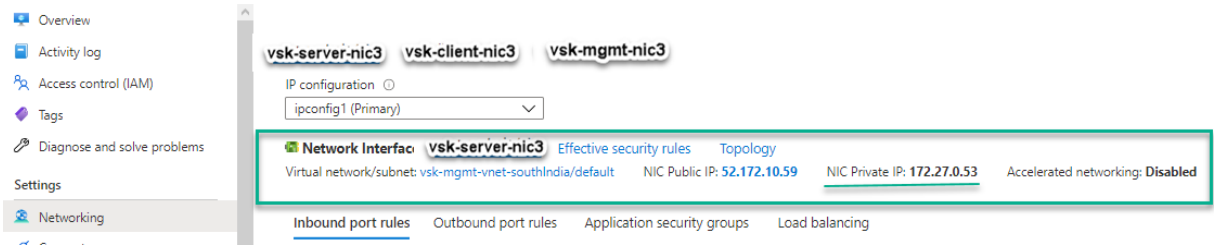
次の例では、ADC の新しいブートストラップシーケンスが記述され、デフォルトのブートストラップがスキップされます。インターフェイスの詳細と IP アドレスとサブネットマスクを次のように指定します。

- 管理インターフェイス（eth2）、NSIP（172.27.2.53）、およびサブネットマスク（255.255.255.0）
- クライアント側インターフェイス（eth1）、VIP（172.27.1.53）、およびサブネットマスク（255.255.255.0）
- サーバー側インターフェイス (eth0)、SNIP (172.27.0.53)、およびサブネットマスク (255.255.255.0)


```
<NS-PRE-BOOT-CONFIG>

    <NS-BOOTSTRAP>
        <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
        <NEW-BOOTSTRAP-SEQUENCE>YES</NEW-BOOTSTRAP-SEQUENCE>

        <MGMT-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth2 </INTERFACE-NUM>
            <IP> 172.27.2.53 </IP>
            <SUBNET-MASK> 255.255.255.0 </SUBNET-MASK>
        </MGMT-INTERFACE-CONFIG>

        <CLIENT-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth1 </INTERFACE-NUM>
            <IP> 172.27.1.53 </IP>
            <SUBNET-MASK> 255.255.255.0 </SUBNET-MASK>
            </CLIENT-INTERFACE-CONFIG>

        <SERVER-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth0 </INTERFACE-NUM>
            <IP> 172.27.0.53 </IP>
            <SUBNET-MASK> 255.255.255.0 </SUBNET-MASK>
        </SERVER-INTERFACE-CONFIG>

    </NS-BOOTSTRAP>

</NS-PRE-BOOT-CONFIG>
```

NetScaler VPX インスタンスが 3 つのネットワークインターフェイスで作成されていることがわかります。**Azure Portal > VM** インスタンス **>** ネットワークに移動し、次の図に示すように 3 つの NIC のネットワークプロパティを確認します。





ADC CLI で`show nsip`コマンドを実行し、&`lt`;NS-BOOTSTRAP&`gt`;セクションで指定された新しいブートストラップシーケンスが適用されていることを確認できます。「show route」コマンドを実行して、サブネットマスクを確認できます。

```
> sh ns ip
        Ipaddress        Traffic Domain  Type             Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----             ----     ---      ----     -------  ------
1)      172.27.2.53      0               NetScaler IP     Active   Enabled  Enabled  NA       Enabled
2)      172.27.0.53      0               SNIP             Active   Enabled  Enabled  NA       Enabled
3)      172.27.1.53      0               VIP              Active   Enabled  Enabled  Enabled  Enabled
 Done
> sh vlan

1)      VLAN ID: 1
        Link-local IPv6 addr: fe80::20d:3aff:fec9:c26c/64
        Interfaces : 0/1 1/1 LO/1

2)      VLAN ID: 10     VLAN Alias Name:
        Interfaces : 1/2
        IPs :
             172.27.2.53        Mask: 255.255.255.0
 Done
> sh route
        Network          Netmask          Gateway/OwnedIP  VLAN    State   Traffic Domain  Type
        -------          -------          ---------------  -----   -----   --------------  ----
1)      0.0.0.0          0.0.0.0          172.27.2.1       0       UP      0               STATIC
2)      127.0.0.0        255.0.0.0        127.0.0.1        0       UP      0               PERMANENT
3)      172.27.0.0       255.255.255.0    172.27.0.53      0       UP      0               DIRECT
4)      172.27.1.0       255.255.255.0    172.27.1.53      0       UP      0               DIRECT
5)      172.27.2.0       255.255.255.0    172.27.2.53      0       UP      0               DIRECT
6)      169.254.0.0      255.255.0.0      172.27.0.1       0       UP      0               STATIC
7)      168.63.129.16    255.255.255.255  172.27.0.1       0       UP      0               STATIC
8)      169.254.169.254  255.255.255.255  172.27.0.1       0       UP      0               STATIC
 Done
```

GCP のカスタムブートストラップの例

次の例では、ADC の新しいブートストラップシーケンスが記述され、デフォルトのブートストラップがスキップされます。インターフェイスの詳細と IP アドレスとサブネットマスクを次のように指定します。

- 管理インターフェイス（eth2）、NSIP（10.128.4.31）、およびサブネットマスク（255.255.255.0）
- クライアント側インターフェイス（eth1）、VIP（10.128.0.43）、およびサブネットマスク（255.255.255.0）
- サーバ側インターフェイス（eth0）、SNIP（10.160.0.75）、およびサブネットマスク（255.255.255.0）


```
<NS-PRE-BOOT-CONFIG>
    <NS-BOOTSTRAP>
        <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
        <NEW-BOOTSTRAP-SEQUENCE>YES</NEW-BOOTSTRAP-SEQUENCE>
        <MGMT-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth2 </INTERFACE-NUM>
            <IP> 10.128.4.31 </IP>
            <SUBNET-MASK> 255.255.255.0 </SUBNET-MASK>
        </MGMT-INTERFACE-CONFIG>
        <CLIENT-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth1 </INTERFACE-NUM>
            <IP> 10.128.0.43 </IP>
            <SUBNET-MASK> 255.255.255.0 </SUBNET-MASK>
        </CLIENT-INTERFACE-CONFIG>
        <SERVER-INTERFACE-CONFIG>
            <INTERFACE-NUM> eth0 </INTERFACE-NUM>
            <IP> 10.160.0.75 </IP>
            <SUBNET-MASK> 255.255.255.0 </SUBNET-MASK>
        </SERVER-INTERFACE-CONFIG>
    </NS-BOOTSTRAP>
</NS-PRE-BOOT-CONFIG>
```

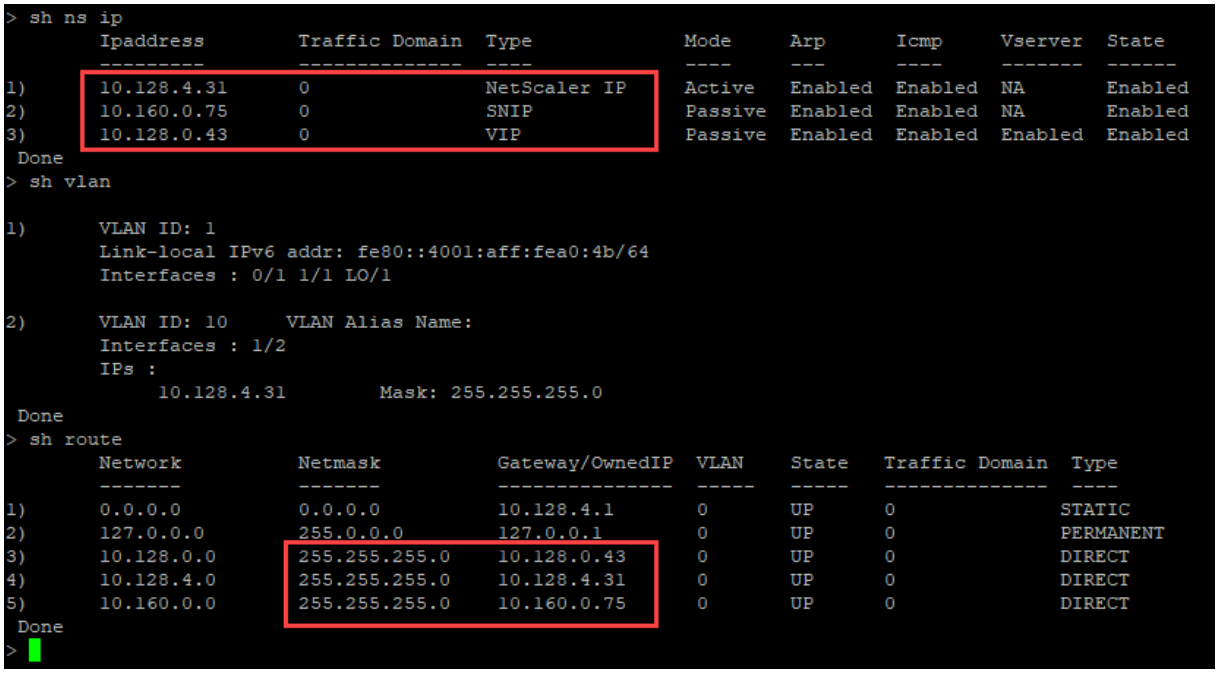
カスタムブートストラップを使用して GCP ポータルで VM インスタンスを作成した後、ネットワークインターフェイスのプロパティを次のように確認できます。

1. カスタムブートストラップ情報を指定して、作成したインスタンスを選択します。
2. ネットワークインターフェイスのプロパティに移動し、NIC の詳細を次のように確認します。

Network interfaces

Name	Network	Subnetwork	Primary internal IP	Alias IP ranges	External IP	Network Tier	IP forwarding	Network details
nic0	default	default	vsk-defnw-st-ip1 (10.160.0.75)	–	34.93.216.90 (ephemeral)	Premium	Off	View details
nic1	vsk-vpc-network-1	asia-south1-subnet-1	vsk-vpc-nw1-st-ip1 (10.128.0.43)	–	35.244.40.113 (ephemeral)	Premium		View details
nic2	vsk-vpc-network-2	asia-south1-subnet-5	vsk-nw2-st-ip-1 (10.128.4.31)	–	34.93.202.214 (ephemeral)	Premium		View details

ADC CLI で`show nsip`コマンドを実行し、`<NS-BOOTSTRAP>`セクションで指定された新しいブートストラップシーケンスが適用されていることを確認できます。「show route」コマンドを実行して、サブネットマスクを確認できます。



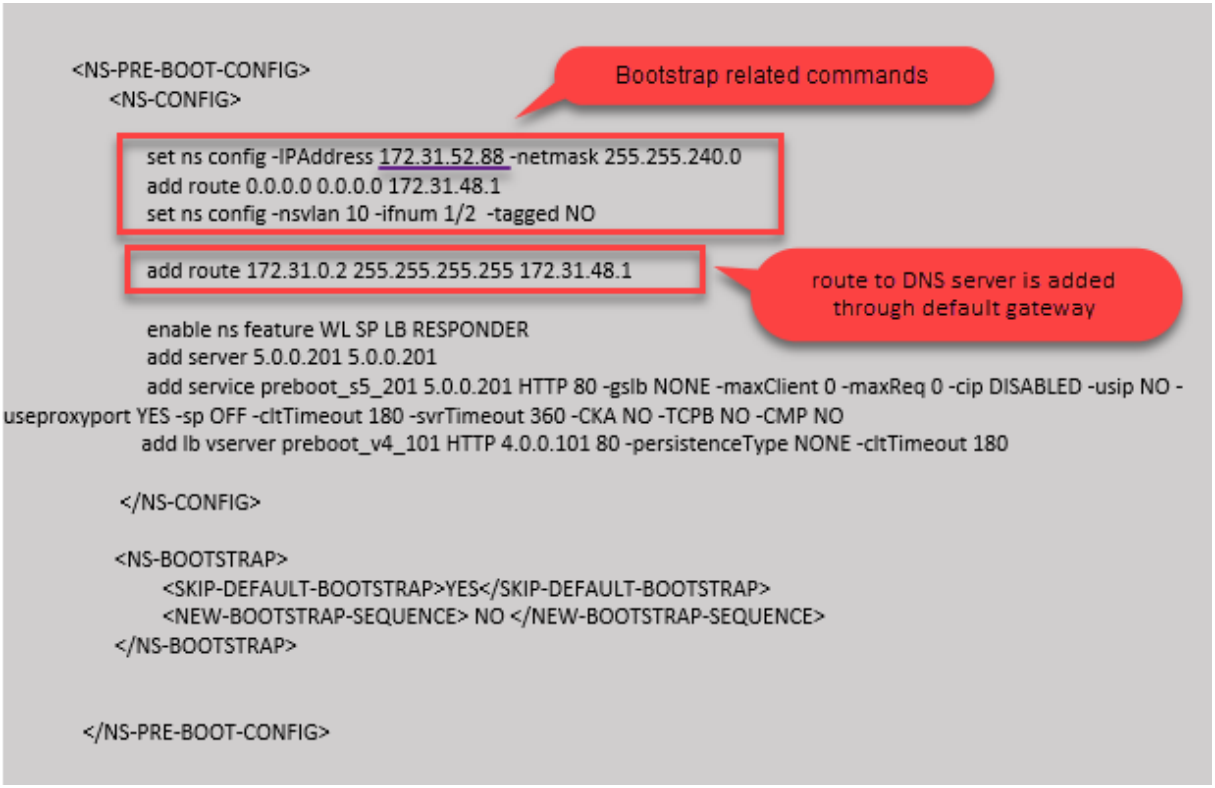
方法 **3: <NS-CONFIG>** セクションにブートストラップ関連のコマンドを指定して、カスタムブートストラップ

ブートストラップ関連のコマンドについては、<NS-CONFIG>セクションを参照してください。<NS-BOOTSTRAP>セクションでブートストラップコマンドを実行するには、<NS-CONFIG>セクションで<NEW-BOOTSTRAP-SEQUENCE>を「No」と指定する必要があります。NSIP、デフォルトルート、および NSVLAN を割り当てるコマンドも指定する必要があります。さらに、使用するクラウドに関連するコマンドも提供します。

カスタムブートストラップを提供する前に、クラウドインフラストラクチャが特定のインターフェイス構成をサポートしていることを確認してください。

AWS のカスタムブートストラップの例

この例では、ブートストラップ関連のコマンドを<NS-CONFIG>セクションで提供しています。<NS-BOOTSTRAP>セクションは、デフォルトのブートストラップがスキップされ、<NS-CONFIG>セクションで提供されるカスタムブートストラップ情報が実行されることを示します。NSIP の作成、デフォルトルートの追加、および NSVLAN の追加を行うコマンドも指定する必要があります。

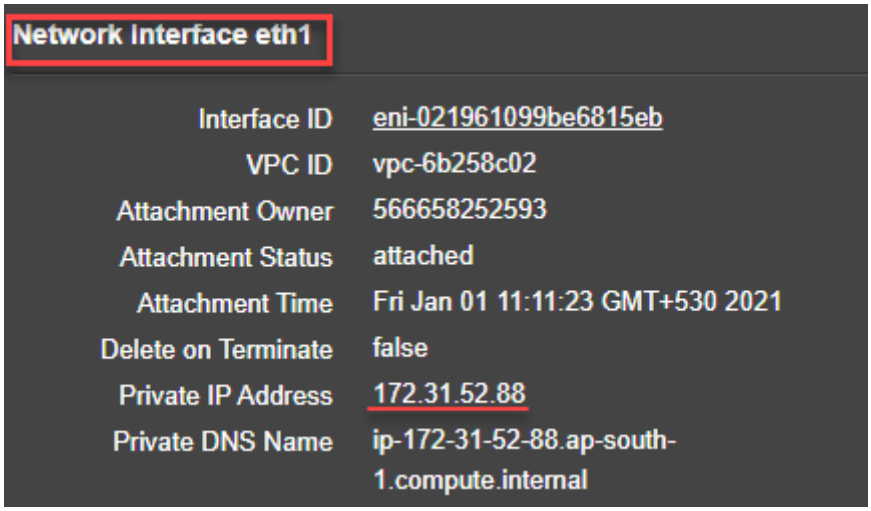


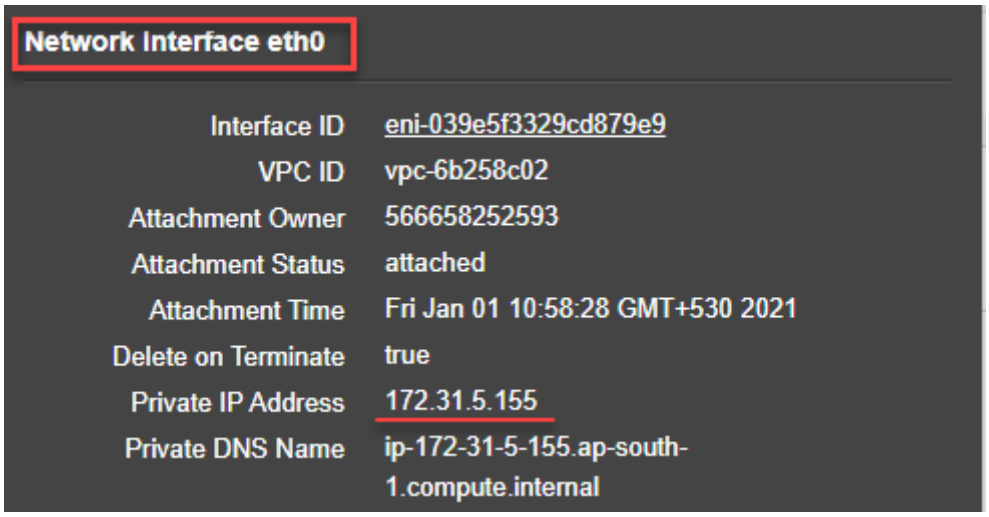
前のスクリーンショットに示した設定をここからコピーできます。

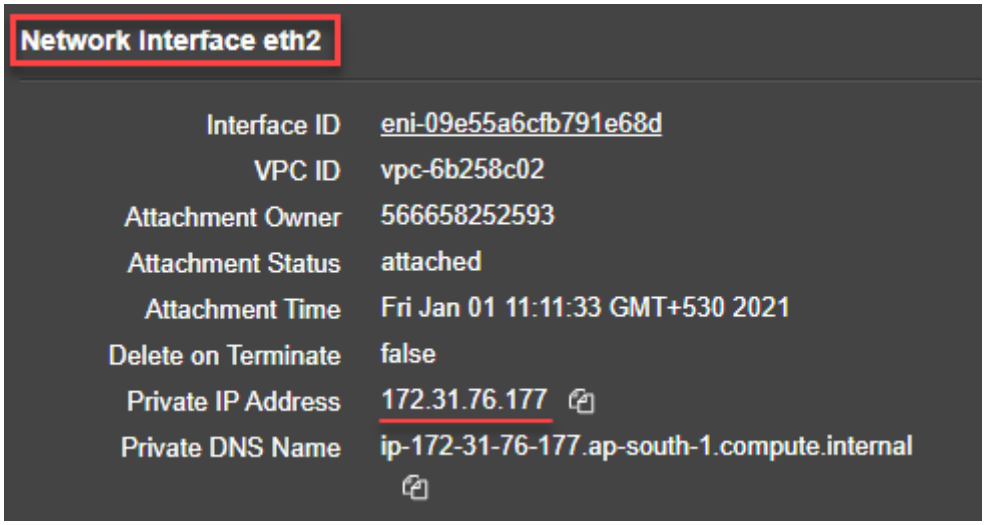
```
<NS-PRE-BOOT-CONFIG>
    <NS-CONFIG>

        set ns config -IPAddress 172.31.52.88 -netmask 255.255.240.0
        add route 0.0.0.0 0.0.0.0 172.31.48.1
        set ns config -nsvlan 10 -ifnum 1/2  -tagged NO
        add route 172.31.0.2 255.255.255.255 172.31.48.1

        enable ns feature WL SP LB RESPONDER
        add server 5.0.0.201 5.0.0.201
        add service preboot_s5_201 5.0.0.201 HTTP 80 -gslb NONE -
           maxClient 0 -maxReq 0 -cip DISABLED -usip NO -
           useproxyport YES -sp OFF -cltTimeout 180 -svrTimeout 360 -
           CKA NO -TCPB NO -CMP NO
        add lb vserver preboot_v4_101 HTTP 4.0.0.101 80 -
           persistenceType NONE -cltTimeout 180

    </NS-CONFIG>

    <NS-BOOTSTRAP>
     <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
     <NEW-BOOTSTRAP-SEQUENCE> NO </NEW-BOOTSTRAP-SEQUENCE>
    </NS-BOOTSTRAP>


</NS-PRE-BOOT-CONFIG>
```


VM インスタンスの作成後、AWS ポータルで、ネットワークインターフェイスのプロパティを次のように確認できます。

1. [**AWS Portal] > [EC2** インスタンス] に移動し、カスタムブートストラップ情報を指定して作成したインスタンスを選択します。
2. [説明] タブでは、次の図に示すように、各ネットワークインタフェースのプロパティを確認できます。







ADC CLI で`show nsip`コマンドを実行し、ADC アプライアンスの初回起動時に NetScalerVPX インスタンスに適用されるネットワークインターフェイスを確認できます。

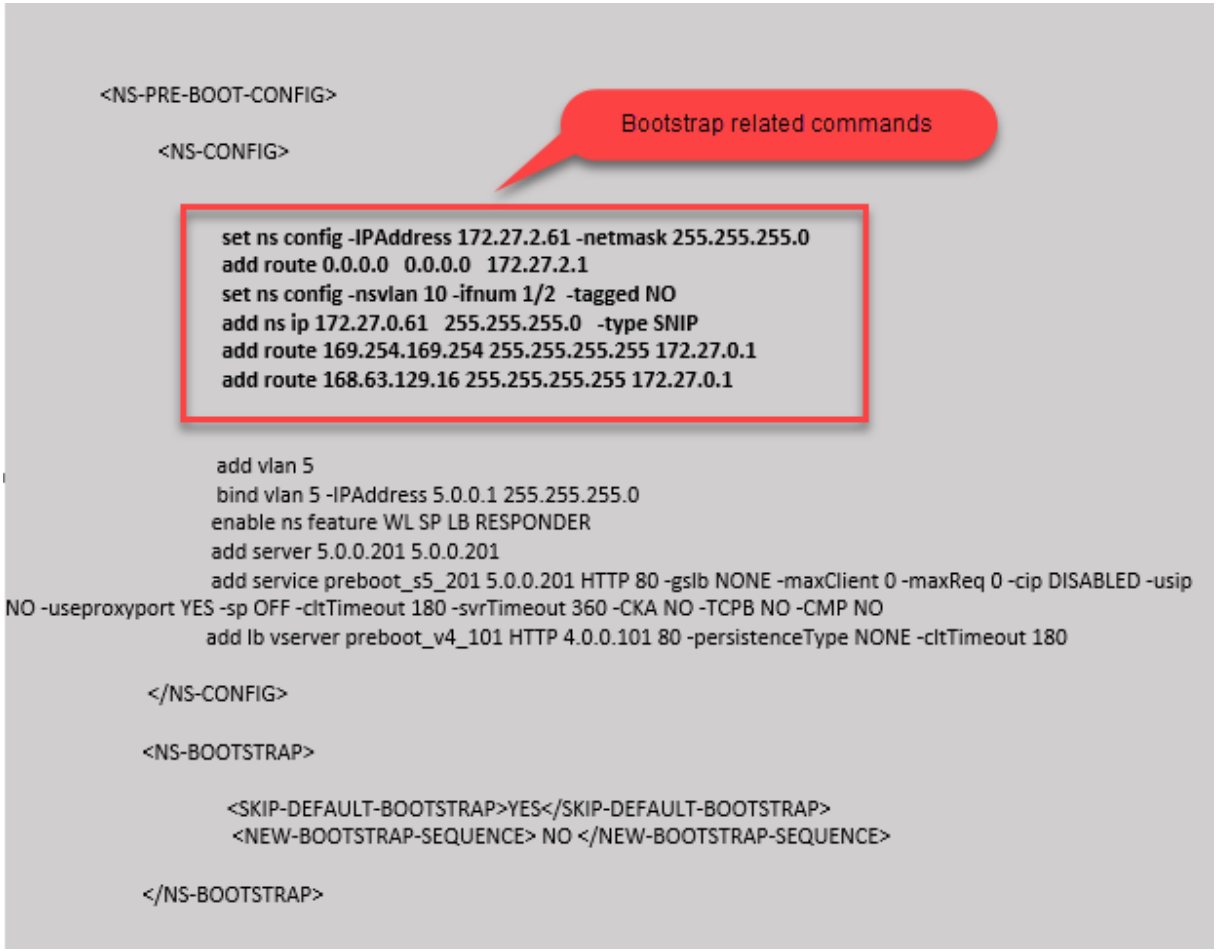
```
> sh ns ip
        Ipaddress        Traffic Domain  Type             Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----             ----     ---      ----     -------  ------
1)      172.31.52.88     0               NetScaler IP     Active   Enabled  Enabled  NA       Enabled
2)      4.0.0.101        0               VIP              Active   Enabled  Enabled  Enabled  Enabled
 Done
> sh vlan

1)      VLAN ID: 1
        Link-local IPv6 addr: fe80::839:e2ff:feaf:4a9e/64
        Interfaces : 1/1 1/3 LO/1

2)      VLAN ID: 10     VLAN Alias Name:
        Interfaces : 1/2
        IPs :
             172.31.52.88       Mask: 255.255.240.0
 Done
> sh route
        Network          Netmask          Gateway/OwnedIP  VLAN    State   Traffic Domain  Type
        -------          -------          ---------------  -----   -----   --------------  ----
1)      0.0.0.0          0.0.0.0          172.31.48.1      0       UP      0               STATIC
2)      127.0.0.0        255.0.0.0        127.0.0.1        0       UP      0               PERMANENT
3)      172.31.48.0      255.255.240.0    172.31.52.88     0       UP      0               DIRECT
4)      172.31.0.2       255.255.255.255  172.31.48.1      0       UP      0               STATIC
 Done
>
```

Azure のカスタムブートストラップの例

この例では、ブートストラップ関連のコマンドを<NS-CONFIG>セクションで提供しています。<NS-BOOTSTRAP>セクションは、デフォルトのブートストラップがスキップされ、<NS-CONFIG>セクションで提供されるカスタムブートストラップ情報が実行されることを示します。

> 注
>
> Azure クラウドの場合、インスタンスメタデータサーバー (IMDS) と DNS サーバーはプライマリインターフェイス (Eth0) を介してのみアクセスできます。したがって、Eth0 インターフェイスが管理インターフェイス

（NSIP）として使用されない場合、Eth0 インターフェイスは、少なくとも IMDS または DNS アクセスを動作させるには、SNIP として設定する必要があります。Eth0 のゲートウェイを経由する IMDS エンドポイント（169.254.169.254）および DNS エンドポイント（168.63.129.16）へのルートも追加する必要があります。

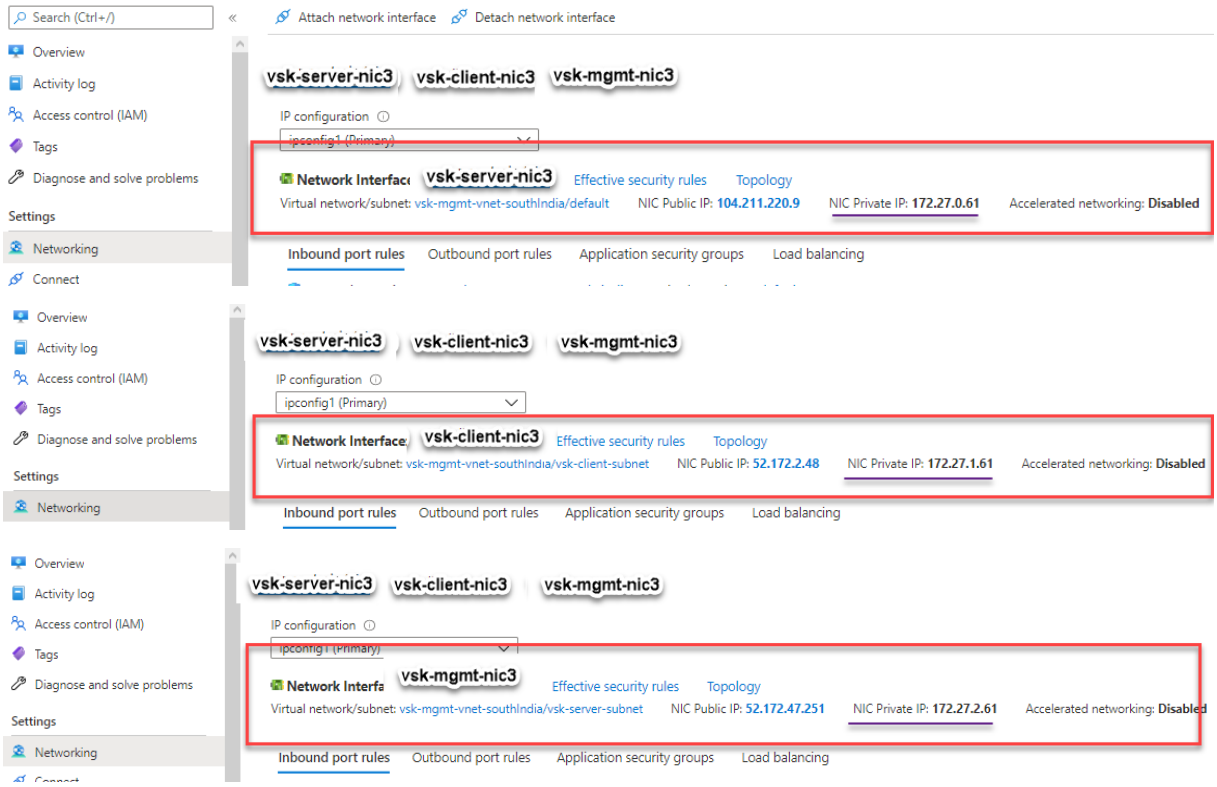



```
<NS-PRE-BOOT-CONFIG>

   <NS-CONFIG>

        set ns config -IPAddress 172.27.2.61 -netmask 255.255.255.0
        add route 0.0.0.0   0.0.0.0   172.27.2.1
        set ns config -nsvlan 10 -ifnum 1/2  -tagged NO
        add ns ip 172.27.0.61   255.255.255.0   -type SNIP
        add route 169.254.169.254 255.255.255.255 172.27.0.1
        add route 168.63.129.16 255.255.255.255 172.27.0.1

        add vlan 5
        bind vlan 5 -IPAddress 5.0.0.1 255.255.255.0
        enable ns feature WL SP LB RESPONDER
        add server 5.0.0.201 5.0.0.201
        add service preboot_s5_201 5.0.0.201 HTTP 80 -gslb NONE -
           maxClient 0 -maxReq 0 -cip DISABLED -usip NO -useproxyport
            YES -sp OFF -cltTimeout 180 -svrTimeout 360 -CKA NO -TCPB
```

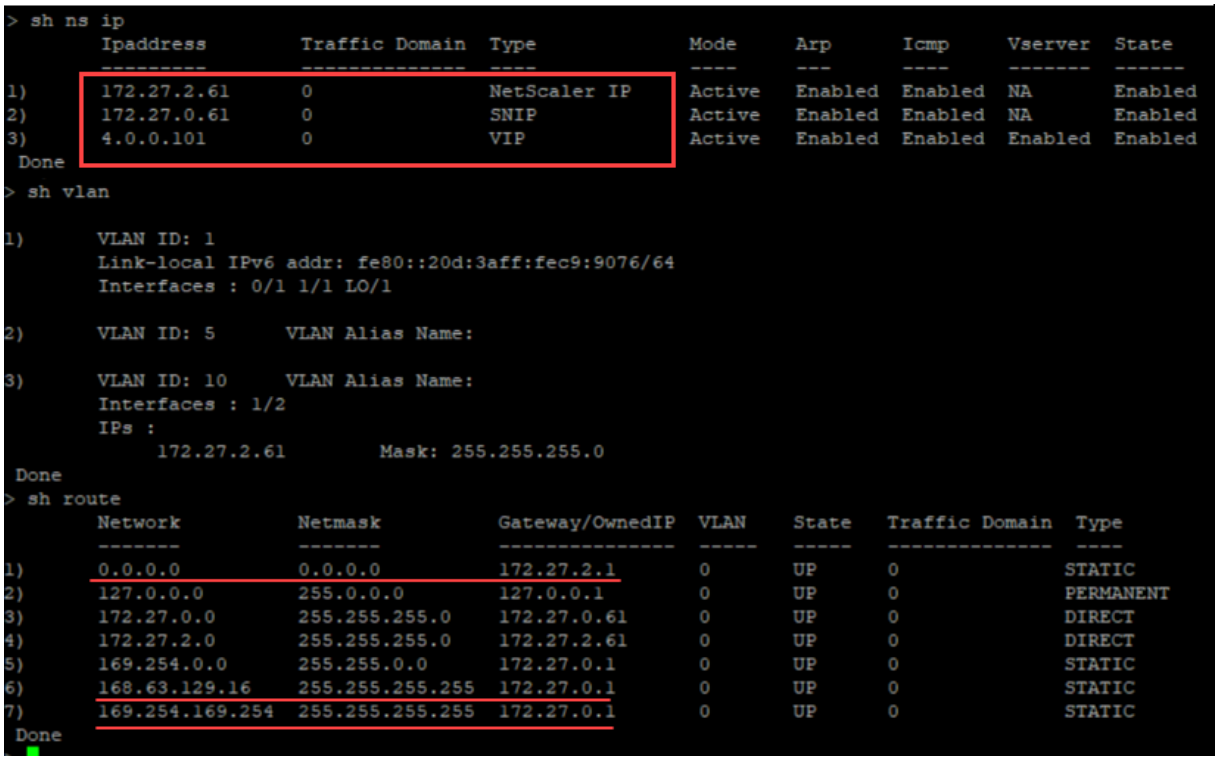


```
                NO -CMP NO
17          add lb vserver preboot_v4_101 HTTP 4.0.0.101 80 -
                persistenceType NONE -cltTimeout 180
18
19      </NS-CONFIG>
20
21      <NS-BOOTSTRAP>
22
23      <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
24      <NEW-BOOTSTRAP-SEQUENCE> NO </NEW-BOOTSTRAP-SEQUENCE>
25
26      </NS-BOOTSTRAP>
27
28  </NS-PRE-BOOT-CONFIG>
```

NetScaler VPX インスタンスが 3 つのネットワークインターフェイスで作成されていることがわかります。**Azure Portal > VM** インスタンス **>** ネットワークに移動し、次の図に示すように 3 つの NIC のネットワークプロパティを確認します。

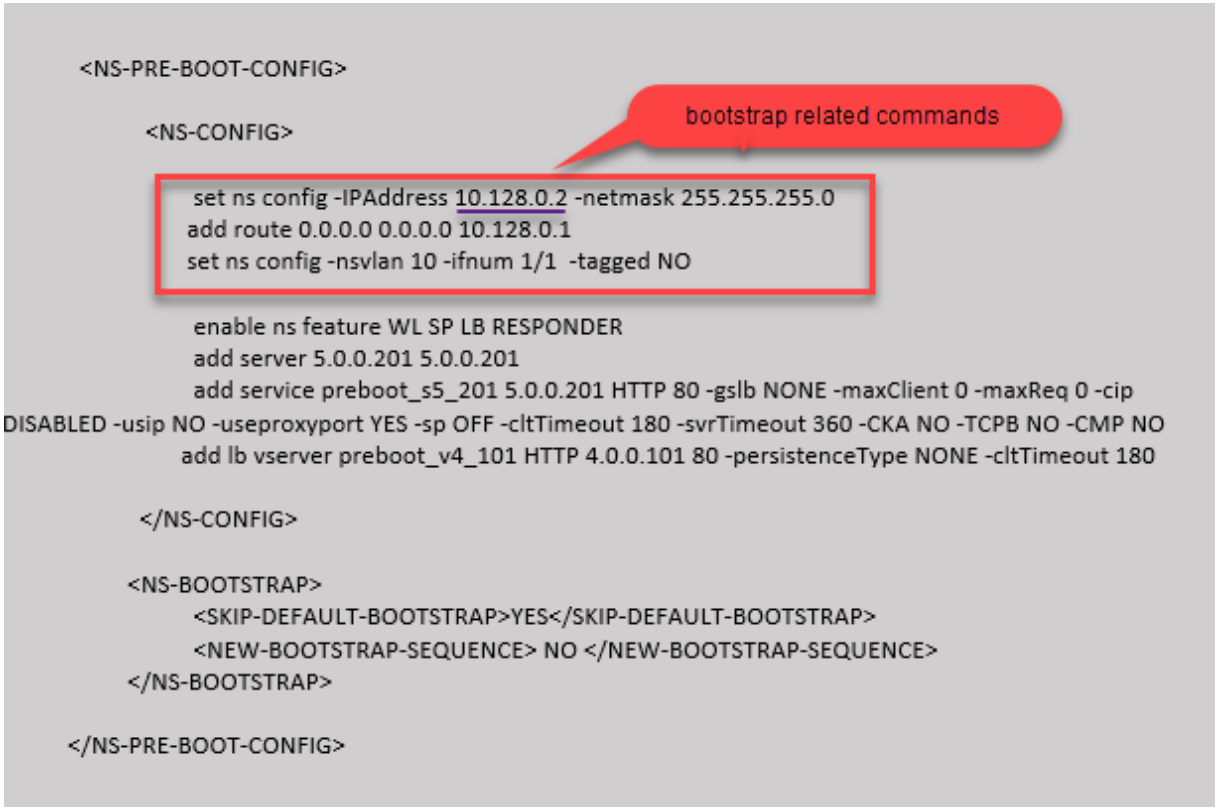


ADC CLI で`show nsip`コマンドを実行し、`<NS-BOOTSTRAP>`セクションで指定された新しいブートストラップシーケンスが適用されていることを確認できます。「show route」コマンドを実行して、サブネットマスクを確認できます。



GCP のカスタムブートストラップの例

この例では、ブートストラップ関連のコマンドを<NS-CONFIG>セクションで提供しています。<NS-BOOTSTRAP>セクションは、デフォルトのブートストラップがスキップされ、<NS-CONFIG>セクションで提供されるカスタムブートストラップ情報が適用されることを示します。



前のスクリーンショットに示した設定をここからコピーできます。

```
<NS-PRE-BOOT-CONFIG>

    <NS-CONFIG>

        set ns config -IPAddress 10.128.0.2 -netmask 255.255.255.0
        add route 0.0.0.0 0.0.0.0 10.128.0.1
        set ns config -nsvlan 10 -ifnum 1/1  -tagged NO

        enable ns feature WL SP LB RESPONDER
        add server 5.0.0.201 5.0.0.201
        add service preboot_s5_201 5.0.0.201 HTTP 80 -gslb NONE -
            maxClient 0 -maxReq 0 -cip DISABLED -usip NO -useproxyport
             YES -sp OFF -cltTimeout 180 -svrTimeout 360 -CKA NO -TCPB
             NO -CMP NO
        add lb vserver preboot_v4_101 HTTP 4.0.0.101 80 -
            persistenceType NONE -cltTimeout 180

    </NS-CONFIG>

    <NS-BOOTSTRAP>
        <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
        <NEW-BOOTSTRAP-SEQUENCE> NO </NEW-BOOTSTRAP-SEQUENCE>
    </NS-BOOTSTRAP>

</NS-PRE-BOOT-CONFIG>
```


カスタムブートストラップを使用して GCP ポータルで VM インスタンスを作成した後、ネットワークインターフェイスのプロパティを次のように確認できます。

1. カスタムブートストラップ情報を指定して、作成したインスタンスを選択します。
2. [Network Interface] プロパティに移動し、図に示すように NIC の詳細を確認します。

Network interfaces

Name	Network	Subnetwork	Primary internal IP	Alias IP ranges	External IP
nic0	default	default	10.160.0.74	–	34.93.9.79 (ephemeral)
nic1	vsk-vpc-network-1	asia-south1-subnet-1	asia-south1-subnet1-10-128-0-2 (10.128.0.2)	–	34.93.245.110 (ephemeral)
nic2	vsk-vpc-network-2	asia-south1-subnet-5	10.128.4.30	–	34.93.146.248 (ephemeral)

ADC CLI で**show nsip**コマンドを実行し、**ADC** アプライアンスの最初の起動時に前の<NS-CONFIG>セクションで説明した設定が適用されていることを確認できます。

```
> sh ns ip
        Ipaddress        Traffic Domain  Type             Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----             ----     ---      ----     -------  ------
1)      10.128.0.2       0               NetScaler IP     Active   Enabled  Enabled  NA       Enabled
2)      4.0.0.101        0               VIP              Active   Enabled  Enabled  Enabled  Enabled
 Done
> sh vlan

1)      VLAN ID: 1
        Link-local IPv6 addr: fe80::4001:aff:fea0:4a/64
        Interfaces : 0/1 1/2 LO/1

2)      VLAN ID: 10     VLAN Alias Name:
        Interfaces : 1/1
        IPs :
             10.128.0.2         Mask: 255.255.255.0
 Done
> sh route
        Network          Netmask          Gateway/OwnedIP  VLAN    State   Traffic Domain  Type
        -------          -------          ---------------  -----   -----   --------------  ----
1)      0.0.0.0          0.0.0.0          10.128.0.1       0       UP      0               STATIC
2)      127.0.0.0        255.0.0.0        127.0.0.1        0       UP      0               PERMANENT
3)      10.128.0.0       255.255.255.0    10.128.0.2       0       UP      0               DIRECT
 Done
```

AWS および Azure での NIC のアタッチとデタッチによる影響

AWS と Azure には、ネットワークインターフェイスをインスタンスにアタッチし、ネットワークインターフェイスをインスタンスからデタッチするオプションが用意されています。インターフェイスをアタッチまたはデタッチすると、インターフェイスの位置が変わることがあります。そのため、Citrix では NetScaler VPX インスタンスからインターフェイスをデタッチしないことをお勧めします。カスタムブートストラップが構成されているときにインターフェイスをデタッチまたは接続すると、NetScaler VPX インスタンスは、管理インターフェイスの位置で新しく使用可能なインターフェイスのプライマリ IP を NSIP として再割り当てします。デタッチしたインターフェイスの後に使用可能なインターフェイスがない場合は、最初のインターフェイスが NetScaler VPX インスタンスの管理インターフェイスになります。

たとえば、NetScaler VPX インスタンスは、Eth0（SNIP）、Eth1（NSIP）、および Eth2（VIP）の 3 つのインターフェイスで起動されます。管理インターフェイスであるインスタンスから Eth1 インターフェイスをデタッチすると、

ADC は次の使用可能なインターフェイス（Eth2）を管理インターフェイスとして設定します。そのため、NetScaler VPX インスタンスには引き続き Eth2 インターフェイスのプライマリ IP を介してアクセスされます。Eth2 も使用できない場合は、残りのインターフェイス（Eth0）が管理インターフェイスになります。そのため、NetScaler VPX インスタンスには引き続きアクセスできます。

Eth0 (SNIP)、Eth1 (VIP)、Eth2 (NSIP) の異なるインターフェイスの割り当てを考えてみましょう。Eth2（NSIP）をデタッチすると、Eth2 の後に新しいインターフェイスが使用できないため、最初のインターフェイス（Eth0）が管理インターフェイスになります。

パブリッククラウドプラットフォームでの **SSL-TPS** パフォーマンスを向上させる

October 17, 2024

パケットエンジン (PE) の重みを均等に分散することで、AWS と GCP クラウドで SSL-TPS のパフォーマンスを向上させることができます。この機能を有効にすると、HTTP スループットが約 10 ～12% わずかに低下する可能性があります。

AWS および GCP クラウドでは、10～16 個の vCPU を持つ NetScaler ADC VPX インスタンスでは、PE の重みがデフォルトで均等に分散されるため、パフォーマンスの向上は見られません。

> 注
>
> Azure クラウドでは、PE の重みはデフォルトで均等に分散されます。この機能によって Azure インスタンスのパフォーマンスは向上しません。

NetScaler CLI を使用して **PE** モードを構成する

PE モードを設定したら、設定変更を有効にするためにシステムをリブートする必要があります。

コマンドプロンプトで入力します：

```
1   set cpuparam pemode [CPUBOUND | Default]
```

PE モードが CPUBOUND に設定されている場合、PE の重みは均等に分散されます。PE モードが DEFAULT に設定されている場合、PE の重みはデフォルト値に設定されます。PE モードが DEFAULT に設定されている場合、PE の重みはデフォルト値に設定されます。

> 注
>
> このコマンドはノード固有です。高可用性またはクラスタセットアップでは、各ノードでコマンドを実行する必要があります。CLIP でコマンドを実行すると、次のエラーが発生します。`CLIP では操作は許可されていません`

設定されている PE モードの状態を表示するには、次のコマンドを実行します。


```
show cpuparam
```

例

```
> show cpuparam
    Pemode:  CPUBOUND
 Done
```

クラウド内の **NetScaler ADC** アプライアンスの初回起動時に **PE** モード構成を適用する

クラウド内の NetScaler ADC アプライアンスの初回起動時に PE モード構成を適用するには、カスタムスクリプトを使用して`/nsconfig/.cpubound.conf`ファイルを作成する必要があります。詳細については、「クラウド内の NetScaler アプライアンスの初回起動時に NetScaler VPX 構成を適用する」を参照してください。

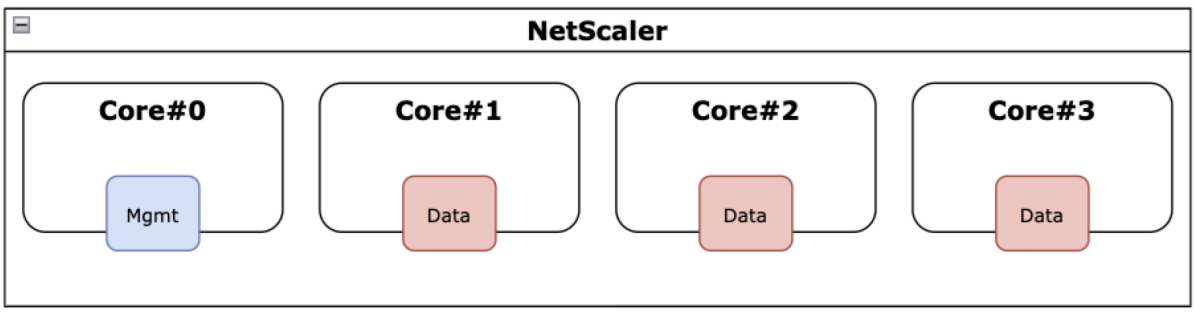
パブリッククラウド上の **NetScaler VPX** 同時マルチスレッドを構成する

October 17, 2024

NetScaler は、管理とデータプレーン機能にさまざまな専用コアを使用します。通常、1 つのコアが管理プレーン機能に割り当てられます。使用可能な残りのコアはデータプレーン機能に割り当てられます。

以下の画像は、4 コアの NetScaler VPX を簡略化した図を示しています。

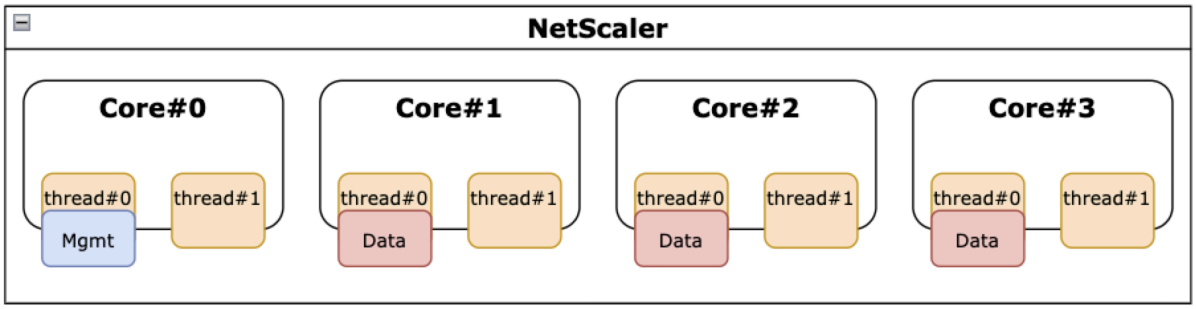
図 1: インライン展開 図 1: 4 コアシステムでの NetScaler 管理とデータプレーンワークロード



上の図は、使用可能なコア全体にわたる NetScaler 機能の分布を示していますが、基盤となるハードウェアを必ずしも正確に表しているわけではありません。最新の x86 CPU のほとんどは、Intel ハイパースレッディング（HT）または AMD 同時マルチスレッディング（SMT）として商業的に知られている機能により、物理コアあたり 2 つの論理コアを備えています。

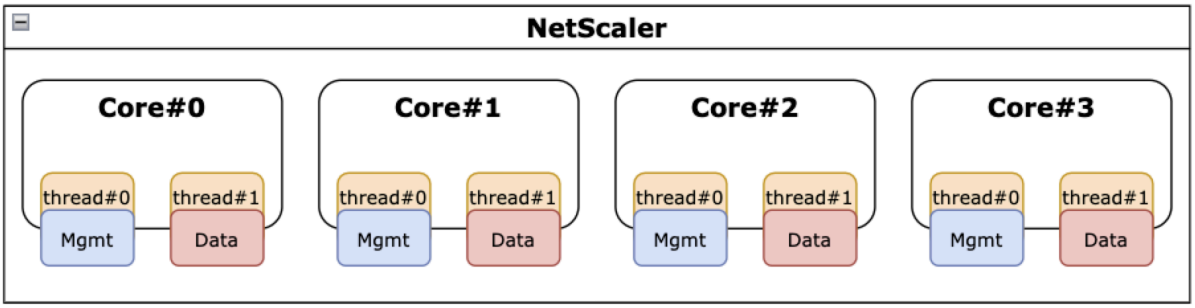
次の画像は、SMT が無効になっている最新の CPU 上で実行されている NetScaler VPX を示しています。各 CPU コアは、一般にスレッドと呼ばれる 2 つ以上の論理 CPU に分割されます。各スレッドには、それぞれ独自の複製リソースセットとパーティション化されたリソースの一部があり、兄弟スレッドと共有リソースをめぐって競合します。

図 2：図 2：SMT を無効にした 4 コア/8 スレッドシステムでの NetScaler 管理とデータプレーンワークロード



次の画像は、SMT が有効になっている最新の CPU 上で実行されている NetScaler VPX を示しています。

図 5. 図 3：SMT が有効になっている 4 コアシステムでの NetScaler 管理とデータプレーンワークロード



SMT を有効にすると、次の点で NetScaler のパフォーマンスが向上します：

- すべての物理コアでデータプレーン機能を実行しています。
- 管理プレーン機能を兄弟スレッドに移動します。
- 管理プレーン機能がデータプレーン機能のパフォーマンスを損なうことを防ぐために、柔軟なリソース制限メカニズムを導入しました。

SMT サポートマトリックス

SMT をサポートする VPX プラットフォーム、クラウドインスタンスタイプ、NetScaler のバージョンを次の表に示します。

VPX プラットフォーム	インスタンスタイプ	NetScaler VPX バージョン
———	————————————	————
AWS	M5, m5n, c5, c5n	14.1-12.x およびそれ以降
Azure	ハイパースレッディングを使用するすべてのインスタンスファミリー (DS_v4 など)	14.1-12.x およびそれ以降
GCP	e2 インスタンス	14.1-12.x およびそれ以降

> 注
>
> SMT 機能を有効にすると、サポートされているタイプの NetScaler VPX パフォーマンスが向上します。

制限事項

SMT 機能により、NetScaler アプライアンスが利用できる仮想 CPU の数が実質的に倍増します。NetScaler アプライアンスがライセンス制限を使用できるようにするには、ライセンス制限を考慮する必要があります。

たとえば、図 3 に示されている NetScaler VPX について考えてみます。スループットベースのライセンスを使用する場合、8 個の vCPU を有効にするには、SMT 機能を備えた 10 Gbps 以上のライセンスが必要です。以前は、4 つの vCPU を有効にするには 1 Gbps のライセンスで十分でした。vCPU ライセンスを使用する場合、正常に動作するためには、2 倍の数の vCPU のライセンスをチェックアウトするように NetScaler VPX を構成する必要があります。このトピックに関する詳細なガイダンスについては、NetScaler テクニカルサポートにお問い合わせください。

SMT を設定して下さい

SMT 機能を有効にする前に、プラットフォームがこの機能をサポートしていることを確認してください。前のセクションのサポートマトリックスの表を参照してください。

SMT 機能を有効にするには、次の手順に従います:

1. 「/nsconfig」ディレクトリの下に名前 `.smt_handling` の空のファイルを作成します。
2. 現在の設定を保存します。
3. NetScaler VPX インスタンスを再起動します。

```
nscli> shell touch /nsconfig/.smt_handling
  Done
nscli> reboot
Are you sure you want to restart NetScaler (Y/N)?   [N]:Y
Done
```

4. 再起動後、NetScaler は機能が使用可能で有効であることを示します。

```
smt_handling and smt_handling_active are set to  “1”

> shell sysctl -a | grep smt_handling
netscaler.smt_handling_platform: 1
netscaler.smt_handling: 1
netscaler.smt_handling_active: 1
```

SMT 機能を無効にするには、次の手順に従います:

1. `.smt_handling` ファイルを削除します。
2. NetScaler VPX インスタンスを再起動します。


```
shell rm -f /nsconfig/.smt_handling
  Done

reboot

Are you sure you want to restart NetScaler (Y/N)?   [N]:Y
Done
```

3. 再起動後、NetScaler は機能が使用可能だが無効になっていることを示します。

```
> shell sysctl -a | grep smt_handling
netscaler.smt_handling_platform: 1
netscaler.smt_handling: 0
netscaler.smt_handling_active: 0
```

トラブルシューティング

`sysctl` シェルコマンドを実行して、SMT 機能のステータスを確認します。

```
```
  > shell sysctl -a | grep smt_handling
  >
```
```

このコマンドは、次の出力のいずれかを返すことができます。

- SMT 機能がありません。

  `sysctl`コマンドは出力を返しません。

- SMT 機能はサポートされていません。

  SMT 機能は次のいずれかの理由でサポートされていません:

  - お使いの NetScaler VPX は 13.1-48.x または 14.1-12.x より古いです。
  - お使いのクラウドは SMT をサポートしていません。
  - お使いの VM インスタンスタイプは SMT をサポートしていません。たとえば、vCPU 数が 8 を超えています。

    ```
    > shell sysctl -a | grep smt_handling
    netscaler.smt_handling_platform: 0(indicates not supported)
    netscaler.smt_handling: 0 (indicates not enabled)
    netscaler.smt_handling_active: 0 (indicates not active)
    ```

- SMT 機能はサポートされていますが、有効になっていません。

```
> shell sysctl -a | grep smt_handling
netscaler.smt_handling_platform: 1 (available)
```



```
3       netscaler.smt_handling: 0          (not enabled)
4       netscaler.smt_handling_active: 0   (not active)
```

NetScaler VPX インスタンスをベアメタルサーバーにインストールする

October 17, 2024

ベアメタルとは、クラウド環境に完全に統合された、物理的に分離された完全専用の物理サーバーです。シングルテナントサーバーとも呼ばれます。シングルテナントを使用すると、ノイズの多いネイバー効果を回避できます。ベアメタルでは、自分が唯一のユーザーなので、ノイズの多いネイバー効果は発生しません。

ハイパーバイザーがインストールされたベアメタルサーバーは、サーバー上に仮想マシンを作成するための管理スイートを提供します。ハイパーバイザーはアプリケーションをネイティブに実行しません。その目的は、ワークロードを個別の仮想マシンに仮想化して、仮想化の柔軟性と信頼性を高めることです。

NetScaler VPX インスタンスをベアメタルサーバーにインストールするための前提条件

ベアメタルサーバーは、それぞれのハイパーバイザーのすべてのシステム要件を満たすクラウドベンダーから入手する必要があります。

NetScaler VPX インスタンスをベアメタルサーバーにインストールします

NetScaler VPX インスタンスをベアメタルサーバーにインストールするには、まずクラウドベンダーから十分なシステムリソースを備えたベアメタルサーバーを入手する必要があります。そのベアメタルサーバーでは、NetScaler VPX インスタンスを展開する前に、Linux KVM、VMware ESX、Citrix Hypervisor、Microsoft Hyper-V などのサポートされているハイパーバイザーのいずれかをインストールして構成する必要があります。

NetScaler VPX インスタンスでサポートされているさまざまなハイパーバイザーと機能のリストの詳細については、「サポート マトリックスと使用ガイドライン」を参照してください。

さまざまなハイパーバイザーに NetScaler ADC VPX インスタンスをインストールする方法の詳細については、それぞれのドキュメントを参照してください。

- **Citrix Hypervisor:** 「Citrix Hypervisor に NetScaler VPX インスタンスをインストールする」を参照してください。
- **VMware ESX:** 「VMware ESX に NetScaler VPX インスタンスをインストールする」を参照してください。
- **Microsoft Hyper-V:** 「Microsoft Hyper-V サーバーに NetScaler VPX インスタンスをインストールする」を参照してください。
- **Linux KVM** プラットフォーム**:** 「Linux-KVM プラットフォームに NetScaler VPX インスタンスをインストールする」を参照してください。

Citrix Hypervisor に NetScaler ADC VPX インスタンスをインストールする

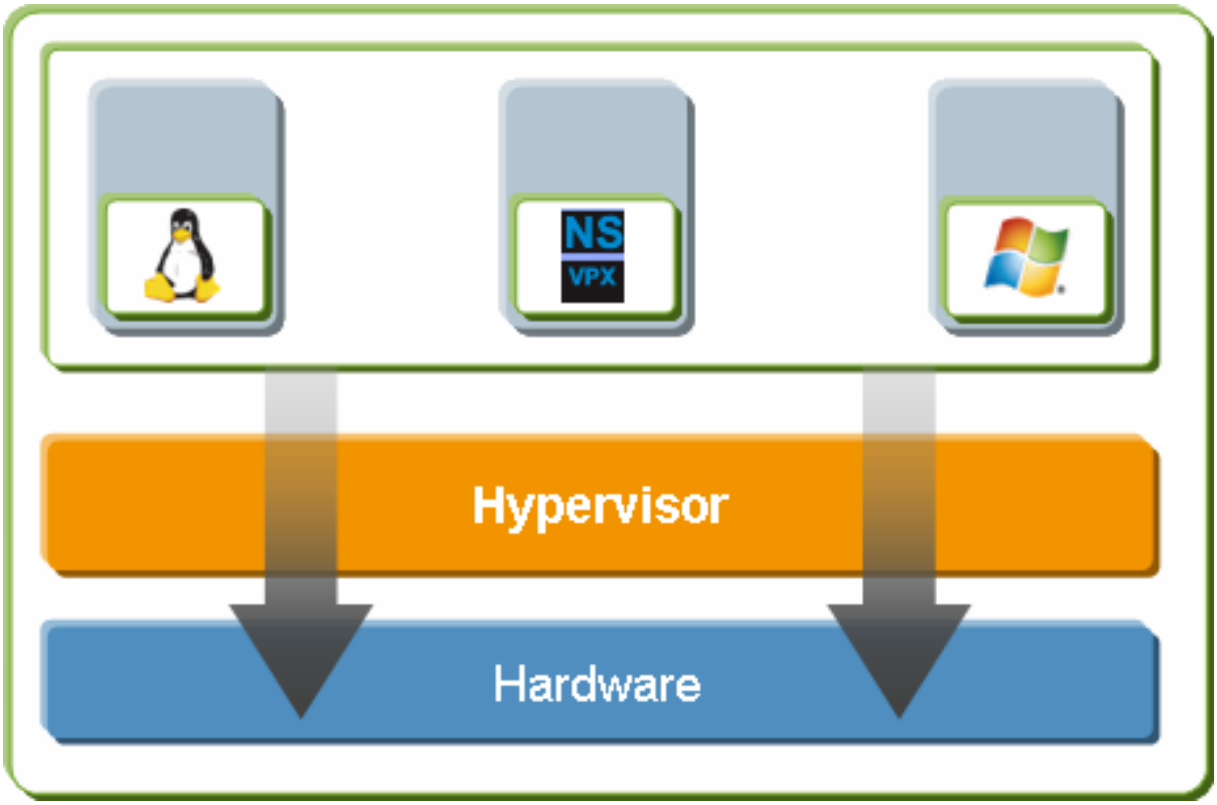
October 17, 2024

Citrix Hypervisor に VPX インスタンスをインストールするには、まず適切なシステムリソースを持つマシンに Hypervisor をインストールする必要があります。NetScaler VPX インスタンスのインストールを実行するには、Citrix XenCenter を使用します。Citrix XenCenter は、ネットワーク経由で Hypervisor ホストに接続できるリモートマシンにインストールする必要があります。

Hypervisor の詳細については、Citrix Hypervisor のドキュメントを参照してください。

次の図は、Hypervisor 上の NetScaler ADC VPX インスタンスのベアメタルソリューションアーキテクチャを示しています。

図. Citrix Hypervisor 上の NetScaler VPX インスタンス



Hypervisor に NetScaler ADC VPX インスタンスをインストールするための前提条件

仮想アプライアンスのインストールを開始する前に、次の操作を行います。

- 最小要件を満たすハードウェアに Hypervisor バージョン 6.0 以降をインストールします。
- 最小システム要件を満たす管理ワークステーションに XenCenter をインストールします。

- 仮想アプライアンスのライセンスファイルを取得します。仮想アプライアンス ライセンスの詳細については、『NetScaler ライセンス ガイド』を参照してください。

Hypervisor のハードウェア要件

次の表は、NetScaler VPX インスタンスを実行するハイパーバイザープラットフォームの最小ハードウェア要件を示しています。

テーブル **1**. nCore VPX インスタンスを実行する Hypervisor の最小システム要件

コンポーネント	条件
CPU	仮想化アシスト (Intel-VT) が有効になっている 64 ビット x86 CPU が 2 つ以上あります。NetScaler VPX インスタンスを実行するには、Hypervisor ホストで仮想化のハードウェアサポートを有効にする必要があります。仮想化サポートの BIOS オプションが無効になっていないことを確認してください。詳細については、BIOS のドキュメントを参照してください。
RAM	3 GB
ディスク領域	40 GB のディスク容量を持つローカル接続ストレージ (PATA、SATA、SCSI)。注:Hypervisor のインストールでは、Hypervisor ホストコントールドメインに 4 GB のパーティションが作成されます。残りのスペースは、NetScaler VPX インスタンスやその他の仮想マシンに使用できます。
NIC	1 Gbps NIC × 1、推奨：2 つの 1 Gbps NIC

Hypervisor のインストールについては、http://support.citrix.com/product/xens/の Hypervisor のドキュメントを参照してください。

次の表に、Hypervisor が各 nCore VPX 仮想アプライアンスに提供する必要がある仮想コンピューティングリソースを示します。

テーブル **2**. nCore VPX インスタンスの実行に必要な最小仮想コンピューティングリソース

コンポーネント	条件
————	—-
メモリ	2 GB
仮想 CPU（VCPU）	2
仮想ネットワークインターフェイス	2

> 注
>
> NetScaler VPX インスタンスを本番環境で使用する場合、スケジューリング動作とネットワーク遅延を改善するために、（仮想マシンプロパティの）CPU 優先度を最高レベルに設定することを Citrix では推奨しています。

XenCenter のシステム要件

XenCenter は、Windows のクライアントアプリケーションです。Hypervisor ホストと同じマシンでは実行できません。最小システム要件と XenCenter のインストールについて詳しくは、Hypervisor に関する次のドキュメントを参照してください。

- システム要件
- インストール

XenCenter を使用して **NetScaler VPX** インスタンスをハイパーバイザーにインストールする

Hypervisor と XenCenter をインストールして構成したら、XenCenter を使用して Hypervisor に仮想アプライアンスをインストールできます。インストールできる仮想アプライアンスの数は、Hypervisor を実行しているハードウェアで使用可能なメモリの量によって異なります。

XenCenter を使用して Hypervisor に NetScaler ADC VPX インスタンスをインストールするには、次の手順に従います。

1. ワークステーションで **XenCenter** を起動します。
2. ［サーバー］メニューの［追加］を選択します。
3. [新規サーバーの追加] ダイアログボックスのホスト名テキストボックスに、接続するハイパーバイザーの IP アドレスまたは DNS 名を入力します。
4. ［**User Name**］ボックスと［**Password**］ボックスに管理者資格情報を入力して、［**Connect**］をクリックします。Hypervisor 名がナビゲーションペインに表示され、Hypervisor が接続されていることを示します。
5. ナビゲーションペインで、NetScaler VPX インスタンスをインストールする Hypervisor の名前をクリックします。
6. ［**VM**］メニューの［**Import**］を選択します。
7. インポートダイアログボックスのインポートファイル名で、NetScaler VPX インスタンス`.xva`イメージファイルを保存した場所を参照します。[エクスポートされた仮想マシン] オプションが選択されていることを確認し、[次へ] をクリックします。
8. 仮想アプライアンスをインストールするハイパーバイザーを選択し、[次へ] をクリックします。
9. 仮想アプライアンスを保存するローカルストレージリポジトリを選択して［**Import**］をクリックし、インポート処理を開始します。

10. 必要に応じて、仮想ネットワークインターフェイスを追加、変更、または削除できます。完了したら［**Next**］をクリックします。
11. [完了] をクリックしてインポートプロセスを完了します。

    > 注
    >
    > インポート処理の状態を参照するには、［**Log**］タブをクリックします。

12. 別の仮想アプライアンスをインストールする場合は、手順 5 〜11 を繰り返します。

> 注
>
> VPX インスタンスの初期構成後、アプライアンスを最新のソフトウェアリリースにアップグレードする場合は、「システムソフトウェアのアップグレードまたはダウングレード」を参照してください。

シングルルート **I/O** 仮想化 **(SR-IOV)** ネットワークインターフェイスを使用するように **VPX** インスタンスを構成する

October 17, 2024

NetScaler VPX インスタンスを Citrix Hypervisor にインストールして構成したら、SR-IOV ネットワークインターフェイスを使用するように仮想アプライアンスを構成できます。

次の NIC がサポートされています。

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

制限事項

Citrix Hypervisor は、SR-IOV インターフェイスの一部の機能をサポートしていません。Intel 82599、Intel X710、および Intel XL710 NIC の制限は、次のセクションに記載されています。

Intel 82599 NIC の制限事項

Intel 82599 NIC は次の機能をサポートしていません。

- L2 モード切り替え
- クラスタリング
- 管理パーティション化［共有 VLAN モード］

- 高可用性［アクティブ/アクティブモード］
- ジャンボフレーム
- クラスター環境の IPv6 プロトコル

Intel X710 10G および **Intel XL710 40G NIC** の制限事項

Intel X710 10 G と Intel XL710 40G NIC には次の制限があります。

- L2 モードの切り替えはサポートされていません。
- 管理パーティショニング（共有 VLAN モード）はサポートされていません。
- クラスタでは、XL710 NIC がデータ・インタフェースとして使用されている場合、ジャンボフレームはサポートされません。
- インターフェイスが切断され、再接続されると、インターフェイスリストが順序変更されます。
- スピート、デュプレックス、オートネゴシエーションなどの Interface パラメーター構成はサポートされません。
- Intel X710 10 G と Intel XL710 40G NIC の両方で、インターフェイスは 40/x インターフェイスとして表示されます。
- VPX インスタンスでサポートできる Intel X710/XL710 SR-IOV インターフェイスは 16 個までです。

> 注
>
> Intel X710 10G および Intel XL710 40G NIC が IPv6 をサポートするには、Citrix Hypervisor ホストで次のコマンドを入力して、仮想機能（VF）のトラストモードを有効にします。
>
> ```
> # ip link set <PNIC> <VF> trust on
> ```
>
> 例
>
> ```
> # ip link set ens785f1 vf 0 trust on
> ```

Intel 82599 NIC の前提条件

Citrix Hypervisor ホストで、次のことを確認してください。

- インテル 82599 NIC (NIC) をホストに追加します。
- **/etc/modprobe.d/blacklist.conf** ファイルに次のエントリを追加して、`ixgbevf`ドライバを一覧表示することを禁止します。

  blacklist ixgbevf

- **/etc/modprobe.d/blacklist.conf** ファイルで、以下のエントリを追加して、SR-IOV Virtual Functions（VF）を有効にします。

  options ixgbe max_vfs=*<number_of_VFs>*

  ここで、*<number_VFs>* は、作成する SR-IOV VF の数です。

- SR-IOV サーバーが BIOS で有効になっていることを確認します。

> 注
>
> IXGBE ドライバーのバージョン 3.22.3 をお勧めします。

Citrix Hypervisor ホストを使用して、**Intel 82599 SR-IOV VF** を **NetScaler VPX** インスタンスに割り当てます

Intel 82599 SR-IOV VF を NetScaler VPX インスタンスに割り当てるには、次の手順に従います。

1. Citrix Hypervisor ホストで、次のコマンドを使用して SR-IOV VF を Citrix ADC VPX インスタンスに割り当てます。

   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>*

   各項目の意味は次のとおりです:

   - <*Xen host UUID*>Citrix Hypervisor の UUID です。
   - <*NetScaler VM UUID*> は、NetScaler VPX インスタンスの UUID です。
   - <*interface name*> は SR-IOV VF のインターフェイスです。
   - <*MAC address* > は SR-IOV VF の MAC アドレスです。

   > 注
   >
   > args: mac= パラメータで使用する MAC アドレスを指定します。指定しない場合、`iovirt`スクリプトはランダムに MAC アドレスを生成して割り当てます。また、リンクアグリゲーションモードで SR-IOV VF を使用する場合は、必ず MAC アドレスを 00:00:00:00:00 と指定します。

2. NetScaler VPX インスタンスを起動します。

Citrix Hypervisor ホストを使用して、**Intel 82599 SR-IOV VF** を **NetScaler VPX** インスタンスに割り当て解除します

正しくない SR-IOV VF を割り当てた場合、または割り当てられた SR-IOV VF を変更する場合は、SR-IOV VF を Citrix ADC VPX インスタンスに割り当て解除して再割り当てする必要があります。

NetScaler VPX インスタンス に割り当てられた SR-IOV ネットワークインターフェイスの割り当てを解除するには、次の手順に従います。

1. Citrix Hypervisor ホストで、次のコマンドを使用して SR-IOV VF を Citrix ADC VPX インスタンスに割り当て、NetScaler VPX インスタンスを再起動します。

xe host-call-plugin plugin=iovirt **host-uuid**=<*Xen_host_UUID*> **fn**=unassign_all **args:uuid**=<*Netscaler_VM_U*

各項目の意味は次のとおりです:

- <*Xen_host_UUID*>-Citrix Hypervisor ホストの UUID。
- <*Netscaler_VM_UUID*>-NetScaler VPX インスタンスの UUID

2. NetScaler VPX インスタンスを起動します。

Citrix Hypervisor ホストを使用して、**Intel X710/XL710 SR-IOV VF** を **NetScaler VPX** インスタンスに割り当てます

Intel X710/XL710 SR-IOV VF を NetScaler VPX インスタンスに割り当てるには、次の手順に従います。

1. Citrix Hypervisor ホストで次のコマンドを実行して、ネットワークを作成します。

```
xe network-create name-label=<network-name>
```

例

```
xe network-create name-label=SR-IOV-NIC-18 8ee59b73-7319-6998-
   cd69-b9fa3e8d7503
```

2. SR-IOV ネットワークを構成する NIC の PIF ユニバーサル一意識別子 (UUID) を決定します。

```
xe pif-list

             uuid ( RO)  : e2874343-f1de-1fa7-8fef-98547c348783
             device ( RO): eth18
currently-attached ( RO): true
               VLAN ( RO): -1
       network-uuid ( RO): f865bd85-44dd-b865-ab65-dcd6ae28c16e
```

3. ネットワークを SR-IOV ネットワークとして設定します。次のコマンドは、新しく作成された SR-IOV ネットワークの UUID も返します。

```
xe network-sriov-create network-uuid=<network-uuid> pif-uuid=<
   physical-pif-uuid>
```

例

```
xe network-sriov-create network-uuid=8ee59b73-7319-6998-cd69-
   b9fa3e8d7503 pif-uuid=e2874343-f1de-1fa7-8fef-98547
   c3487831629b44f-832a-084e-d67d-5d6d314d5e0f
```

SR-IOV ネットワークパラメーターの詳細情報を取得するには、次のコマンドを実行します。

```
[root@citrix-XS82-TOPO ~]# xe network-sriov-param-list uuid=1629
   b44f-832a-084e-d67d-5d6d314d5e0f

```



```
                uuid ( RO): 1629b44f-832a-084e-d67d-5d6d314d5e0f
         physical-PIF ( RO): e2874343-f1de-1fa7-8fef-98547c348783
          logical-PIF ( RO): 85d52771-5814-c62d-45fa-f37b536144ff
     requires-reboot ( RO): false
remaining-capacity ( RO): 32
```

4. 仮想インターフェイス (VIF) を作成し、ターゲット VM にアタッチします。

```
xe vif-create device=0 mac=b2:61:fc:ae:00:1d network-uuid=8
   ee59b73-7319-6998-cd69-b9fa3e8d7503 vm-uuid=b507e8a6-f5ca-18
   eb-561d-308218a9dd68
3e1e2e58-b2ad-6dc0-61d4-1d149c9c6466
```

> 注
>
> VM の NIC インデックス番号は 0 で始まる必要があります。

VM UUID を見つけるには、次のコマンドを使用します。

```
[root@citrix-XS82-TOPO ~]# xe vm-list
uuid ( RO): b507e8a6-f5ca-18eb-561d-308218a9dd68
 name-label ( RW): sai-vpx-1
power-state ( RO): halted
```

Citrix Hypervisor ホストを使用して **NetScaler** インスタンスから **Intel X710/XL710 SR-IOV VF** を削除します

NetScaler VPX インスタンスから Intel X710/XL710 SR-IOV VF を削除するには、次の手順に従います。

1. 破棄する VIF の UUID をコピーします。
2. VIF を破棄するには、Citrix Hypervisor ホスト上で以下のコマンドを実行します。

```
xe vif-destroy uuid=<vif-uuid>
```

例

```
[root@citrix-XS82-TOPO ~]# xe vif-destroy uuid=3e1e2e58-b2ad-6
   dc0-61d4-1d149c9c6466
```

SR-IOV インターフェイスでのリンクアグリゲーションの設定

SR-IOV 仮想機能 (VF) をリンクアグリゲーションモードで使用するには、作成した仮想機能のスプーフィングチェックを無効にする必要があります。

Citrix Hypervisor ホストで、次のコマンドを使用してスプーフィングチェックを無効にします。

ip リンクセット ** <interface_name>**vf ** <VF_id>**spoofchk** オフ

各項目の意味は次のとおりです:

- <interface_name> は、インターフェイス名です。
- <VF_id> は、仮想機能 ID です。

作成したすべての仮想機能のスプーフィングチェックを無効にした後、NetScaler VPX インスタンスを再起動し、リンクアグリゲーションを設定します。手順については、「リンク集約の設定」を参照してください。

> 重要:
>
> SR-IOV VF を Citrix ADC VPX インスタンスに割り当てるときは、VF の MAC アドレス 00:00:00:00:00:00 を必ず指定してください。

SR-IOV インターフェイスで VLAN を設定します

SR-IOV 仮想機能に VLAN を設定できます。手順については、「VLAN の設定」を参照してください。

> 重要:
>
> Citrix Hypervisor ホストに VF インターフェイスの VLAN 設定が含まれていないことを確認してください。

VMware ESX に Citrix ADC VPX インスタンスをインストールする

October 17, 2024

NetScaler VPX インスタンスを VMware ESX にインストールする前に、VMware ESX サーバーが適切なシステムリソースを備えたマシンにインストールされていることを確認してください。NetScaler VPX インスタンスを VMware ESXi にインストールするには、VMware vSphere クライアントを使用します。これらのクライアントソフトウェアは、ネットワーク経由で VMware ESX に接続できるリモートマシンにインストールする必要があります。

このセクションでは、以下のトピックについて説明します。

- 前提条件
- VMware ESX に Citrix ADC VPX インスタンスをインストールする

> 重要:
>
> NetScaler VPX インスタンスで標準の VMware Tools をインストールしたり、VMware Tools バージョンをアップグレードしたりすることはできません。NetScaler VPX インスタンス用の VMware ツールは、Citrix ADC ソフトウェアリリースの一部として提供されます。

前提条件

仮想アプライアンスのインストールを開始する前に、次の操作を行います。

- 最小要件を満たすハードウェアに VMware ESX をインストールします。
- 最小システム要件を満たす管理用のワークステーションに VMware Client をインストールします。
- NetScaler VPX アプライアンスセットアップファイルをダウンロードします。
- 仮想スイッチを作成し、物理 NIC を仮想スイッチに接続します。
- ポートグループを追加し、仮想スイッチに接続します。
- ポートグループを VM に接続します。
- VPX ライセンスファイルを入手します。NetScaler VPX インスタンスライセンスの詳細については、「ライセンスの概要」を参照してください。

VMware ESX のハードウェア要件

次の表では、NetScaler VPX nCore 仮想アプライアンスを実行している VMware ESX サーバーの最小システム要件について説明します。

表 1. NetScaler VPX インスタンスを実行する VMware ESX サーバーの最小システム要件

コンポーネント	条件
CPU	仮想化アシスト (Intel-VT) が有効になっている 64 ビット x86 CPU が 2 つ以上あります。NetScaler VPX インスタンスを実行するには、仮想化のハードウェアサポートが VMware ESX ホストで有効になっている必要があります。仮想化サポートの BIOS オプションが無効になっていないことを確認します。詳しくは、BIOS のドキュメントを参照してください。NetScaler 13.1 リリース以降、VMware ESXi ハイパーバイザー上の NetScaler VPX インスタンスは AMD プロセッサをサポートしています。
RAM	2 GB VPX。2 ギガバイトの VPX 重要な展開では、システムがメモリに制約のある環境で動作するため、VPX に 2 GB の RAM を使用することはお勧めしません。これにより、スケール、パフォーマンス、または安定性に関連する問題が発生する可能性があります。推奨されるのは 4 GB の RAM または 8 GB の RAM です。
ディスク領域	ESXi をセットアップするための VMware の最小サーバ要件よりも 20 GB 多くなっています。サーバの最小要件については、VMware のマニュアルを参照してください。

コンポーネント	条件
ネットワーク	1 Gbps NIC (NIC) 1 つ、1 Gbps NIC を 2 つ推奨

VMware ESX のインストールについては、http://www.vmware.com/を参照してください。

SR-IOV ネットワークインターフェイスまたは PCI パススルーをサポートするには、次のプロセッサと設定が有効になっていることを確認してください。

- Intel VT をサポートする Intel ・プロセッサー
- AMD-V をサポートする AMD プロセッサー
- 入出力メモリ管理ユニット (IOMMU) または SR-IOV が BIOS で有効になっています

SR-IOV モードでは次の NIC がサポートされます。

- Mellanox ConnectX-4 NIC（Citrix ADC リリース 13.1-42.x 以降）
- Intel 82599 NIC

次の表に、VMware ESX サーバが各 VPX nCore 仮想アプライアンスに対して提供する必要がある仮想コンピューティングリソースを示します。

表 2. NetScaler VPX インスタンスの実行に必要な最小限の仮想コンピューティングリソース

コンポーネント	条件
———-	——————————————————-
メモリ	4 GB
仮想 CPU（VCPU）	2
仮想ネットワークインターフェイス	ESX では、VPX ハードウェアをバージョン 7 以上にアップグレードすると、最大 10 個の仮想ネットワークインターフェイスをインストールできます。
ディスク領域	20GB

> 注
>
> これは、ハイパーバイザーのディスク要件に加えて必要になります。

VPX 仮想アプライアンスを本番環境で使用するには、完全なメモリ割り当てを予約する必要があります。少なくとも ESX の 1 つの CPU コアの速度に等しい CPU サイクル (MHz) を予約する必要があります。

VMware vSphere クライアントのシステム要件

VMware vSphere Client は、Windows および Linux の各オペレーティングシステムで実行できるクライアントアプリケーションです。VMware ESX サーバと同じマシンでは実行できません。次の表は、最小システム要件を示しています。

表 3. VMware vSphere クライアントインストールの最小システム要件

コンポーネント	条件
オペレーティングシステム	VMware からの詳細な要件については、http://kb.vmware.com/で「vSphere 互換性マトリックス」PDF ファイルを検索してください。
CPU	750 MHz。1 ギガヘルツ (GHz) 以上推奨
RAM	1GB. 2GB を推奨
NIC (NIC)	100Mbps 以上の NIC。

OVF ツール **1.0** のシステム要件

Open Virtualization Format Tool（OVF Tool）は、Windows および Linux システムで実行できるクライアントアプリケーションです。VMware ESX サーバと同じマシンでは実行できません。次の表は、最小システム要件を示しています。

表 4. OVF ツールのインストールに必要な最小システム要件

コンポーネント	条件
オペレーティングシステム	VMware からの詳細な要件については、http://kb.vmware.com/で『OVF ツールユーザーガイド』の PDF ファイルを検索してください。
CPU	最低 750MHz、1GHz 以上推奨
RAM	最小 1 GB、推奨 2 GB
NIC (NIC)	100Mbps 以上の NIC。

OVF のインストールについては、http://kb.vmware.com/で『OVF ツールユーザーガイド』の PDF ファイルを検索してください。

NetScaler VPX セットアップファイルのダウンロード

VMware ESX 用の NetScaler VPX インスタンスセットアップパッケージは、オープン仮想マシン（OVF）フォーマット標準に準拠しています。これらのファイルは、Citrix Web サイトからダウンロードできます。ログオンするには、Citrix アカウントが必要です。Citrix アカウントをお持ちでない場合は、http://www.citrix.comのホームページにアクセスし、［新規ユーザー］リンクをクリックし、指示に従って Citrix アカウントを作成します。

ログオンしたら、Citrix のホームページから次のパスをナビゲートします。

Citrix.com > ダウンロード **> NetScaler >** 仮想アプライアンス。

次のファイルを、ESX サーバーと同じネットワーク上のワークステーションにコピーします。3 つのファイルをすべて同じフォルダーにコピーします。

- NSVPX-ESX-<release number>-<build number>-disk1.vmdk（たとえば、nsvpx-esx-13.0-71.44_nc_64-disk1.vmdk）
- NSVPX-ESX-<release number>-<build number>.ovf（たとえば、nsvpx-ESX-13.0-71.44_nc_64.ovf）
- NSVPX-ESX-<release number>-<build number>.mf（たとえば、nsvpx-esx-13.0-71.44_nc_64.mf）

VMware ESX に Citrix ADC VPX インスタンスをインストールする

VMware ESX をインストールして構成したら、VMware vSphere Client を使用して VMware ESX サーバーに仮想アプライアンスをインストールします。インストールできる仮想アプライアンスの数は、VMware ESX を実行するハードウェアで使用可能なメモリの量によって決まります。

VMware vSphere クライアントを使用して NetScaler VPX インスタンスを VMware ESX にインストールするには、以下の手順に従ってください。

1. ワークステーション上で VMware vSphere Client を起動します。
2. [**IP address / Name**］テキストボックスに、接続する VMware ESX サーバーの IP アドレスを入力します。
3. [ユーザー名] テキストボックスと [パスワード] テキストボックスに、管理者の認証情報を入力し、[ログイン] をクリックします。
4. [**File**］メニューの［**Deploy OVF Template**］を選択します。
5. [**OVF** テンプレートのデプロイ］ダイアログボックスの［ファイルからデプロイ］で、NetScaler VPX インスタンスセットアップファイルを保存した場所を参照し、.ovf ファイルを選択して、［次へ］をクリックします。
6. 仮想アプライアンス OVF テンプレートに示されるネットワークを、ESX ホストで構成したネットワークにマップします。［**Next**］をクリックして、VMware ESX への仮想アプライアンスのインストールを開始します。インストールが完了すると、ポップアップウィンドウによって正常にインストールされたことが通知されます。
7. これで、NetScaler VPX インスタンスを起動する準備ができました。ナビゲーションペインで、インストールした Citrix ADC VPX インスタンスを選択し、右クリックメニューから［パワーオン］を選択します。
8. 仮想マシンを起動したら、コンソールから Citrix ADC IP、ネットマスク、およびゲートウェイアドレスを設定します。設定が完了したら、コンソールで [**Save and Quit**] オプションを選択します。
9. 別の仮想アプライアンスをインストールするには、ステップ 6 からステップ 8 までを繰り返します。

> 注
>
> デフォルトでは、NetScaler VPX インスタンスは E1000 ネットワークインターフェイスを使用します。
>
> インストール後、vSphere クライアントまたは vSphere Web Client を使用して VMware ESX 上の仮想アプライアンスを管理できます。
>
> VMware ESX で VLAN タグ付けを有効にするには、vSwitch でポート グループの VLAN ID をすべて (4095) に設定します。vSwitch で VLAN ID を設定する詳細な手順については、VMware のドキュメントを参照して

ください。

VMware vMotion を使用して **Citrix ADC VPX** インスタンスを移行する

VMware vSphere vMotion を使用して、NetScaler VPX インスタンスを移行できます。

使用上のガイドラインに従ってください。

- VMware は、PCI パススルーおよび SR-IOV インターフェイスで構成された仮想マシンでは vMotion 機能をサポートしていません。
- サポートされているインターフェイスは、E1000 と VMXNET3 です。VPX インスタンスで vMotion を使用するには、サポートされているインターフェイスでインスタンスが設定されていることを確認します。
- VMware vMotion を使用してインスタンスを移行する方法の詳細については、VMware のドキュメントを参照してください。

VMXNET3 ネットワークインターフェイスを使用するように **NetScaler VPX** インスタンスを構成する

October 17, 2024

VMware ESX に NetScaler VPX インスタンスをインストールして構成したら、VMware vSphere Web クライアントを使用して、VMXNET3 ネットワークインターフェイスを使用するように仮想アプライアンスを構成できます。

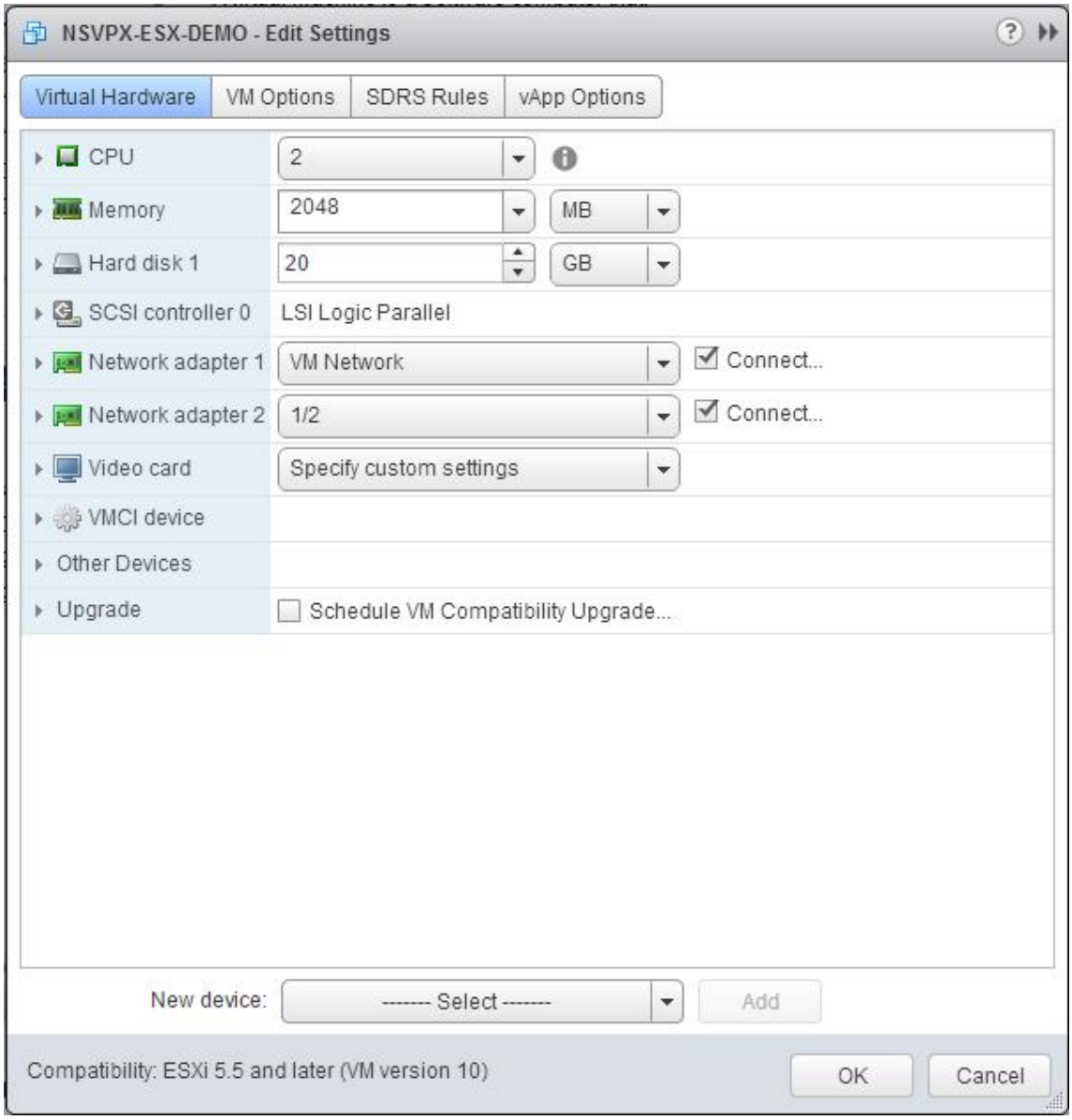
VMware vSphere Web クライアントを使用して VMXNET3 ネットワークインターフェイスを使用するように NetScaler VPX インスタンスを構成するには:

1. vSphere Web クライアントで、ホストとクラスタを選択します。
2. 次のように、NetScaler VPX インスタンスの互換性設定を ESX にアップグレードします。

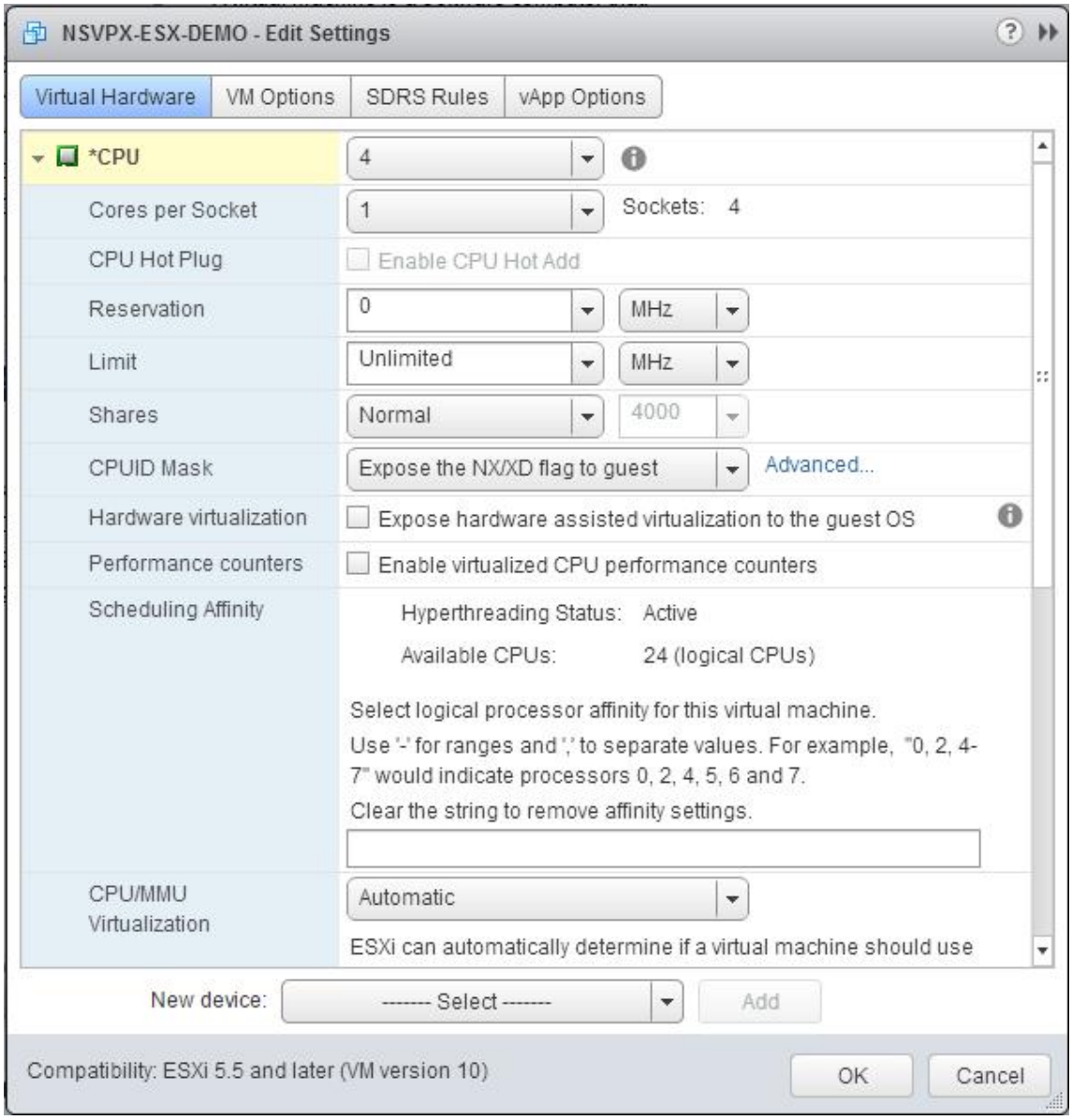
   a. NetScaler VPX インスタンスの電源を切ります。

   b. NetScaler VPX インスタンスを右クリックし、「互換性」>「仮想マシンの互換性のアップグレード」を選択します。

   c. 「仮想マシンの互換性の設定」ダイアログボックスで、「互換性」ドロップダウンリストから「ESXi 5.5 以降」を選択し、「OK」をクリックします。
3. NetScaler VPX インスタンスを右クリックし、[設定の編集] をクリックします。



4. [<virtual_appliance> 設定の編集] ダイアログボックスで、[CPU] セクションをクリックします。



5. [CPU] セクションで、以下を更新します。

   - CPU の数
   - ソケット数
   - 予約
   - 上限
   - 共有

   値を次のように設定します:

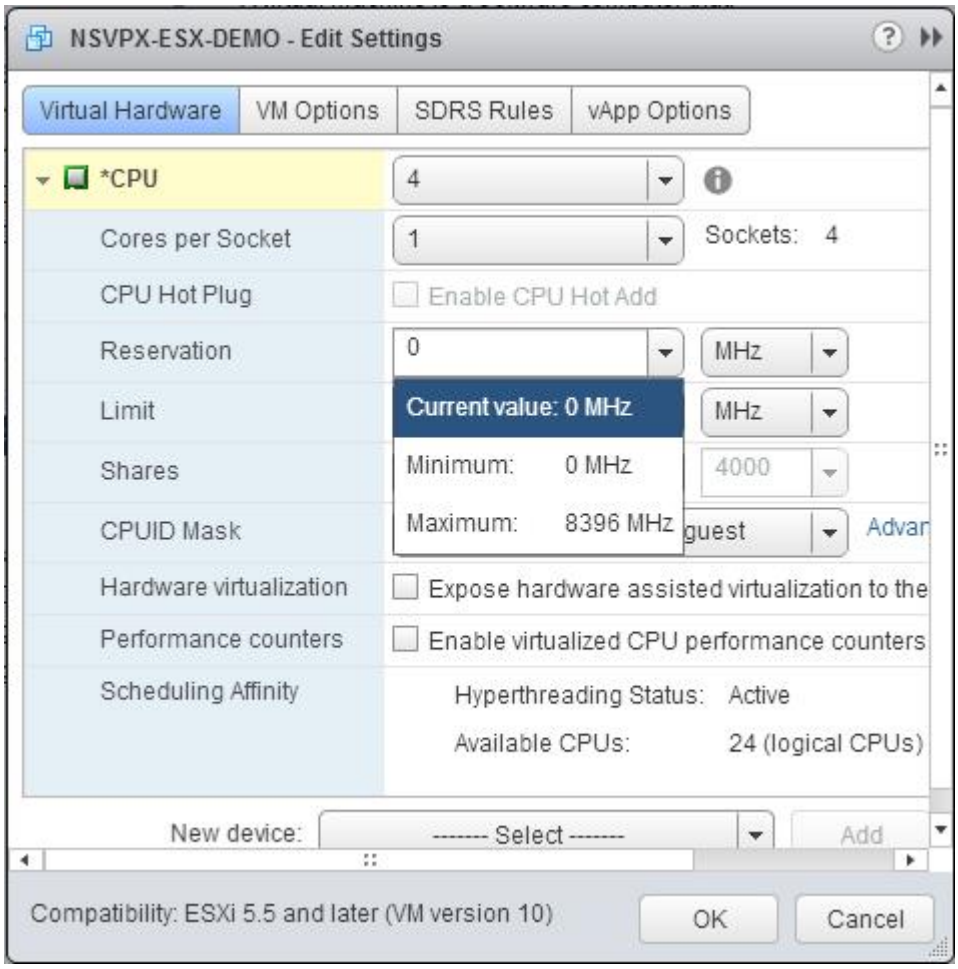
   a. CPU ドロップダウンリストで、仮想アプライアンスに割り当てる CPU の数を選択します。

   b. [予約] ドロップダウンリストで、最大値として表示される番号を選択します。

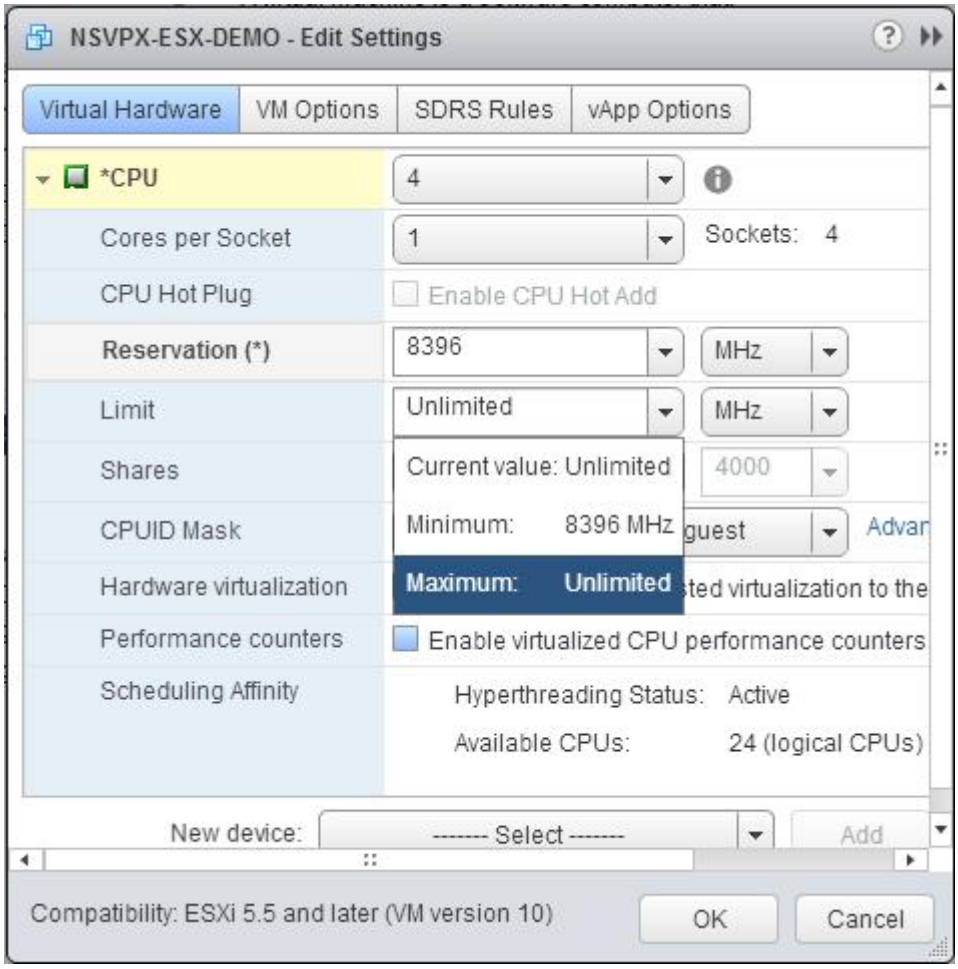
c. （オプション）[CPU ホットプラグ] フィールドで、[CPU ホットアドを有効にする] チェックボックスをオンまたはオフにします。

> 注
>
> Citrix では、デフォルト (無効) を受け入れることをお勧めします。

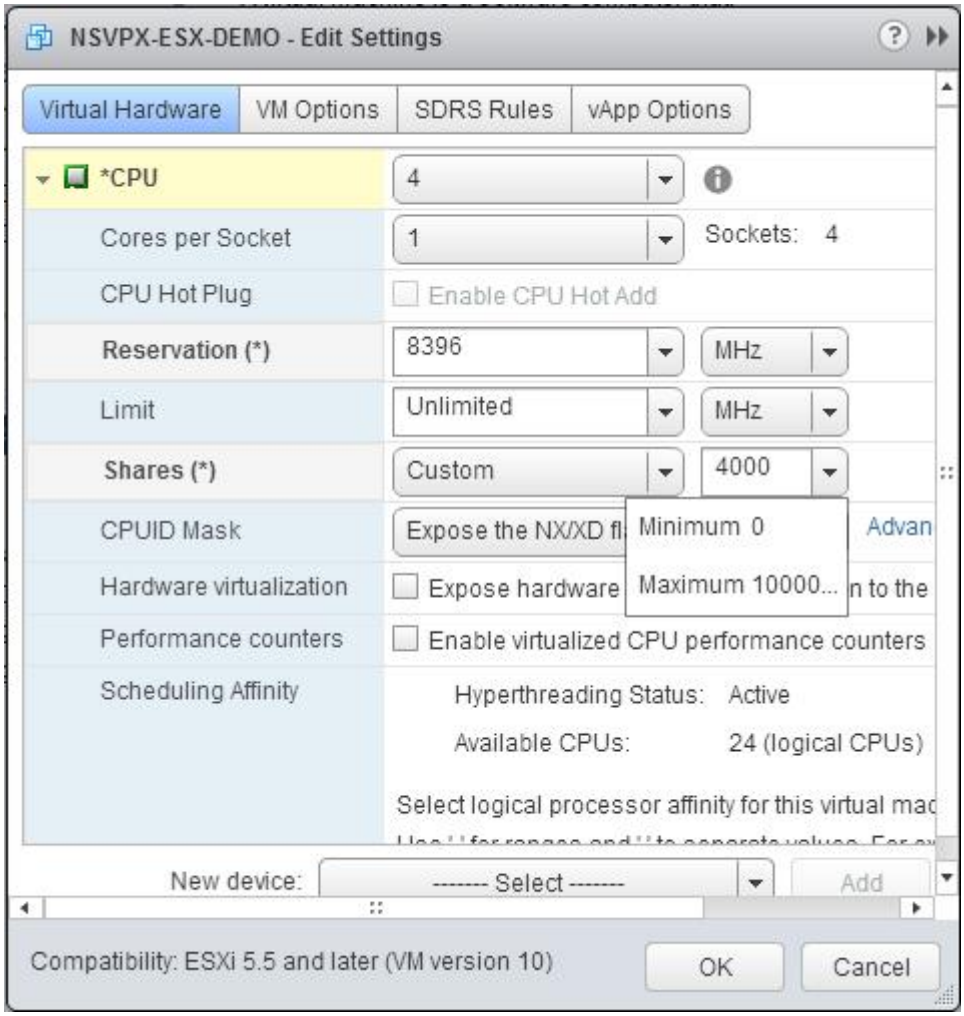
d. [制限] ドロップダウンリストで、最大値として表示されている数値を選択します。



e. [制限] ドロップダウンリストで、最大値として表示されている数値を選択します。



f. [共有] ドロップダウンリストで、[カスタム] を選択し、最大値として表示される数を指定します。



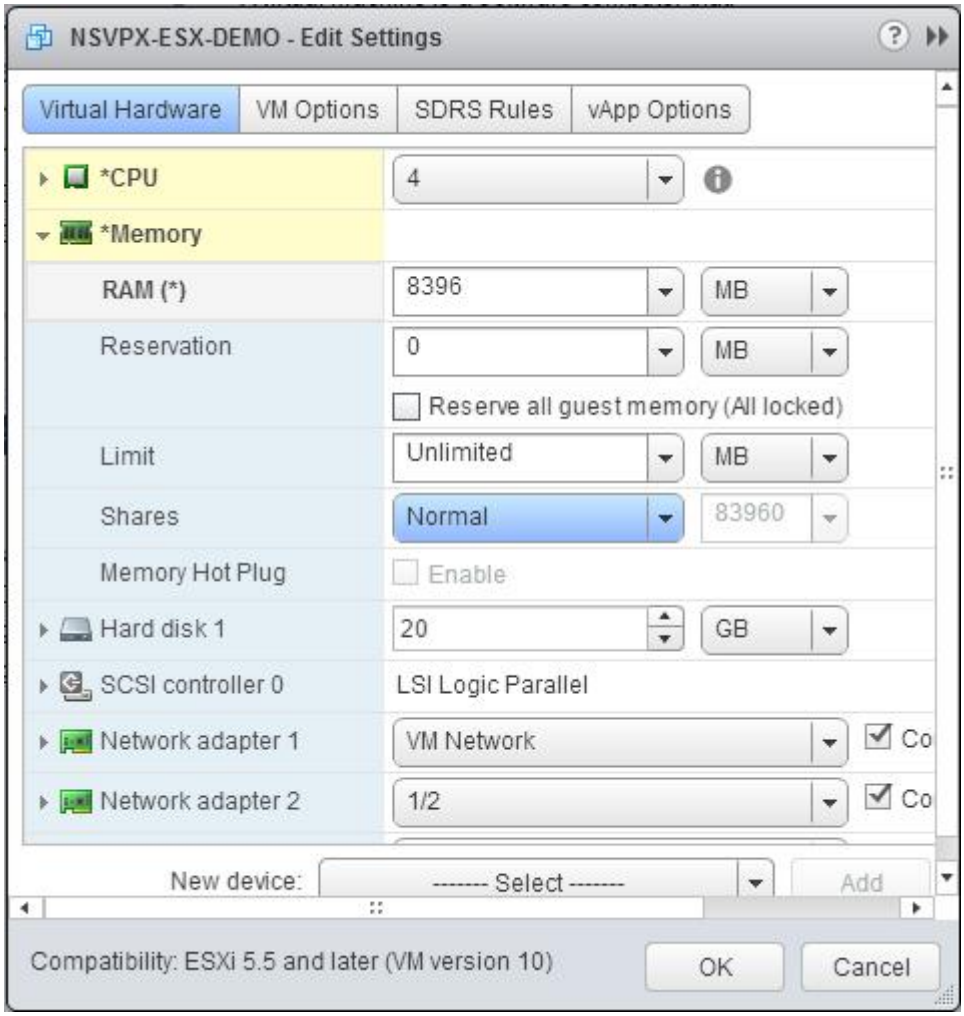
6. [メモリ] セクションで、次の項目を更新します。

   - メモリのサイズ
   - 予約
   - 上限
   - 共有

   値を次のように設定します:

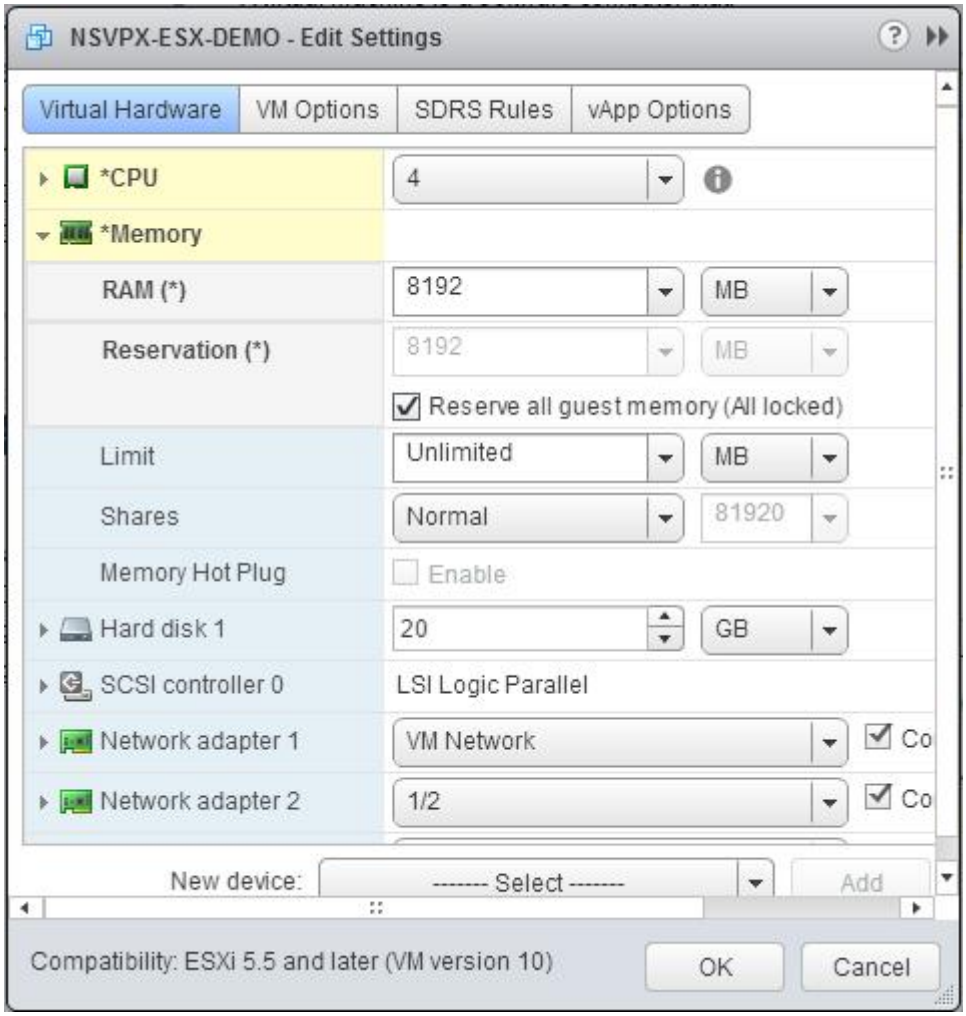
   a. [RAM] ドロップダウンリストで、RAM のサイズを選択します。これは vCPU の数 x 2 GB でなければなりません。たとえば、vCPU の数が 4 の場合、RAM は 4 x 2 GB = 8 GB でなければなりません。

   > 注
   >
   > NetScaler VPX アプライアンスの Advanced または Premium エディションの場合は、各 vCPU に 4 GB の RAM を割り当てるようにしてください。たとえば、vCPU の数が 4 の場合、RAM = 4×4GB = 16GB になります。

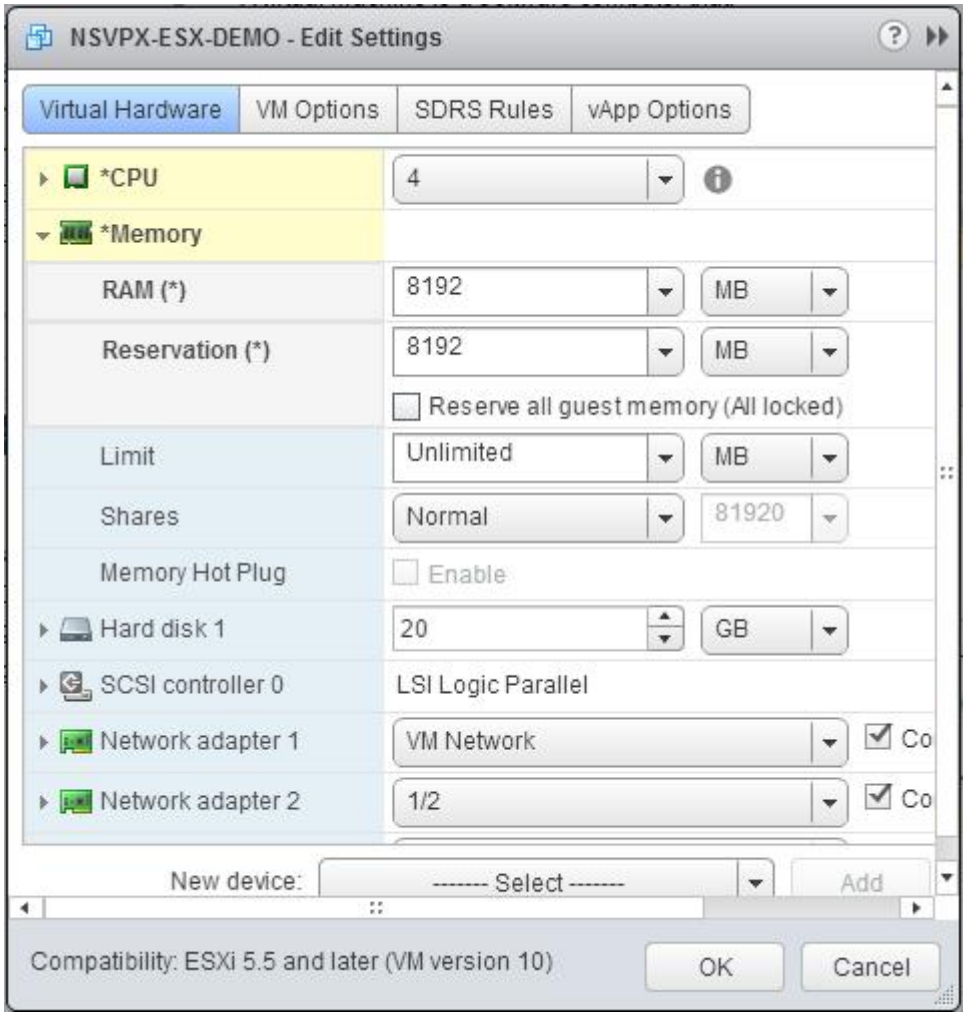


b. [予約] ドロップダウンリストで、メモリ予約の値を入力し、[すべてのゲストメモリを予約する (すべてのロック済み)] チェックボックスをオンにします。メモリ予約は vCPU の数 x 2 GB である必要があります。たとえば、vCPU の数が 4 の場合、メモリ予約は 4 x 2 GB = 8 GB である必要があります。

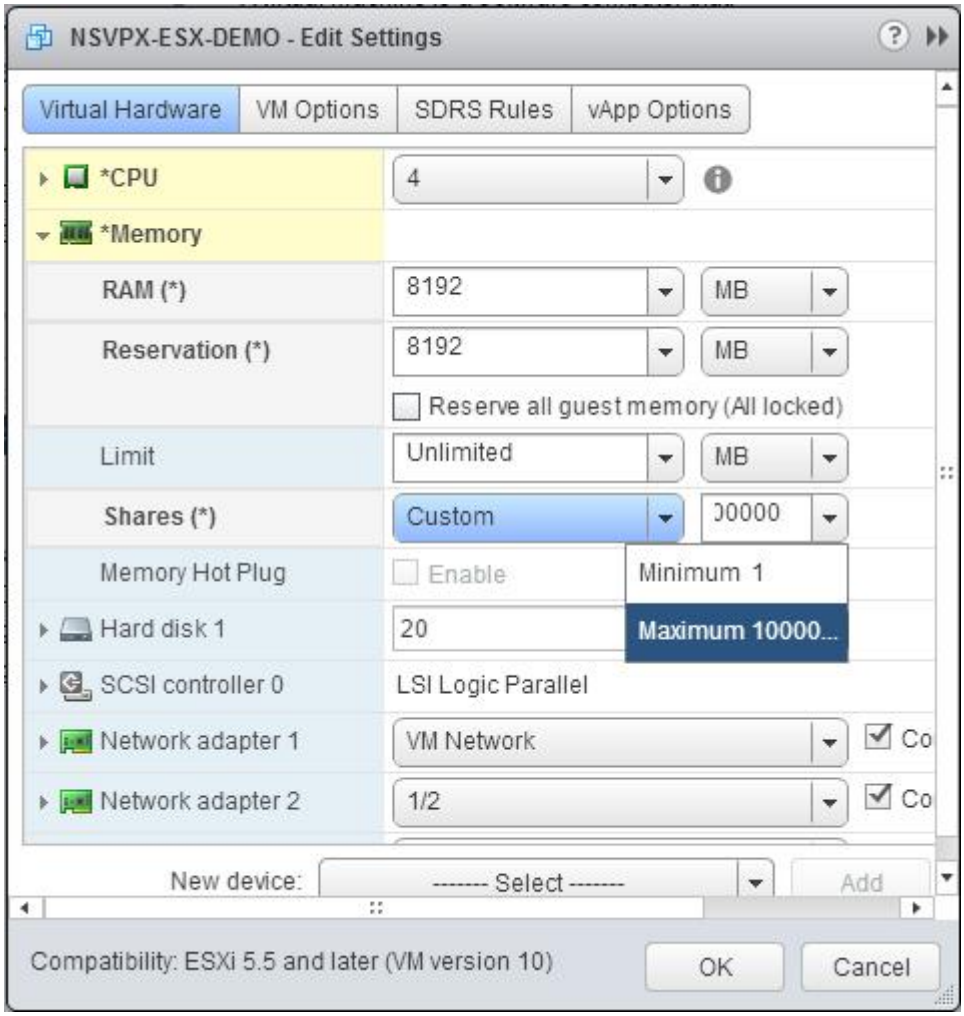
> 注
>
> NetScaler VPX アプライアンスの Advanced または Premium エディションの場合は、各 vCPU に 4 GB の RAM を割り当てるようにしてください。たとえば、vCPU の数が 4 の場合、RAM = 4×4GB = 16GB になります。



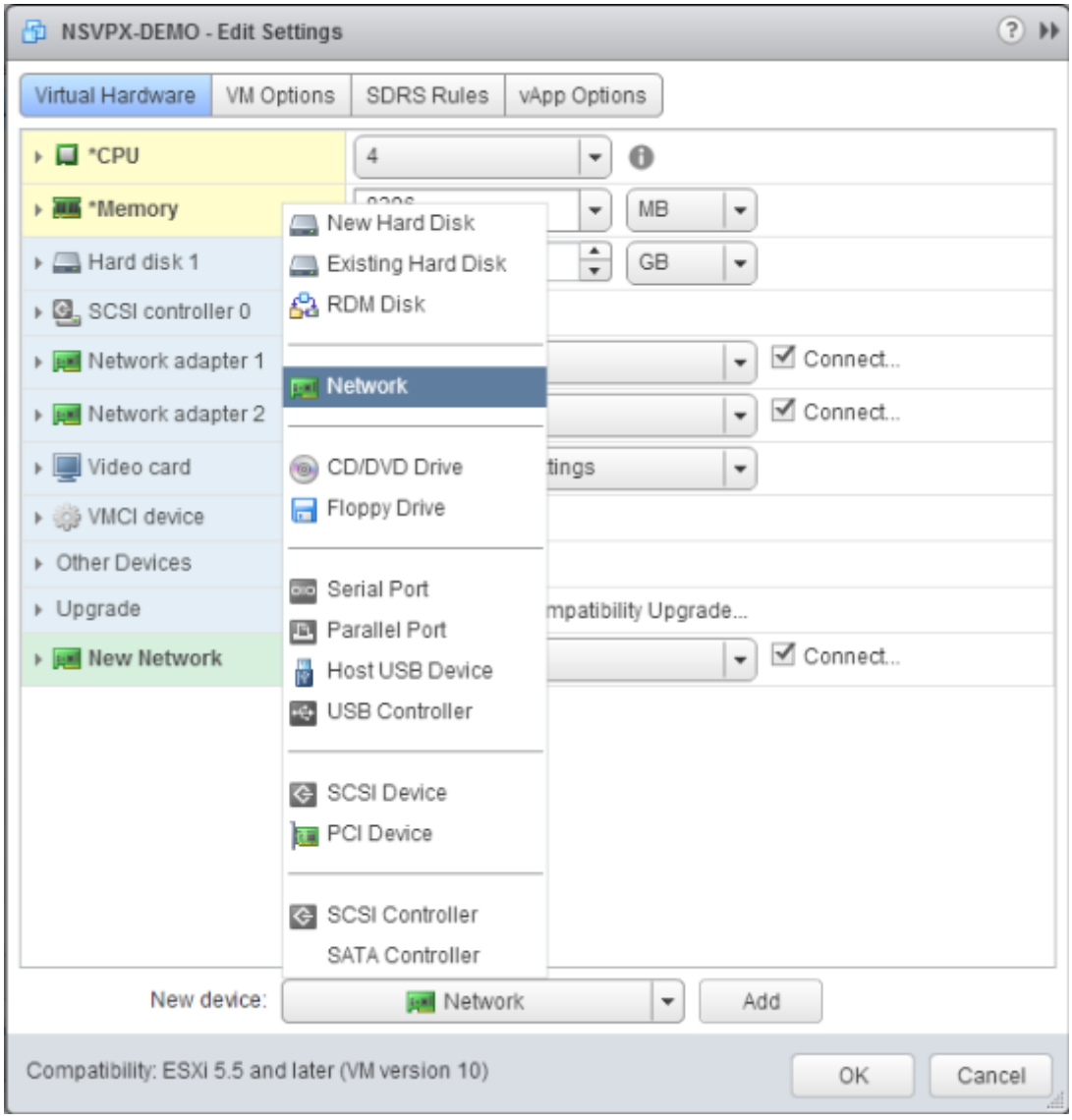
c. [制限] ドロップダウンリストで、最大値として表示されている数値を選択します。



d. [共有] ドロップダウンリストで、[カスタム] を選択し、最大値として表示される数を指定します。



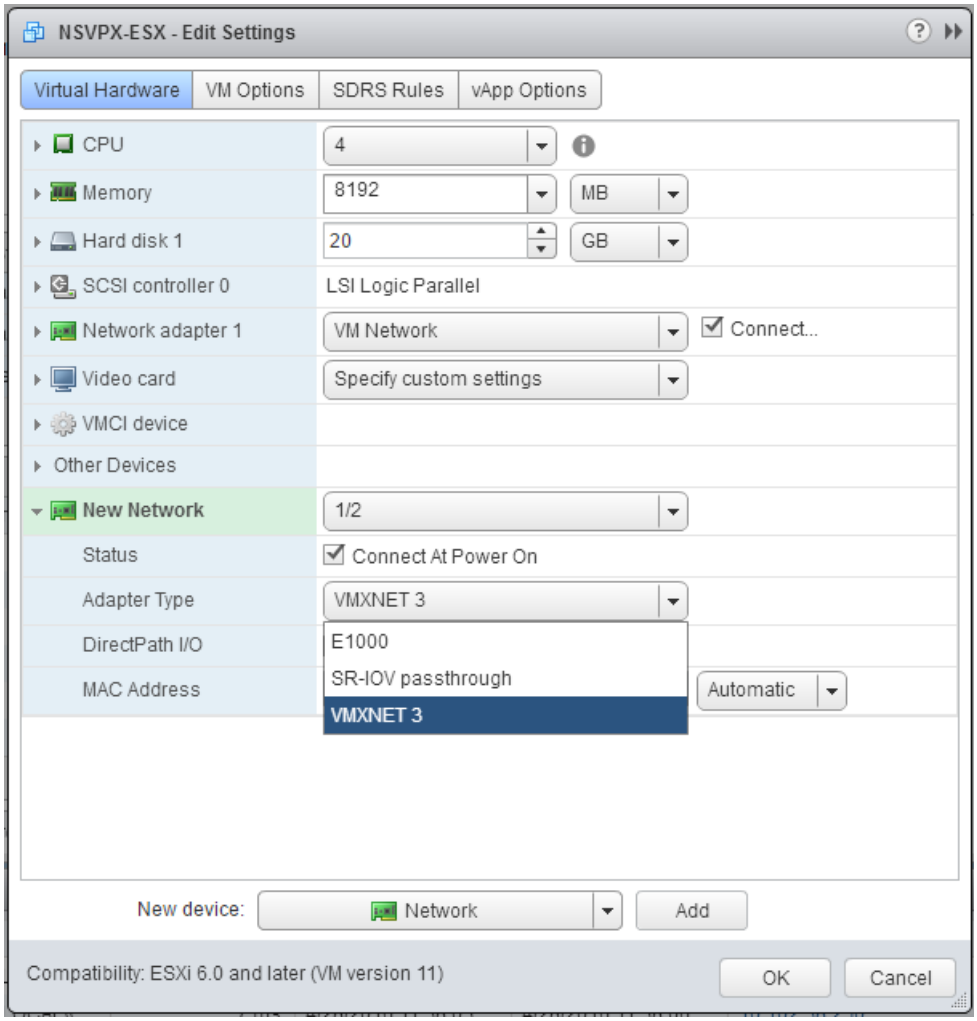
7. VMXNET3 ネットワークインターフェイスを追加します。[新しいデバイス] ドロップダウンリストから [ネットワーク] を選択し、[追加] をクリックします。



8. [New Network] セクションのドロップダウンリストからネットワークインターフェイスを選択し、次の操作を行います。

   a. [アダプタ タイプ] ドロップダウン リストで、[VMXNET3] を選択します。

   > 重要:
   >
   > デフォルトの E1000 ネットワークインターフェイスと VMXNET3 は共存できないため、E1000 ネットワークインターフェイスの削除を確認して、VMXNET3（0/1）を管理インターフェイスとして使用します。



9. [作成] または [**OK]** をクリックします。
10. NetScaler VPX インスタンスをパワーオンします。
11. NetScaler VPX インスタンスの電源が入ったら、次のコマンドを使用して構成を確認できます。

    > インターフェイスの概要を表示

出力には、設定したすべてのインターフェイスが表示されている必要があります。

```
> show interface summary
-------------------------------------------------------------------------------

      Interface  MTU        MAC                Suffix
-------------------------------------------------------------------------------

1     0/1        1500       00:0c:29:89:1d:0e  NetScaler Vir...
   rface, VMXNET3
2     1/1        9000       00:0c:29:89:1d:18  NetScaler Vir...
   rface, VMXNET3
3     1/2        9000       00:0c:29:89:1d:22  NetScaler Vir...
   rface, VMXNET3
```



```
8    4     LO/1        9000        00:0c:29:89:1d:0e    Netscaler Loopback
     interface
```

> 注
>
> VMXNET3 インターフェイスを追加して NetScaler VPX アプライアンスを再起動すると、VMware ESX ハイパーバイザーによって NIC が VPX アプライアンスに提示される順序が変更されることがあります。そのため、ネットワークアダプター 1 が常に 0/1 のままであるとは限らず、その結果 VPX アプライアンスに対する管理接続が失われることがあります。この問題を回避するには、ネットワークアダプターの仮想ネットワークを変更します。
>
> これは VMware ESX ハイパーバイザの制限です。

VMXNET3 ネットワークインターフェイスの受信リングサイズの設定

VMware ESX の VMXNET3 ネットワークインターフェイスの受信リングサイズを増やすことができます。リングサイズを大きくすると、トラフィックの突然のバーストが発生したときのパケットドロップが減少します。

> 注
>
> この機能は、リリース 14.1 ビルド 14.x 以降で使用できます。

VMXNET3 ネットワークインターフェイスのリングサイズを設定するには

コマンドプロンプトで入力します:

set interface *id* [-ringsize *positive_integer*]

VMXNET3 インターフェイスに設定できる最大リングサイズは 2048 です。固定リングタイプのみがサポートされています。設定を有効にするには、構成を保存して NetScaler VPX インスタンスを再起動する必要があります。

SR-IOV ネットワークインターフェイスの使用を **NetScaler ADC VPX** インスタンスで構成する

October 17, 2024

VMware ESX に Citrix ADC VPX インスタンスをインストールして構成した後、VMware vSphere Web クライアントを使用して、シングルルート I/O 仮想化（SR-IOV）ネットワークインターフェイスを使用するように仮想アプライアンスを構成できます。

制限事項

SR-IOV ネットワークインターフェイスで構成された NetScaler VPX には、以下の制限事項があります。

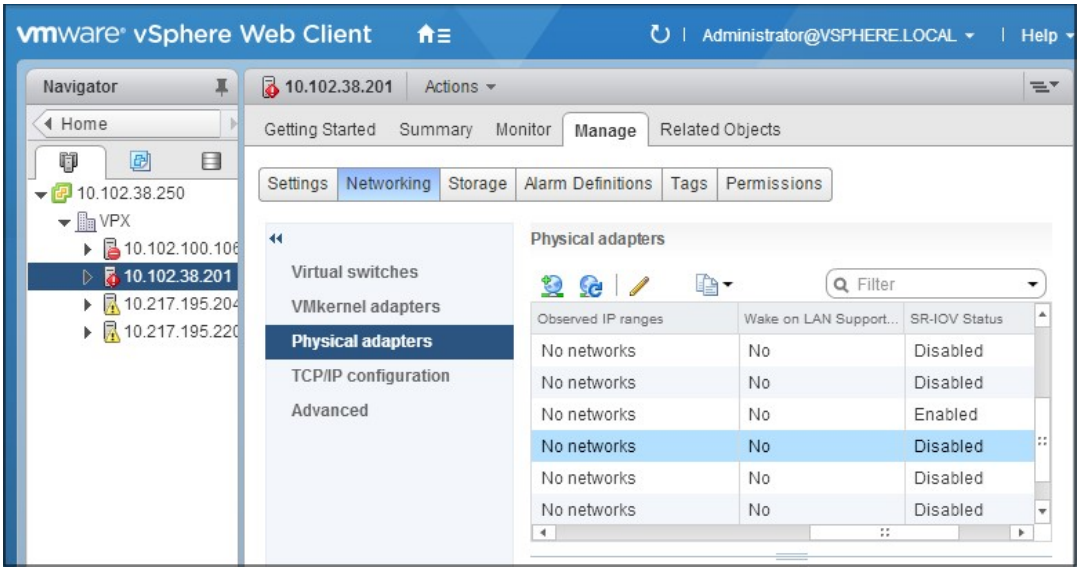
- 次の機能は、ESX VPX 上の Intel 82599 10G NIC を使用する SR-IOV インターフェイスではサポートされていません。
  - L2 モード切り替え
  - スタティックリンク集約および LACP
  - クラスタリング
  - 管理パーティション化［共有 VLAN モード］
  - 高可用性［アクティブ/アクティブモード］
  - ジャンボフレーム
  - IPv6
- KVM VPX 上の Intel 82599 10G NIC を搭載した SR-IOV インターフェイスでは、次の機能はサポートされていません。
  - スタティックリンク集約および LACP
  - L2 モード切り替え
  - クラスタリング
  - 管理パーティション化［共有 VLAN モード］
  - 高可用性 [アクティブ—アクティブモード]
  - ジャンボフレーム
  - IPv6
  - `ip link`コマンドによる SR-IOV VF インターフェイスのハイパーバイザでの VLAN 設定はサポートされていません

前提要件

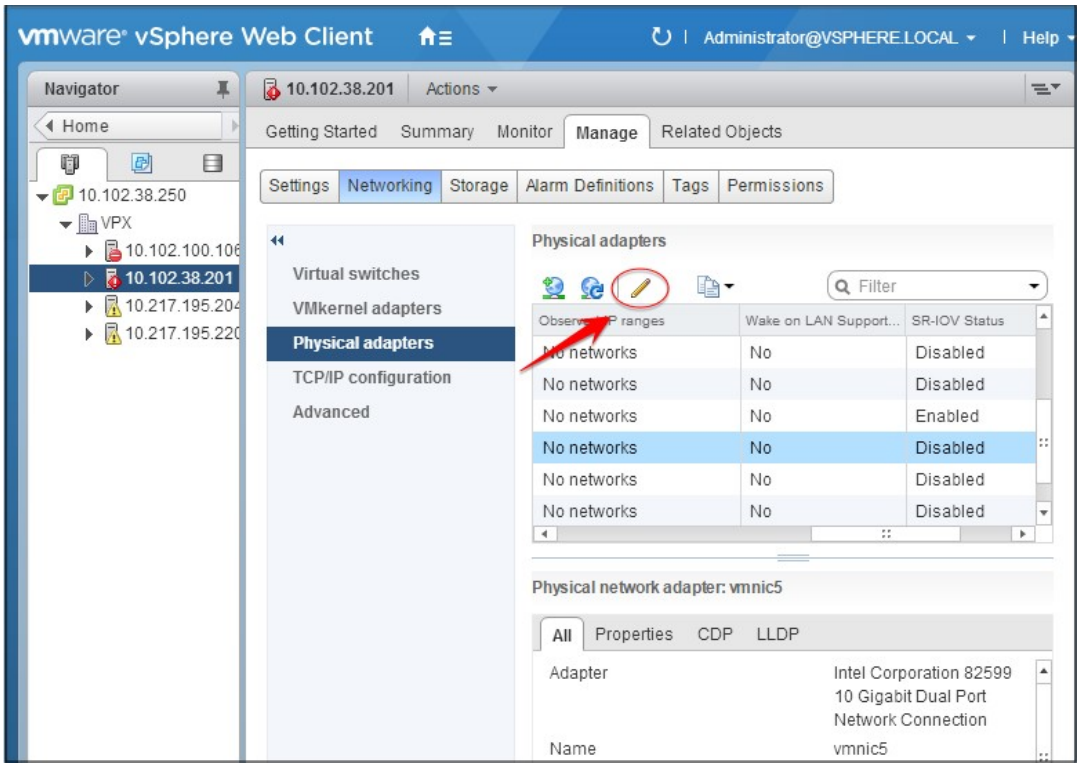
- 必ず ESX ホストに次のいずれかの NIC を追加してください。
  - Intel 82599 NIC、IXGBE ドライバーバージョン 3.7.13.7.14iov 以降が推奨されます。
  - Mellanox ConnectX-4 NIC
- ホスト物理アダプタで SR-IOV を有効にします。

  以下の手順に従って、ホスト物理アダプタで SR-IOV を有効にします。

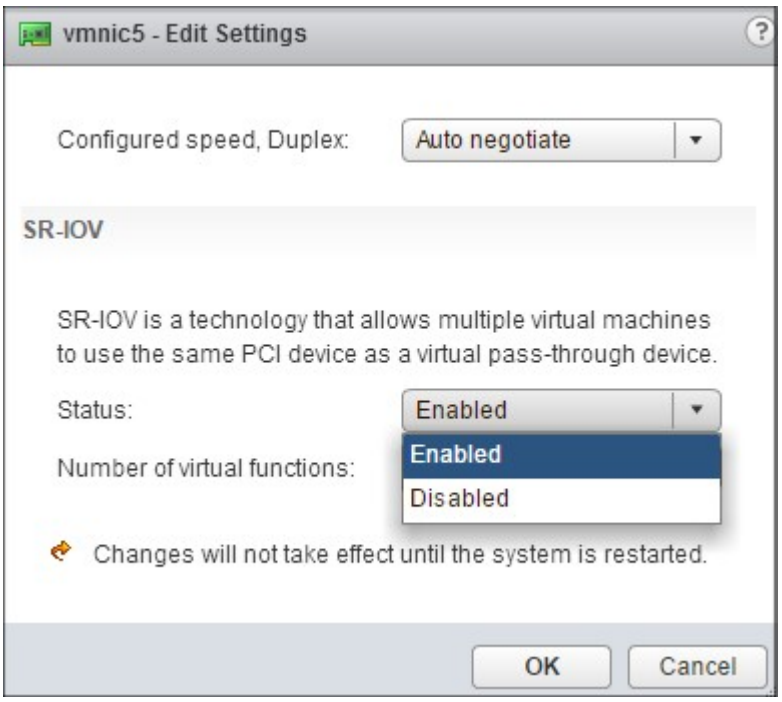
  1. vSphere Web クライアントで、ホストに移動します。
  2. [管理**] > [**ネットワーク] タブで、[物理アダプター] を選択します。[SR-IOV Status］フィールドに、物理アダプターが SR-IOV をサポートしているかどうかが表示されます。



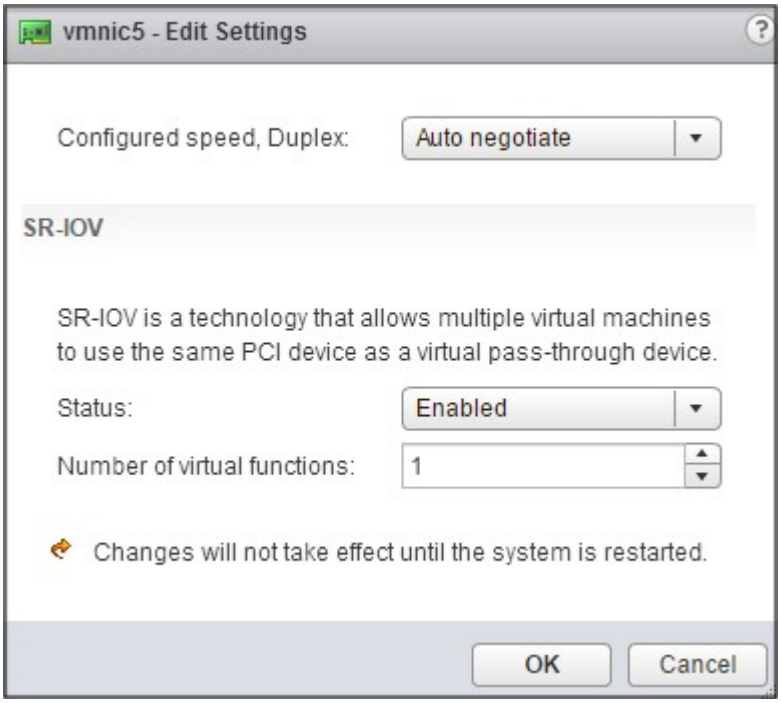
3. 物理アダプタを選択し、鉛筆アイコンをクリックして [設定の編集] ダイアログボックスを開きます。



4.「SR-IOV」で、「ステータス」ドロップダウンリストから「有効」を選択します。



5. [仮想関数の数] フィールドに、アダプタに対して構成する仮想関数の数を入力します。



6. [作成] または [**OK]** をクリックします。
7. ホストを再起動します。

- 分散仮想スイッチ (DVS) と`Portgroups`を作成します。手順については、VMware のドキュメントを参照してください。

> 注
>
> Citrix は、DVS および`Portgroups`でのみ SR-IOV 構成を認定しています。

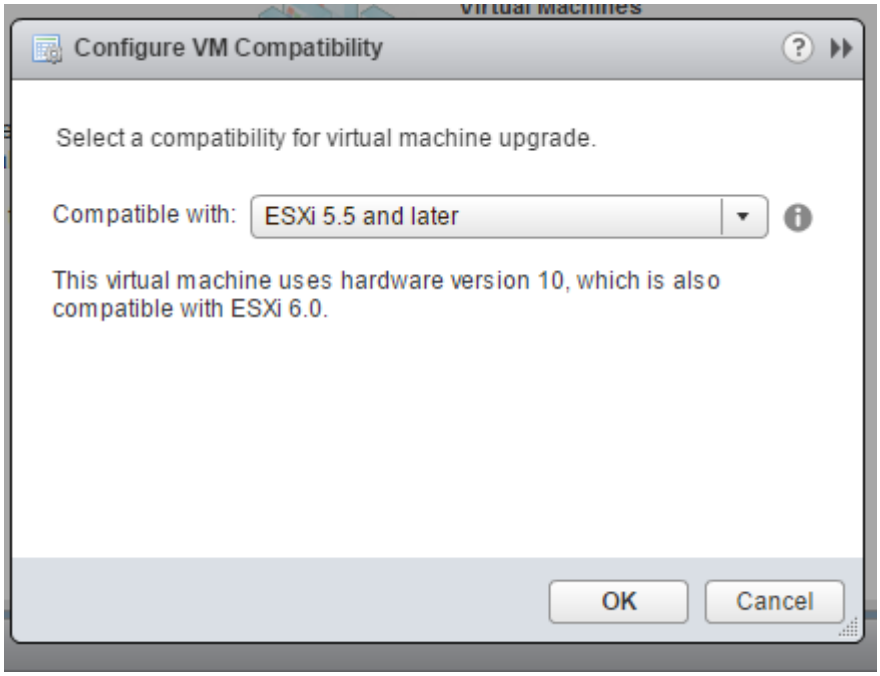
VMware vSphere Web クライアントを使用して **SR-IOV** ネットワークインターフェイスを使用するように **Citrix ADC VPX** インスタンスを構成するには:

1. vSphere Web クライアントで、ホストとクラスタを選択します。
2. 次のように、NetScaler VPX インスタンスの互換性設定を ESX 5.5 以降にアップグレードします。

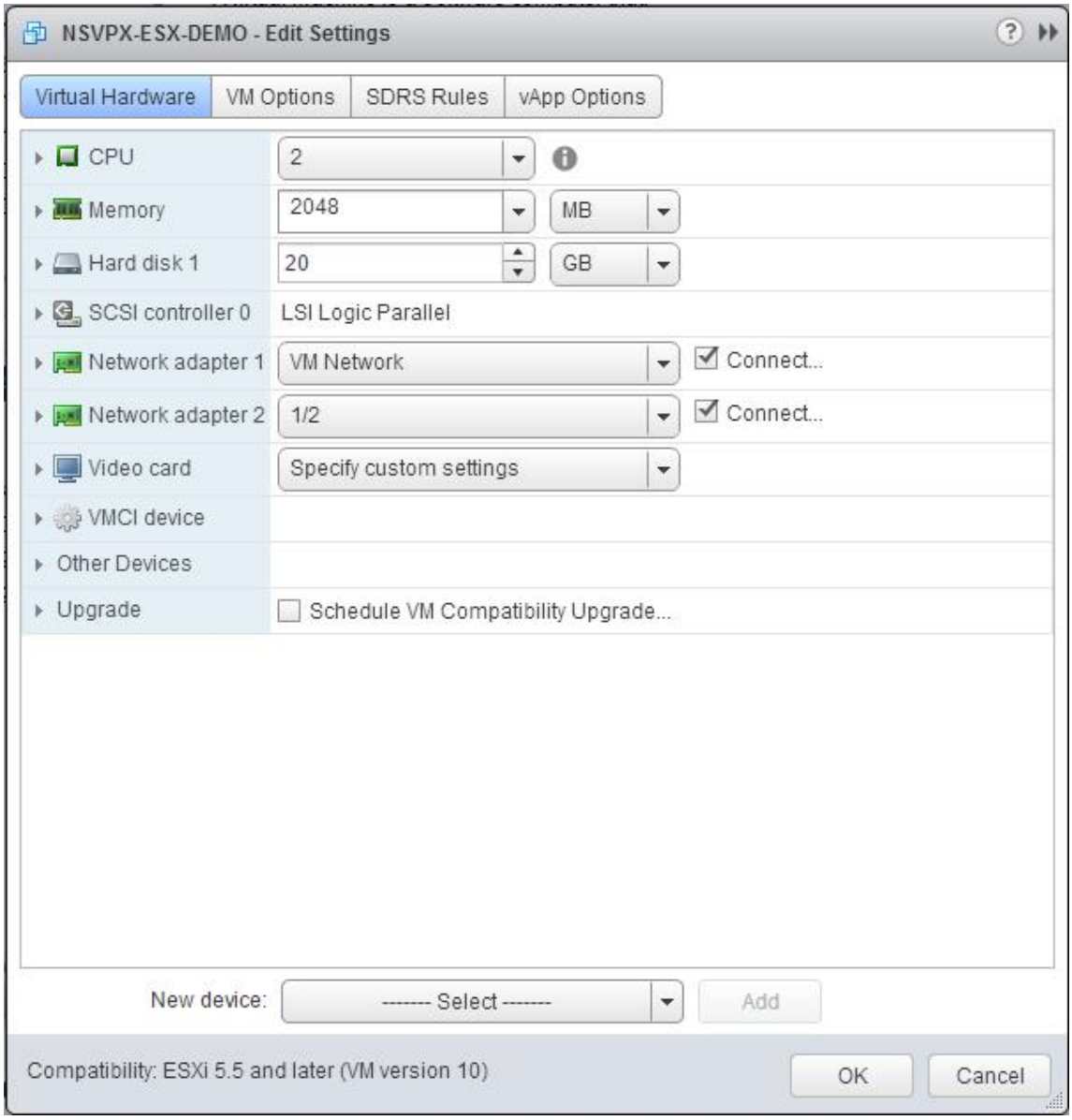
   a. NetScaler VPX インスタンスの電源を切ります。

   b. NetScaler VPX インスタンスを右クリックし、「互換性」**>**「仮想マシンの互換性のアップグレード」を選択します。

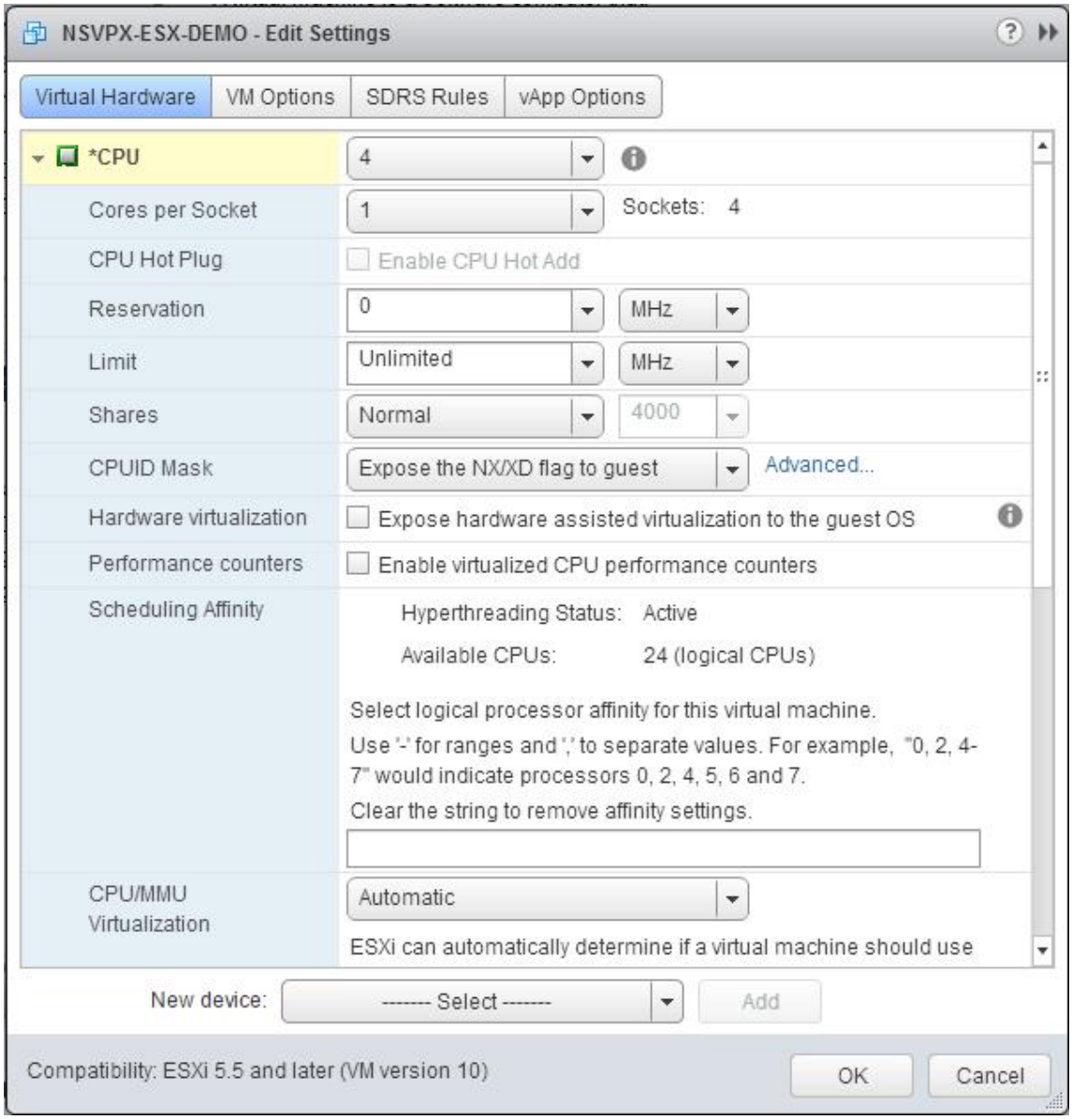
   c. [制限] ドロップダウンリストで、最大値として表示されている数値を選択します。



3. NetScaler VPX インスタンスを右クリックし、**[**設定の編集**]** をクリックします。



4. [**<virtual_appliance> 設定の編集] ダイアログボックスで、[CPU**] セクションをクリックします。



5. [**CPU**] セクションで、次の設定を更新します。

   - CPU の数
   - ソケット数
   - 予約
   - 上限
   - 共有

   値を次のように設定します:

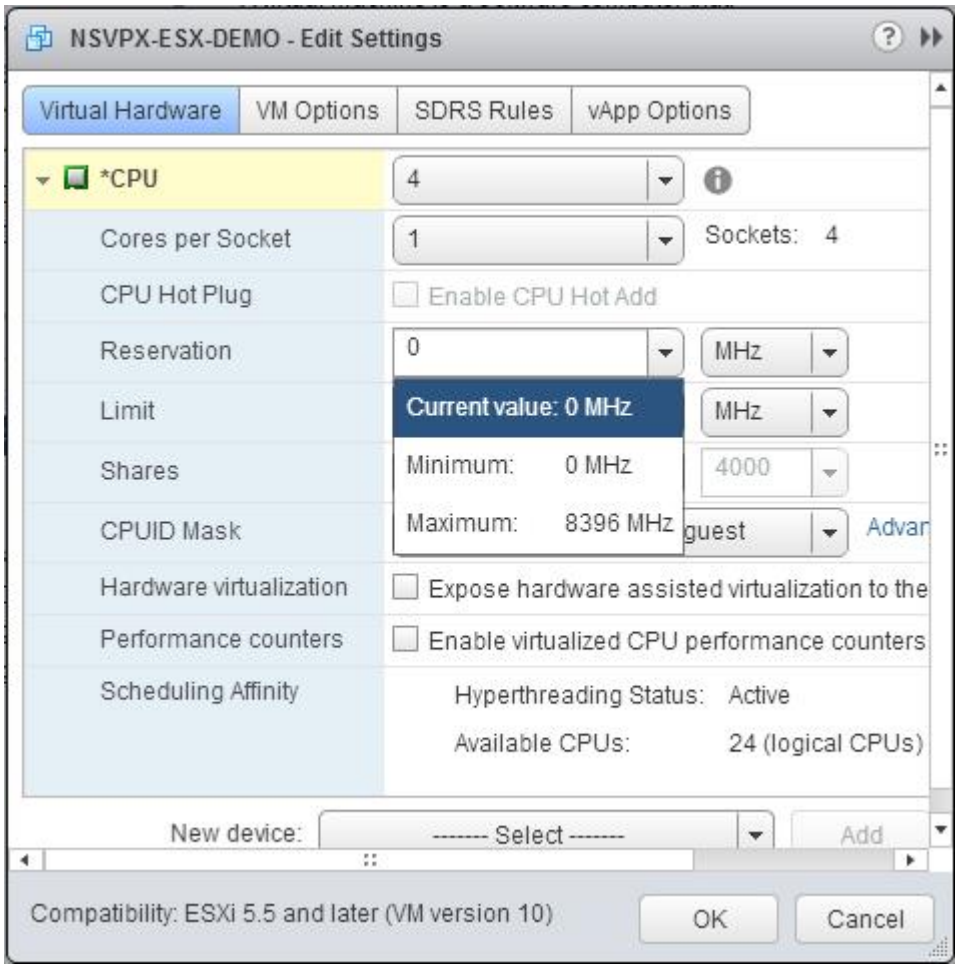
   a. **CPU** ドロップダウンリストで、仮想アプライアンスに割り当てる CPU の数を選択します。

   b. [予約] ドロップダウンリストで、最大値として表示される番号を選択します。

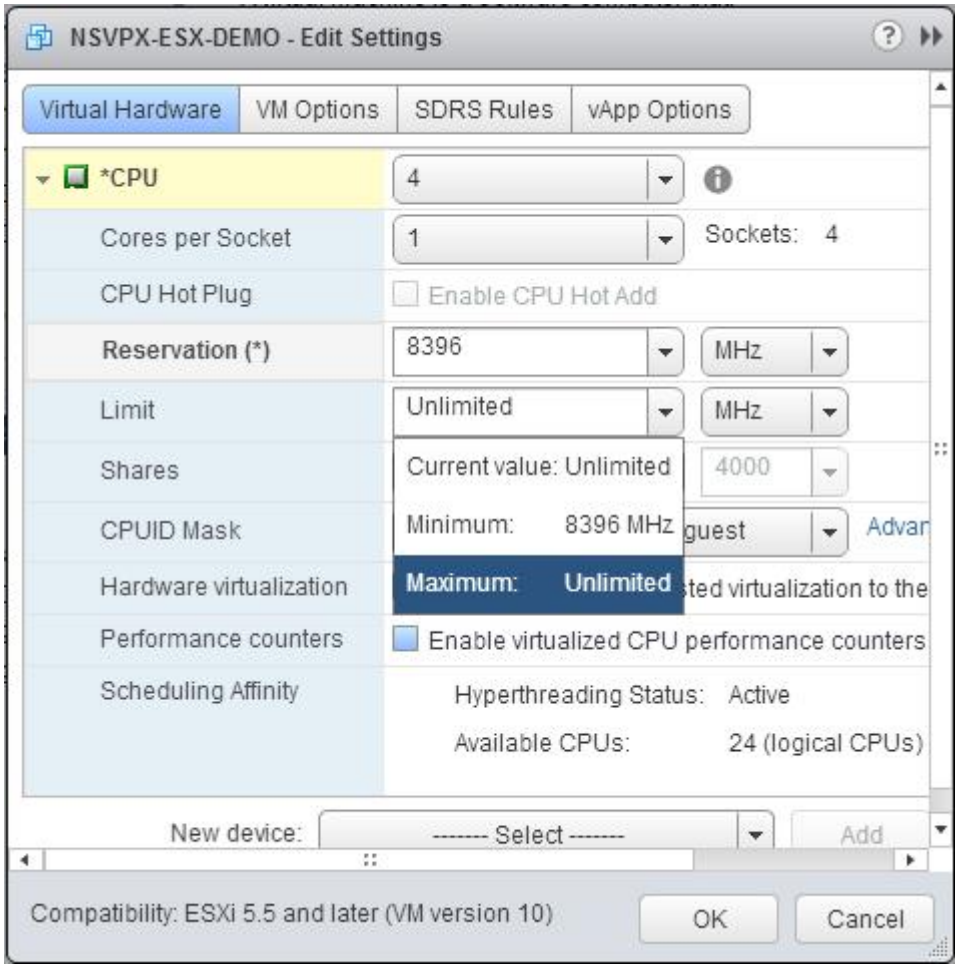
c. (オプション)「**CPU** ホットプラグ」フィールドで、「**CPU** ホットアドを有効にする」チェックボックスをオンまたはオフにします。

> 注
>
> Citrix では、デフォルト (無効) を受け入れることをお勧めします。

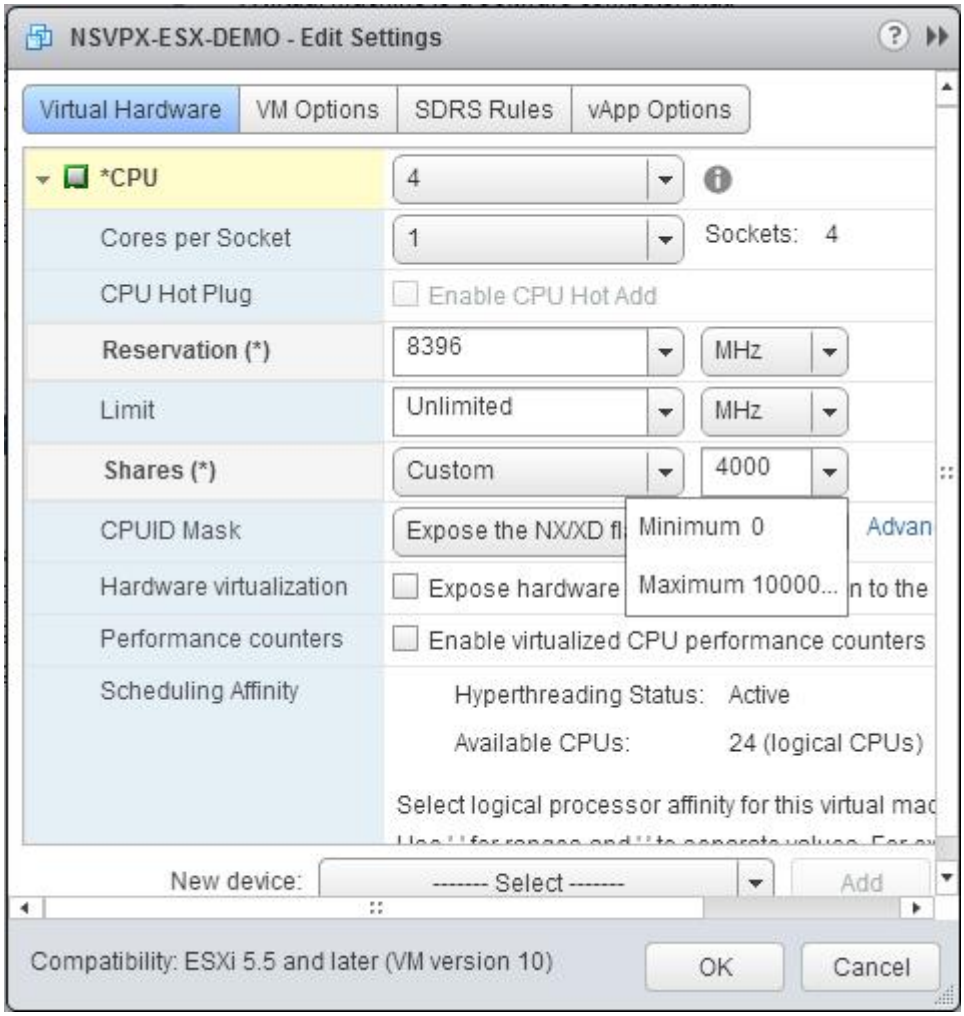
d. [共有] ドロップダウンリストで、[カスタム] を選択し、最大値として表示される数値を選択します。



e. [制限] ドロップダウンリストで、最大値として表示されている数値を選択します。



f. [共有] ドロップダウンリストで、[カスタム] を選択し、最大値として表示される数を指定します。



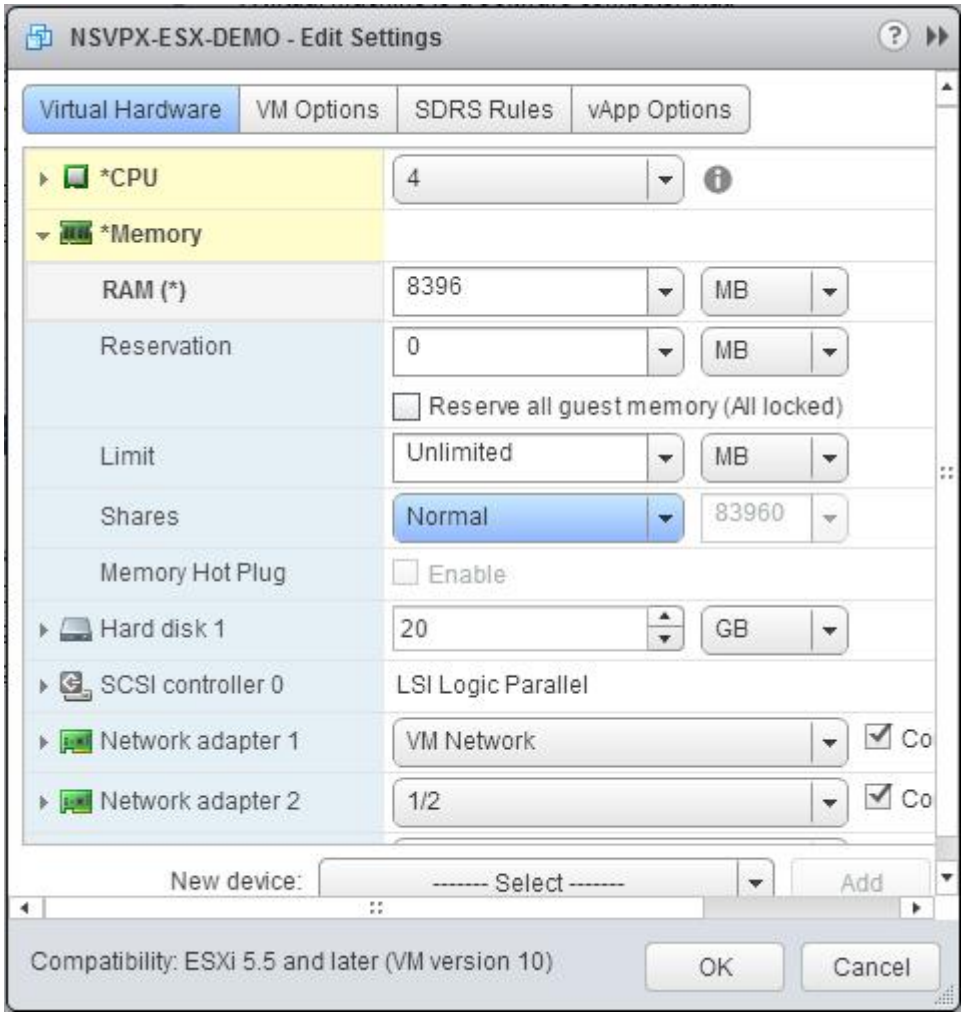
6. [メモリ] セクションで、次の設定を更新します。

   - メモリのサイズ
   - 予約
   - 上限
   - 共有

   値を次のように設定します:

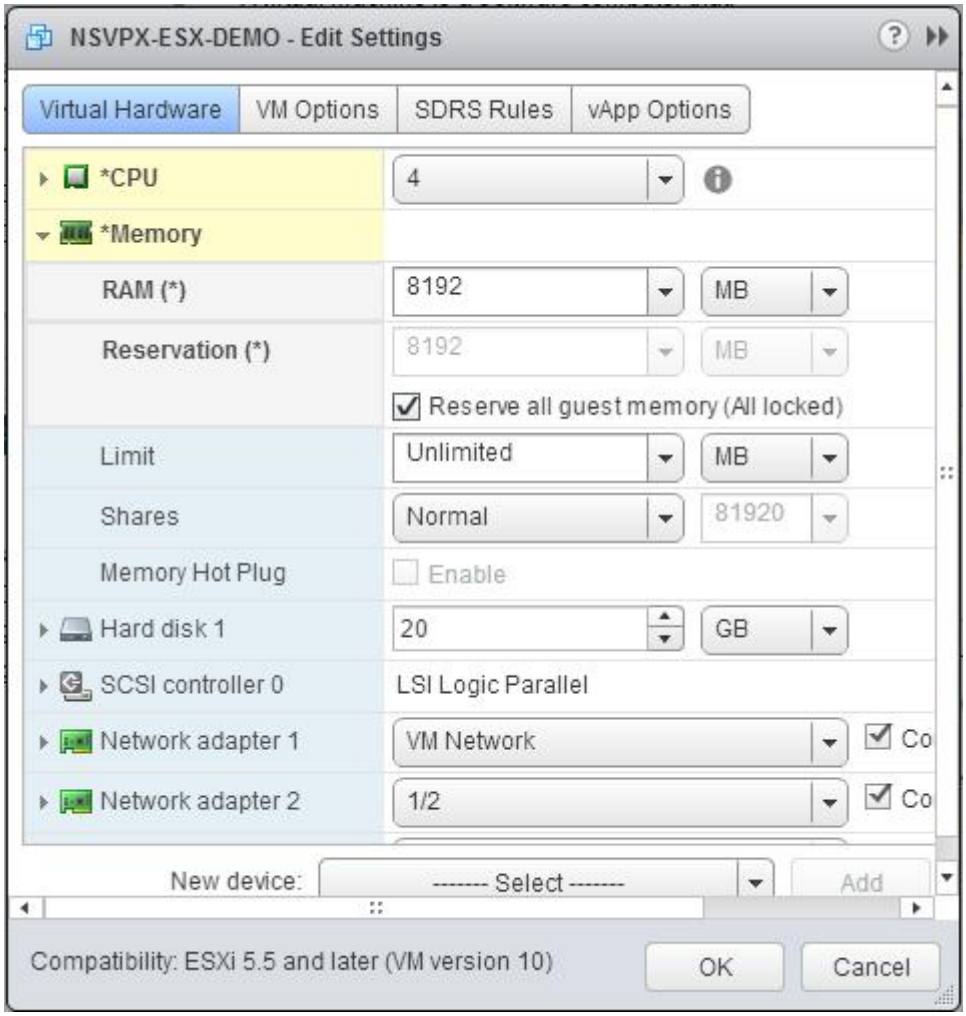
   a. [**RAM**] ドロップダウンリストで、RAM のサイズを選択します。これは vCPU の数 x 2 GB でなければなりません。たとえば、vCPU の数が 4 の場合、RAM = 4×2GB = 8GB になります。

   > 注
   >
   > NetScaler VPX アプライアンスの Advanced または Premium エディションの場合は、各 vCPU に 4 GB の RAM を割り当てるようにしてください。たとえば、vCPU の数が 4 の場合、RAM = 4×4GB = 16GB になります。

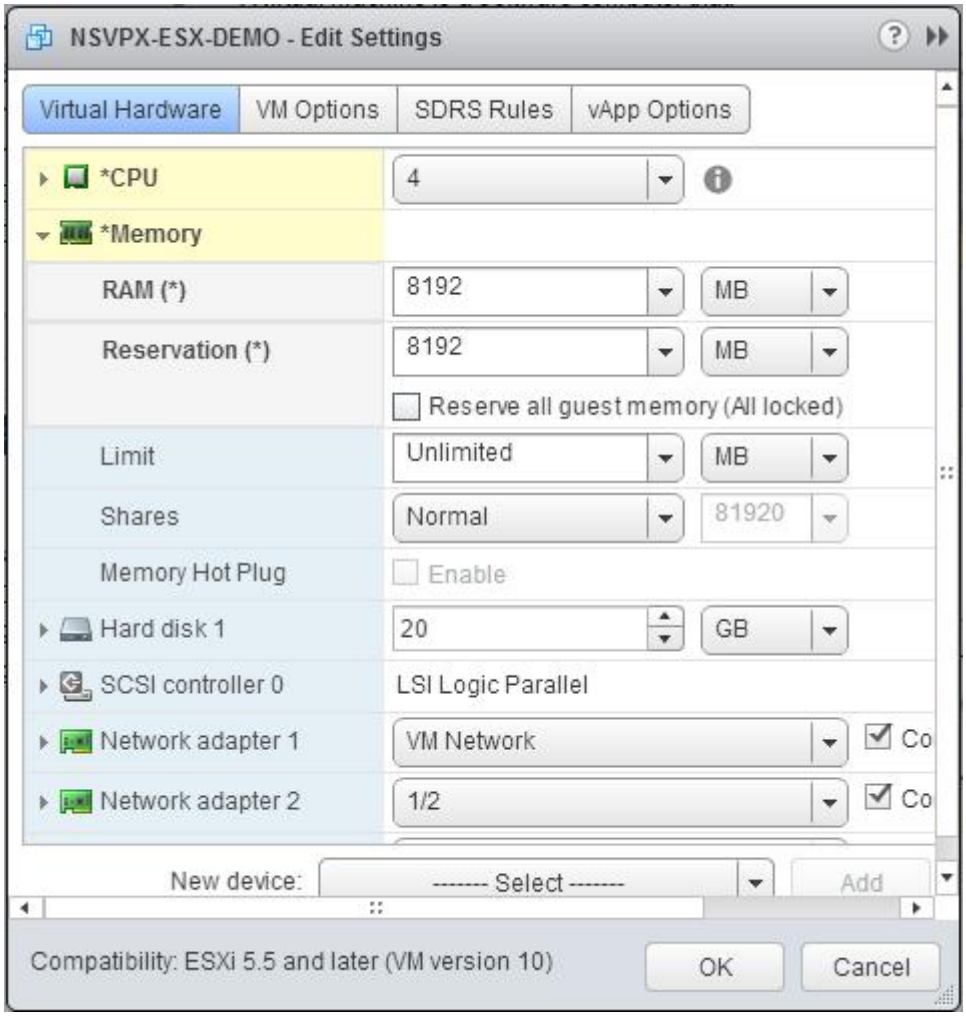


b. [予約] ドロップダウンリストで、メモリ予約の値を入力し、[すべてのゲストメモリを予約する **(**すべてのロック済み**)**] チェックボックスをオンにします。メモリ予約は vCPU の数 x 2 GB である必要があります。たとえば、vCPU の数が 4 の場合、メモリ予約は 4 x 2 GB = 8 GB である必要があります。

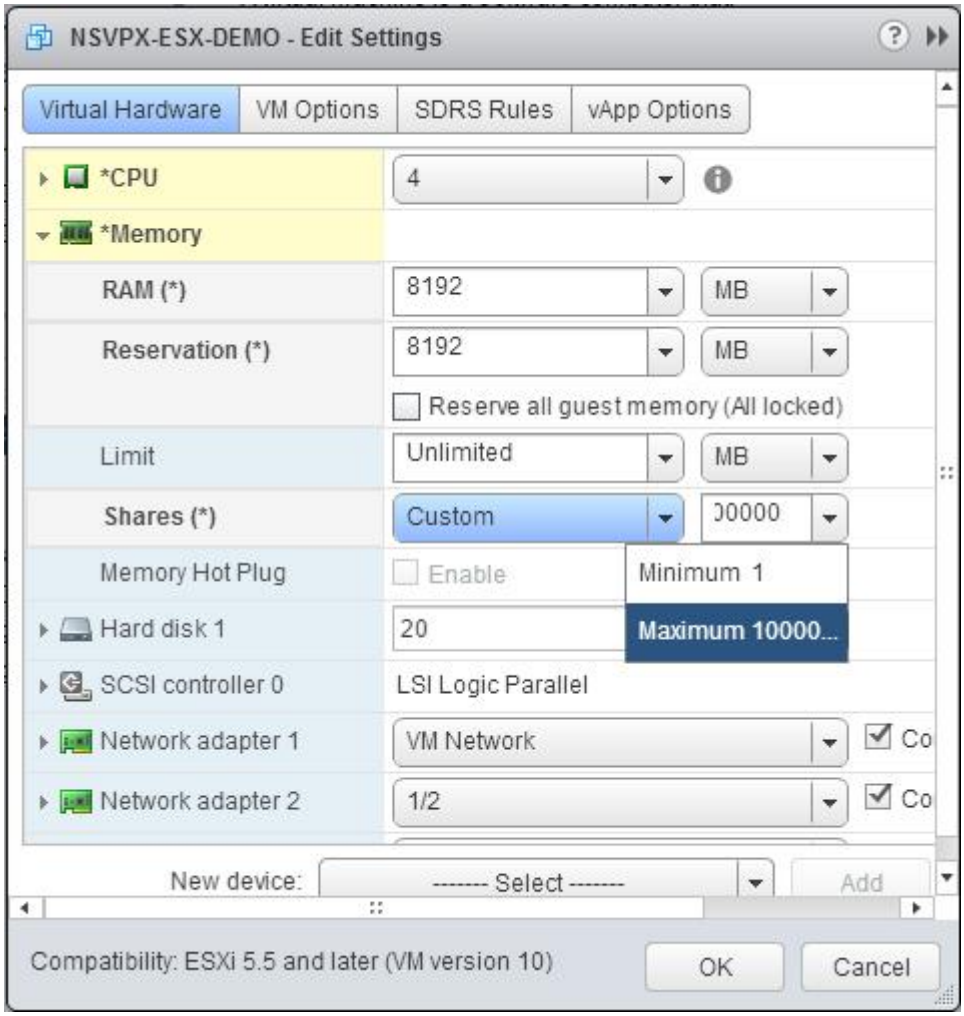
> 注
>
> NetScaler VPX アプライアンスの Advanced または Premium エディションの場合は、各 vCPU に 4 GB の RAM を割り当てるようにしてください。たとえば、vCPU の数が 4 の場合、RAM = 4×4GB = 16GB になります。



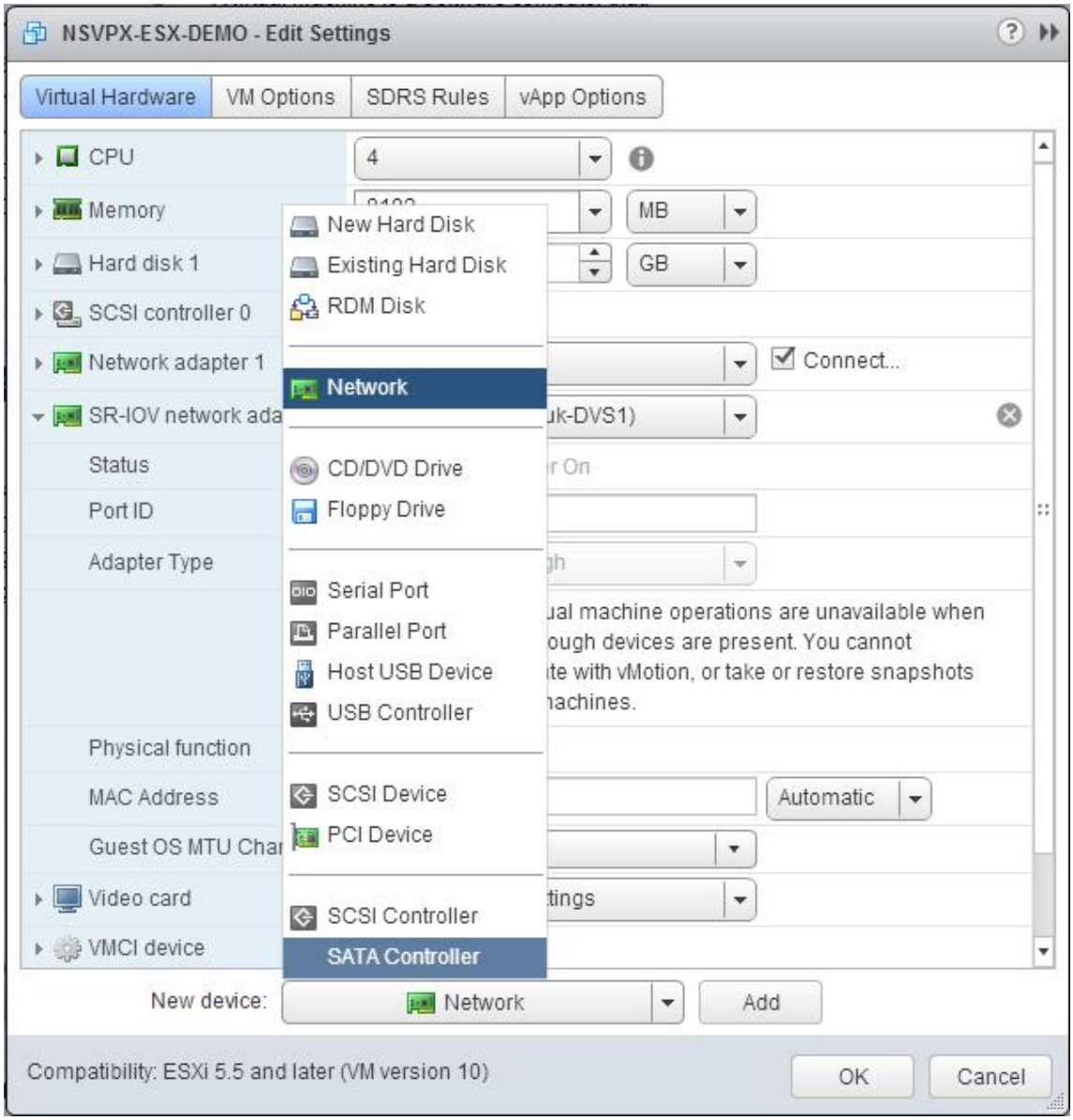
c. [制限】ドロップダウンリストで、最大値として表示されている数値を選択します。



d. [物理機能] ドロップダウンリストで、`Portgroup`にマップされている物理アダプタを選択します。

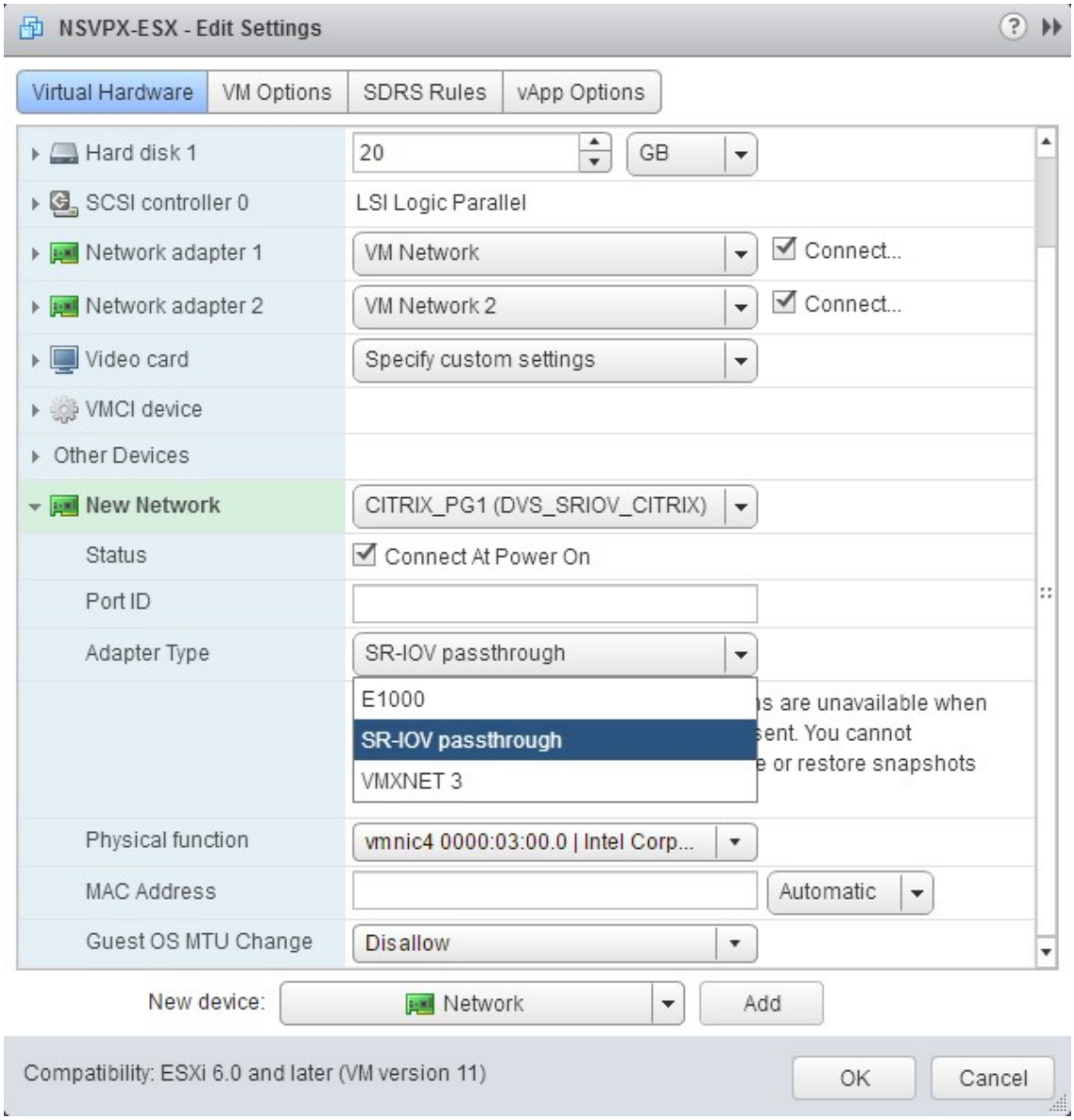


7. SR-IOV ネットワークインターフェイスを追加します。[新しいデバイス] ドロップダウンリストから [ネットワーク] を選択し、[追加] をクリックします。

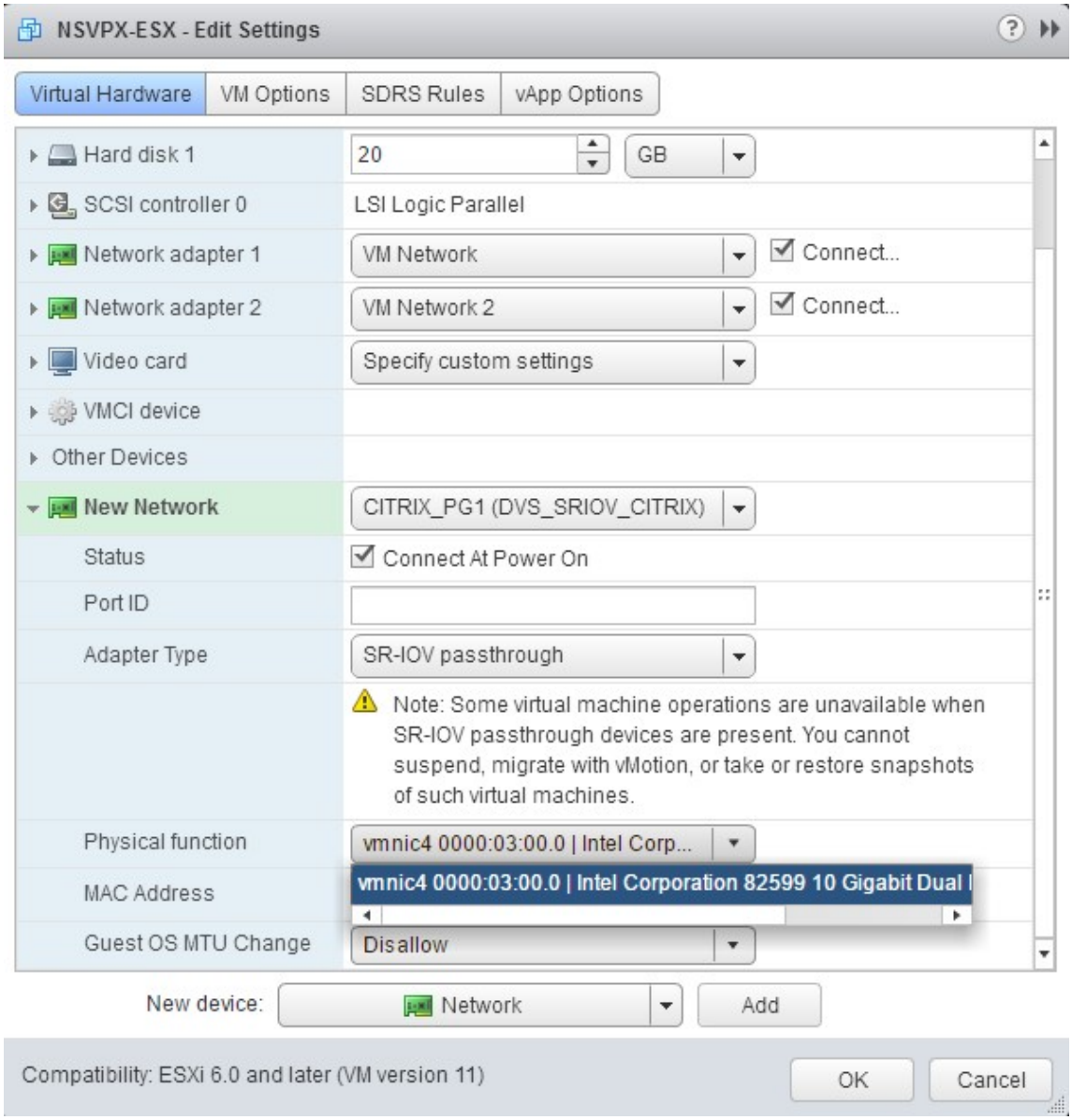


8. [新しいネットワーク] セクションで。ドロップダウンリストから、作成したPortgroupを選択し、次の操作を行います。

   a. b.「ソケットあたりのコア数」ドロップダウンリストで、ソケット数を選択します。



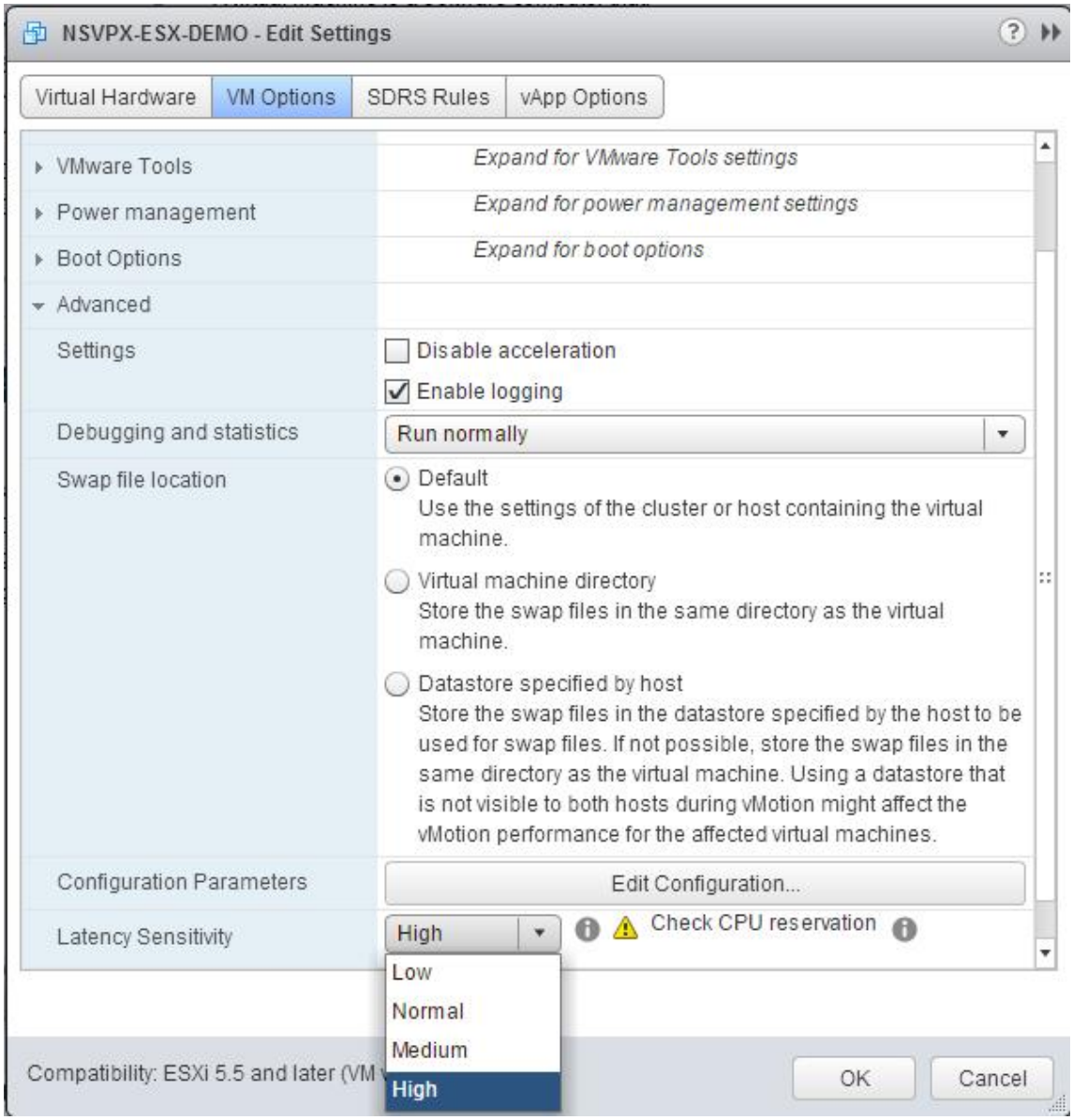
b. [物理機能] ドロップダウンリストで、`Portgroup`にマップされている物理アダプタを選択します。



c. ゲスト **OS** の **MTU** 変更ドロップダウンリストで、「許可」を選択します 。

9. [-設定の編集 <virtual_appliance>] ダイアログボックスで、[仮想マシンオプション] タブをクリックします。

10. [仮想マシンオプション] タブで、[詳細設定] セクションを選択します。[遅延感度] ドロップダウンリストから、[高] を選択します。



11. [作成] または [**OK]** をクリックします。
12. NetScaler VPX インスタンスをパワーオンします。
13. NetScaler VPX インスタンスの電源が入ったら、次のコマンドを使用して構成を確認できます。

    インターフェイスの概要を表示

出力には、設定したすべてのインターフェイスが表示されている必要があります。

```
> show interface summary
-------------------------------------------------------------------------------

      Interface  MTU        MAC                Suffix
-------------------------------------------------------------------------------
```



```
1     0/1         1500        00:0c:29:1b:81:0b    NetScaler Virtual
      Interface
2     10/1        1500        00:50:56:9f:0c:6f    Intel 82599 10G VF
      Interface
3     10/2        1500        00:50:56:9f:5c:1e    Intel 82599 10G VF
      Interface
4     10/3        1500        00:50:56:9f:02:1b    Intel 82599 10G VF
      Interface
5     10/4        1500        00:50:56:9f:5a:1d    Intel 82599 10G VF
      Interface
6     10/5        1500        00:50:56:9f:4e:0b    Intel 82599 10G VF
      Interface
7     LO/1        1500        00:0c:29:1b:81:0b    Netscaler Loopback
      interface
 Done
> show inter 10/1
1)      Interface 10/1 (Intel 82599 10G VF Interface) #1
        flags=0xe460 <ENABLED, UP, UP, HAMON, 802.1q>
        MTU=1500, native vlan=55, MAC=00:50:56:9f:0c:6f, uptime 0
           h21m53s
        Actual: media FIBER, speed 10000, duplex FULL, fctl NONE,
           throughput 10000
        LLDP Mode: NONE,                LR Priority: 1024

        RX: Pkts(838020742) Bytes(860888485431) Errs(0) Drops(2527)
           Stalls(0)
        TX: Pkts(838149954) Bytes(860895860507) Errs(0) Drops(0)
           Stalls(0)
        NIC: InDisc(0) OutDisc(0) Fctls(0) Stalls(0) Hangs(0) Muted
           (0)
        Bandwidth thresholds are not set.
 Done
```

SR-IOV モードでの SSL アクセラレーションにインテル QAT を使用するように ESX ハイパーバイザー上の NetScaler VPX を構成する

October 17, 2024

VMware ESX ハイパーバイザー上の NetScaler VPX インスタンスは、Intel QuickAssist Technology（QAT）を使用して NetScaler SSL パフォーマンスを高速化できます。インテル QAT を使用すると、レイテンシーの高い暗号処理をすべてチップにオフロードできるため、1 つまたは複数のホスト CPU を解放して他のタスクを実行できるようになります。

以前は、NetScaler データパスの暗号化処理はすべて、ホスト vCPU を使用するソフトウェアで行われていました。

> 注
>
> 現在、NetScaler VPX はインテル QAT ファミリーの C62x チップモデルのみをサポートしています。この機能は、NetScaler リリース 14.1 ビルド 8.50 以降でサポートされています。

前提条件

- ESX ホストには、1 つ以上のインテル C62x (QAT) チップが搭載されています。
- NetScaler VPX は VMware ESX のハードウェア要件を満たしています。詳細については、「VMware ESX に NetScaler VPX インスタンスをインストールする」を参照してください。

制限事項

個々の VM 用に暗号ユニットや帯域幅を予約する規定はありません。Intel QAT ハードウェアで使用可能なすべての暗号ユニットは、QAT ハードウェアを使用するすべての VM で共有されます。

インテル **QAT** を使用するためのホスト環境のセットアップ

1. インテルが提供する C62x シリーズ (QAT) チップモデル用 VMware ドライバーを VMware ホストにダウンロードしてインストールします。インテルパッケージのダウンロードとインストール手順の詳細については、VMware 用インテルクイックアシストテクノロジードライバーを参照してください。
2. ESX ホストで SR-IOV を有効にします。
3. 仮想マシンを作成します。仮想マシンを作成するときは、パフォーマンス要件を満たす適切な数の PCI デバイスを割り当てます。

> 注
>
> 各 C62x (QAT) チップには、最大 3 つの個別の PCI エンドポイントを設定できます。各エンドポイントは VF の論理的な集合であり、チップの他の PCI エンドポイントと帯域幅を均等に共有します。各エンドポイントには、最大 16 個の PCI デバイスとして表示される VF を最大 16 個設定できます。これらのデバイスを VM に追加すると、QAT チップを使用して暗号アクセラレーションを実行できます。

注意事項

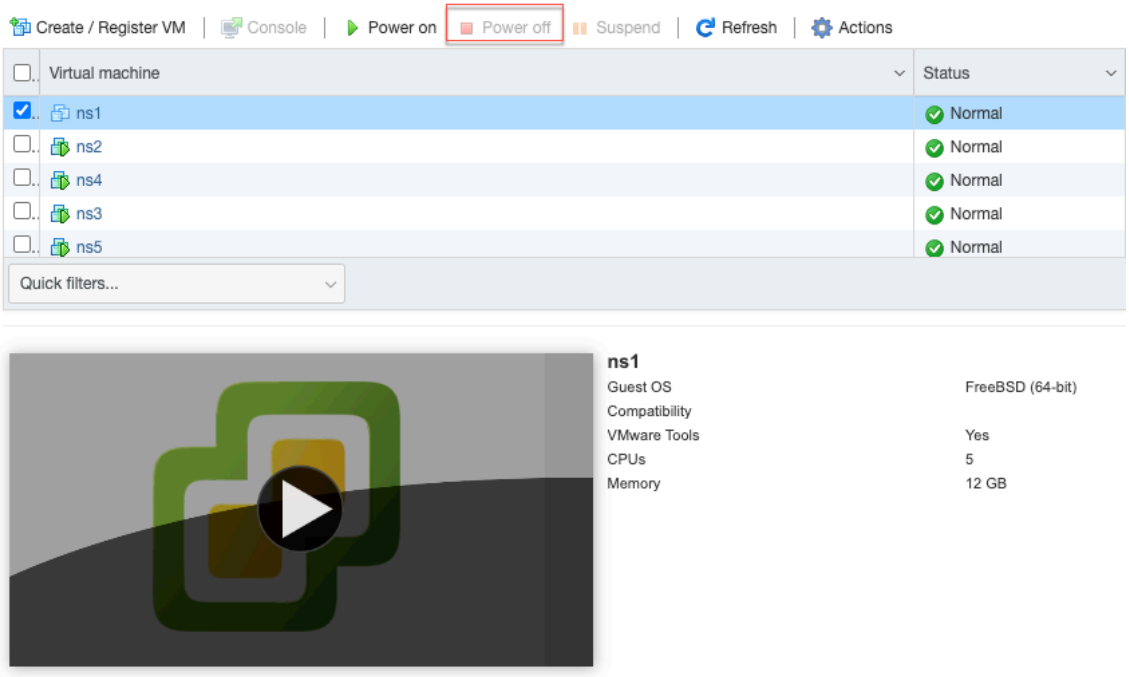
- 仮想マシンの暗号化要件が複数の QAT PCI エンドポイント/チップを使用することである場合は、対応する PCI デバイス/VF をラウンドロビン方式で選択して対称的に配信することをお勧めします。
- 選択する PCI デバイスの数は、ライセンスされている vCPU の数 (管理 vCPU 数は含まない) と同じにすることをお勧めします。利用可能な vCPU 数よりも多くの PCI デバイスを追加しても、必ずしもパフォーマンスが向上するわけではありません。

例

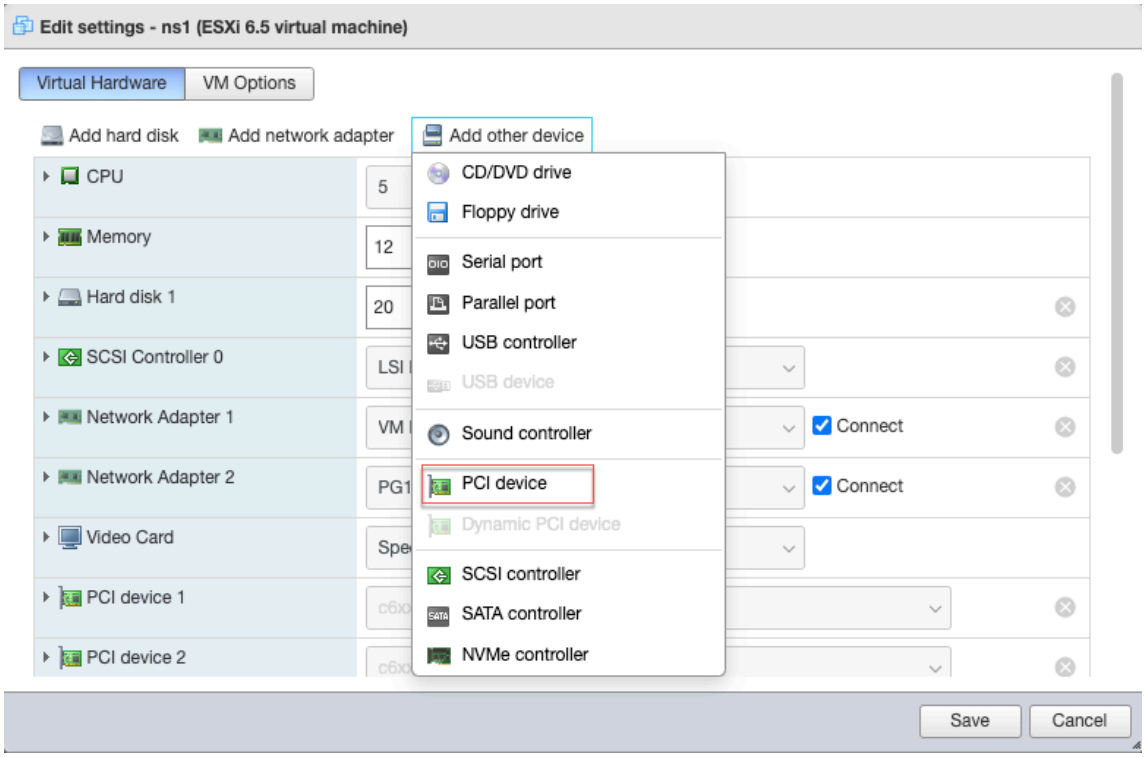
3 つのエンドポイントを持つ 1 つの Intel C62x チップを搭載した ESX ホストを考えてみましょう。6 個の vCPU を搭載した VM をプロビジョニングする場合、各エンドポイントから 2 つの VF を選択し、それらを VM に割り当てます。このような割り当てにより、仮想マシンの暗号ユニットを効果的かつ均等に分散できます。使用可能な vCPU の合計のうち、デフォルトで 1 つの vCPU が管理プレーン用に予約され、残りの vCPU はデータプレーン PE で使用できます。

vSphere ウェブクライアントを使用して **QAT** 仮想マシンを **VPX** に割り当てる

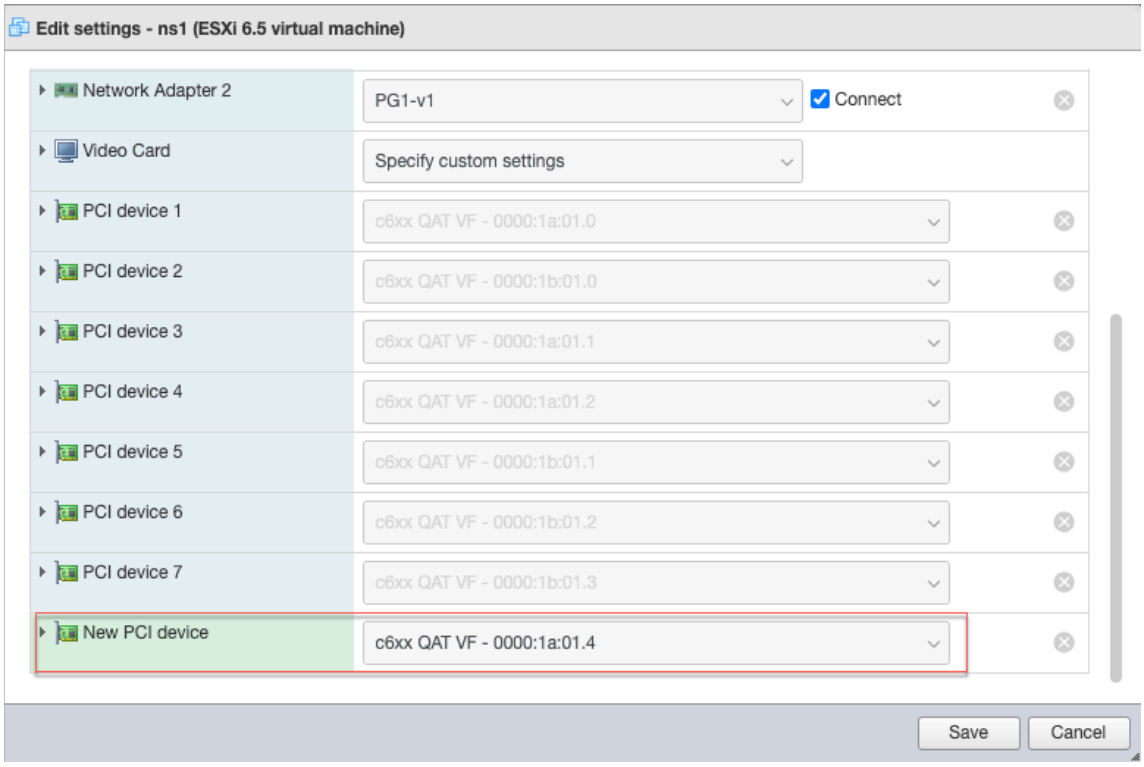
1. vSphere Web Client で、仮想マシンが配置されている ESX ホストに移動し、[パワーオフ] をクリックします。



2. [アクション] **>** **[**設定の編集**]** **>** **[**他のデバイスの追加] に移動し、[PCI デバイス] を選択します。



3. 新しく追加した PCI デバイスに、c6xx QAT VF を割り当て、設定を保存します。



4. VM を再度パワーオンします。
5. NetScaler CLI で`stat ssl`コマンドを実行して SSL の概要を表示し、QAT VF を VPX に割り当てた後に

SSL カードを確認します。

展開について

このデプロイメントは、次のコンポーネント仕様でテストされました:

- **NetScaler VPX** バージョンとビルド:14.1—8.50
- **VMware ESXi** バージョン**:**7.0.3 (ビルド 20036589)
- VMware 用インテル **C62x QAT** ドライバーバージョン:1.5.1.54

E1000 から **SR-IOV** または **VMXNET3** ネットワークインターフェイスへの **NetScaler VPX** の移行

October 17, 2024

2018 年 5 月 24 日

E1000 ネットワークインターフェイスを使用する既存の NetScaler VPX インスタンスを、SR-IOV または VMXNET3 ネットワークインターフェイスを使用するように構成できます。

既存の NetScaler VPX インスタンスを SR-IOV ネットワーク インターフェイスを使用するように構成するには、「SR-IOV ネットワーク インターフェイスを使用するように NetScaler VPX インスタンスを構成する」を参照してください。

VMXNET3 ネットワーク インターフェイスを使用するように既存の NetScaler VPX インスタンスを構成するには、「VMXNET3 ネットワーク インターフェイスを使用するように NetScaler VPX インスタンスを構成する」を参照してください。

PCI パススルーネットワークインターフェイスを使用するように **NetScaler VPX** インスタンスを構成する

October 17, 2024

概要

VMware ESX Server に NetScaler VPX インスタンスをインストールして構成したら、vSphere Web Client を使用して、PCI パススルーネットワークインターフェイスを使用するように仮想アプライアンスを構成できます。

PCI パススルー機能では、ゲスト仮想マシンからホストに接続された物理 PCI および PCIe デバイスに直接アクセスできます。

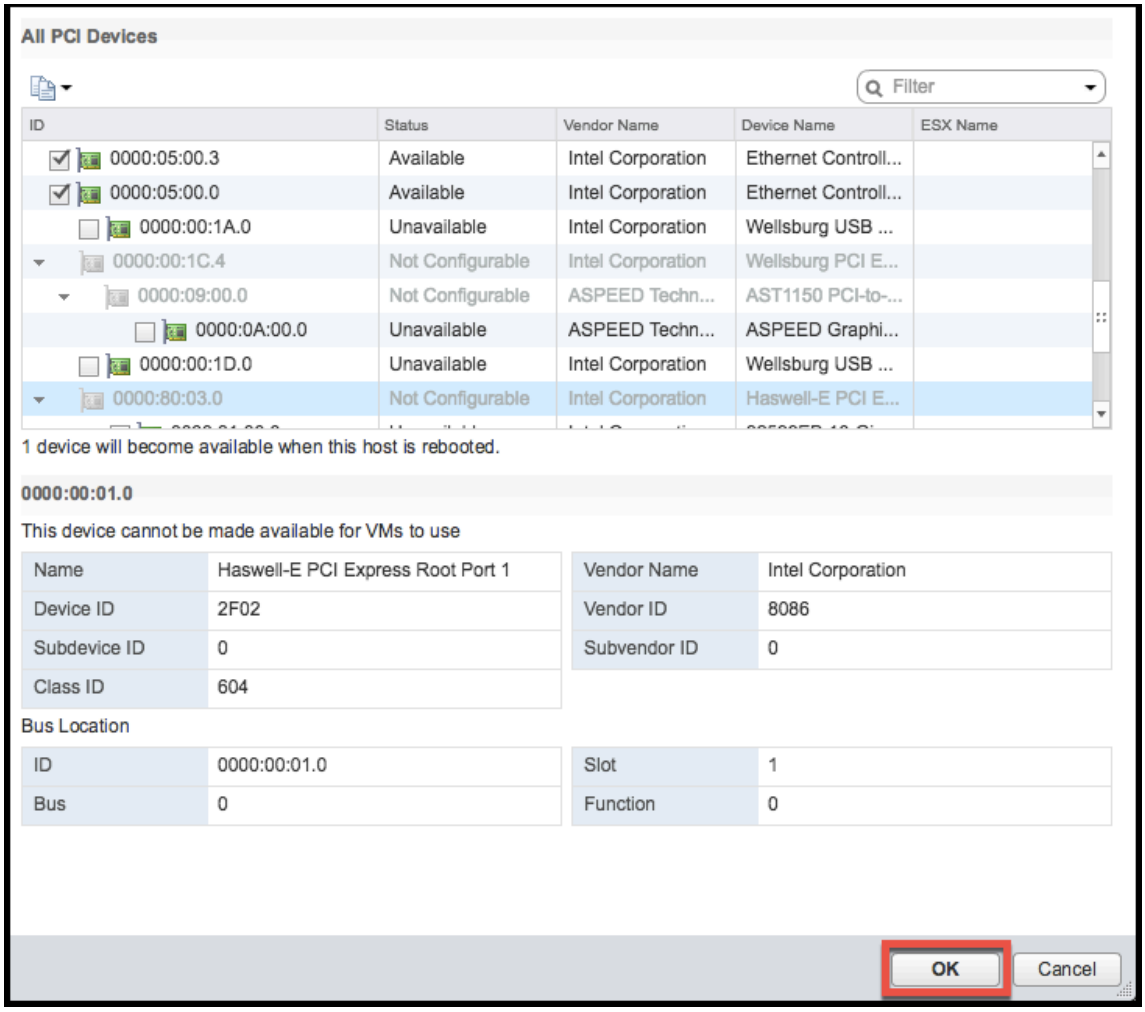
前提条件

- ホストの Intel XL710 NIC のファームウェアバージョンは、5.04 です。
- ホストに接続され構成されている PCI パススルーデバイス
- サポートされている NIC:
  - Intel X710 10G NIC
  - Intel XL710 デュアルポート 40G NIC
  - Intel XL710 シングルポート 40G NIC
  - Intel XXV710 デュアルポート 25G NIC

ホスト上のパススルー・デバイスの構成

仮想マシンでパススルー PCI デバイスを構成する前に、ホストマシン上でそれを構成する必要があります。ホストでパススルーデバイスを構成するには次の手順を実行します。

1. vSphere Web クライアントのナビゲーターパネルからホストを選択します。
2. 管理 > 設定 > **PCI** デバイス をクリックします。すべての利用可能なパススルーデバイスが表示されます。
3. 構成するデバイスを右クリックし、[**Edit**] をクリックします。
4. 「**PCI** デバイスの可用性の編集」ウィンドウが表示されます。
5. パススルーに使用するデバイスを選択し、[**OK**] をクリックします。

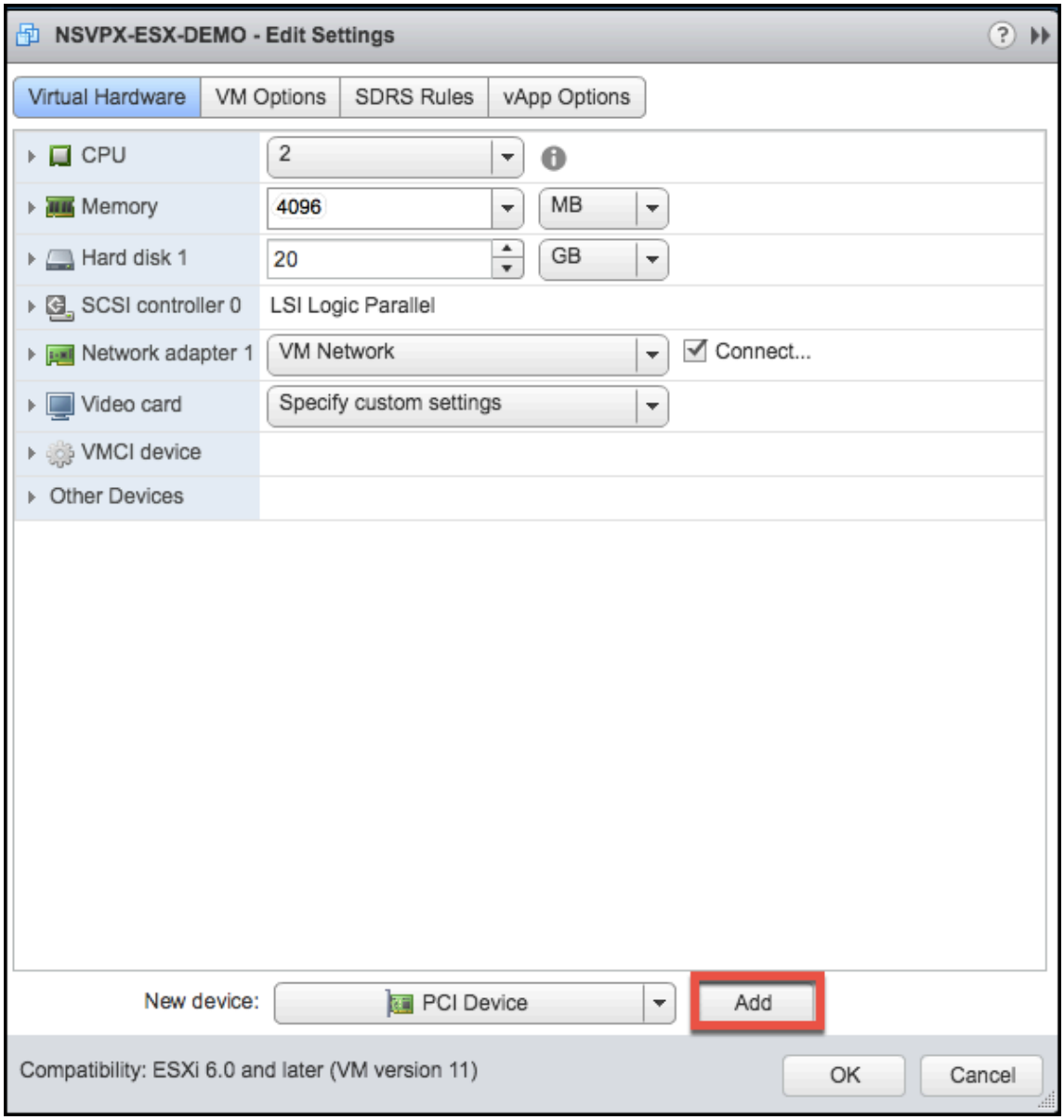


6. ホストマシンを再起動します。

NetScaler VPX インスタンスでパススルーデバイスを構成する

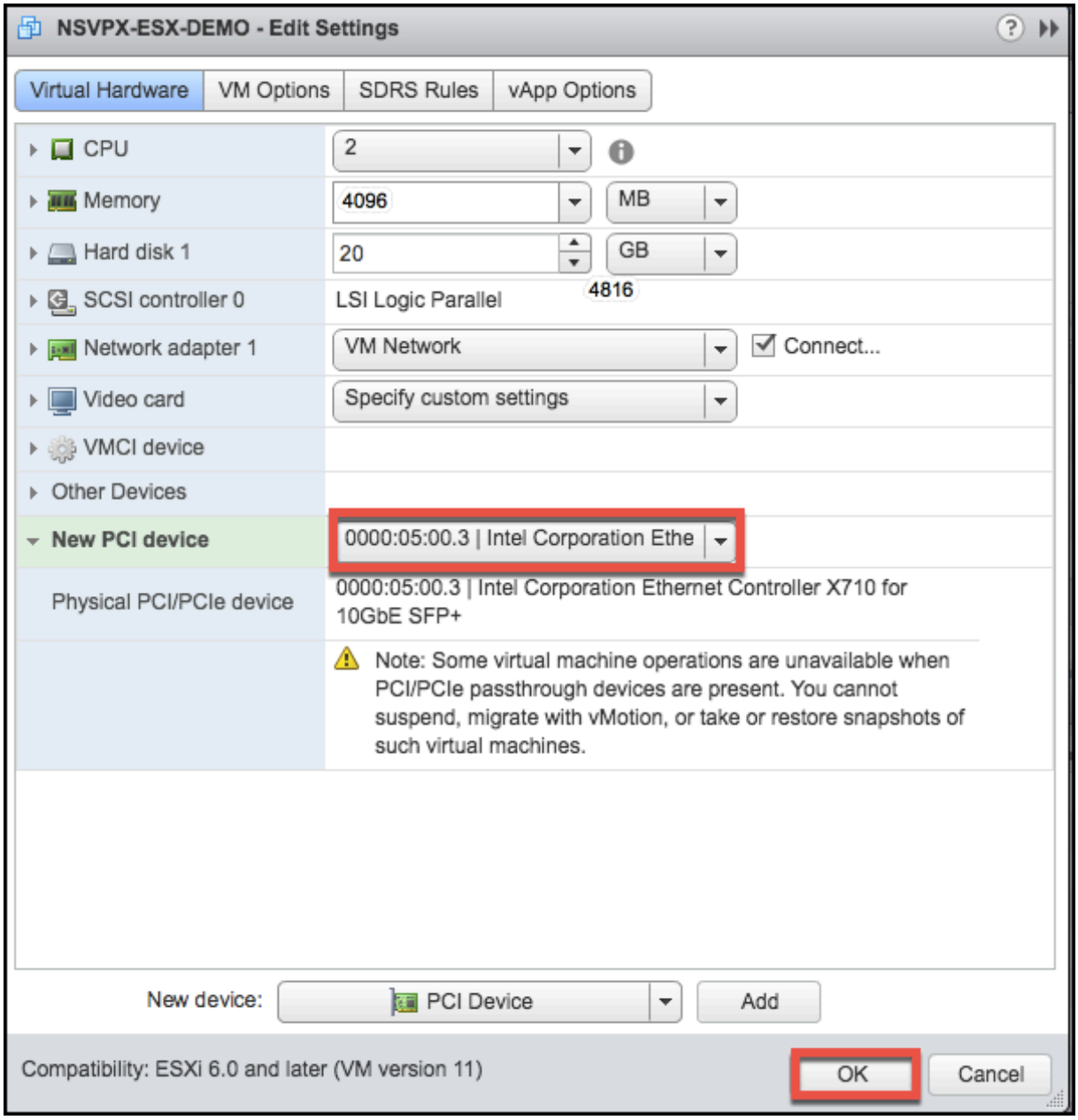
次の手順に従って、NetScaler VPX インスタンスでパススルー PCI デバイスを構成します。

1. 仮想マシンの電源を切ります。
2. 仮想マシンを右クリックし、[設定の編集] を選択します。
3. [仮想ハードウェア] タブで、[新しいデバイス] ドロップダウンメニューから [PCI デバイス] を選択し、[追加] をクリックします。



4. [**New PCI device**] を展開し、ドロップダウンリストから仮想マシンに接続するパススルーデバイスを選択し、[**OK**] をクリックします。

> 注
>
> VMXNET3 ネットワークインターフェイスと PCI パススルーネットワークインターフェイスは共存できません。



1. ゲスト仮想マシンの電源を入れます。

PCI パススルーネットワークインターフェイスを使用するように NetScaler VPX を構成する手順を完了しました。

VMware ESX ハイパーバイザーでの NetScaler ADC アプライアンスの初回起動時に NetScaler ADC VPX の構成を適用する

October 17, 2024

NetScaler VPX 構成は、VMware ESX ハイパーバイザーでの NetScaler ADC アプライアンスの初回起動時に適用できます。Therefore in certain cases, a specific setup or VPX instance is brought up in much lesser time.

プレブートユーザーデータとその形式について詳しくは、クラウドでの NetScaler ADC アプライアンスの初回起動時に NetScaler ADC VPX 構成を適用するを参照してください。

> 注
>
> To bootstrap using preboot user data in ESX, default gateway config must be passed in <NS-CONFIG> section. For more information on the content of the <NS-CONFIG> tag, see Sample-<NS-CONFIG>-section.

Sample <NS-CONFIG> section:

```
<NS-PRE-BOOT-CONFIG>

    <NS-CONFIG>
        add route 0.0.0.0 0.0.0.0  10.102.38.1
    </NS-CONFIG>

    <NS-BOOTSTRAP>
            <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
            <NEW-BOOTSTRAP-SEQUENCE>YES</NEW-BOOTSTRAP-SEQUENCE>

        <MGMT-INTERFACE-CONFIG>
                <INTERFACE-NUM> eth0 </INTERFACE-NUM>
                <IP> 10.102.38.216 </IP>
                <SUBNET-MASK> 255.255.255.0 </SUBNET-MASK>
        </MGMT-INTERFACE-CONFIG>
    </NS-BOOTSTRAP>

</NS-PRE-BOOT-CONFIG>
```

How to provide preboot user data on ESX hypervisor

Web クライアントまたは vSphere クライアントから ESX ハイパーバイザーのプレブートユーザーデータを提供するには、次の 2 つの方法があります。

- Using CD/DVD ISO
- Using OVF Property

Provide user data using CD/DVD ISO

VMware vSphere クライアントを使用すると、CD/DVD ドライブを使用して ISO イメージとしてユーザーデータを VM に注入できます。

CD/DVD ISO を使用してユーザーデータを提供するには、次の手順に従います。

1. プレブートユーザーデータコンテンツを含むファイル名userdataでファイルを作成します。For more information on the content of the <NS-CONFIG> tag, see Sample <NS-CONFIG > section.

   > 注
   >
   > ファイル名は厳密に userdataとして使用する必要があります。

2. Store the userdata file in a folder, and build an ISO image using the folder.

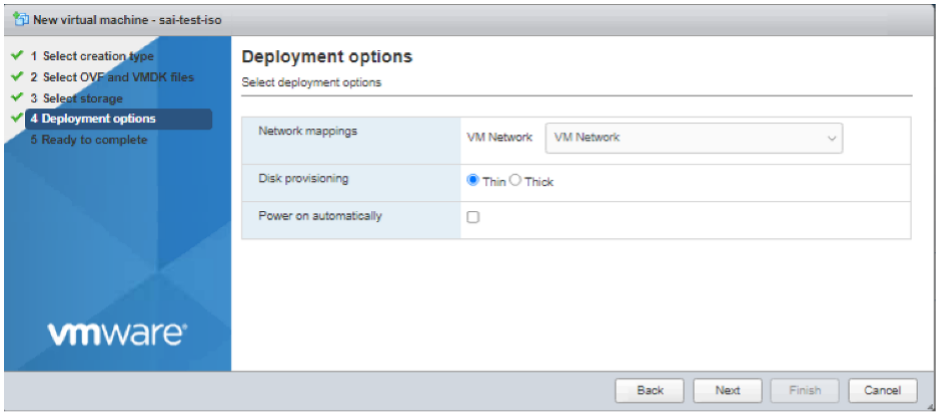
   You can build an ISO image with userdata file by the following two methods:

   - Using any image processing tool such as PowerISO.
   - Using mkisofs command in Linux.

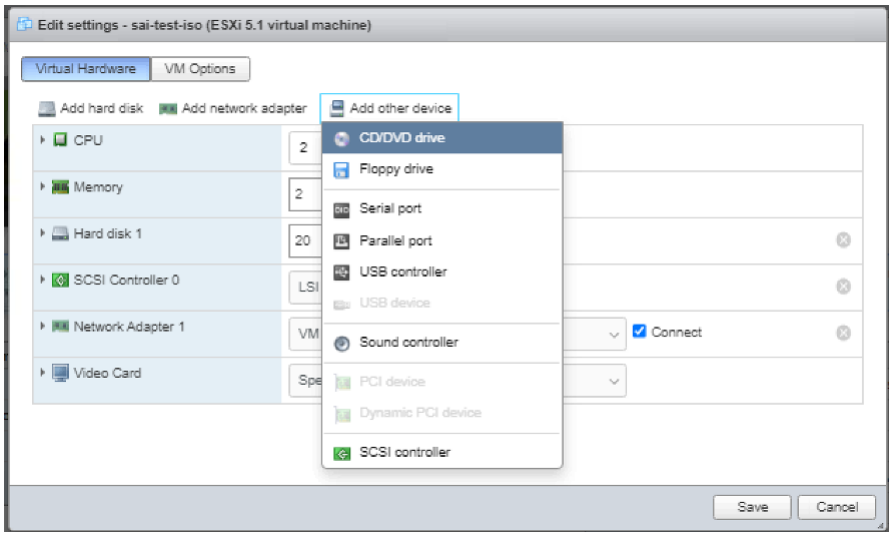
   The following sample configuration shows how to generate an ISO image using the mkisofs command in Linux.

```
root@ubuntu:~/sai/14jul2021# ls -l total 4
drwxr-xr-x 2 root root 4096 Jul 14 12:32 esx_preboot_userdata
root@ubuntu:~/sai/14jul2021#
root@ubuntu:~/sai/14jul2021# ls -l esx_preboot_userdata/total 4
-rw-r--r-- 1 root root 3016 Jul 14 12:32 userdata
root@ubuntu:~/sai/14jul2021# mkisofs -o esx_preboot_userdata.iso
    ./esx_preboot_userdata
I: -input-charset not specified, using utf-8 (detected in locale
    settings)
Total translation table size: 0
Total rockridge attributes bytes: 0
Total directory bytes: 112
Path table size(bytes): 10
Max brk space used 0
176 extents written (0 MB)
root@ubuntu:~/sai/14jul2021# ls -lh
total 356K
drwxr-xr-x 2 root root 4.0K Jul 14 12:32 esx_preboot_userdata
-rw-r--r-- 1 root root 352K Jul 14 12:34 esx_preboot_userdata.
   iso

root@ubuntu:~/sai# ls preboot_userdata_155_193 userdata
root@ubuntu:~/sai# mkisofs -o preboot_userdata_155_193.iso ./
   preboot_userdata_155_193
I: -input-charset not specified, using utf-8 (detected in locale
    settings)
Total translation table size: 0
Total rockridge attributes bytes: 0
Total directory bytes: 112
Path table size(bytes): 10
Max brk space used 0
176 extents written (0 MB)
```

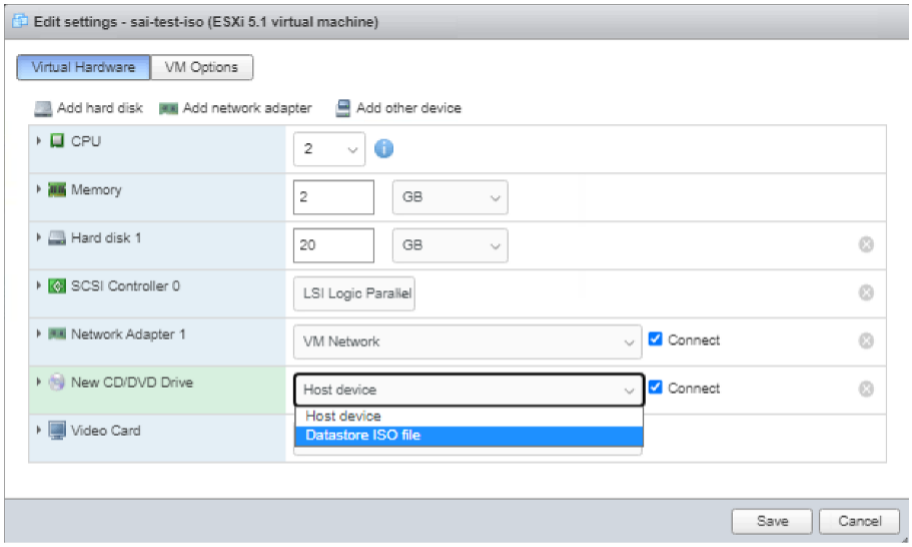

3. 標準の展開プロセスを使用して NetScaler ADC VPX インスタンスをプロビジョニングし、仮想マシンを作成します。But do not power on the VM automatically.



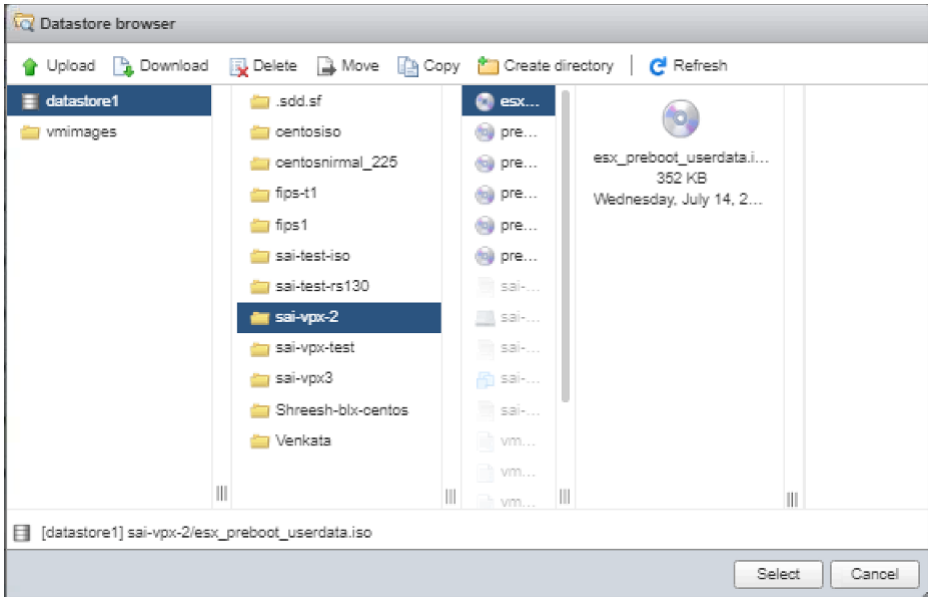
4. After the VM is successfully created, attach the ISO file as CD/DVD drive to the VM.



5. Navigate to **New CD/DVD Drive** and choose **Datastore ISO file** from the drop-down menu.



6. Select a Datastore in the vSphere Client.



7. Power on the VM.

ESX Web クライアントの **OVF** プロパティを使用してユーザーデータを提供

Follow these steps to provide user data using OVF property.

1. Create a file with user data content.


```
root@ubuntu:~/sai/14jul2021# cat esx_userdata.xml
<NS-PRE-BOOT-CONFIG>
    <NS-CONFIG>
        add route 0.0.0.0 0.0.0.0 10.102.38.1
    </NS-CONFIG>

    <NS-BOOTSTRAP>
            <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
            <NEW-BOOTSTRAP-SEQUENCE>YES</NEW-BOOTSTRAP-SEQUENCE>

        <MGMT-INTERFACE-CONFIG>
                <INTERFACE-NUM> eth0 </INTERFACE-NUM>
                <IP>    10.102.38.219 </IP>
                <SUBNET-MASK> 255.255.255.0 </SUBNET-MASK>
        </MGMT-INTERFACE-CONFIG>
    </NS-BOOTSTRAP>
</NS-PRE-BOOT-CONFIG>
```

2. Encode the user data content with Base64 encoding. You can perform the Base64 encoding using the following two methods:

   - In Linux, use the following command:

     ```
     base64 <userdata-filename>  >   <outuput-file>
     ```

     例

     ```
     base64 esx_userdata.xml  > esx_userdata_b64
     ```

     ```
     root@ubuntu:~/sai/14jul2021# base64 esx_userdata.xml > esx_userdata_b64
     root@ubuntu:~/sai/14jul2021#
     root@ubuntu:~/sai/14jul2021# cat esx_userdata_b64
     PE5TLVBSRS1CT09ULUNPTkZJRz4KICAgIDxOUy1DT05GSUc+CglhZGQgcm91dGUgMC4wLjAuMCAw
     LjAuMC4wIDEwLjEwMi4zOC4xCiAgICA8L05TLUNPTkZJRz4KCiAgICA8TlMtQk9PVFNUUkFQPgog
     ICAgICAgICAgICA8U0tJUC1ERUZBVUxULUJPT1RTVFJBUD5ZRVM8L1NLSVAtREVGQVVMVC1CT09U
     U1RSQVA+CiAgICAgICAgICAgIDxORVctQk9PVFNUUkFQLVNFUVVFTkNFPllFUzwvTkVXLUJPT1RT
     VFJBUC1TRVFVRU5DRT4KCiAgICAgICAgPE1HTVQtSU5URVJGQUNFLUNPTkZJRz4KICAgICAgICAg
     ICAgICAgIDxJTlRFUkZBQ0UtTlVNPiBldGgwIDwvSU5URVJGQUNFLU5VTT4KICAgICAgICAgICAg
     ICAgIDxJUD4gICAgMTAuMTAyLjM4LjIxOSA8L0lQPgogICAgICAgICAgICAgICAgPFNVQk5FVC1N
     QVNLPiAyNTUuMjU1LjI1NS4wIDwvU1VCTkVULU1BU0s+CiAgICAgICAgPC9NR01ULUlOVEVSRkFD
     RS1DT05GSUc+CiAgICA8L05TLUJPT1RTVFJBUD4KPC9OUy1QUkUtQk9PVC1DT05GSUc+Cg==
     ```

   - Use online tools to encode user data content, for example, Base64 Encode and Decode.

3. ESX ハイパーバイザー上の NetScaler ADC **VPX** インスタンスの **OVF** テンプレートに製品セクションを含めます。

   Sample Product section:

   ```
   <ProductSection>

     <Info>Information about the installed software</Info>
     <Product>NSVPX-VSK Template</Product>
     <Vendor>Citrix</Vendor>
     <VendorUrl>www.citrix.com</VendorUrl>
     <Category> Preboot Userdata </Category>
   ```



```
8
9       <Property ovf:key="guestinfo.userdata" ovf:type="string" ovf:
            userConfigurable="true" ovf:value="">
10
11        <Label>Userdata</Label>
12        <Description> Userdata for ESX VPX </Description>
13      </Property>
14
15    </ProductSection>
```

4. Provide the base64 encoded user data as the `ovf:value` for `guestinfo.userdata` property in the Product section.

```
1   <ProductSection>
2
3     <Info>Information about the installed software</Info>
4     <Product>NSVPX-VSK Template</Product>
5     <Vendor>Citrix</Vendor>
6     <VendorUrl>www.citrix.com</VendorUrl>
7     <Category> Preboot Userdata </Category>
8     <Property ovf:key="guestinfo.userdata" ovf:type="string" ovf:
          userConfigurable="true"
9       ovf:value="PE5TLVBSRS1CT09ULUNPTkZJRz4KICAgIDxOUy1DT05GSUc+
            CglhZGQgcm91dGUgMC4wLjAuMCAw
10      LjAuMC4wIDEwLjEwMi4zOC4xCiAgICA8L05TLUNPTkZJRz4KCiAgICA8TlMtQk9PVFNUUk
11      ICAgICAgICAgICA8U0tJUC1ERUZBVUxULUJPT1RTVFJBUD5ZRVM8L1NLSVAtREVGQVVMVC
12      U1RSQVA+
            CiAgICAgICAgICAgIDxORVctQk9PVFNUUkFQLVNFUVVFTkNFPllFUzwvTkVXLUJPT1R
13      VFJBUC1TRVFVRU5DRT4KCiAgICAgICAgPE1HTVQtSU5URVJGQUNFLUNPTkZJRz4KICAgIC
14      ICAgICAgIDxJTlRFUkZBQ0UtTlVNPiBldGgwIDwvSU5URVJGQUNFLU5VTT4KICAgICAgIC
15      ICAgIDxJUD4gICAgMTAuMTAyLjM4LjIxOSA8L0lQPgogICAgICAgICAgICAgICAgPFNVQk
16      QVNLPiAyNTUuMjU1LjI1NS4wIDwvU1VCTkVULU1BU0s+
            CiAgICAgICAgPC9NR01ULUlOVEVSRkFD
17      RS1DT05GSUc+
            CiAgICA8L05TLUJPT1RTVFJBUD4KPC9OUy1QUkUtQk9PVC1DT05GSUc+
            Cg==">
18
19        <Label>Userdata</Label>
20        <Description> Userdata for ESX VPX </Description>
21      </Property>
22
23    </ProductSection>
```

5. Use the modified OVF template with Product section for the VM deployment.


```
Please change the default NSROOT password.
Enter new password:
Please re-enter your password:
 Done
> sh ns ver
        NetScaler NS13.0: Build 83.9005.nc, Date: Jul 13 2021, 02:56:05   (64-bit)
 Done
> sh ns ip
        Ipaddress        Traffic Domain  Type             Mode     Arp      Icmp     Vserver  S
tate
        ---------        --------------  ----             ----     ---      ----     -------  -
-----
1)      10.102.38.219    0               NetScaler IP     Active   Enabled  Enabled  NA       E
nabled
 Done
> sh route
        Network          Netmask          Gateway/OwnedIP  VLAN    State   Traffic Domain  Type
        -------          -------          ---------------  -----   -----   --------------  ----
1)      0.0.0.0          0.0.0.0          10.102.38.1      0       UP      0               STATI
C
2)      127.0.0.0        255.0.0.0        127.0.0.1        0       UP      0               PERMA
NENT
3)      10.102.38.0      255.255.255.0    10.102.38.219    0       UP      0               DIREC
T
 Done
```

ESX vSphere クライアントの **OVF** プロパティを使用してユーザーデータを提供

ESX vSphere クライアントから OVF プロパティを使用してユーザーデータを提供するには、次の手順に従います。

1. Create a file with user data content.

```
root@ubuntu:~/sai/14jul2021# cat esx_userdata.xml
<NS-PRE-BOOT-CONFIG>
    <NS-CONFIG>
        add route 0.0.0.0 0.0.0.0 10.102.38.1
    </NS-CONFIG>

    <NS-BOOTSTRAP>
            <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
            <NEW-BOOTSTRAP-SEQUENCE>YES</NEW-BOOTSTRAP-SEQUENCE>

        <MGMT-INTERFACE-CONFIG>
                <INTERFACE-NUM> eth0 </INTERFACE-NUM>
                <IP>    10.102.38.219 </IP>
                <SUBNET-MASK> 255.255.255.0 </SUBNET-MASK>
        </MGMT-INTERFACE-CONFIG>
    </NS-BOOTSTRAP>
</NS-PRE-BOOT-CONFIG>
```

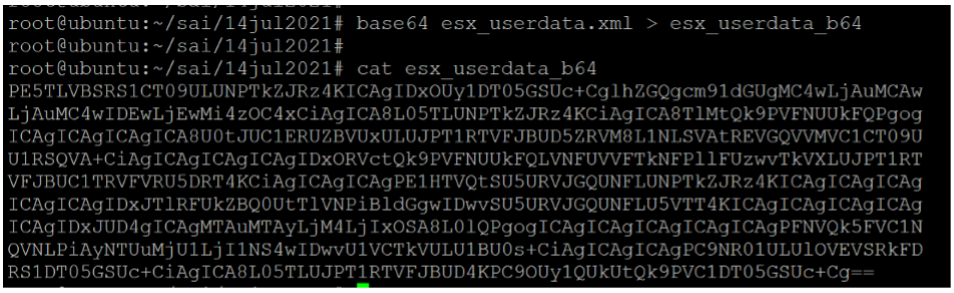
2. Encode the user data content with Base64 encoding. You can perform the Base64 encoding using the following two methods:

   - In Linux, use the following command:

```
base64 <userdata-filename>  >   <outuput-file>
```

   例


```
base64 esx_userdata.xml  > esx_userdata_b64
```



- Use online tools to encode user data content, for example, Base64 Encode and Decode.

3. ESX ハイパーバイザー上の NetScaler ADC **VPX** インスタンスの **OVF** テンプレートに製品セクションを含めます。

   Sample Product section:

```
<ProductSection>

  <Info>Information about the installed software</Info>
  <Product>NSVPX-VSK Template</Product>
  <Vendor>Citrix</Vendor>
  <VendorUrl>www.citrix.com</VendorUrl>
  <Category> Preboot Userdata </Category>

  <Property ovf:key="guestinfo.userdata" ovf:type="string" ovf:
     userConfigurable="true" ovf:value="">

    <Label>Userdata</Label>
    <Description> Userdata for ESX VPX </Description>
  </Property>

</ProductSection>
```

4. Provide the base64 encoded user data as the `ovf:value` for `guestinfo.userdata` property in the Product section.

```
<ProductSection>

  <Info>Information about the installed software</Info>
  <Product>NSVPX-VSK Template</Product>
  <Vendor>Citrix</Vendor>
  <VendorUrl>www.Citrix.com</VendorUrl>
  <Category> Preboot Userdata </Category>
  <Property ovf:key="guestinfo.userdata" ovf:type="string" ovf:
     userConfigurable="true"
    ovf:value="PE5TLVBSRS1CT09ULUNPTkZJRz4KICAgIDxOUy1DT05GSUc+
       CglhZGQgcm91dGUgMC4wLjAuMCAw
    LjAuMC4wIDEwLjEwMi4zOC4xCiAgICA8L05TLUNPTkZJRz4KICAgICA8TlMtQk9PVFNUUk
```



```
11        ICAgICAgICAgICA8U0tJUC1ERUZBVUxULUJPT1RTVFJBUD5ZRVM8L1NLSVAtREVGQVVMVC
12        U1RSQVA+
              CiAgICAgICAgICAgIDxORVctQk9PVFNUUkFQLVNFUVVFTkNFPllFUzwvTkVXLUJPT1R
13        VFJBUC1TRVFVRU5DRT4KCiAgICAgICAgPE1HTVQtSU5URVJGQUNFLUNPTkZJRz4KICAgICA
14        ICAgICAgIDxJTlRFUkZBQ0UtTlVNPiBldGgwIDwvSU5URVJGQUNFLU5VTT4KICAgICAgIC
15        ICAgIDxJUD4gICAgMTAuMTAyLjM4LjIxOSA8L0lQPgogICAgICAgICAgICAgICAgICAgPFNVQk
16        QVNLPiAyNTUuMjU1LjI1NS4wIDwvU1VCTkVULU1BU0s+
              CiAgICAgICAgPC9NR01ULUlOVEVSRkFD
17        RS1DT05GSUc+
              CiAgICA8L05TLUJPT1RTVFJBUD4KPC9OUy1QUkUtQk9PVC1DT05GSUc+
              Cg==">
18
19        <Label>Userdata</Label>
20        <Description> Userdata for ESX VPX </Description>
21      </Property>
22
23    </ProductSection>
```

5. 次のように ovf:transport="com.vmware.guestInfo"、プロパティを仮想ハードウェアセクションに追加します。

```
1    <VirtualHardwareSection  ovf:transport="com.vmware.guestInfo">
```

6. Use the modified OVF template with Product section for the VM deployment.

```
Please change the default NSROOT password.
Enter new password:
Please re-enter your password:
 Done
> sh ns ver
        NetScaler NS13.0: Build 83.9005.nc, Date: Jul 13 2021, 02:56:05   (64-bit)
 Done
> sh ns ip
        Ipaddress        Traffic Domain  Type             Mode     Arp      Icmp     Vserver  S
tate
        ---------        --------------  ----             ----     ---      ----     -------  -
-----
1)      10.102.38.219    0               NetScaler IP     Active   Enabled  Enabled  NA       E
nabled
 Done
> sh route
        Network          Netmask          Gateway/OwnedIP  VLAN    State   Traffic Domain  Type
        -------          -------          ---------------  -----   -----   --------------  ----
1)      0.0.0.0          0.0.0.0          10.102.38.1      0       UP      0               STATI
C
2)      127.0.0.0        255.0.0.0        127.0.0.1        0       UP      0               PERMA
NENT
3)      10.102.38.0      255.255.255.0    10.102.38.219    0       UP      0               DIREC
T
 Done
```


AWS の VMware クラウドに Citrix ADC VPX インスタンスをインストールする

October 17, 2024

AWS 上の VMware クラウド (VMC) を使用すると、必要な数の ESX ホストを使用してクラウドソフトウェア定義データセンター (SDDC) を AWS 上に作成できます。AWS 上の VMC は、NetScaler VPX デプロイをサポートしています。VMC は、オンプレミスの vCenter と同じユーザー・インタフェースを提供します。ESX ベースの Citrix ADC VPX デプロイメントと同じように機能します。

前提条件

仮想アプライアンスのインストールを開始する前に、次の操作を行います。

- 1 つの VMware SDDC が少なくとも 1 つのホストに存在している必要があります。
- NetScaler VPX アプライアンスセットアップファイルをダウンロードします。
- 仮想マシンが接続する適切なネットワークセグメントを VMware SDDC 上に作成します。
- VPX ライセンスファイルを入手します。NetScaler VPX インスタンス ライセンスの詳細については、*NetScaler VPX* ライセンス ガイド (</en-us/licensing/licensing-guide-for-netscaler.html>) を参照してください。

VMware クラウドのハードウェア要件

次の表に、VMware SDDC が各 VPX nCore 仮想アプライアンスに対して提供する必要がある仮想コンピューティングリソースを示します。

表 1. NetScaler VPX インスタンスの実行に必要な最小限の仮想コンピューティングリソース

コンポーネント	条件
メモリ	2 GB
仮想 CPU（VCPU）	2
仮想ネットワークインターフェイス	VMware SDDC では、VPX ハードウェアをバージョン 7 以上にアップグレードすると、最大 10 個の仮想ネットワーク インターフェイスをインストールできます。
ディスク領域	20GB

> 注
>
> これは、ハイパーバイザーのディスク要件に加えて必要になります。

VPX 仮想アプライアンスを実稼働で使用するには、フルメモリ割り当てを予約する必要があります。

OVF ツール 1.0 のシステム要件

Open Virtualization Format Tool（OVF Tool）は、Windows および Linux システムで実行できるクライアントアプリケーションです。次の表は、最小システム要件を示しています。

表 2. OVF ツールのインストールに必要な最小システム要件

コンポーネント	条件
オペレーティングシステム	VMware からの詳細な要件については、http://kb.vmware.com/で『OVF ツールユーザーガイド』の PDF ファイルを検索してください。
CPU	最低 750MHz、1GHz 以上推奨
RAM	最小 1 GB、推奨 2 GB
NIC	100Mbps 以上の NIC。

OVF のインストールについては、http://kb.vmware.com/で『OVF ツールユーザーガイド』の PDF ファイルを検索してください。

NetScaler VPX セットアップファイルのダウンロード

VMware ESX 用の NetScaler VPX インスタンスセットアップパッケージは、オープン仮想マシン（OVF）フォーマット標準に準拠しています。これらのファイルは、Citrix Web サイトからダウンロードできます。ログオンするには、Citrix アカウントが必要です。Citrix アカウントをお持ちでない場合は、http://www.citrix.comのホームページにアクセスしてください。［新しいユーザー］リンクをクリックし、指示に従って新しい Citrix アカウントを作成します。

ログオンしたら、Citrix のホームページから次のパスをナビゲートします。

Citrix.com > ダウンロード **> NetScaler >** 仮想アプライアンス。

次のファイルを、ESX サーバーと同じネットワーク上のワークステーションにコピーします。3 つのファイルをすべて同じフォルダーにコピーします。

- NSVPX-ESX-<release number>-<build number>-disk1.vmdk（例えば、NSVPX-ESX-13.0-79.64-disk1.vmdk）
- NSVPX-ESX-<release number>-<build number>.ovf（例えば、NSVPX-ESX-13.0-79.64.ovf）
- NSVPX-ESX-<release number>-<build number>.mf（例えば、NSVPX-ESX-13.0-79.64.mf）

VMware クラウドへの Citrix ADC VPX インスタンスのインストール

VMware SDDC をインストールして設定したら、SDDC を使用して VMware クラウドに仮想アプライアンスをインストールできます。インストールできる仮想アプライアンスの数は、SDDC で使用可能なメモリの量によって異なります。

NetScaler VPX インスタンスを VMware クラウドにインストールするには、次の手順に従います。

1. ワークステーションで VMware SDDC を開きます。
2. [ユーザー名] テキストボックスと [パスワード] テキストボックスに、管理者の認証情報を入力し、[ログイン] をクリックします。
3. ［**File**］メニューの［**Deploy OVF Template**］を選択します。
4. ［**OVF** テンプレートのデプロイ］ダイアログボックスの［ファイルからデプロイ］で、NetScaler VPX インスタンスセットアップファイルを保存した場所を参照し、.ovf ファイルを選択して、［次へ］をクリックします。

   > 注：デフォルトでは、NetScaler VPX インスタンスは E1000 ネットワークインターフェイスを使用します。VMXNET3 インターフェイスで ADC を展開するには、E1000 ではなく VMXNET3 インターフェイスを使用するように OVF を変更します。

5. 仮想アプライアンス OVF テンプレートに表示されるネットワークを、VMware SDDC で設定したネットワークにマッピングします。[次へ] をクリックして VMware SDDC への仮想アプライアンスのインストールを開始します。
6. これで、NetScaler VPX インスタンスを起動する準備ができました。ナビゲーションペインで、インストールした Citrix ADC VPX インスタンスを選択し、右クリックメニューから［パワーオン］を選択します。コンソールポートをエミュレートするには、［**Console**］タブをクリックします。
7. 別の仮想アプライアンスをインストールする場合は、ステップ 6 から繰り返します。
8. 管理ネットワークとして選択した同じセグメントから管理 IP アドレスを指定します。ゲートウェイには同じサブネットが使用されます。
9. VMware SDDC では、ネットワークセグメントに属するすべてのプライベート IP アドレスに対して NAT ルールとファイアウォールルールを明示的に作成する必要があります。

Microsoft Hyper-V サーバーに NetScaler VPX インスタンスをインストールします

October 17, 2024

NetScaler VPX インスタンスを Microsoft Windows Server にインストールするには、まず、十分なシステムリソースを備えたマシンに、Hyper-V ロールを有効にした Windows Server をインストールする必要があります。Hyper-V の役割をインストールするときは、仮想ネットワークを作成するために Hyper-V で使用されるサーバー

上の NIC を必ず指定してください。一部の NIC は、ホスト用に確保できます。Hyper-V マネージャーを使用して NetScaler VPX インスタンスのインストールを実行します。

Hyper-V 用の NetScaler ADC VPX インスタンスは、仮想ハードディスク（VHD）形式で配信されます。CPU、ネットワークインターフェイス、ハードディスクのサイズと形式などの要素について、デフォルト構成が格納されています。NetScaler VPX インスタンスをインストールしたら、仮想アプライアンスにネットワークアダプタを構成し、仮想 NIC を追加してから、NetScaler IP アドレス、サブネットマスク、およびゲートウェイを割り当てて、仮想アプライアンスの基本構成を完了できます。

VPX インスタンスの初期構成後、アプライアンスを最新のソフトウェアリリースにアップグレードする場合は、NetScaler VPX スタンドアロンアプライアンスのアップグレードを参照してください。

> 注
>
> ISIS（Intermediate System-to-Intermediate System）プロトコルは、Hyper-V 2012 プラットフォーム上でホストされる NetScaler VPX 仮想アプライアンスではサポートされません。

NetScaler VPX インスタンスを Microsoft サーバーにインストールするための前提条件

仮想アプライアンスのインストールを開始する前に、次の操作を行います。

- Windows サーバーで Hyper-V ロールを有効にします。詳しくは、http://technet.microsoft.com/en-us/library/ee344837(WS.10).aspxを参照してください。
- 仮想アプライアンスセットアップファイルをダウンロードします。
- NetScaler VPX インスタンスのライセンスファイルを取得します。NetScaler VPX インスタンスライセンスの詳細については、https://support.citrix.com/s/article/CTX255959-how-to-allocate-and-install-citrix-netscaler-vpx-licenses?language=en_USの『*NetScaler VPX* ライセンスガイド』を参照してください。

Microsoft のサーバハードウェア要件

次の表は、Microsoft サーバーの最小システム要件を示しています。

表 1. Microsoft サーバーの最小システム要件

コンポーネント	条件
CPU	1.4GHz 64 ビットプロセッサ
RAM	8 GB
ディスク領域	32GB 以上

次の表は、各仮想コンピューティングリソースの一覧です。NetScaler VPX インスタンス。

表 2. NetScaler VPX インスタンスの実行に必要な最小限の仮想コンピューティングリソース

コンポーネント	条件
RAM	4 GB
仮想 CPU	2
ディスク領域	20GB
仮想ネットワーク インターフェイス	1

NetScaler VPX セットアップファイルをダウンロードする

Hyper-V 用の NetScaler ADC VPX インスタンスは、仮想ハードディスク（VHD）形式で配信されます。これらのファイルは、Citrix Web サイトからダウンロードできます。ログインするには Citrix アカウントが必要です。Citrix アカウントをお持ちでない場合は、http://www.citrix.comのホームページにアクセスし、［サインイン］**>**［マイアカウント］**>**［**Citrix** アカウントの作成］の順にクリックし、手順に従って Citrix アカウントを作成します。

NetScaler VPX インスタンスのセットアップファイルをダウンロードするには、次の手順に従います。

1. Web ブラウザーで、http://www.citrix.com/に移動します。
2. ユーザー名とパスワードを使用してサインインします。
3. ［**Downloads**］をクリックします。
4. 「製品の選択」ドロップダウンメニューで、「**NetScaler**（**NetScaler ADC**）」を選択します。
5. 「**NetScaler** リリース **X.X**」**>**「仮想アプライアンス」で、「**NetScaler VPX** リリース **X.X**」をクリックします。
6. 圧縮ファイルをサーバーにダウンロードします。

NetScaler VPX インスタンスを **Microsoft** のサーバーにインストールします

Microsoft Server で Hyper-V ロールを有効にし、仮想アプライアンスファイルを抽出したら、Hyper-V Manager を使用して NetScaler ADC VPX インスタンスをインストールできます。仮想マシンをインポートしてから仮想 NIC を構成し、Hyper-V によって作成された仮想ネットワークに関連付ける必要があります。

最大 8 つの仮想 NIC を構成できます。物理 NIC が DOWN になっても、同じホスト（サーバー）上の他の仮想アプライアンスと通信できるため、仮想アプライアンスでは仮想 NIC は UP と見なされます。

> 注
>
> 仮想アプライアンスの実行中は、設定を変更することができません。仮想アプライアンスをシャットダウンしてから変更を行います。

Hyper-V マネージャーを使用して **NetScaler VPX** インスタンスを **Microsoft** サーバーにインストールするには:

1. Hyper-V マネージャーを起動するには、[スタート] ボタンをクリックし、[管理ツール] をポイントして、[**Hyper-V** マネージャー] をクリックします。
2. ナビゲーションペインの **Hyper-V Manage**r で、NetScaler VPX インスタンスをインストールするサーバーを選択します。
3. [アクション] メニューで、[仮想マシンのインポート] をクリックします。
4. [仮想マシンのインポート] ダイアログボックスの [場所] で、NetScaler VPX インスタンスのソフトウェアファイルを含むフォルダーのパスを指定し、[仮想マシンをコピー（新しい一意の **ID** を作成）] を選択します。このフォルダーは、Snapshots フォルダー、Virtual Hard Disks フォルダー、および Virtual Machines フォルダーを格納する親フォルダーです。

> 注
>
> 圧縮ファイルを受け取った場合は、フォルダーへのパスを指定する前に、フォルダーにファイルを展開することを確認します。

1. [インポート] をクリックします。
2. インポートした仮想アプライアンスが [仮想マシン] の下に表示されていることを確認します。
3. 別の仮想アプライアンスをインストールするには、手順 **2** ～**6** を繰り返します。

> 重要**:**
>
> 手順 **4** で、必ずファイルを別のフォルダーに解凍してください。

Hyper-V で **NetScaler ADC VPX** インスタンスを自動プロビジョニングする

NetScaler VPX インスタンスの自動プロビジョニングはオプションです。自動プロビジョニングを実行しない場合は、NetScaler 仮想アプライアンスによって IP アドレスなどを構成するためのオプションが提供されます。

Hyper-V で NetScaler ADC VPX インスタンスを自動プロビジョニングするには、次の手順に従います。

1. 例に示されている説明に従い、XML ファイルを使用して ISO9660 準拠の ISO イメージを作成します。xml ファイルの名前が **userdata** であることを確認してください。

   XML ファイルから ISO ファイルを作成するには、以下を使用します。

   • PowerISO などの任意の画像処理ツール。

   • Linux の `mkisofs`コマンド。

```
<?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`"
```



```
6
7    oe:id=""
8
9    xmlns=`"http://schemas.dmtf.org/ovf/environment/1`">
10
11   <PlatformSection>
12
13   <Kind>HYPER-V</Kind>
14
15   <Version>2013.1</Version>
16
17   <Vendor>CITRIX</Vendor>
18
19   <Locale>en</Locale>
20
21   </PlatformSection>
22
23   <PropertySection>
24
25   <Property oe:key="com.citrix.netscaler.ovf.version" oe:value="
        1.0"/>
26
27   <Property oe:key="com.citrix.netscaler.platform" oe:value="
        NS1000V"/>
28
29   <Property oe:key="com.citrix.netscaler.orch\_env" oe:value="
        cisco-orch-env"/>
30
31   <Property oe:key="com.citrix.netscaler.mgmt.ip" oe:value="
        10.102.100.122"/>
32
33   <Property oe:key="com.citrix.netscaler.mgmt.netmask" oe:value="
        255.255.255.128"/>
34
35   <Property oe:key="com.citrix.netscaler.mgmt.gateway" oe:value="
        10.102.100.67"/></PropertySection>
36
37   </Environment>
```

2. ISO イメージを Hyper-V Server にコピーします。
3. インポートした仮想アプライアンスを選択し、[アクション] メニューで [設定] を選択します。仮想アプライアンスを選択し、右クリックして [設定] を選択することもできます。選択した仮想アプライアンスの [設定] ウィンドウが表示されます。
4. 「設定」ウィンドウの「ハードウェア」セクションで、「**IDE Controller**」をクリックします。
5. 右側のウィンドウペインで、「**DVD Drive** 」を選択し、「追加」をクリックします。DVD ドライブは、左側のウィンドウペインの **IDE** コントローラセクションに追加されます。
6. 手順 5 で追加した **DVD** ドライブを選択します。右側のウィンドウペインで [イメージファイル] ラジオボタンを選択し、[参照] をクリックして、手順 2 で Hyper-V サーバにコピーした ISO イメージを選択します。

7. ［適用］をクリックします。

> 注
>
> 次の場合、仮想アプライアンスインスタンスはデフォルトの IP アドレスで起動します。

- DVD ドライブがアタッチされているのに、ISO ファイルが提供されていない。
- ISO ファイルにはユーザーデータファイルは含まれていません。
- ユーザーデータのファイル名または形式が正しくありません。

NetScaler VPX インスタンスで仮想 NIC を構成するには、次の手順に従います。

1. インポートした仮想アプライアンスを選択し、[アクション] メニューで [設定] を選択します。
2. [設定] ダイアログボックスの左ペインの <virtual appliance name>[ハードウェアの追加] をクリックします。
3. 右ペインで、デバイスのリストから［ネットワークアダプター］を選択します。
4. ［追加］をクリックします。
5. 左ペインに［**Network Adapter** (not connected)］が表示されていることを確認します。
6. 左ペインでネットワークアダプターを選択します。
7. 右側のペインの [ネットワーク] メニューから、アダプターを接続する仮想ネットワークを選択します。
8. 使用する他のネットワークアダプタの仮想ネットワークを選択するには、手順 **6** と **7** を繰り返します。
9. ［適用］をクリックしてから、［**OK**］をクリックします。

NetScaler VPX インスタンスを構成するには:

1. 前にインストールした仮想アプライアンスを右クリックし、［開始］を選択します。
2. 仮想アプライアンスをダブルクリックして、コンソールにアクセスします。
3. 仮想アプライアンスの NetScaler IP アドレス、サブネットマスク、およびゲートウェイを入力します。

仮想アプライアンスの基本構成が完了しました。Web ブラウザーで IP アドレスを入力して、仮想アプライアンスにアクセスします。

> 注
>
> 仮想マシン（VM）テンプレートを使用して、SCVMM を使用して NetScaler ADC VPX インスタンスをプロビジョニングすることもできます。
>
> NetScaler VPX インスタンスで Microsoft Hyper-V NIC チーミングソリューションを使用する場合、詳細については、記事 CTX224494 を参照してください。

Linux-KVM プラットフォームへの Citrix ADC VPX インスタンスのインストール

October 17, 2024

Linux-KVM プラットフォーム用の Citrix ADC VPX を設定するには、グラフィカル仮想マシンマネージャ（仮想マネージャ）アプリケーションを使用できます。Linux-KVM コマンドラインを使用する場合は、`virsh` プログラムを使用できます。

KVM Module および QEMU のような仮想化ツールを使って、適切なハードウェアにホスト Linux オペレーティングシステムをインストールする必要があります。ハイパーバイザー上で展開できる仮想マシン（VM）の数はアプリケーション要件および選択されたハードウェアにより異なります。

NetScaler VPX インスタンスをプロビジョニングしたら、より多くのインターフェイスを追加できます。

制限事項と使用ガイドライン

一般的な推奨事項

予測できない動作を回避するには、次の推奨事項を適用します。

- VPX 仮想マシンに関連付けられている VNet インターフェイスの MTU を変更しないでください。インターフェイスモードや CPU などの構成パラメータを変更する前に、VPX VM をシャットダウンします。
- VPX VM を強制的にシャットダウンしないでください。つまり、[強制オフ] コマンドは使用しないでください。
- ホスト Linux 上で指定された任意の構成は、Linux ディストリビューションの設定によってそのまま維持されたり、維持されなかったりします。これらの構成を維持するよう選択して、ホスト Linux オペレーティングシステムのリブートにおける一貫した動作を確保できます。
- NetScaler パッケージはプロビジョニングされた各 NetScaler VPX インスタンスに対して一意である必要があります。

制限事項

- KVM 上で動作する VPX インスタンスのライブマイグレーションはサポートされていません。

Linux-KVM プラットフォームに **Citrix ADC VPX** インスタンスをインストールするための前提条件

October 17, 2024

NetScaler VPX インスタンスで実行されている Linux-KVM サーバーの最小システム要件を確認します。

CPU 要件:

- Intel VT-X プロセッサーに含まれるハードウェア仮想化機能を備えた 64 ビット x86 プロセッサー。

CPU が Linux ホストをサポートしているかどうかをテストするには、ホスト Linux シェルプロンプトで次のコマンドを入力します。

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

前の拡張機能の **BIOS** 設定が無効になっている場合は、BIOS でそれらを有効にする必要があります。

- ホスト Linux に 2 つ以上の CPU コアを指定します。
- プロセッサ速度に対する特定の推奨設定はありませんが、速度が速ければ速いほど VM アプリケーションのパフォーマンスはよくなります。

メモリ **(RAM)** 要件:

ホスト Linux カーネルに対して 4GB 以上。VM が必要とするメモリを追加します。

ハードディスク要件:

ホスト Linux カーネルおよび VM 要件の領域を計算します。単一の NetScaler VPX VM は 20GB のディスク領域を必要とします。

ソフトウェア要件

使用されるホストカーネルは、リリース 2.6.20 以降で、すべての仮想化ツールがある 64 ビットの Linux カーネルである必要があります。3.6.11-4 以降といったより新しいカーネルを推奨します。

Red Hat、CentOS、Fedora などの多くの Linux ディストリビューションでは、カーネルのバージョンと関連する仮想化ツールのテストが行われています。

ゲスト **VM** のハードウェア要件

NetScaler VPX でサポートされるハードディスクの種類は IDE と virtIO です。ハードディスクの種類は、NetScaler パッケージに含まれる XML ファイルで構成されています。

ネットワーク要件

NetScaler VPX は、VirtIO 準仮想化、SR-IOV、および PCI パススルーネットワークインターフェイスをサポートします。

サポートされるネットワークインターフェースの詳細については、以下を参照してください。

- 仮想マシンマネージャーを使用して NetScaler VPX インスタンスをプロビジョニングします
- SR-IOV ネットワークインターフェースを使用するように NetScaler VPX インスタンスを構成する
- PCI パススルーネットワークインターフェイスを使用するように NetScaler VPX インスタンスを構成する

ソースインターフェイスおよびモード

ソースデバイスの種類は、Bridge または MacVTap のいずれかにできます。MacvTap では、VEPA モード、ブリッジ、プライベート、パススルーの 4 つのモードが可能です。次のように、使用できるインターフェイスのタイプとサポートされているトラフィックタイプを確認します。

ブリッジ:

- Linux Bridge。
- 正しい設定を選択したり、`IPtable` サービスを無効にしたりしないと、ホスト Linux の`Ebtables`および`iptables`設定によってブリッジのトラフィックがフィルタリングされることがあります。

MacV タップ **(VEPA** モード**)**:

- パフォーマンスは Bridge より向上します。
- 同じ下位デバイスからのインターフェイスを VM 間で共有できます。
- 仮想マシン間通信（同じ
- 下位のデバイスは、アップストリームスイッチまたはダウンストリームスイッチが VEPA モードをサポートしている場合にのみ可能です。

MacVTap (プライベートモード**)**:

- パフォーマンスは Bridge より向上します。
- 同じ下位デバイスからのインターフェイスを VM 間で共有できます。
- 同じ下位デバイスを使った内部 VM 通信を実行できません。

MacVTap (ブリッジモード):

- Bridge と比べてよい。
- 同じ下位デバイスのインターフェイスは、VM 間で共有できます。
- 下位のデバイスリンクがアップしている場合は、同じ下位デバイスを使用する VM 間通信が可能です。

MacVTap (パススルーモード**)**:

- Bridge と比べてよい。
- 同じ下位デバイスのインターフェイスは、仮想マシン間で共有できません。
- 1 つの VM のみ、下位デバイスを使用できます。

> 注
>
> VPX インスタンスによる最高のパフォーマンスを得るには、ソース インターフェイスで `gro` および `lro` 機能がオフになっていることを確認します。

送信元インターフェイスのプロパティ

ソースインターフェイスの Generic-receive-offload（`gro`）および大規模受信オフロード（`lro`）機能をオフにします。`gro`および`lro`機能をオフにするには、ホスト Linux シェルプロンプトで次のコマンドを実行します。

ethtool -K eth6 gro をオフにします ethool -K eth6 lro オフ

例:

```
[root@localhost ~]# ethtool -K eth6

                    Offload parameters for eth6:

                                        rx-checksumming: on

                                        tx-checksumming: on

                    scatter-gather: on

                    tcp-segmentation-offload: on

                    udp-fragmentation-offload: off

                    generic-segmentation-offload: on

                    generic-receive-offload: off

                    large-receive-offload: off

                    rx-vlan-offload: on

                    tx-vlan-offload: on

                    ntuple-filters: off

                    receive-hashing: on

[root@localhost ~]#
```

例:

次の例のように、ホスト Linux ブリッジをソースデバイスとして使用する場合、ホストとゲスト VM を接続する仮想インターフェイスである VNet インターフェイスでlro機能をオフにする必要があります。

```
[root@localhost ~]# brctl show eth6_br

bridge name     bridge  id               STP enabled interfaces

eth6_br         8000.00e0ed1861ae          no         eth6

                                                      vnet0

                                                      vnet2

[root@localhost ~]#
```

上記の例では、2 つの仮想インターフェイスは eth6_br から派生し、vnet0 および vnet2 として表されます。次のコマンドを実行して、これらのインターフェイスのgro機能とlro機能をオフにします。


```
        ethtool -K vnet0 gro off
                ethtool -K vnet2 gro off
                ethtool -K vnet0 lro off
                         ethtool -K vnet2 lro off
```

無差別モード

次の機能を動作させるには、無差別モードを有効にする必要があります。

- L2 モード
- マルチキャストトラフィック処理
- ブロードキャスト
- IPV6 トラフィック
- 仮想 MAC
- 動的ルーティング

次のコマンドを使用して、無差別モードを有効にします。

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

[root@localhost ~]#
```

必要なモジュール

ネットワークパフォーマンスを向上させるには、Linux ホストに vhost_net モジュールが存在することを確認してください。vhost_net モジュールの存在を確認するには、Linux ホストで次のコマンドを実行します。

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

vhost_net がまだ実行されていない場合は、次のコマンドを入力して実行します。

```
modprobe vhost\_net
```


OpenStack を使用して Citrix ADC VPX インスタンスをプロビジョニングする

October 17, 2024

OpenStack 環境で Citrix ADC VPX インスタンスをプロビジョニングするには、**Nova** ブートコマンド（OpenStack CLI）または Horizon（OpenStack ダッシュボード）を使用します。

VPX インスタンスをプロビジョニングします。オプションで、コンフィグドライブのデータを使用します。「構成ドライブ」とは、インスタンスの起動時に CD-ROM デバイスとしてアタッチされる特殊な構成ドライブを指します。この構成ドライブは、管理 IP アドレス、ネットワークマスク、デフォルトゲートウェイなど、ネットワーク構成を渡すためや、顧客スクリプトを注入するために使用できます。

NetScaler アプライアンスでは、デフォルトの認証方式はパスワードベースです。現在、OpenStack 環境上の Citrix ADC VPX インスタンスでは、SSH キーペア認証メカニズムがサポートされています。

キーペア (公開鍵と秘密キー) は、公開鍵暗号化メカニズムを使用する前に生成されます。Horizon、Windows 用 Puttygen.exe、Linux 環境用`ssh-keygen` など、さまざまなメカニズムを使用して、キーペアを生成できます。キーペアの生成について詳しくは、それぞれの方式のオンラインドキュメントを参照してください。

キーペアが利用可能になったら、権限のあるユーザーがアクセスできる安全な場所に秘密鍵をコピーします。OpenStack では、Horizon または Nova ブートコマンドを使用して、VPX インスタンスにパブリックキーをデプロイできます。OpenStack を使用して VPX インスタンスをプロビジョニングすると、まず特定の BIOS 文字列を読み取って、インスタンスが OpenStack 環境で起動していることを検出します。この文字列は「OpenStack Foundation」であり、Red Hat Linux ディストリビューションの場合は、/etc/nova/release に保存されます。これは、KVM ハイパーバイザープラットフォームに基づくすべての OpenStack 実装で利用できる標準的なメカニズムです。ドライブには特定の OpenStack ラベルが必要です。

ネットワ ― ク構成、カスタムスクリプト、および SSH キーペアが提供されている場合は、構成ドライブが検出されると、インスタンスがそれらを読み取ろうとします。

ユーザーデータファイル

NetScaler VPX インスタンスは、ユーザーデータファイルとも呼ばれるカスタマイズされた OVF ファイルを使用して、ネットワーク構成、カスタムスクリプトを注入します。このファイルは、構成ドライブの一部として提供されます。次に、カスタマイズされた OVF ファイルの例を示します。

```
```
  <?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="openstack-orch-env"/>
<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>
 <cs:ScriptSection>
   <cs:Version>1.0</cs:Version>
     <ScriptSettingSection xmlns="http://schemas.citrix.com/openstack" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <Scripts>
        <Script>
              <Type>shell</Type>
              <Parameter>X Y</Parameter>
             <Parameter>Z</Parameter>
             <BootScript>before</BootScript>
               <Text>
                     #!/bin/bash
                     echo "Hi, how are you" $1 $2 >> /var/sample.txt
              </Text>
        </Script>
        <Script>
              <Type>python</Type>
              <BootScript>after</BootScript>
               <Text>
                    #!/bin/python
print("Hello");
              </Text>
        </Script>
  <Script>
              <Type>perl</Type>
              <BootScript>before</BootScript>
               <Text>
                    !/usr/bin/perl
my $name = "VPX";
print "Hello, World $name !\n" ;
              </Text>
        </Script>
        <Script>
             <Type>nscli</Type>
             <BootScript>after</BootScript>
             <Text>
```



```
                    add vlan 33
    bind vlan 33 -ifnum 1/2
                   </Text>
                </Script>
             </Scripts>
          </ScriptSettingSection>
     </cs:ScriptSection>
   </Environment>
```

前のOVFファイルでは、「PropertySection」はNetScalerネットワーク構成に使用され、\<cs:ScriptSection> はすべてのスクリプトを囲むために使用されます。 \ \</Scripts> タグは、すべてのスクリプトをまとめるのに使われます。 各スクリプトは\<Script> \</Script>タグの間に定義されています。 各スクリプトタグには、従属するフィールドやタグがあります。

a) <Type>: スクリプトタイプの値を指定します。指定可能な値: Shell/Perl/Python/NSLCI（NetScaler CLI スクリプトの場合）

b) <Parameter>: スクリプトにパラメーターを指定します。各スクリプトでは、複数の <Parameter> タグを使用できます。

c) <BootScript>: スクリプト実行ポイントを指定します。このタグに指定できる値: 前/後。「before」は、PE がアップする前にスクリプトを実行することを指定します。「after」は、PE が起動した後にスクリプトが実行されることを指定します。

d) <Text>: スクリプトの内容を貼り付けます。

> 注
>
> 現在、VPX インスタンスはスクリプトのサニタイズを処理しません。管理者は、スクリプトの有効性を確認する必要があります。
>
> すべてのセクションを表示する必要はありません。空の「PropertySection」を使用して最初のブート時に実行するスクリプトのみを定義するか、
>
> OVF ファイル（ユーザーデータファイル）の必要なセクションが入力されたら、そのファイルを使用して VPX インスタンスをプロビジョニングします。

ネットワーク構成

ネットワーク構成の一部として、VPX インスタンスは以下を読み込みます。

- 管理 IP アドレス
- ネットワークマスク
- デフォルトゲートウェイ

パラメーターは、正常に読み取られると、インスタンスをリモートで管理できるように NetScaler 構成に移入されます。パラメーターが読み取られない場合、または構成ドライブが存在しない場合は、インスタンスが以下のデフォルトの処理を実行します。

- DHCP から IP アドレス情報を取得する。
- DHCP で障害が発生するか、タイムアウトした場合、インスタンスはデフォルトのネットワーク設定（192.168.100.1/16）で起動します。

カスタマースクリプト

VPX インスタンスでは、初期プロビジョニング中にカスタムスクリプトを実行できます。アプライアンスは、シェル、Perl、Python、および Citrix ADC CLI コマンドタイプのスクリプトをサポートしています。

SSH キーペア認証

VPX インスタンスは、インスタンスメタデータの一部として構成ドライブ内で利用可能なパブリックキーをその「authorized_keys」ファイルにコピーします。これにより、ユーザーが秘密キーを使用してインスタンスにアクセスできるようになります。

> 注
>
> SSH キーが提供されると、デフォルトの認証情報（nsroot/nsroot）は機能しなくなります。パスワードベースのアクセスが必要な場合は、それぞれの SSH プライベートキーでログオンし、手動でパスワードを設定します。

はじめに

OpenStack 環境で VPX インスタンスをプロビジョニングする前に、.tgz ファイルから.qcow2ファイルを抽出してビルドします。

qcow2 イメージからの OpenStack イメージ。以下の手順を実行します：

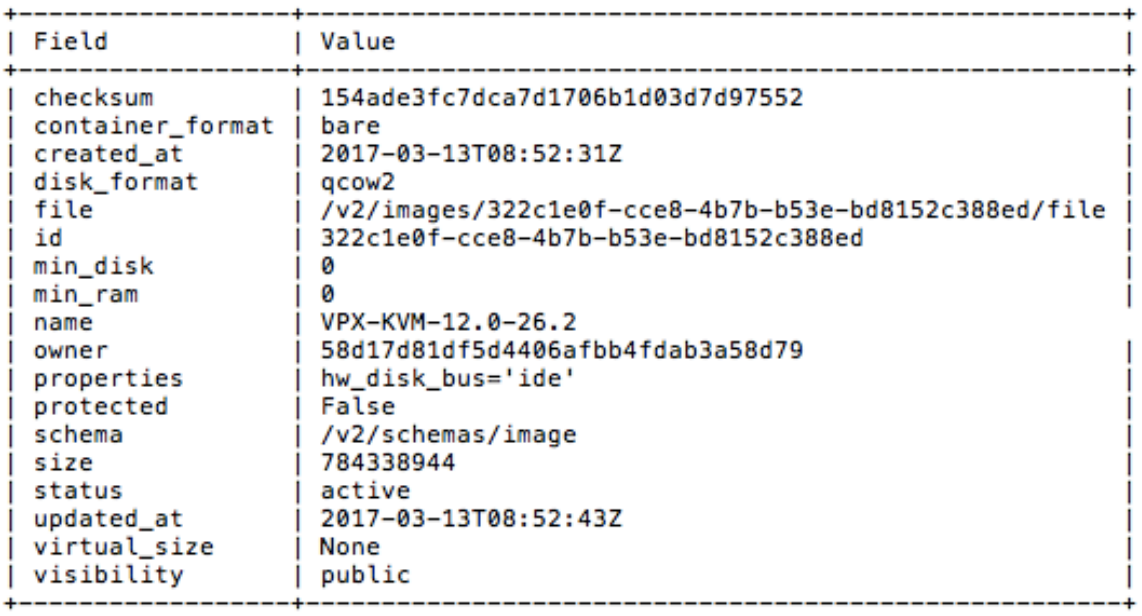
1. 次のコマンドを入力して、.tqzファイルから.qcow2ファイルを抽出します。

```
tar xvzf <TAR file>
tar xvzf <NSVPX-KVM-12.0-26.2_nc.tgz>
NSVPX-KVM.xml
NSVPX-KVM-12.0-26.2_nc.qcow2
```

2. 次のコマンドを入力して、手順 1 で抽出した.qcoz2ファイルを使用して OpenStack イメージをビルドします。

```
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>

glance image-create --name="NS-VPX-12-0-26-2" --property
    hw_disk_bus=ide --ispublic=
true --container-format=bare --disk-format=qcow2< NSVPX-KVM
    -12.0-26.2_nc.qcow2
```


図 **1**: 次の図に、glance image-create コマンドの出力例を示します。



```
+------------------+--------------------------------------------------------+
| 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                                                 |
+------------------+--------------------------------------------------------+
```

VPX インスタンスのプロビジョニング

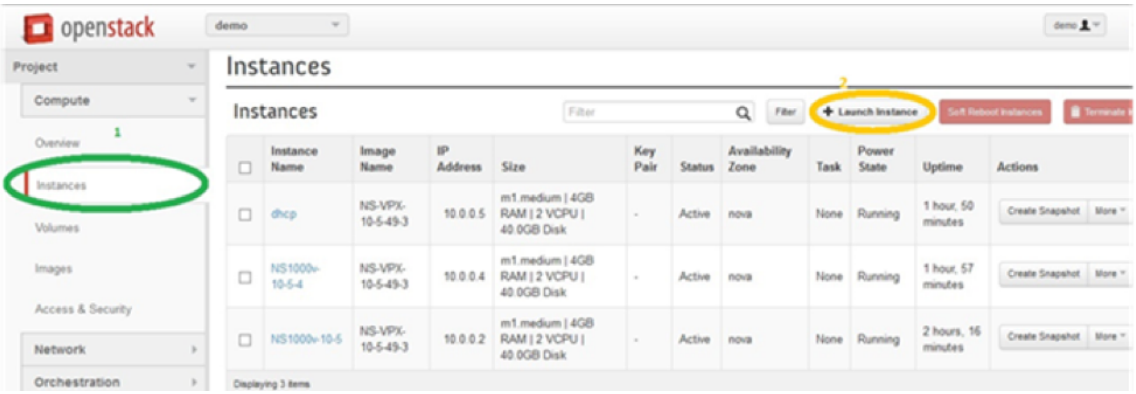
VPX インスタンスをプロビジョニングするには、次のいずれかの方法を使用します。

- Horizon（OpenStack ダッシュボード）
- Nova boot コマンド（OpenStack CLI）

OpenStack ダッシュボードを使用して **VPX** インスタンスをプロビジョニングする

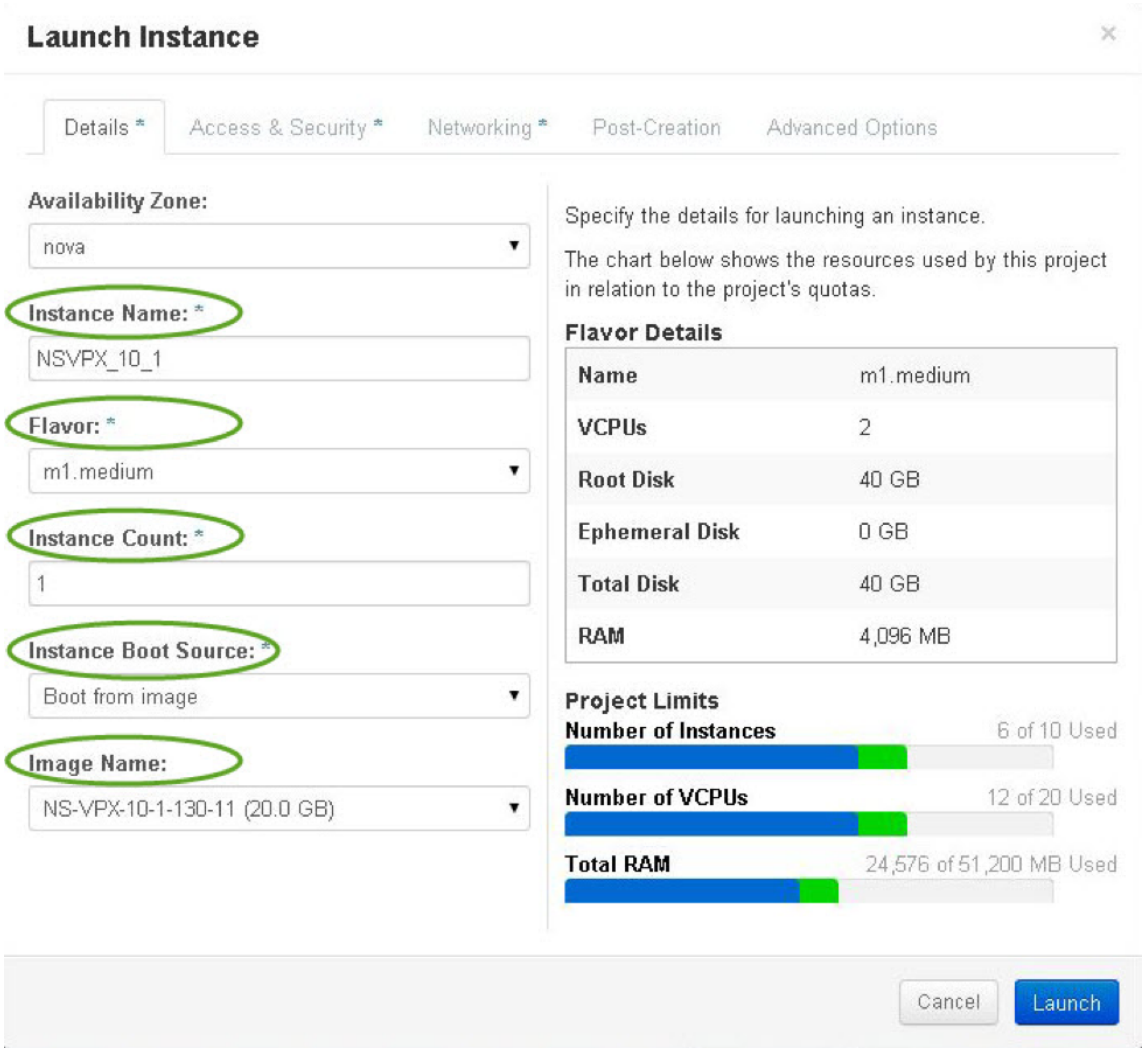
Horizon を使用して VPX インスタンスをプロビジョニングするには、次の手順に従います。

1. OpenStack ダッシュボードにログオンします。
2. ダッシュボードの左側にある [プロジェクト] パネルで、[インスタンス] を選択します。
3. [インスタンス] パネルで、[インスタンスの起動] をクリックして、[インスタンスの起動] ウィザードを開きます。



4. インスタンスの起動ウィザードで、次のような詳細を入力します。

    a) Instance Name - インスタンス名
    b) Flavor - インスタンスのフレーバー（種類）
    c) Instance Count - インスタンスの数
    d) Instance Boot Source - インスタンスの起動ソース
    e) イメージ名



5. 次の手順を実行して、Horizon を介して新しいキーペアか既存のキーペアを展開します。

   a) 既存のキーペアがない場合は、既存の方式を使用してキーを作成します。既存のキーがある場合は、この手順はスキップします。

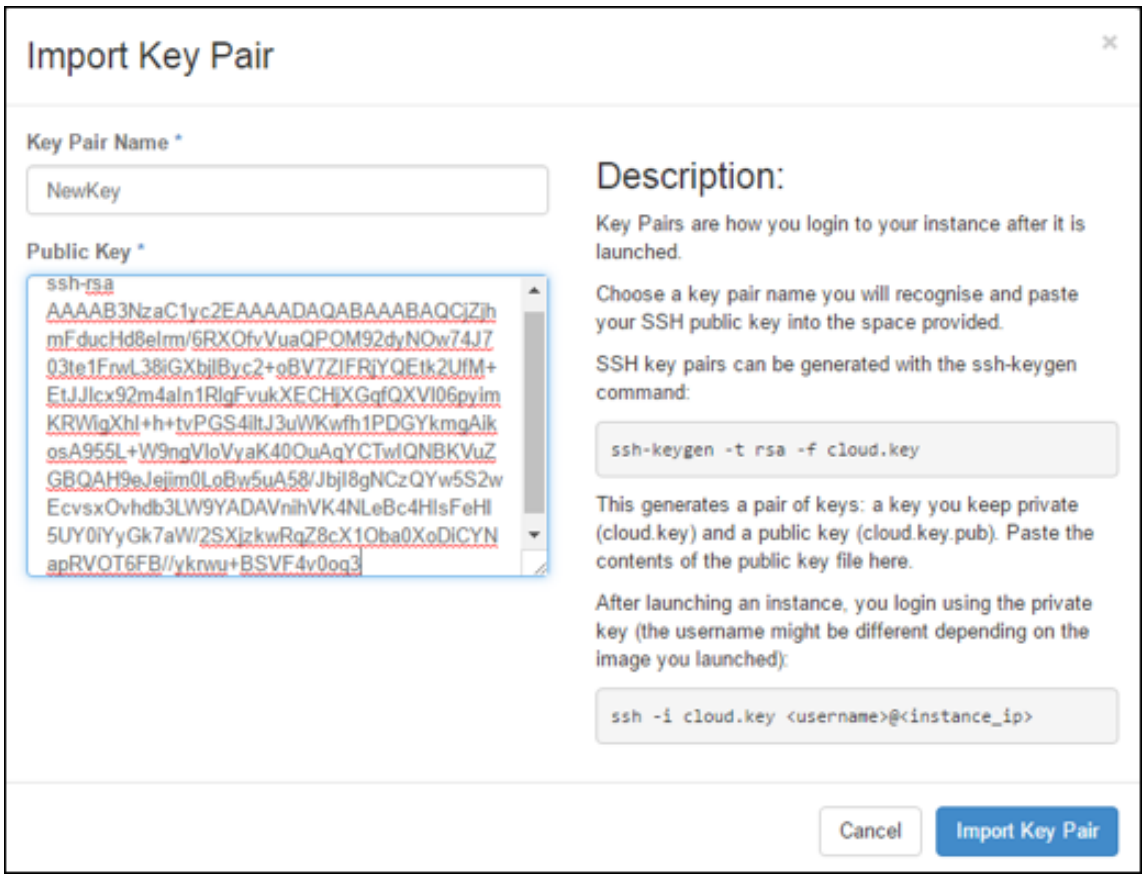
   b) 公開キーの内容をコピーします。

   c) [**Horizon] > [**インスタンス**] > [**新しいインスタンスの作成**]** の順に選択します。

   d) [アクセスとセキュリティ] をクリックします。

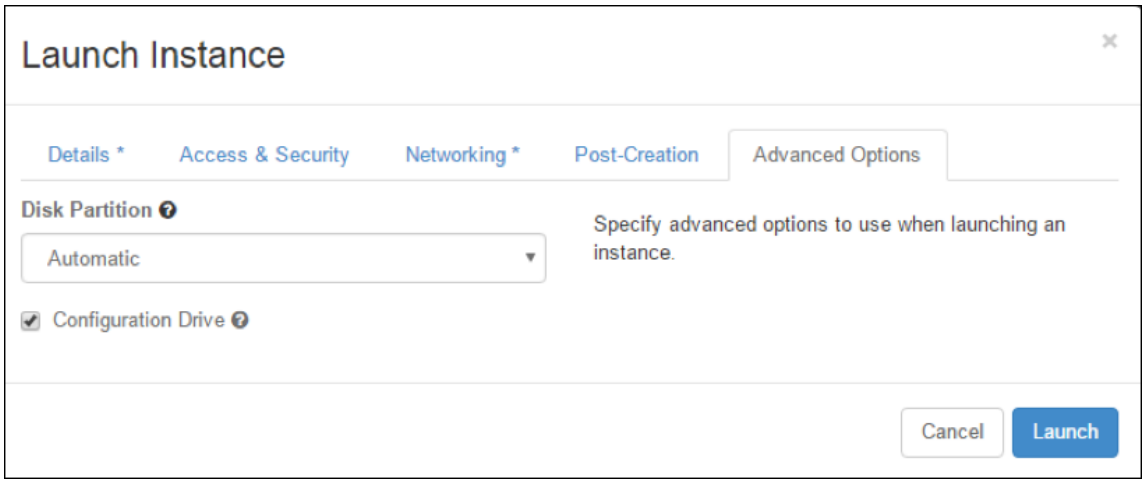
   e) [**Key Pair**] ドロップダウンメニューの隣にある [+] 記号をクリックし、表示されるパラメータの値を入力します。

   f) 公開鍵の内容を 公開鍵 ボックスに貼り付け、鍵に名前を付け、[鍵 ペアのインポート] をクリックします。



6. ウィザードの [ポスト作成] タブをクリックします。[カスタマイズスクリプト] で、ユーザーデータファイルのコンテンツを追加します。ユーザーデータファイルには、VPX インスタンスの IP アドレス、ネットマスクとゲートウェイの詳細、およびカスタマースクリプトが含まれます。

7. キーペアを選択またはインポートした後、config-drive オプションをチェックし、**Launch** をクリックします。



OpenStack CLI を使用して **VPX** インスタンスをプロビジョニングする

OpenStack CLI を使用して VPX インスタンスをプロビジョニングするには、次の手順に従います。

1. qcow2 からイメージを作成するには、次のコマンドを入力します。

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

2. インスタンスを作成するイメージを選択するには、次のコマンドを入力します。

   ```
   openstack image list | more
   ```

3. 特定のフレーバーのインスタンスを作成するには、次のコマンドを入力して、リストからフレーバー ID/名前を選択します。

   ```
   openstack flavor list
   ```

4. NIC を特定のネットワークに接続するには、次のコマンドを入力して、ネットワークリストからネットワーク ID を選択します。

   ```
   openstack network list
   ```

5. インスタンスを作成するには、次のコマンドを入力します。

   ```
   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
      -3efd44b761b9
   6  VPX-ToT
   ```

図 **2**: 次の図は、出力例を示しています。

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	[{u'name': u'default'}]
status	BUILD
updated	2017-03-13T10:33:00Z
user_id	a6347b33916b4eb1b1f76360a9c8f935

仮想マシンマネージャーを使用して **NetScaler VPX** インスタンスをプロビジョニングします

October 17, 2024

Virtual Machine Manager は、VM ゲストを管理するためのデスクトップツールです。これによって新しい VM ゲストおよびさまざまな種類のストレージを作成し、仮想ネットワークを管理できます。組み込み VNC ビューアーにより VM ゲストのグラフィカルコンソールにアクセスして、ローカルまたはリモートでパフォーマンス統計を閲覧できます。

優先 Linux ディストリビューションをインストールした後、KVM 仮想化を有効にして、仮想マシンのプロビジョニングを処理できます。

仮想マシンマネージャーを使用して NetScaler VPX インスタンスをプロビジョニングする場合、2 つのオプションがあります。

- 手動で IP アドレス、ゲートウェイ、およびネットマスクを入力する
- IP アドレス、ゲートウェイ、ネットマスクを自動的に割り当てる (自動プロビジョニング)

NetScaler VPX インスタンスのプロビジョニングには、次の 2 種類のイメージを使用できます。

- RAW
- QCOW2

NetScaler VPX RAW イメージを QCOW2 イメージに変換して、NetScaler VPX インスタンスをプロビジョニングできます。RAW イメージを QCOW2 イメージに変換するには、次のコマンドを入力します。

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

例えば**:**

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

KVM での一般的な NetScaler VPX 展開には、次の手順があります。

- NetScaler VPX インスタンスを自動プロビジョニングするための前提条件の確認
- RAW イメージを使用した NetScaler VPX インスタンスのプロビジョニング
- QCOW2 イメージを使用した NetScaler VPX インスタンスの Provisioning
- Virtual Machine Manager を使用した VPX インスタンスへのインターフェイスの追加

NetScaler VPX インスタンスの自動プロビジョニングの前提条件を確認する

自動プロビジョニングはオプション機能であり、CDROM ドライブからのデータの使用を伴います。この機能が有効になっている場合は、初期セットアップ時に、NetScaler VPX インスタンスの管理 IP アドレス、ネットワークマスク、およびデフォルトゲートウェイを入力する必要があります。

VPX インスタンスを自動プロビジョニングする前に、次のタスクを完了する必要があります。

1. カスタマイズされたオープン仮想化形式 (OVF) XML ファイルまたはユーザーデータファイルを作成します。
2. オンラインアプリケーション（たとえば、PowerISO）を使用して、OVF ファイルを ISO イメージに変換します。
3. セキュアコピー (SCP) ベースのツールを使用して、ISO イメージを KVM ホストにマウントします。

サンプル **OVF XML** ファイル:

次に、OVF XML ファイルの内容の例を示します。このファイルをサンプルとして使用して、ファイルを作成することができます。

```
<?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>

```



```
15    <Kind></Kind>
16
17    <Version>2016.1</Version>
18
19    <Vendor>VPX</Vendor>
20
21    <Locale>en</Locale>
22
23    </PlatformSection>
24
25    <PropertySection>
26
27    <Property oe:key="com.citrix.netscaler.ovf.version" oe:value="1.0"/>
28
29    <Property oe:key="com.citrix.netscaler.platform" oe:value="NSVPX"/>
30
31    <Property oe:key="com.citrix.netscaler.orch\_env" oe:value="KVM"/>
32
33    <Property oe:key="com.citrix.netscaler.mgmt.ip" oe:value="10.1.2.22"
          />
34
35    <Property oe:key="com.citrix.netscaler.mgmt.netmask" oe:value="
          255.255.255.0"/>
36
37    <Property oe:key="com.citrix.netscaler.mgmt.gateway" oe:value="
          10.1.2.1"/>
38
39    </PropertySection>
40
41    </Environment>
```

前述の OVF XML ファイルでは、NetScaler ネットワーク構成に「PropertySection」が使用されています。ファイルを作成するときには、この例の最後で強調表示されている、パラメーターの値を指定します。

- 管理 IP アドレス
- ネットマスク
- Gateway

> 重要
>
> OVF ファイルが適切に XML 形式になっていない場合、VPX インスタンスにはファイルに指定されている値ではなく、デフォルトのネットワーク構成が割り当てられます。

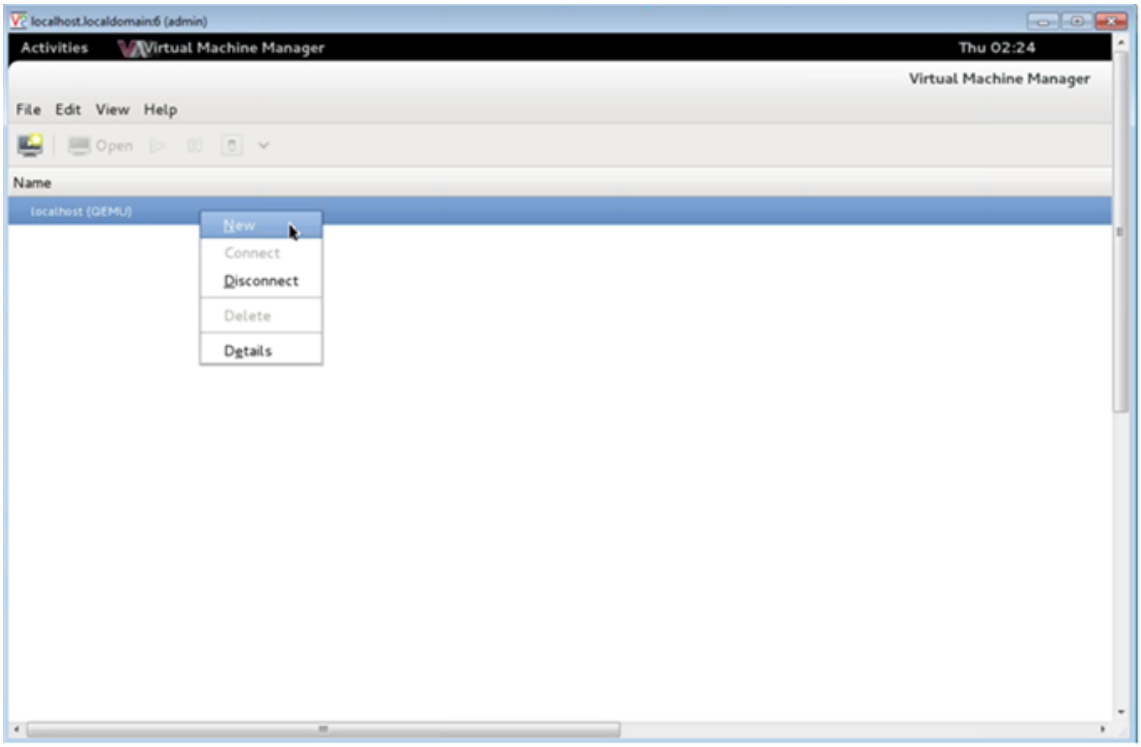
RAW イメージを使用して **NetScaler VPX** インスタンスをプロビジョニングします

Virtual Machine Manager では、RAW イメージを使用して NetScaler VPX インスタンスをプロビジョニングできます。

仮想マシンマネージャーを使用して NetScaler VPX インスタンスをプロビジョニングするには、次の手順に従います。

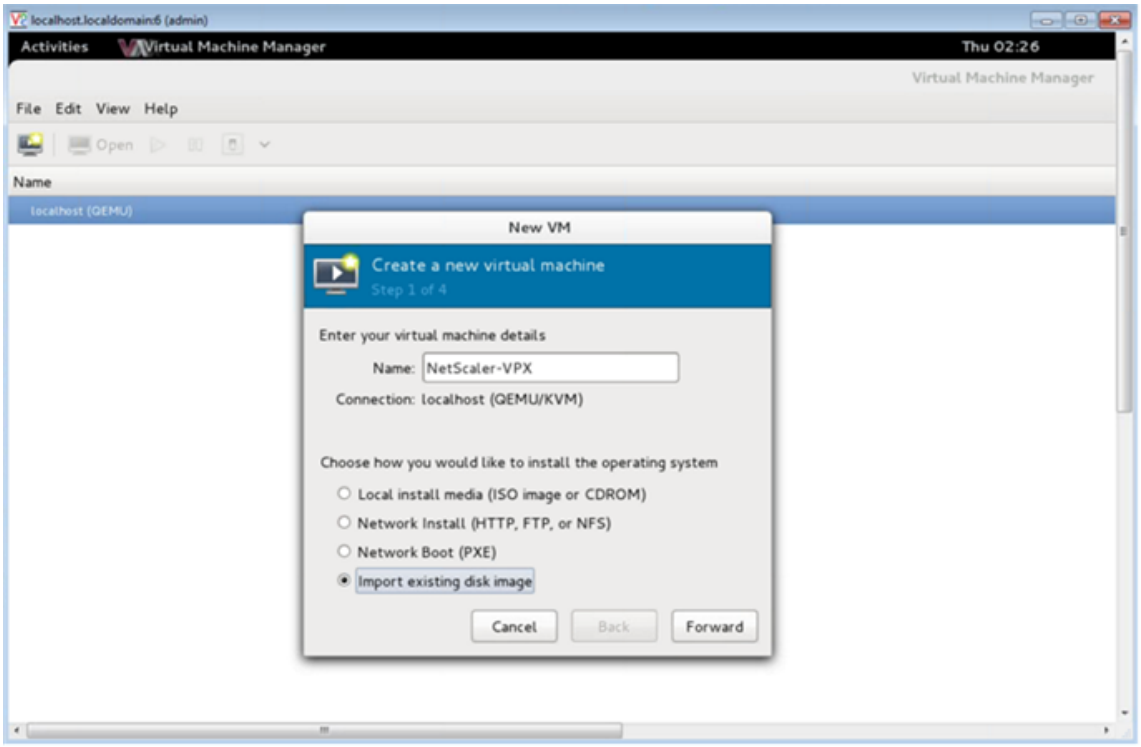
1. 仮想マシンマネージャー (アプリケーション > システムツール > バーチャルマシンマネージャー) を開き、[認証] ウィンドウにログオン資格情報を入力します。

2. 新しい NetScaler VPX インスタンスを作成するには、  アイコンをクリックするか、**localhost (QEMU)** を右クリックします。

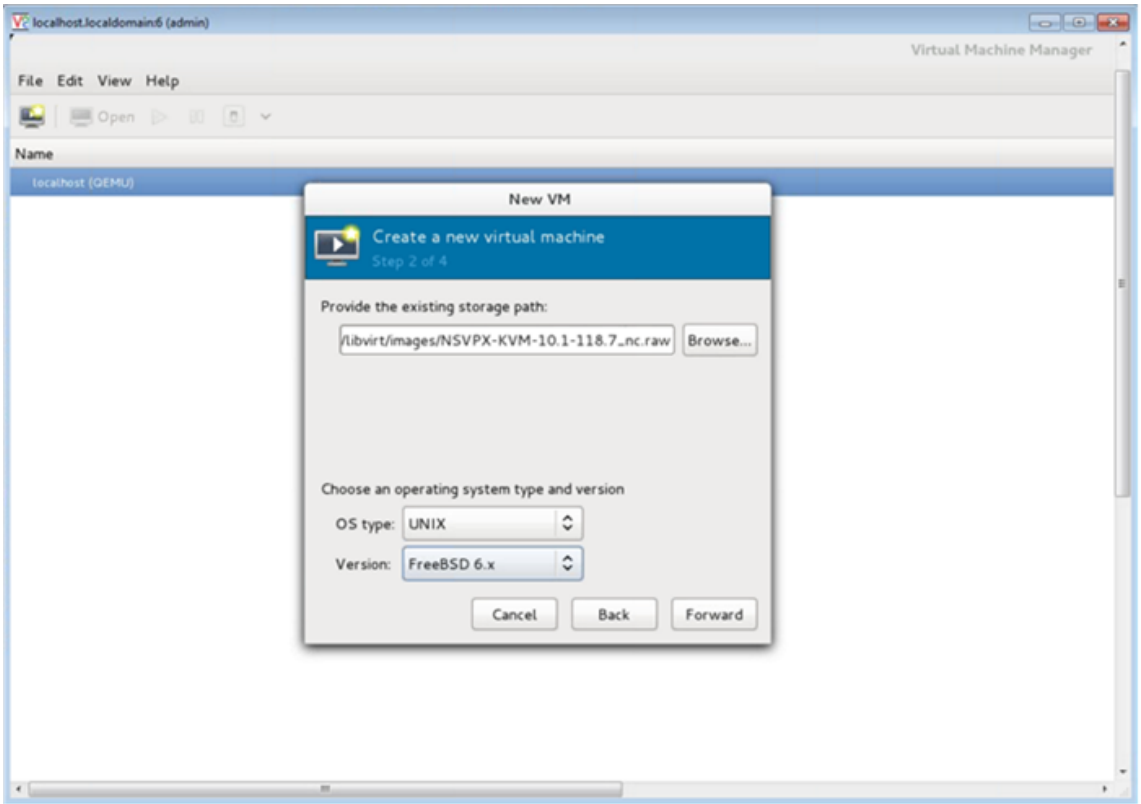


3. 名前テキストボックスに、新しい仮想マシンの名前 (たとえば、NetScaler-VPX) を入力します。

4. ［新規 **VM** ］ウィンドウの［オペレーティングシステムのインストール方法を選択］で［既存のディスクイメージをインポートする］を選択し、［転送］をクリックします。

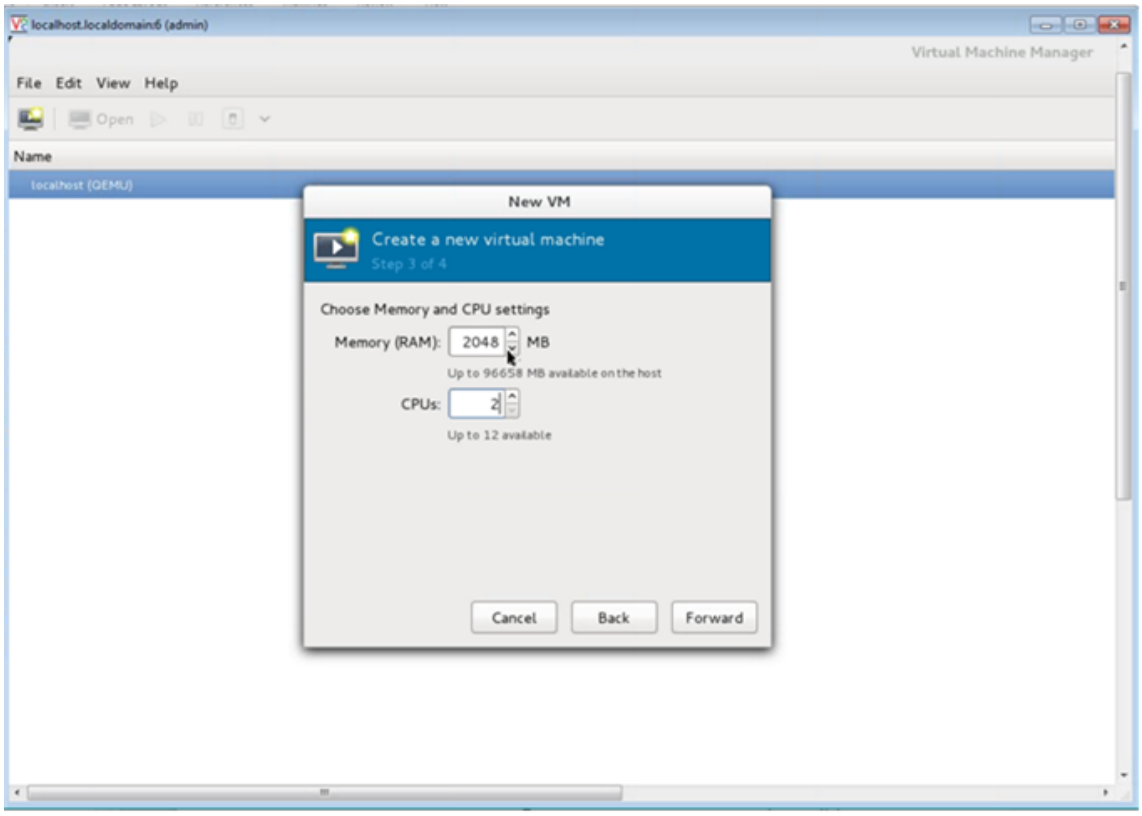


5. 「既存のストレージパスを指定」フィールドで、画像へのパスをナビゲートします。オペレーティングシステム種類に UNIX、バージョンとして FreeBSD 6.x を選択します。次に、「進む」をクリックします。

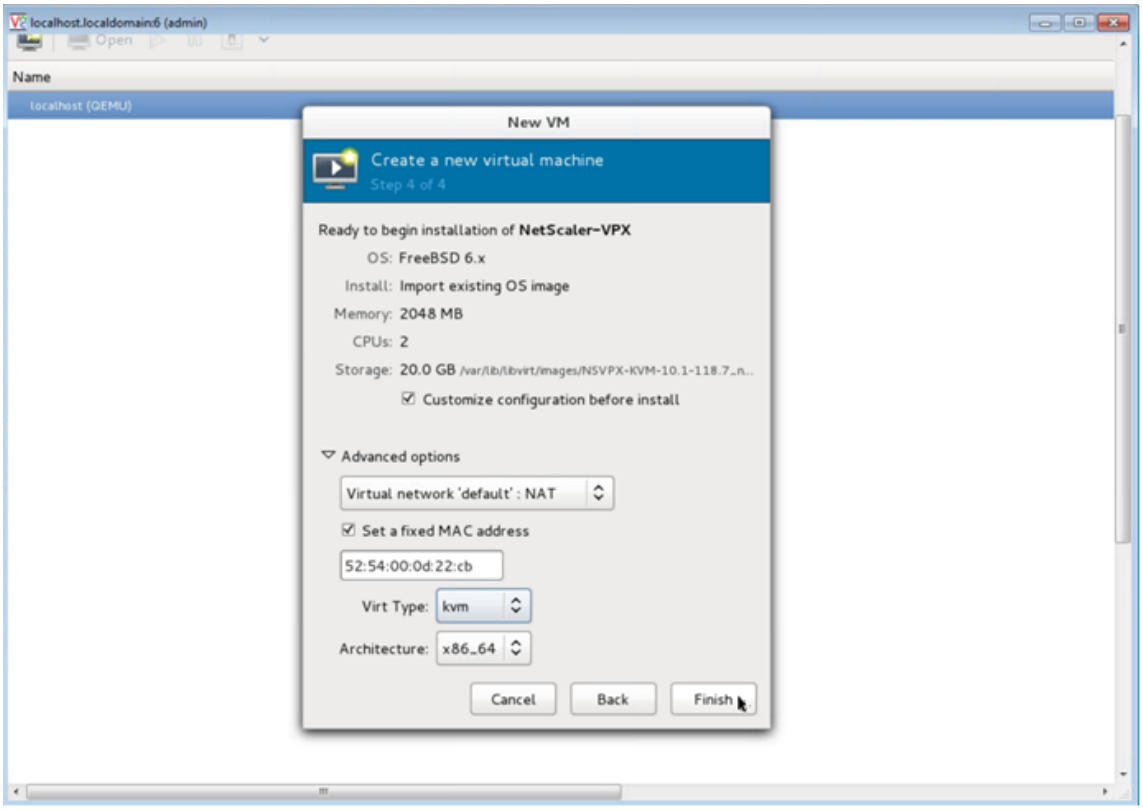


6.「メモリと **CPU** の設定を選択」で次の設定を選択し、「転送」をクリックします。

- メモリ (RAM) –2048MB
- CPU –2

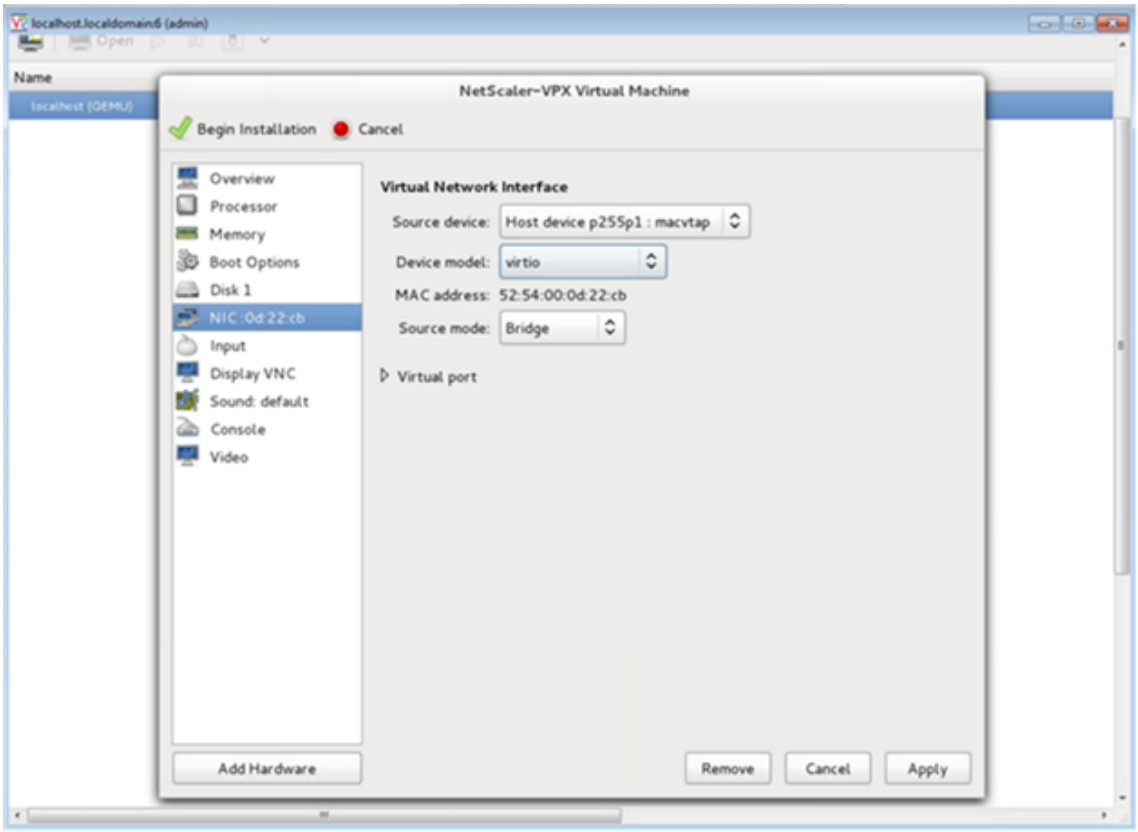


7. [インストール前に構成をカスタマイズする] チェックボックスをオンにします。オプションで、[詳細オプション**]** で MAC アドレスをカスタマイズできます。選択した **Virt** タイプが KVM で、選択されたアーキテクチャが x86_64 であることを確認します。[完了] をクリックします。



8. NIC を選択し、次の構成を指定します。

   - ソースデバイス：`ethX macvtap` またはブリッジ
   - デバイスモデル—`virtio`
   - ソースモード - Bridge



9. [適用] をクリックします。
10. VPX インスタンスを自動プロビジョニングする場合は、このドキュメントの「**CDROM** ドライブを接続して自動 **Provisioning** を有効にする」セクションを参照してください。それ以外の場合は、[インストレーションを開始**]** をクリックします。KVM で Citrix ADC VPX をプロビジョニングしたら、インターフェイスを追加できます。

QCOW2 イメージを使用して **Citrix ADC VPX** インスタンスをプロビジョニングする

仮想マシンマネージャーを使用すると、QCOW2 イメージを使用して NetScaler VPX インスタンスをプロビジョニングできます。

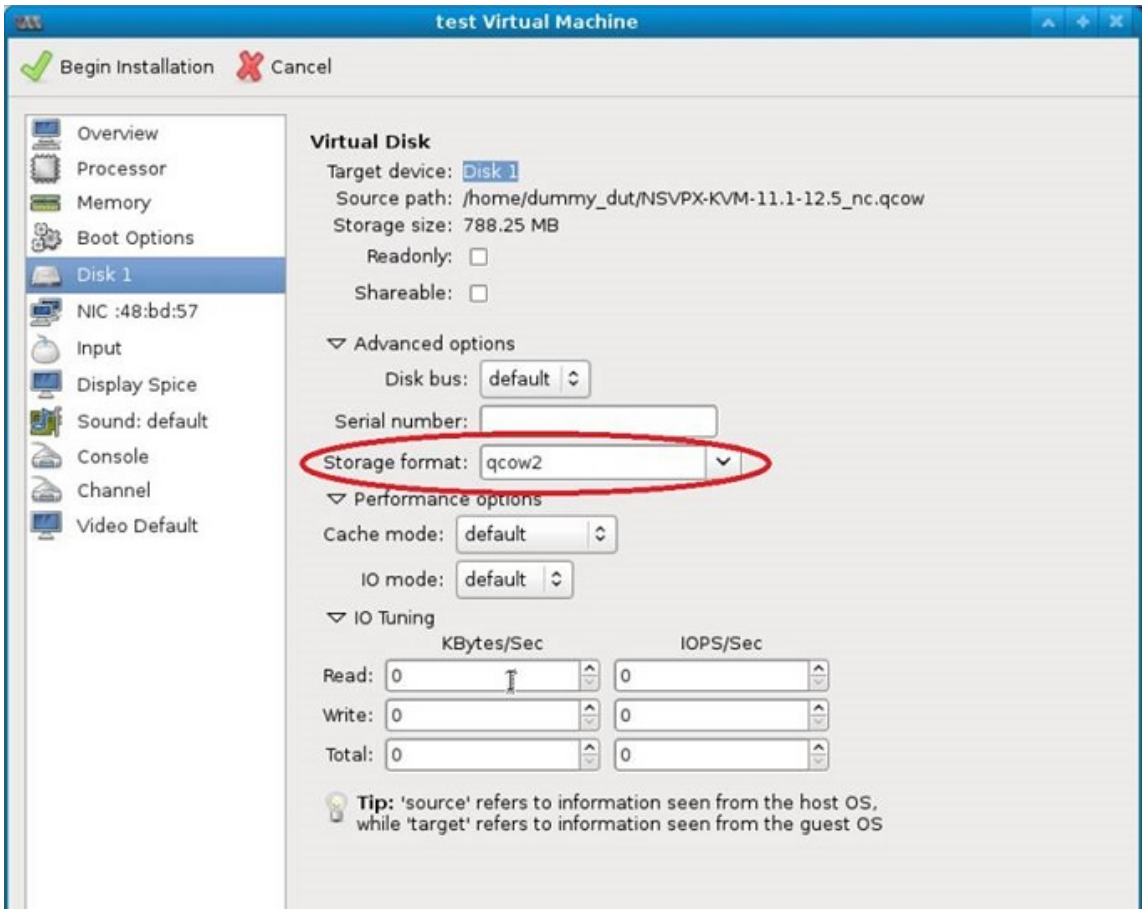
QCOW2 イメージを使用して Citrix ADC VPX インスタンスをプロビジョニングするには、次の手順に従います。

1. RAW イメージを使用した Citrix ADC ****VPX** インスタンスのプロビジョニングの手順 **1**～ステップ **8** に従います **。

   > 注
   >
   > ステップ **5** で **qcow2** イメージが選択されていることを確認してください。

2. **Disk 1** を選択し、[詳細オプション] をクリックします。

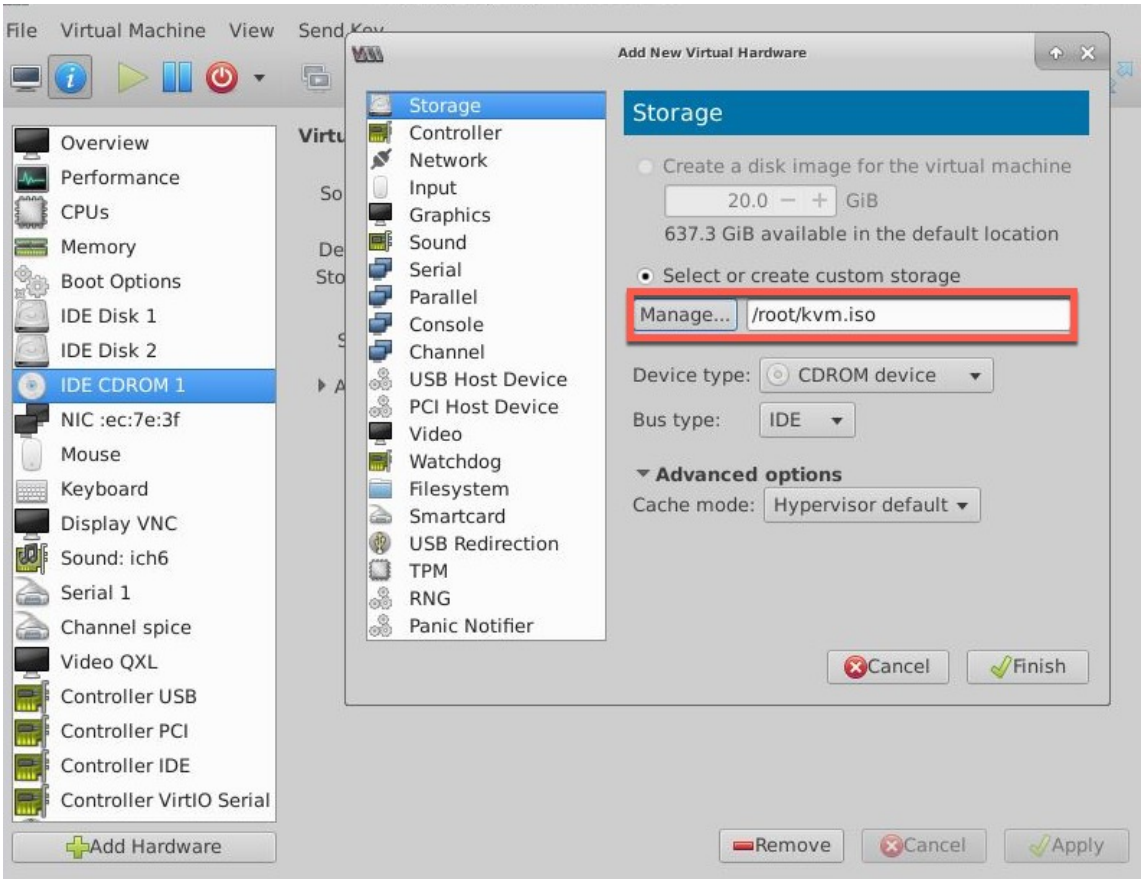
3. [ストレージ形式] ドロップダウンリストから [**qcow2**] を選択します。



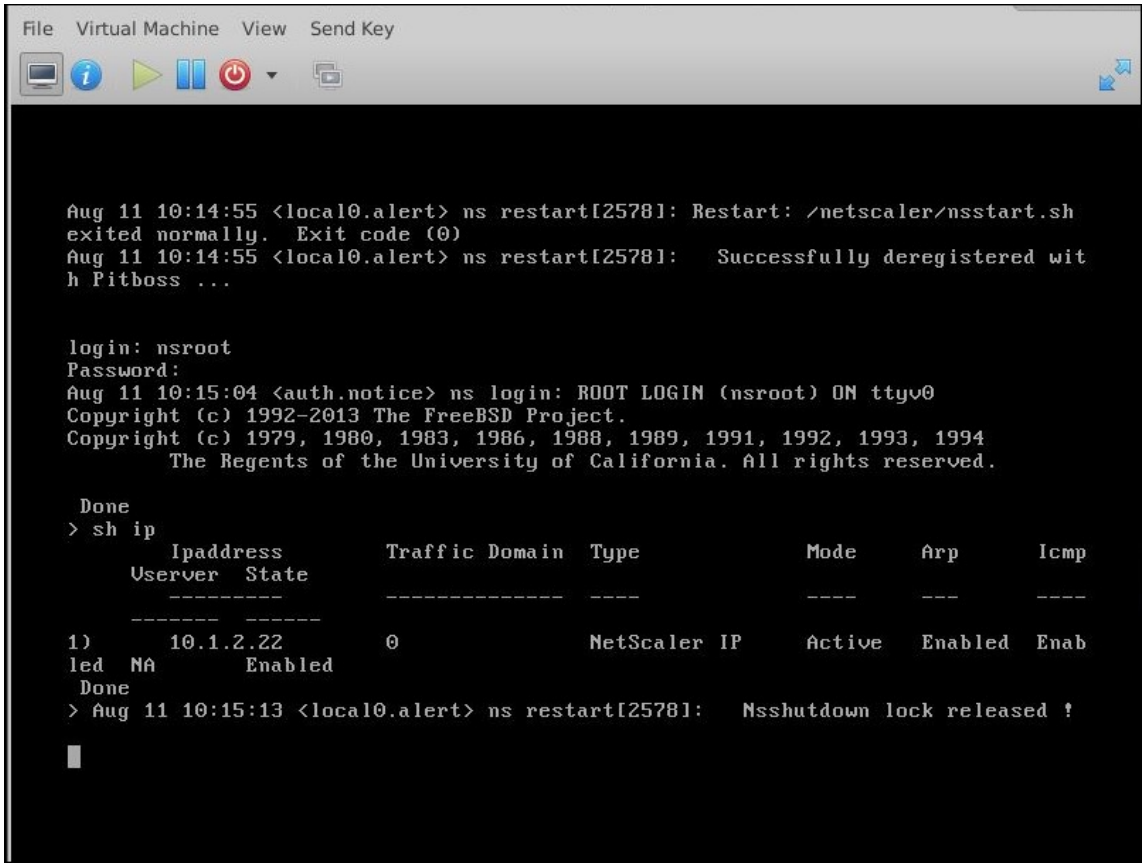
4. [**Apply**] をクリックし、次に [**Begin Installation**] をクリックします。KVM で Citrix ADC VPX をプロビジョニングしたら、インターフェイスを追加できます。

CD-ROM ドライブを接続して自動プロビジョニングを有効にする

1. ハードウェアの追加 **>** ストレージ **>** デバイスタイプ **> CDROM** デバイスをクリックします。
2. [管理] をクリックし、[NetScaler VPX インスタンスの自動プロビジョニングの前提条件] セクションでマウントした正しい ISO ファイルを選択し、[完了] をクリックします。NetScaler VPX インスタンスの [Resources] の下に新しい CDROM が作成されます。



3. VPX インスタンスの電源をオンにすると、スクリーンショットの例で示すように、OVF ファイルで提供されているネットワーク構成を使用して自動プロビジョニングが行われます。



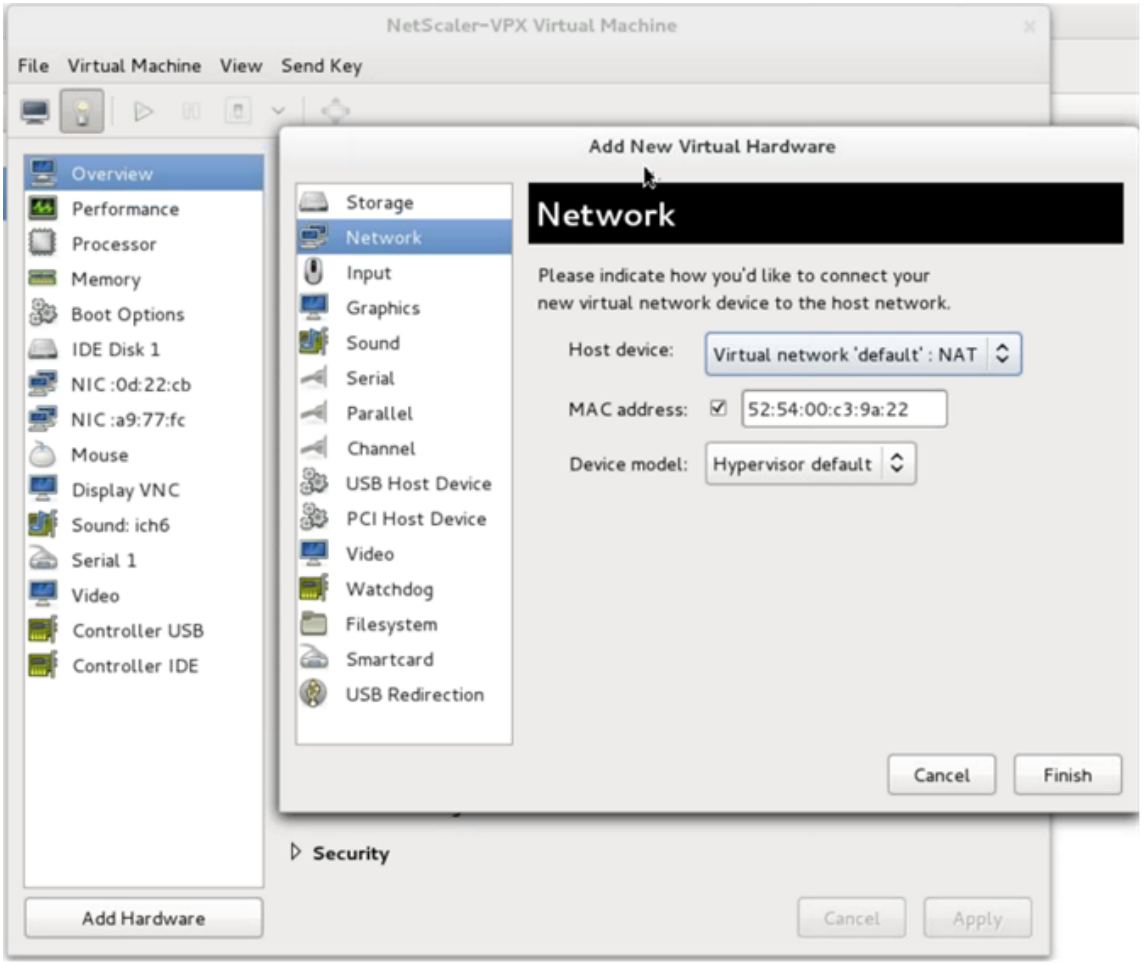
4. 自動プロビジョニングが失敗した場合、インスタンスはデフォルトの IP アドレス（192.168.100.1）で起動します。その場合は、初期設定を手動で完了する必要があります。詳細については、「ADC を初めて構成する」を参照してください。

仮想マシンマネージャーを使用して、**NetScaler VPX** インスタンスにインターフェイスを追加する

KVM で NetScaler VPX インスタンスをプロビジョニングしたら、インターフェイスを追加できます。

インターフェイスを追加するには、次の手順を実行します。

1. KVM の上で動作している NetScaler VPX インスタンスをシャットダウンします。
2. VPX インスタンスを右クリックし、ポップアップメニューから［**Open**］を選択します。
3. ヘッダーの  アイコンをクリックすると、仮想ハードウェアの詳細が表示されます。
4. [ハードウェアの追加] をクリックします。［**Add New Virtual Hardware**］ウィンドウで、ナビゲーションメニューから［**Network**］を選択します。



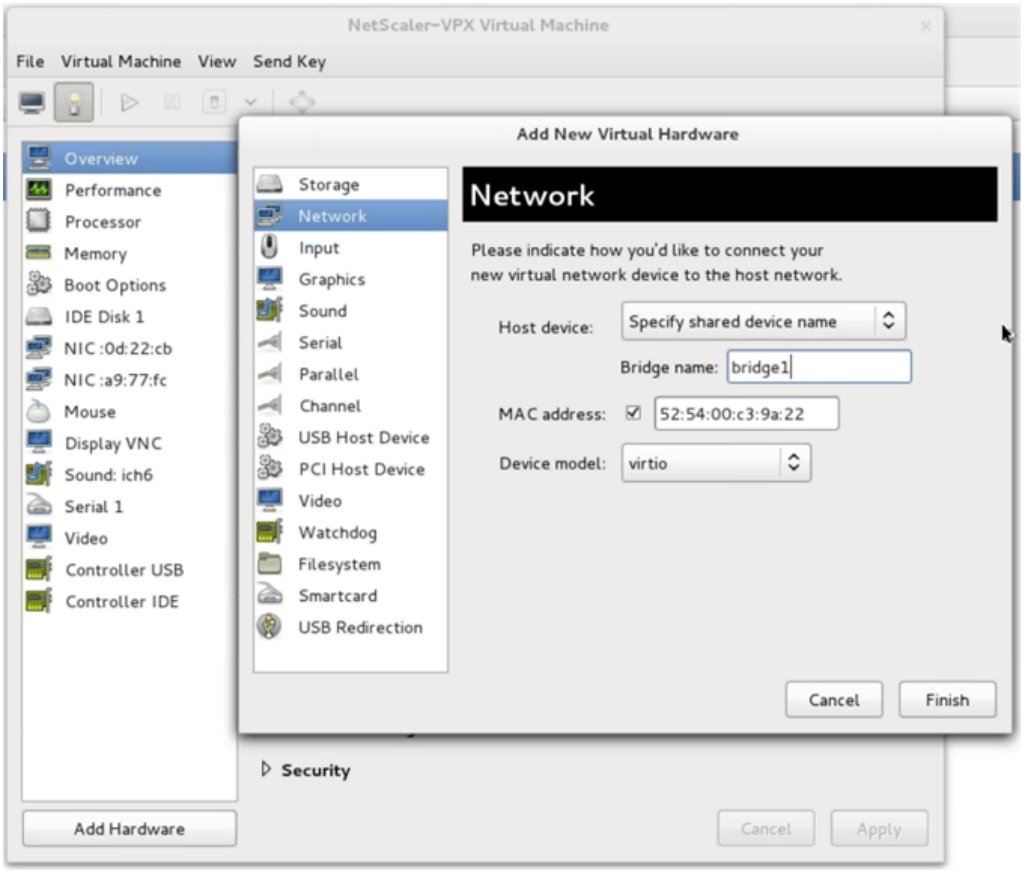
5. [**Host Device**] フィールドで、物理インターフェイスの種類を選択します。ホストデバイスの種類は、Bridge または MacVTap のいずれかにできます。macvTap の場合、VEPA モード、ブリッジ、プライベート、パススルーの 4 つのモードが可能です。

   a) Bridge の場合

      i. Host device - [Specify shared device name] オプションを選択します。

      ii. KVM ホストで構成される Bridge 名を指定します。

      > 注
      >
      > KVM ホストに Linux ブリッジを設定し、物理インターフェイスをブリッジにバインドし、ブリッジを UP 状態にしていることを確認します。



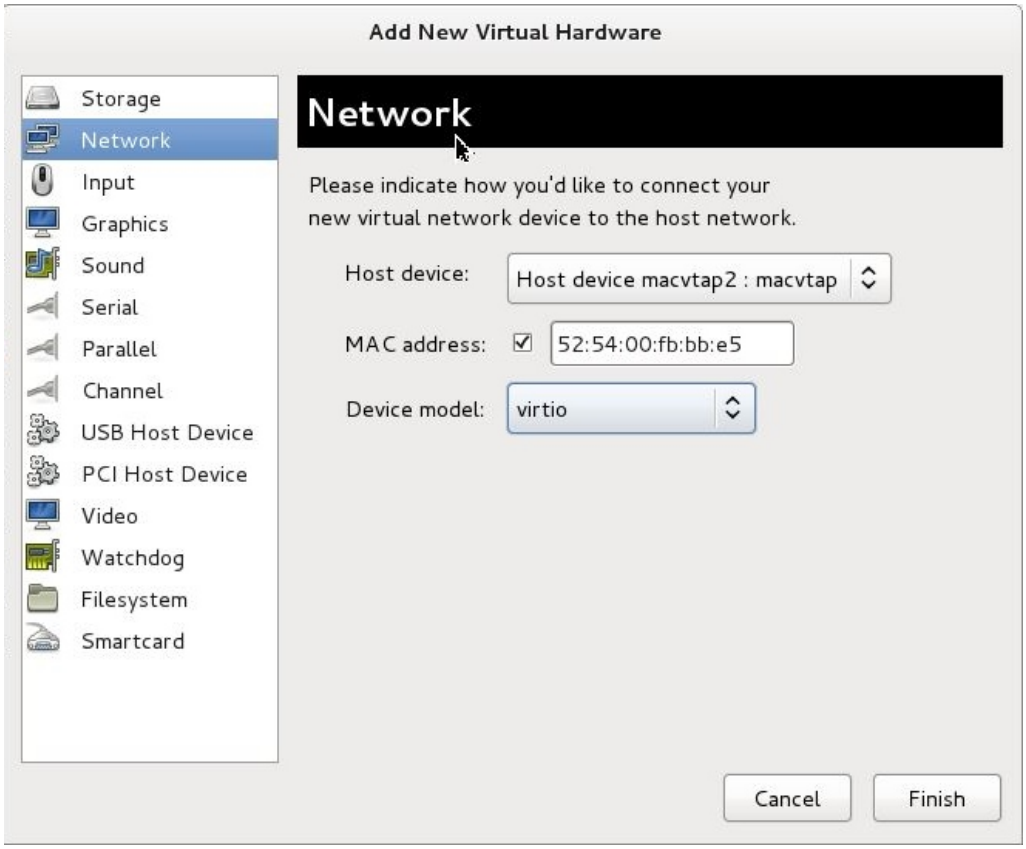
iii. デバイスモデル—`virtio`。

iv. [完了] をクリックします。

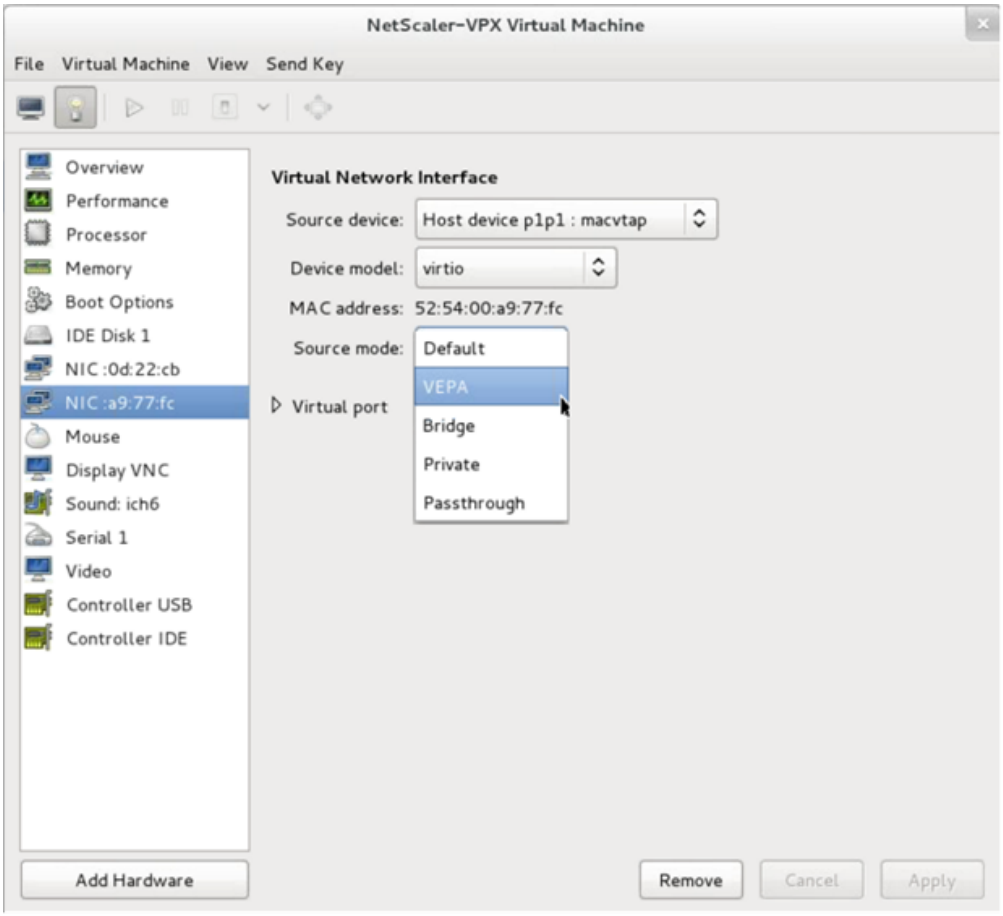
b) MacVTap 用

i. Host device - メニューからの物理インターフェイス

ii. デバイスモデル—`virtio`。



iii. ［完了］をクリックします。ナビゲーションペインで新しく追加された NIC を見ることができます。



iv. 新しく追加された NIC を選択して、この NIC の Source モードを選択します。利用可能なモードは VEPA、Bridge、Private、および Passthrough です。インターフェイスとモードについて詳しくは、「ソースインターフェイスおよびモード」を参照してください

v. ［適用］をクリックします。

6. VPX インスタンスを自動プロビジョニングする場合は、このドキュメントの「自動プロビジョニングを有効にするための構成ドライブの追加」セクションを参照してください。それ以外の場合は、VPX インスタンスをパワーオンして初期構成を手動で完了します。

> 重要:
>
> スピード、デュプレックス、オートネゴシエーションなどの Interface パラメーター構成はサポートされません。

SR-IOV ネットワークインターフェースを使用するように NetScaler VPX インスタンスを構成する

October 17, 2024

Linux-KVM プラットフォームで実行される NetScaler VPX インスタンスは、次の NIC でシングルルート I/O 仮想化（SR-IOV）を使用して構成できます。

- Intel 82599 10G
- Intel X710 10G
- Intel XL710 40G
- インテル X722 10G

このセクションでは、次の方法について説明します。

- SR-IOV ネットワークインターフェイスを使用するように NetScaler VPX インスタンスを構成する
- SR-IOV インターフェイスで静的 LA/LACP を構成する
- SR-IOV インターフェイスで VLAN を構成する

制限事項

インテル 82599、X710、XL710、X722 の NIC を使用する場合は、制限事項に留意してください。次の機能はサポートされません。

インテル **82599 NIC** の制限事項:

- L2 モード切り替え
- 管理パーティション化（共有 VLAN モード）
- 高可用性（アクティブ/アクティブモード）
- ジャンボフレーム。
- IPv6: SR-IOV インターフェイスが 1 つ以上ある場合は、VPX インスタンスで最大 30 個までの一意の IPv6 アドレスのみを設定できます。
- `ip link`コマンドによる SRIOV VF インターフェイスのハイパーバイザでの VLAN 設定はサポートされていません。
- スピート、デュプレックス、オートネゴシエーションなどの Interface パラメーター構成はサポートされません。

インテル **X710 10G**、インテル **XL710 40G**、インテル **X722 10G NIC** の制限事項:

- L2 モード切り替え
- 管理パーティション化（共有 VLAN モード）

- クラスタでは、XL710 NIC がデータ・インタフェースとして使用されている場合、ジャンボフレームはサポートされません。
- インターフェイスが切断され、再接続されると、インターフェイスリストが順序変更されます。
- スピート、デュプレックス、オートネゴシエーションなどの Interface パラメーター構成はサポートされません。
- インターフェイス名は、Intel X710 10G、Intel XL710 40G、Intel X722 10G NIC の場合は 40/X
- VPX インスタンスでは、最大 16 個のインテル XL710/X710/X722 SRIOV または PCI パススルーインターフェイスをサポートできます。

> 注
>
> Intel X710 10G、Intel XL710 40G、および Intel X722 10G NIC で IPv6 をサポートするには、KVM ホストで次のコマンドを入力して、仮想機能 (VF) の信頼モードを有効にする必要があります。
>
> ```
> # ip link set <PNIC> <VF> trust on
> ```
>
> 例
>
> ```
> # ip link set ens785f1 vf 0 trust on
> ```

前提条件

SR-IOV ネットワークインターフェイスを使用するように NetScaler VPX インスタンスを構成する前に、次の前提条件となるタスクを完了してください。対応するタスクを完了する方法の詳細については、「NIC」列を参照してください。

タスク	Intel 82599 NIC	インテル X710、XL710、X722 NIC
1. NIC を KVM ホストに追加します。	-	-
1. 最新の Intel ドライバーをダウンロードしてインストールします。	IXGBE ドライバー	I40E ドライバー
1. KVM ホスト上のドライバーをブロック リストに追加します。	/etc/modprobe.d/blacklist.conf ファイルに次のエントリを追加します。`blacklist ixgbevf`。IXGBE ドライバーのバージョン 4.3.15 を使用します（推奨）。	/etc/modprobe.d/blacklist.conf ファイルに次のエントリを追加します。`blacklist i40evf`。i40e ドライバーのバージョン 2.0.26 を使用します（推奨）。

タスク	Intel 82599 NIC	インテル X710、XL710、X722 NIC
1. KVM ホストで SR-IOV 仮想機能 (VF) を有効にします。次の 2 つの列の両方のコマンドで、number_of_VFs = 作成する仮想 VF の数。device_name = インターフェイス名です。 1. VF の作成に使用したコマンドを rc.local ファイルに追加して、VF を永続化します。	以前のバージョンのカーネル 3.8 を使用している場合は、次のエントリを /etc/modprobe.d/ixgbe ファイルに追加し、KVM ホストを再起動します。options ixgbe max_vfs=<number_of_VFs>。カーネル 3.8 以降を使用している場合は、次のコマンドを使用して VF を作成します。echo <number_of_VFs> > /sys/**class**/net/<device_name>/device/sriov_numvfs。図 1 の例を参照してください。 図 3 の例を参照してください。	以前のバージョンのカーネル 3.8 を使用している場合は、/etc/modprobe.d/i40e.conf ファイルに次のエントリを追加し、KVM ホストを再起動します。options i40e max_vfs=<number_of_VFs>。カーネル 3.8 以降を使用している場合は、次のコマンドを使用して VF を作成します。echo<number_of_VFs> > /sys/**class**/net/<device_name>/device/sriov_numvfs。図 2 の例を参照してください。 図 3 の例を参照してください。

> 重要:
>
> SR-IOV VF を作成するときは、MAC アドレスを VF に割り当てないようにしてください。

図 **1**: インテル 82599 10G NIC の KVM ホストで SR-IOV VF を有効にする

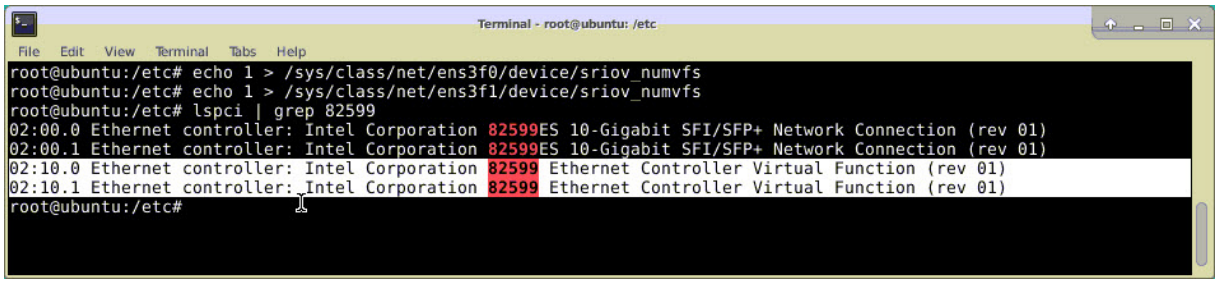


図 **2:** インテル X710 10G および XL710 40G NIC の KVM ホストで SR-IOV VF を有効にする

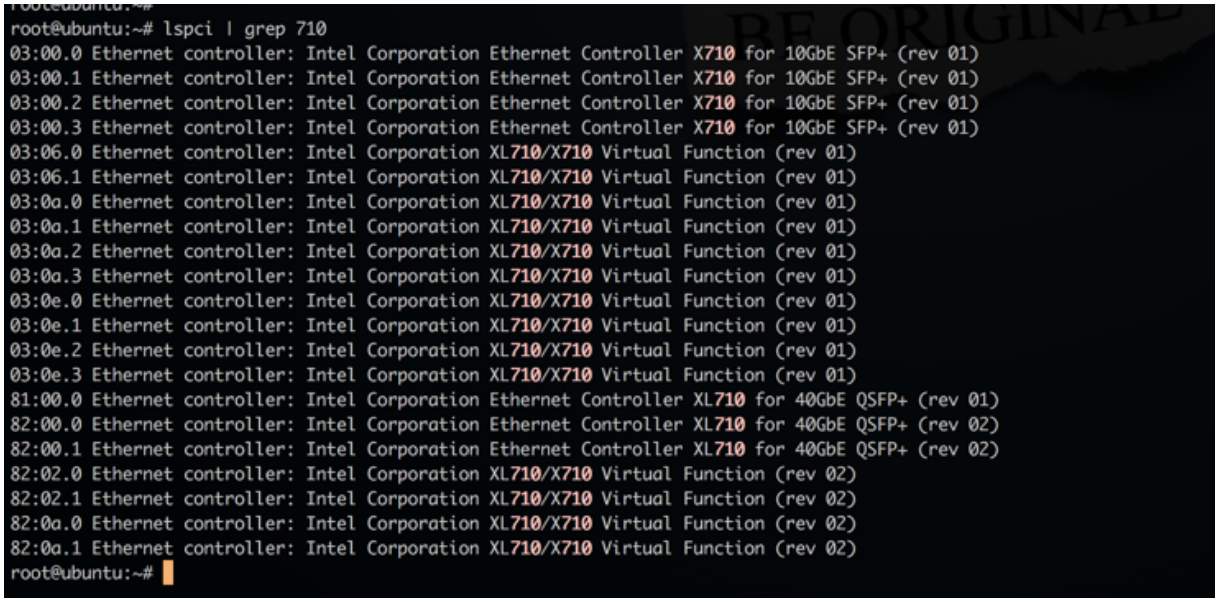


図 **3**: インテル X722 10G NIC の KVM ホストで SR-IOV VF を有効にする

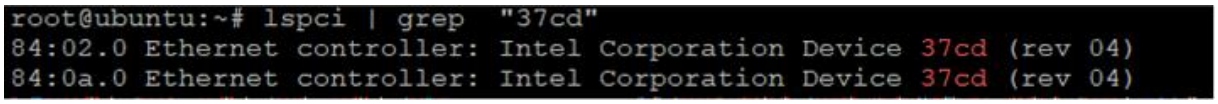
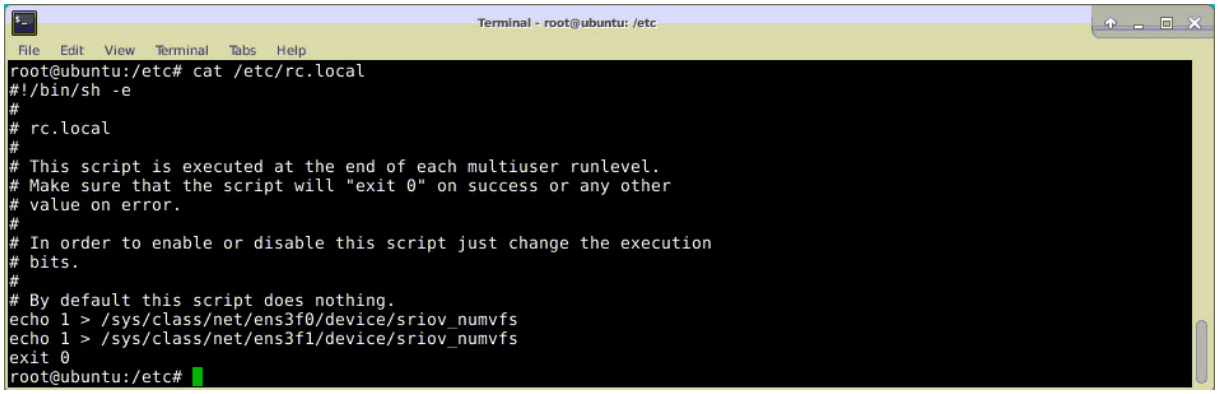


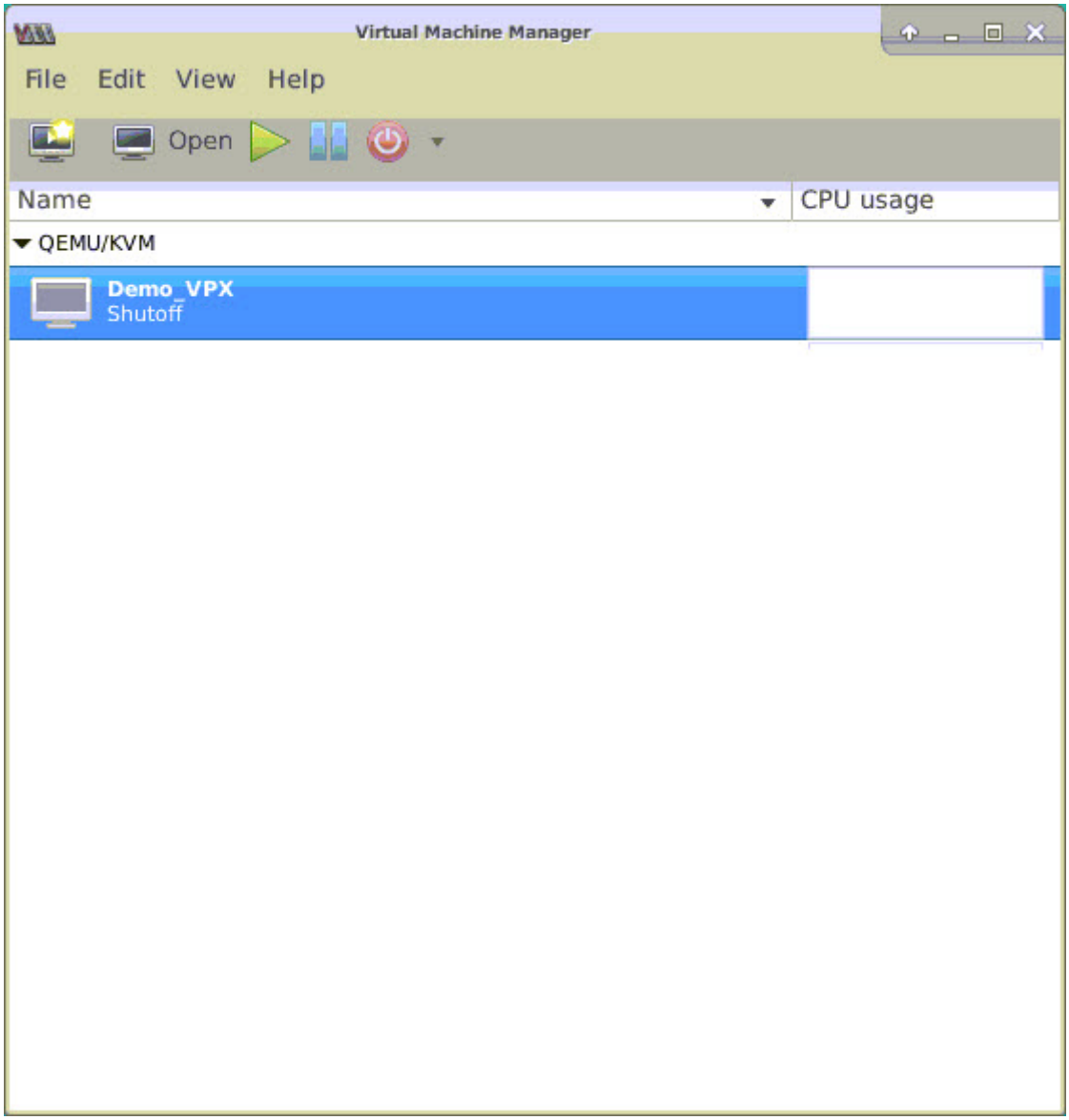
図 **4**: VF を永続的にする



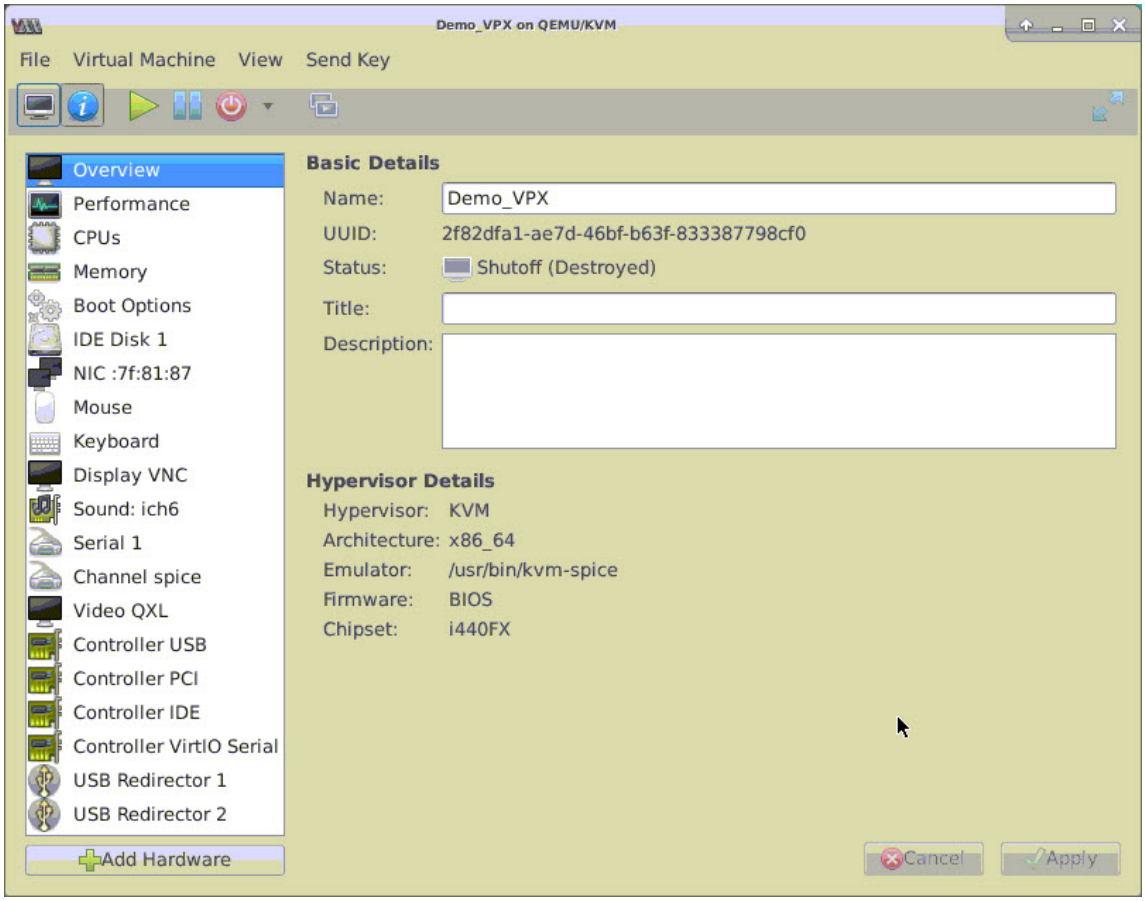
SR-IOV ネットワークインターフェイスの使用を **NetScaler ADC VPX** インスタンスで構成する

仮想マシンマネージャを使用して SR-IOV ネットワークインターフェイスを使用するように NetScaler ADC VPX インスタンスを構成するには、次の手順を実行します。

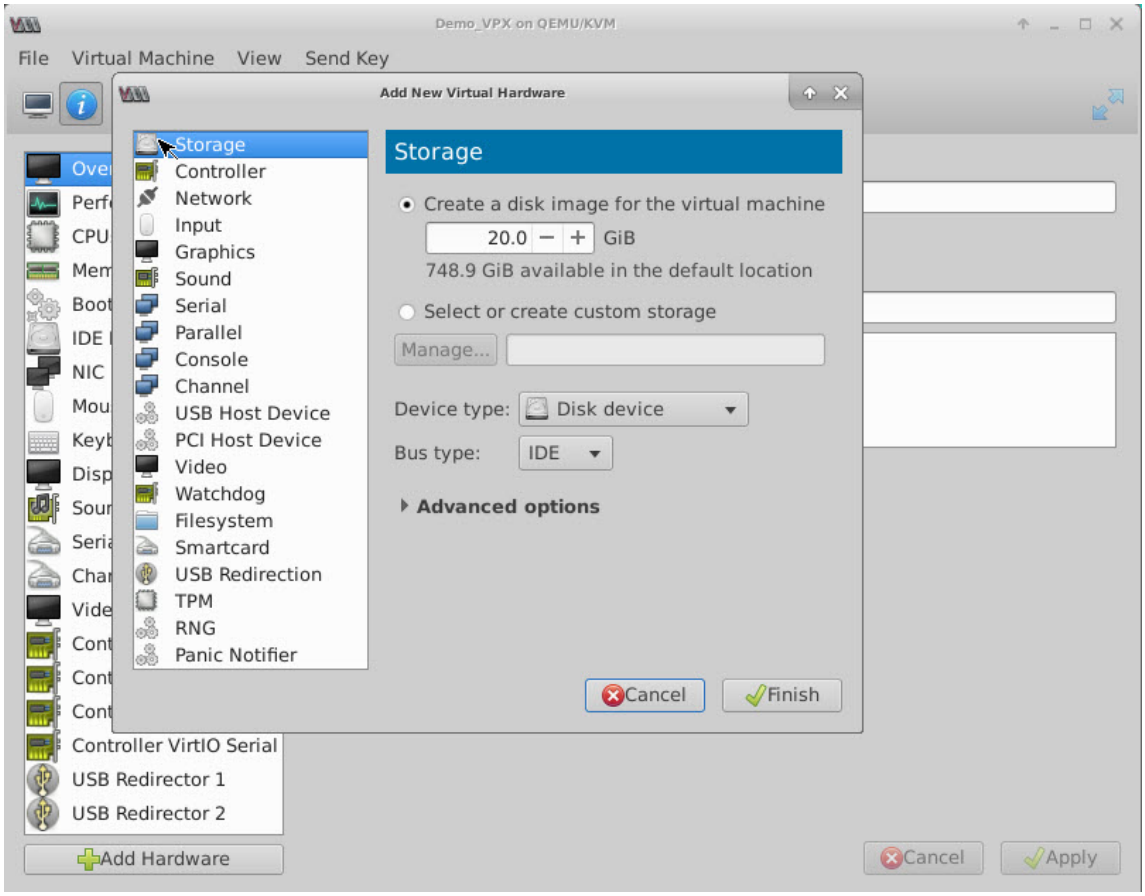
1. NetScaler VPX インスタンスの電源を切ります。
2. NetScaler VPX インスタンスを選択して、［Open］をクリックします。



3. <virtual machine on KVM> ウィンドウで、**i** アイコンを選択します。



4. [ハードウェアの追加] を選択します。



5. [新しい仮想ハードウェアの追加] ダイアログボックスで、次の操作を行います。

   a) [PCI ホストデバイス] を選択します。

   b) [Host Device] セクションで、作成した VF を選択して、[Finish] をクリックします。

図 **4**: インテル 82599 10G NIC の VF

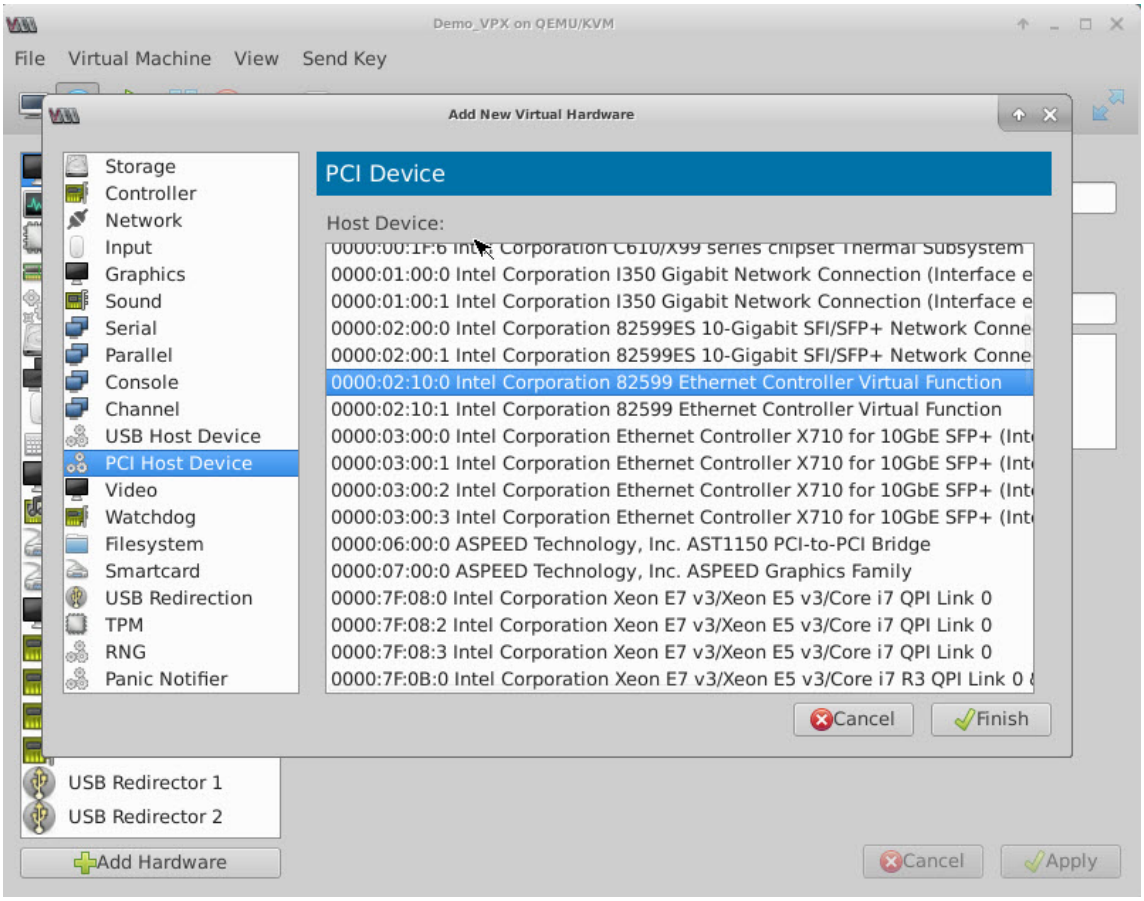


図 **5**: インテル XL710 40G NIC の VF

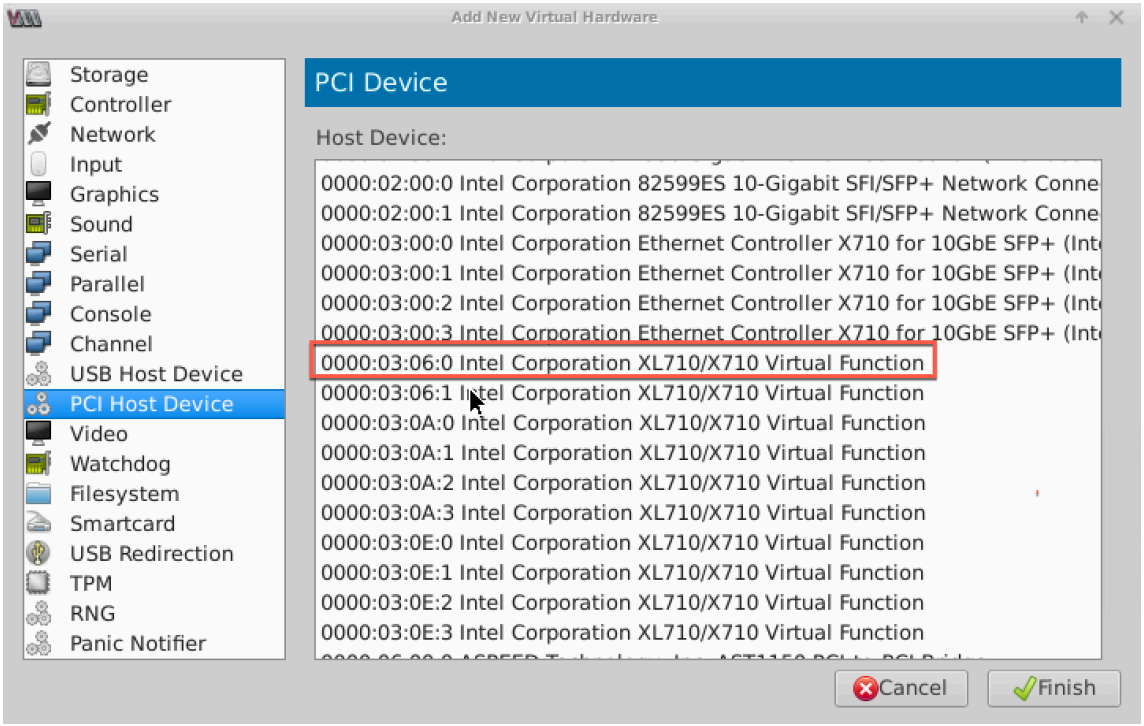
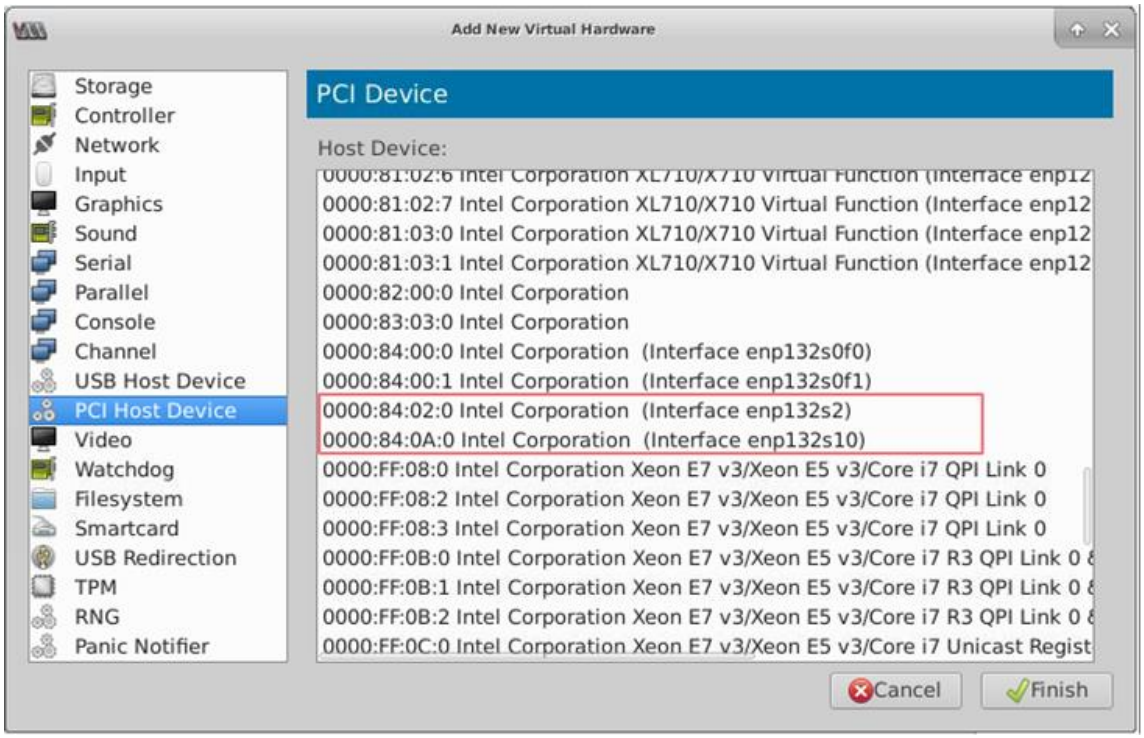


図 **6**: インテル X722 10G NIC の VF



6. 手順 4 と 5 を繰り返し、作成した VF を追加します。
7. NetScaler VPX インスタンスをパワーオンします。
8. NetScaler VPX インスタンスがパワーオンしたら、次のコマンドを使用して構成を確認します。

```
1    show interface summary
```

構成したすべてのインターフェイスが出力に表示されます。

図 **6**: インテル 82599 NIC の出力サマリー

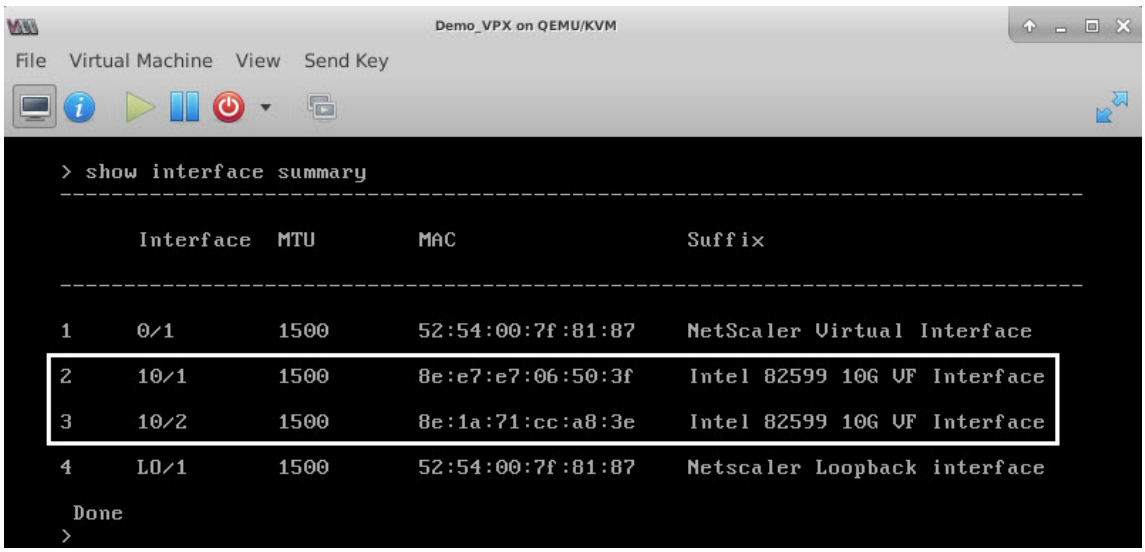
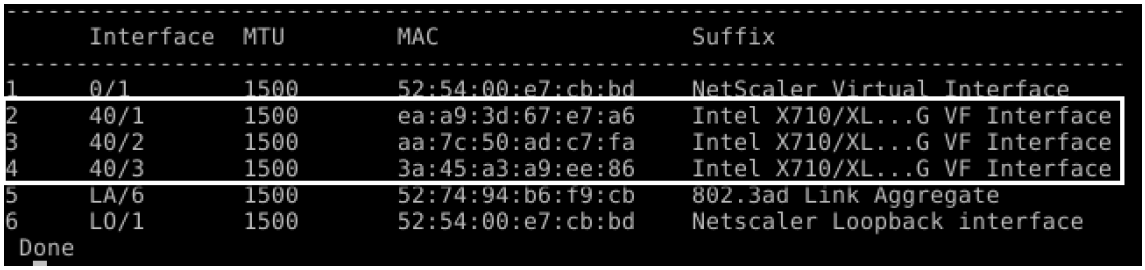


図 **7**. インテル X710 および XL710 NIC の出力サマリー。



SR-IOV インターフェイスでスタティック **LA/LACP** を設定する

> 重要**:**
>
> SR-IOV VF を作成する場合、MAC アドレスを VF に割り当てていないことを確認してください。

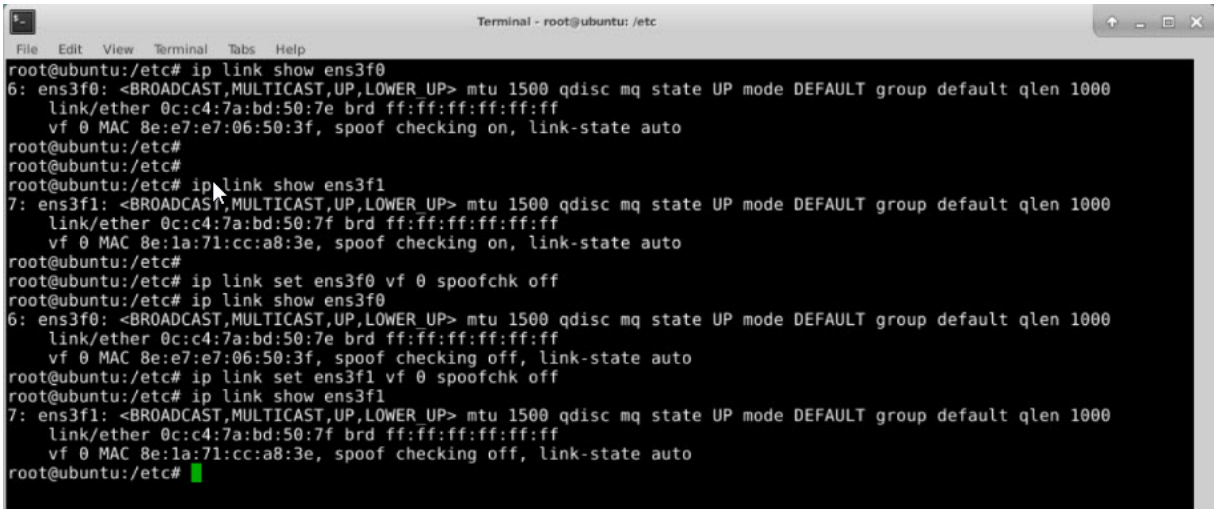
リンクアグリゲーションモードで、SR-IOV VF を使用するには、作成した VF のなりすましチェックを無効にします。KVM ホストでなりすましチェックを無効にするには、以下のコマンドを使用します。

*ip link set \\<**interface**_name\\> vf \\<VF_id \\> spoofchk off*

各項目の意味は次のとおりです:

- Interface_name –インターフェイス名です。
- VF_id –Virtual Function ID です。

例:



作成したすべての VF のなりすましチェックを無効にします。NetScaler VPX インスタンスを再起動し、リンクアグリゲーションを構成します。詳細な手順については、リンク集約の設定を参照してください。

SR-IOV インターフェイスで **VLAN** を構成する

SR-IOV VF で VLAN を構成できます。詳細な手順については 、VLAN の設定を参照してください。

> 重要**:**
>
> KVM ホストに VF インターフェイスの VLAN 設定が含まれていないことを確認してください。

SR-IOV モードでの **SSL** アクセラレーションに **Intel QAT** を使用するように **KVM** ハイパーバイザー上の **NetScaler VPX** を構成します

October 17, 2024

Linux KVM ハイパーバイザー上の NetScaler VPX インスタンスは、Intel クイックアシストテクノロジー（QAT）を使用して NetScaler SSL のパフォーマンスを高速化できます。インテル QAT を使用すると、レイテンシーの高い暗号処理をすべてチップにオフロードできるため、1 つまたは複数のホスト CPU を解放して他のタスクを実行できるようになります。

以前は、NetScaler データパスの暗号化処理はすべて、ホスト vCPU を使用するソフトウェアで行われていました。

> 注
>
> 現在、NetScaler VPX はインテル QAT ファミリーの C62x チップモデルのみをサポートしています。この機能は、NetScaler リリース 14.1 ビルド 8.50 以降でサポートされています。

前提条件

- Linux ホストには、マザーボードに直接統合されているか、外部 PCI カードに追加された Intel QAT C62x チップが搭載されています。

  Intel QAT C62x シリーズ モデル**:** C625、C626、C627、C628。これらの C62x モデルのみに公開キー暗号化 (PKE) 機能が含まれています。その他の C62x バリアントは PKE をサポートしていません。
- NetScaler VPX は、VMware ESX のハードウェア要件を満たしています。詳細については、「Linux KVM プラットフォームに NetScaler VPX インスタンスをインストールする」を参照してください。

制限事項

個々の VM 用に暗号ユニットや帯域幅を予約する規定はありません。Intel QAT ハードウェアで使用可能なすべての暗号ユニットは、QAT ハードウェアを使用するすべての VM で共有されます。

インテル **QAT** を使用するためのホスト環境のセットアップ

1. Linux ホストに C62x シリーズ (QAT) チップモデル用の Intel 提供のドライバーをダウンロードしてインストールします。Intel パッケージのダウンロードとインストール手順の詳細については、Linux 用 Intel QuickAssist テクノロジー・ドライバーを参照してください。readme ファイルはダウンロードパッケージに含まれています。ダウンロード パッケージの一部として readme ファイルが提供されます。このファイルには、パッケージをコンパイルしてホストにインストールする手順が記載されています。

   ドライバをダウンロードしてインストールしたら、次の健全性チェックを行います:

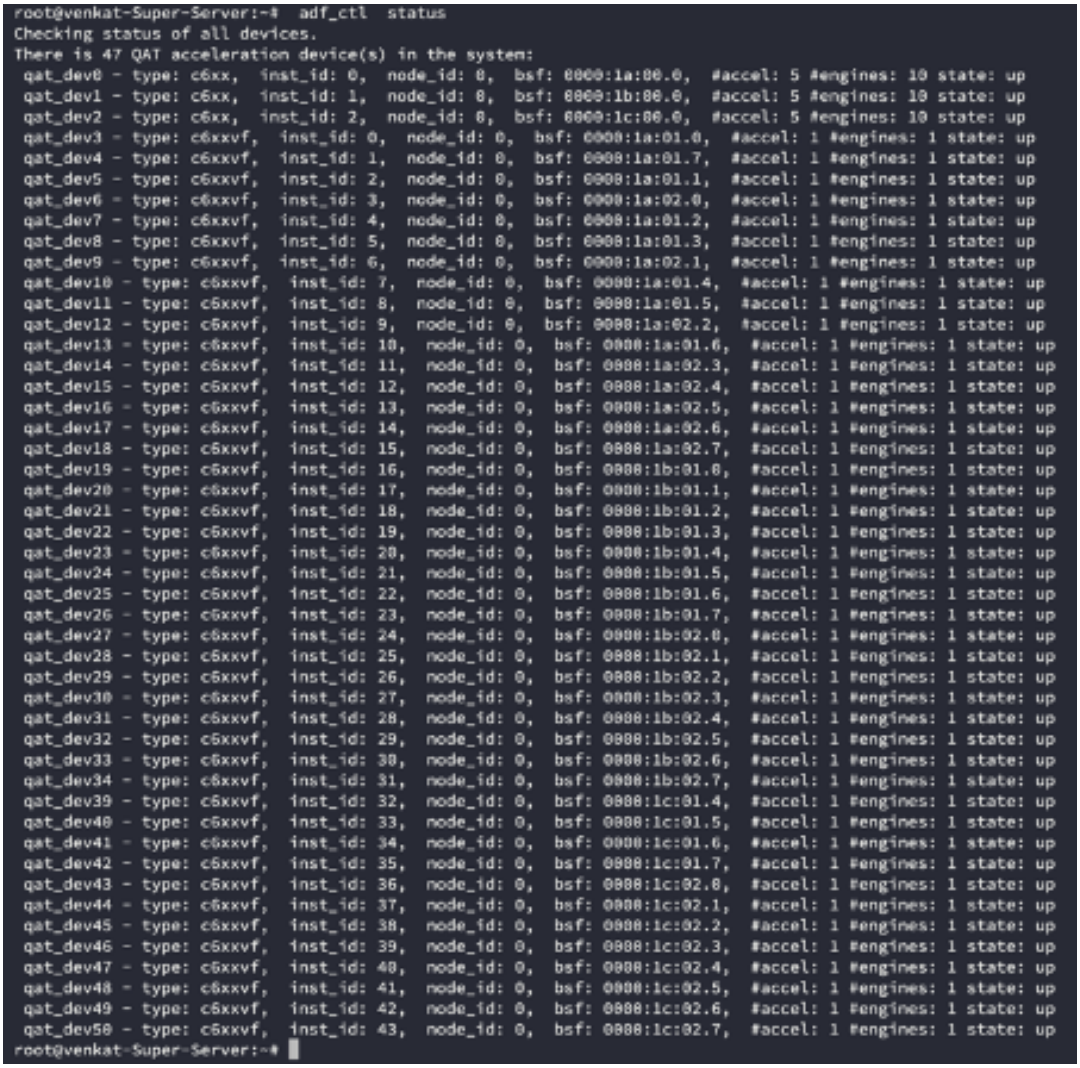
   - C62x チップの数に注意してください。各 C62x チップには、最大 3 つの PCIe エンドポイントがあります。
   - すべてのエンドポイントが稼働していることを確認します。`adf_ctl status` コマンドを実行して、すべての PF エンドポイント (最大 3 つ) のステータスを表示します。

     ```
     root@Super-Server:~# adf_ctl status

     Checking status of all devices.
     There is 51 QAT acceleration device(s) in the system
     qat_dev0 - type: c6xx,  inst_id: 0,  node_id: 0,  bsf:
        0000:1a:00.0,  #accel: 5 #engines: 10 state: up
     qat_dev1 - type: c6xx,  inst_id: 1,  node_id: 0,  bsf:
        0000:1b:00.0,  #accel: 5 #engines: 10 state: up
     qat_dev2 - type: c6xx,  inst_id: 2,  node_id: 0,  bsf:
        0000:1c:00.0,  #accel: 5 #engines: 10 state: up
     ```

   - すべての QAT エンドポイントで SRIOV (VF サポート) を有効にします。

     ```
     root@Super-Server:~# echo 1 > /sys/bus/pci/devices/0000\:1a
        \:00.0/sriov_numvfs
     root@Super-Server:~# echo 1 > /sys/bus/pci/devices/0000\:1b
        \:00.0/sriov_numvfs
     root@Super-Server:~# echo 1 > /sys/bus/pci/devices/0000\:1c
        \:00.0/sriov_numvfs
     ```

   - すべての VF が表示されていることを確認します（エンドポイントあたり 16 VF、合計 48 VF)。
   - adf_ctl status コマンドを実行して、各 Intel QAT チップのすべての PF エンドポイント (最大 3 つ) と VF が稼働していることを確認します。この例では、システムには C62x チップが 1 つしかありません。つまり、合計で 51 のエンドポイント（3+48 の VF）があります。



2. Linux ホストで SR-IOV を有効にしてください。
3. 仮想マシンを作成します。仮想マシンを作成するときは、パフォーマンス要件を満たす適切な数の PCI デバイスを割り当てます。

> 注
>
> 各 C62x (QAT) チップには、最大 3 つの個別の PCI エンドポイントを設定できます。各エンドポイントは VF の論理的な集合であり、チップの他の PCI エンドポイントと帯域幅を均等に共有します。各エンドポイントには、最大 16 個の PCI デバイスとして表示される VF を最大 16 個設定できます。これらのデバイスを VM に追加して、QAT チップを使用して暗号アクセラレーションを行います。

注意事項

- 仮想マシンの暗号化要件が複数の QAT PCI エンドポイント/チップを使用することである場合は、対応する PCI デバイス/VF をラウンドロビン方式で選択して対称分散を行うことをお勧めします。

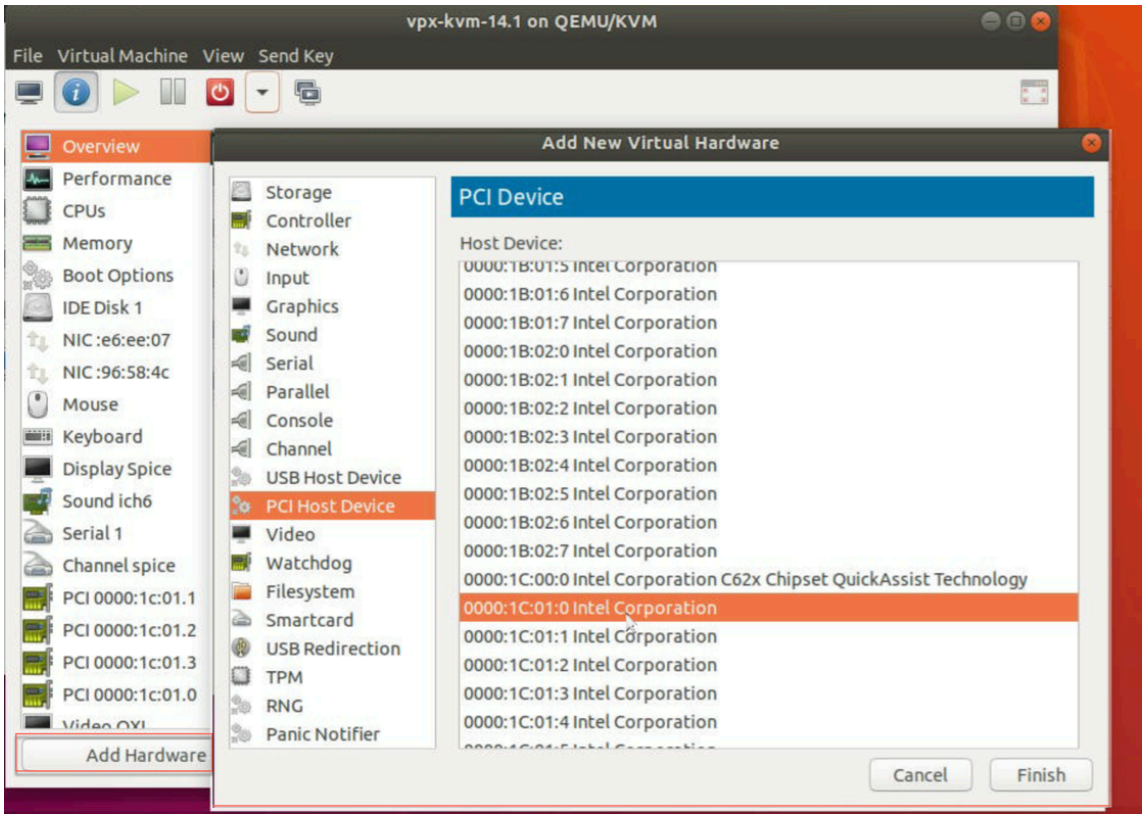
- 選択する PCI デバイスの数は、ライセンスされている vCPU の数と同じにすることをお勧めします (管理 vCPU 数は含まない)。利用可能な vCPU 数よりも多くの PCI デバイスを追加しても、必ずしもパフォーマンスが向上するわけではありません。

  例

  3 つのエンドポイントを備えた 1 つの Intel C62x チップを搭載した Linux ホストを考えてみましょう。6 個の vCPU を搭載した VM をプロビジョニングする場合、各エンドポイントから 2 つの VF を選択し、それらを VM に割り当てます。この割り当てにより、仮想マシンの暗号ユニットを効果的かつ均等に分配できます。使用可能な vCPU の合計のうち、デフォルトで 1 つの vCPU が管理プレーン用に予約され、残りの vCPU はデータプレーン PE で使用できます。

Linux KVM ハイパーバイザーにデプロイされた **NetScaler VPX** に **QAT VF** を割り当てる

1. Linux KVM Virtual Machine Manager で、仮想マシン (NetScaler VPX) の電源がオフになっていることを確認します。
2. [ハードウェアの追加**] > [PCI** ホストデバイス] に移動します。
3. Intel QAT VF を PCI デバイスに割り当てます。



4. [完了] をクリックします。

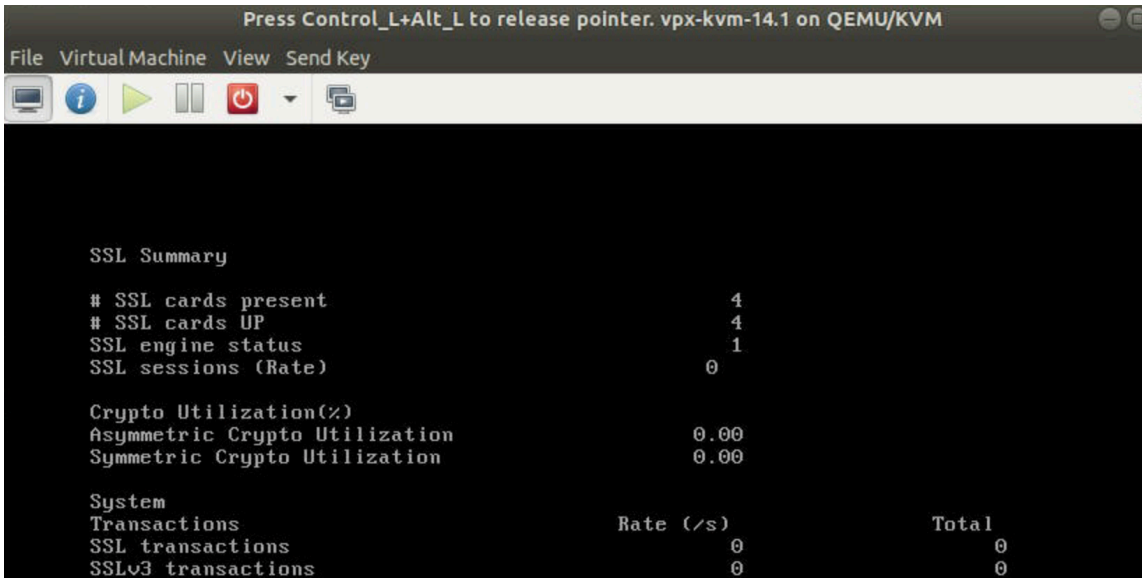
5. 前述の手順を繰り返して、1 つ以上の Intel QAT VF を NetScaler VPX インスタンスに割り当てます。なぜなら、1 つの vCPU が管理プロセス用に予約されているからです。

   仮想マシンあたりの QAT 仮想マシンの数 = 仮想 CPU の数-1

6. Power on the VM.

7. NetScaler CLI で`stat ssl`コマンドを実行して SSL サマリーを表示し、QAT VF を NetScaler VPX に割り当てた後に SSL カードを確認します。

   この例では、5 つの vCPU を使用しました。つまり、4 つのパケットエンジン (PE) です。



展開について

このデプロイメントは、次のコンポーネント仕様でテストされました:

- **NetScaler VPX** バージョンとビルド:14.1—8.50
- **Ubuntu** バージョン**:** 18.04、カーネル 5.4.0-146
- Linux 用インテル **C62x QAT** ドライバーのバージョン:L.4.21.0-00001

PCI パススルーネットワークインターフェイスを使用するように **NetScaler VPX** インスタンスを構成する

October 17, 2024

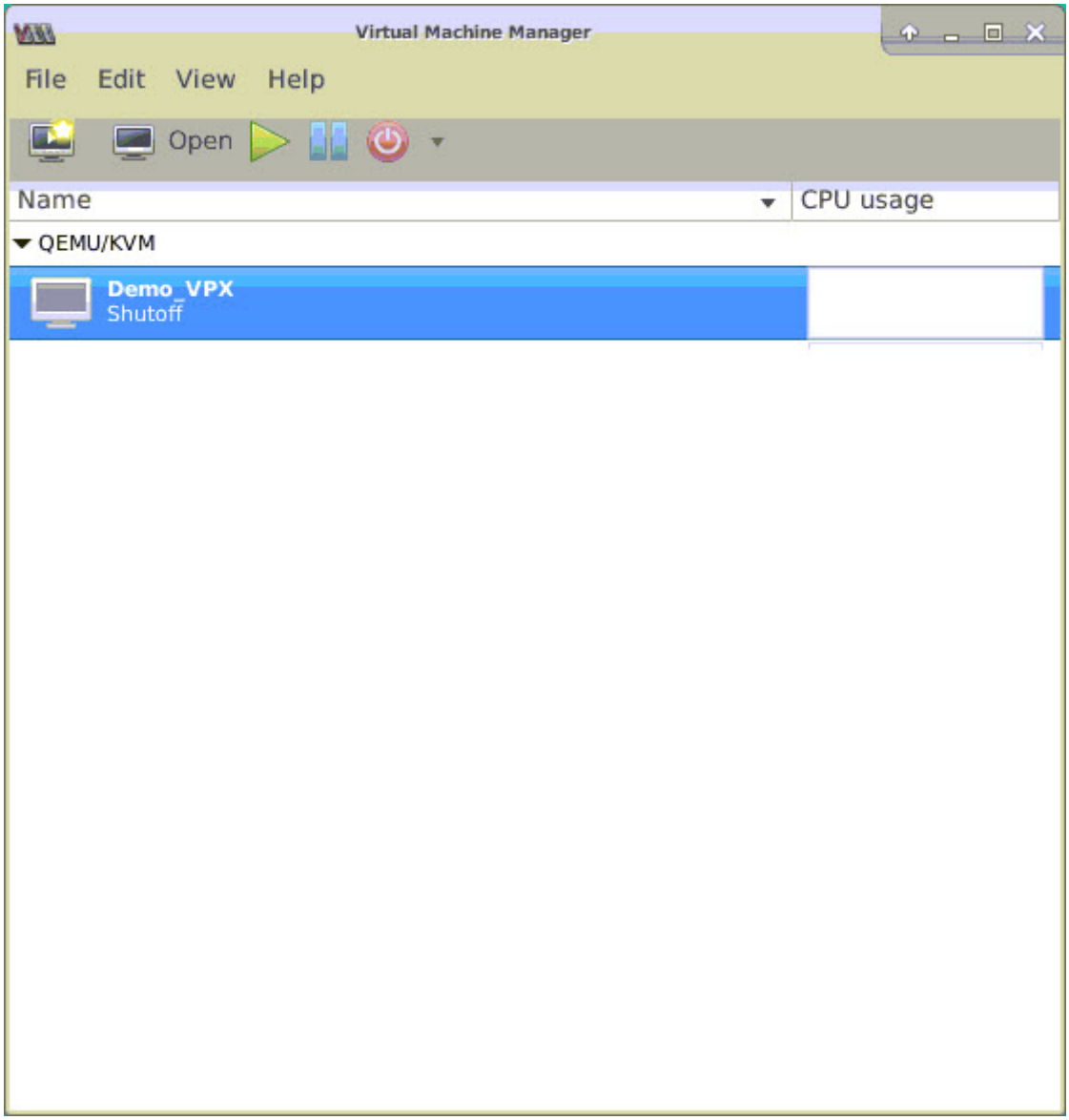
Linux-KVM プラットフォームに NetScaler VPX インスタンスをインストールして構成したら、仮想マシンマネージャーを使用して、PCI パススルーネットワークインターフェイスを使用するように仮想アプライアンスを構成できます。

前提条件

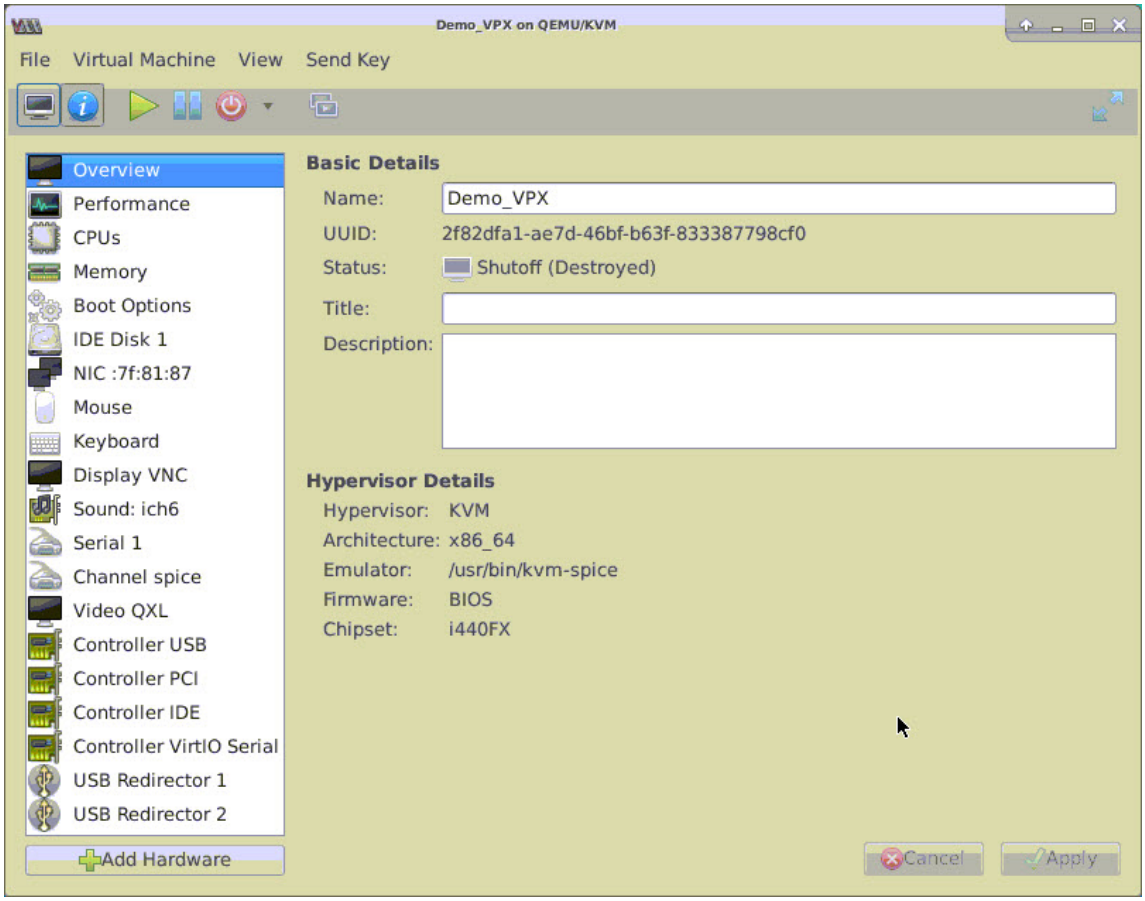
- KVM ホスト上のインテル XL710 NIC (NIC) のファームウェア・バージョンは 5.04 です。
- KVM ホストは、IOMMU（Input–Output Memory Management Unit）と Intel VT をサポートし、これらは KVM ホストの BIOS で有効になっています。KVM ホストで **IOMMU** を有効にするには、**/boot/-grub2/grub.cfg** ファイルに次のエントリを追加します。
- 次のコマンドを実行して KVM ホストを再起動します。**grub2-MKConfig –o /boot/grub2/grub.cfg**

仮想マシンマネージャーを使用して **PCI** パススルーネットワークインターフェイスを使用するように **NetScaler ADC VPX** インスタンスを構成するには：

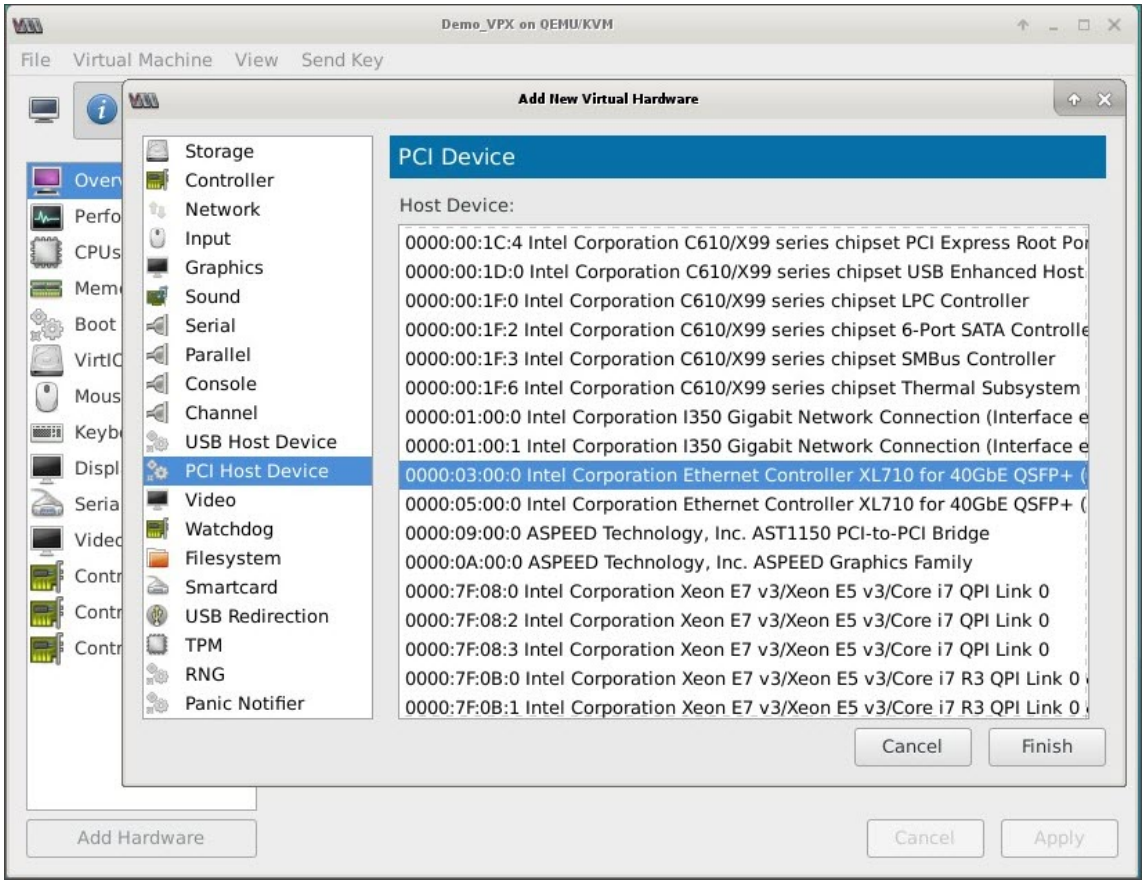
1. NetScaler VPX インスタンスの電源を切ります。
2. NetScaler VPX インスタンスを選択し、［開く］をクリックします。



3. **KVM>** の **virtual_machine** ウィンドウで、**i** アイコンをクリックします。



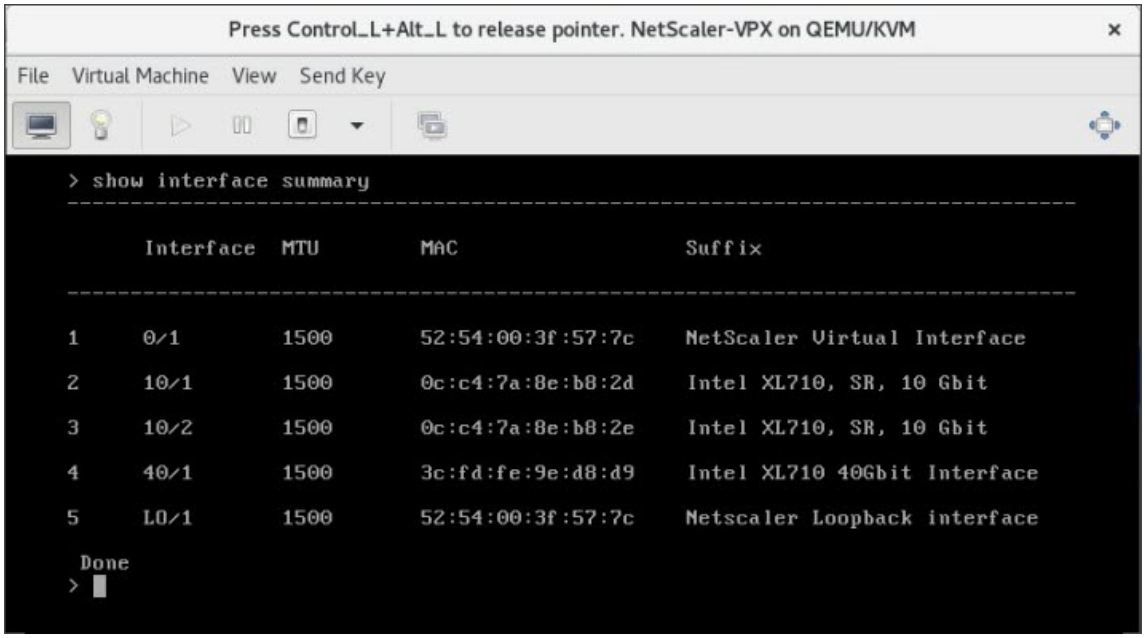
4. [ハードウェアの追加**]** をクリックします。
5. [新しい仮想ハードウェアの追加] ダイアログボックスで、次の操作を行います。
   a. [**PCI** ホストデバイス] を選択します。
   b. [ホストデバイス] セクションで、インテル XL710 物理機能を選択します。
   c. ［完了］をクリックします。



6. 手順 **4** と **5** を繰り返して、インテル XL710 物理関数を追加します。
7. NetScaler VPX インスタンスをパワーオンします。
8. NetScaler VPX インスタンスの電源が入ったら、次のコマンドを使用して構成を確認できます。

```
COMMAND

> show interface summary
```

出力には、設定したすべてのインターフェイスが表示されている必要があります。



virsh プログラムを使用して **NetScaler ADC VPX** インスタンスをプロビジョニングする

October 17, 2024

virshプログラムは VM ゲストを管理するためのコマンドラインツールです。その機能性は Virtual Machine Manager に似ています。これにより VM Guest の状態（開始、停止、一時停止など）を変更でき、新しい Guests およびデバイスをセットアップして、既存の構成を編集できます。virshプログラムは、VM ゲスト管理操作のスクリプト作成にも役立ちます。

virshプログラムを使用して NetScaler ADC VPX をプロビジョニングするには、次の手順に従います。

1. tar コマンドを使用して、NetScaler VPX パッケージを解凍します。nsvpx-kvm-*_nc.tgz パッケージには、次のコンポーネントが含まれています。

   - VPX 属性 [NSVPX-KVM-*_nc.xml] を指定するドメイン XML ファイル
   - NS-VM ディスクイメージ [Checksum.txt] のチェックサム
   - NS-VM Disk Image [NSVPX-KVM-*_nc.raw]

   例

```
tar -xvzf NSVPX-KVM-10.1-117_nc.tgz
NSVPX-KVM-10.1-117_nc.xml
NSVPX-KVM-10.1-117_nc.raw
checksum.txt
```


2. NSVPX-KVM-*_nc.xml XML ファイルを \\<DomainName\\>-NSVPX-KVM-*_nc.xmlという名前のファイルにコピーします。<DomainName> は仮想マシンの名前でもあります。例

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

3. \\<DomainName\\>-NSVPX-KVM-*_nc.xml ファイルを編集して、次のパラメータを指定します。

   - name - 名前を指定します。
   - Mac —MAC アドレスを指定します。

     > 注
     >
     > ドメイン名と MAC アドレスは一意である必要があります。

   - source file: ディスクイメージの絶対ソースパスを指定します。ファイルパスは絶対パスである必要があります。RAW イメージファイルまたは QCOW2 イメージファイルのパスを指定することができます。

     RAW イメージファイルを指定する場合は、次の例のようにディスクイメージソースパスを指定します。

     例

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

     次の例に示すように、絶対 QCOW2 ディスクイメージソースパスを指定し、ドライバタイプを **qcow2** と定義します。

     例

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

4. \\<DomainName\\>-NSVPX-KVM-*_nc.xml ファイルを編集して、ネットワークの詳細を構成します。

   - source dev - インターフェースを指定します。
   - mode - モードを指定します。デフォルトのインターフェイスは **Macvtap** ブリッジです。

   例: モード:macvTap Bridge ターゲットインターフェイスをethxに設定し、モードをブリッジモデルタイプvirtioに設定

```
<interface type='direct'>
    <mac address='52:54:00:29:74:b3'/>
```



```
        <source dev='eth0' mode='bridge'/>
        <target dev='macvtap0'/>
        <model type='virtio'/>
        <alias name='net0'/>
        <address type='pci' domain='0x0000' bus='0x00' slot='0x03'
            function='0x0'/>
      </interface>
```

ここで、eth0 は仮想マシンに接続された物理インターフェイスです。

5. 次のコマンドを使用して、\\<DomainName\\>-NSVPX-KVM-*_nc.xml ファイル内の VM 属性を定義します。

```
virsh define \<DomainName\>-NSVPX-KVM-\*\_nc.xml
```

例

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

6. 次のコマンドを入力して VM を起動します。

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

例

```
virsh start NetScaler-VPX
```

7. コンソール経由でゲスト VM を接続します。

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

例

```
virsh console NetScaler-VPX
```

virsh プログラムを使用して **NetScaler ADC VPX** インスタンスにインターフェイスを追加する

KVM 上で NetScaler VPX をプロビジョニングした後、追加のインターフェイスを付加できます。

インターフェイスを追加するには、次の手順を実行します。

1. KVM の上で動作している NetScaler VPX インスタンスをシャットダウンします。
2. 次のコマンドを使用して、\\<DomainName\\>-NSVPX-KVM-*_nc.xml ファイルを編集します。

```
virsh edit \[\<DomainName\> | \<DomainUUID\>\]
```

3. \\<DomainName\\>-NSVPX-KVM-*_nc.xml ファイルに、次のパラメータを追加します。

a) **MacVTap** 用

- Interface type - インターフェイスの種類として「direct」を指定します。
- MAC アドレス：MAC アドレスを指定し、MAC アドレスがインターフェイス全体で一意であることを確認します。
- source dev - インターフェイス名を指定します。
- mode-モードを指定します。サポートされているモードは、ブリッジ、VEPA、プライベート、パススルーです。
- モデルタイプ-モデルタイプを次のように指定します。`virtio`

例

モード：MacVTap Pass-through

ターゲットインターフェースを次のように設定します `ethx`、モードとして 橋梁、モデルタイプとして ヴィルティオ

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

ここで eth1 は仮想マシンに接続された物理インターフェイスです。

b) ブリッジモード用

> 注
>
> KVM ホストに Linux ブリッジを設定し、物理インターフェイスをブリッジにバインドし、ブリッジを UP 状態にしていることを確認します。

- Interface type - インターフェイスの種類として「bridge」を指定します。
- MAC アドレス：MAC アドレスを指定し、MAC アドレスがインターフェイス全体で一意であることを確認します。
- source dev - ブリッジ名を指定します。
- モデルタイプ-モデルタイプを次のように指定します。`virtio`

例：Bridge Mode

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


NetScaler VPX ゲスト仮想マシンの管理

October 17, 2024

仮想マシンマネージャと `virsh`プログラムを使用して、仮想マシンゲストの起動または停止、新しいゲストとデバイスの設定、既存構成の編集、仮想ネットワークコンピューティング (VNC) によるグラフィカルコンソールへの接続などの管理タスクを実行できます。

仮想マシンマネージャーを使用して VPX ゲスト仮想マシンを管理する

- VM ゲストを一覧表示する

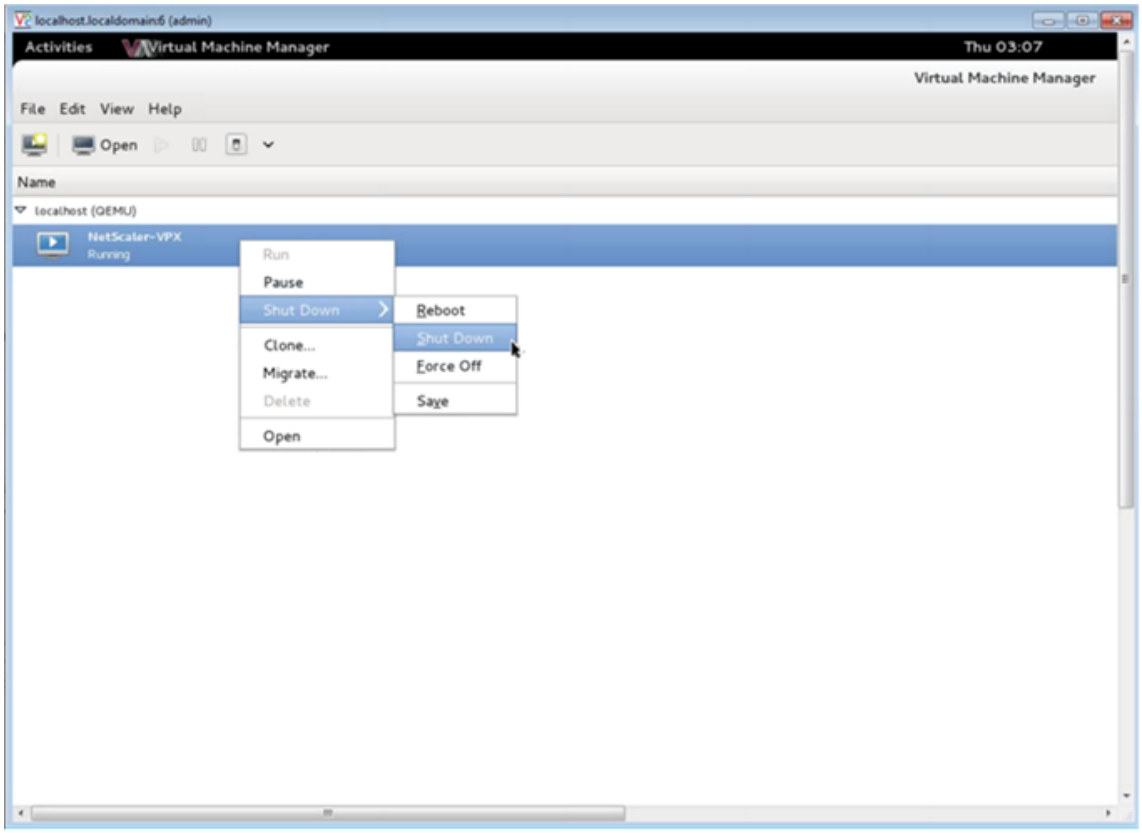
  Virtual Machine Manager のメインウィンドウには、接続される各 VM ホストサーバのすべての VM Guests の一覧が表示されます。各仮想マシンゲストエントリには、仮想マシンの名前と、アイコンに表示されるステータス（実行中、一時停止、またはシャットオフ）が含まれます。

- グラフィカルコンソールを開く

  VM Guest に対してグラフィカルコンソールを開いて、VNC 接続介して物理的ホストと通信するようにマシンと相互通信できます。Virtual Machine Manager でグラフィカルコンソールを開くには、VM Guest エントリーを右クリックして、ポップアップメニューで［オープン］オプションを選択します。

- ゲストの起動とシャットダウン

  Virtual Machine Manager から VM Guest を開始または停止できます。VM の状態を変更するには、VM Guest エントリーを右クリックして、ポップアップメニューで［Run］または［Shut Down］オプションのいずれかを選択します。



- ゲストを再起動

  Virtual Machine Manager から VM Guest を再起動できます。VM を再起動するには、VM Guest エントリーを右クリックして、ポップアップメニューで［Shut Down］>［Reboot］を選択します。

- ゲストを削除する

  デフォルトでは、VM Guest を削除すると XML 構成が消去されます。また、ゲストのストレージファイルを削除できます。これを実行して、完全にそうすることはゲストを消します。

  1. Virtual Machine Manager で、VM Guest エントリーを右クリックします。
  2. ポップアップメニューで［Delete from］を選択します。確認ウィンドウが開きます。

     > 注
     >
     > 削除オプションは、VM ゲストがシャットダウンされている場合にのみ有効になります。

  3. ［削除］をクリックします。
  4. 完全にゲストを消去するには、［Delete Associated Storage Files］チェックボックスをオンにして、関連付けられた.raw ファイルを削除します。

`virsh` プログラムを使用して **NetScaler ADC VPX** ゲスト仮想マシンを管理する

- VM ゲストとその現在の状態を一覧表示します。

  ゲストに関する情報を表示するためにvirshを使用するには

  ```
  virsh list --all
  ```

  コマンド出力はすべてのドメインとその状態を表示します。出力例:

  ```
  Id Name                 State
  ----------------------------------
    0 Domain-0            running
    1 Domain-1            paused
    2 Domain-2            inactive
    3 Domain-3            crashed
  ```

- virshコンソールを開きます。

  ゲスト仮想マシンをコンソールから接続します。

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

  例

  ```
  virsh console NetScaler-VPX
  ```

- ゲストを起動してシャットダウンします。

  Guest は DomainName または Domain-UUID を使って開始できます。

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

  例

  ```
  virsh start NetScaler-VPX
  ```

  ゲストをシャットダウンするには:

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

  例

  ```
  virsh shutdown NetScaler-VPX
  ```

- ゲストを再起動

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

  例

  ```
  virsh reboot NetScaler-VPX
  ```

  ゲストを削除する

  ゲスト仮想マシンを削除するには、削除コマンドを実行する前に、ゲストをシャットダウンして-NSVPX-KVM <DomainName>- * _nc.xml を定義解除する必要があります。


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

例

```
virsh shutdown NetScaler-VPX
virsh undefine NetScaler-VPX
```

> 注
>
> 削除コマンドではディスク イメージ ファイルは削除されないため、手動で削除する必要があります。

OpenStack 上で SR-IOV を使用して NetScaler VPX インスタンスをプロビジョニングします

October 17, 2024

OpenStack で、シングルルート I/O 仮想化（Single-Root I/O Virtualization：SR-IOV）テクノロジを使用する高パフォーマンスの NetScaler VPX インスタンスを展開できます。

OpenStack で、3 つの手順で、SR-IOV テクノロジを使用する NetScaler VPX インスタンスを展開できます。

- ホスト上で SR-IOV Virtual Functions（VF）を有効にします。
- VF を構成し、OpenStack で使用できるようにします。
- OpenStack で NetScaler VPX をプロビジョニングします。

前提条件

次のことを確認してください:

- インテル 82599 NIC (NIC) をホストに追加します。
- 最新の IXGBE ドライバーをダウンロードしてインストールします。
- ホスト上の IXGBEVF ドライバをブロックリストします。/etc/modprobe.d/blacklist.conf ファイルに次のエントリを追加します。ブロックリスト `ixgbevf`

> 注
>
> `ixgbe`ドライバーのバージョンは 5.0.4 以上でなければなりません。

ホストで SR-IOV VF を有効にする

SR-IOV VF を有効にするには、次のいずれかの手順を実行します。

- <number_of_VFs>3.8 より前のカーネルバージョンを使用している場合は、/etc/modprobe.d/ixgbe ファイルに次のエントリを追加し、ホストを再起動します。オプション ixgbe max_vfs=
- カーネル 3.8 以降のバージョンを使用している場合、以下のコマンドを使用して VF を作成します。

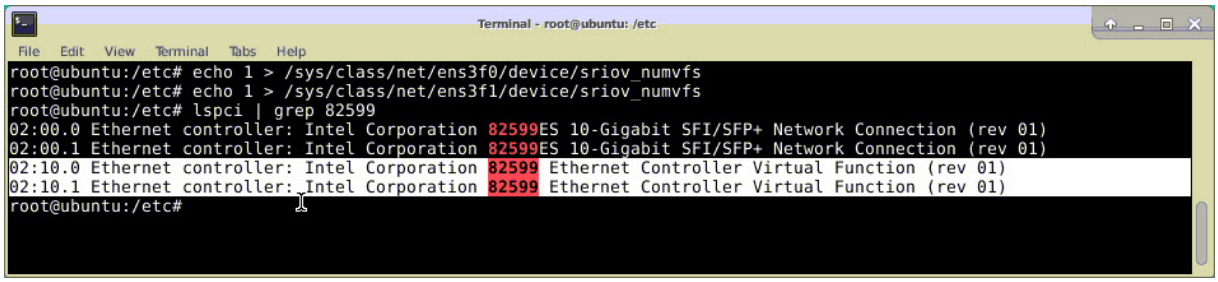
```
1       echo <number_of_VFs> > /sys/class/net/<device_name>/device/
           sriov_numvfs
```

各項目の意味は次のとおりです:

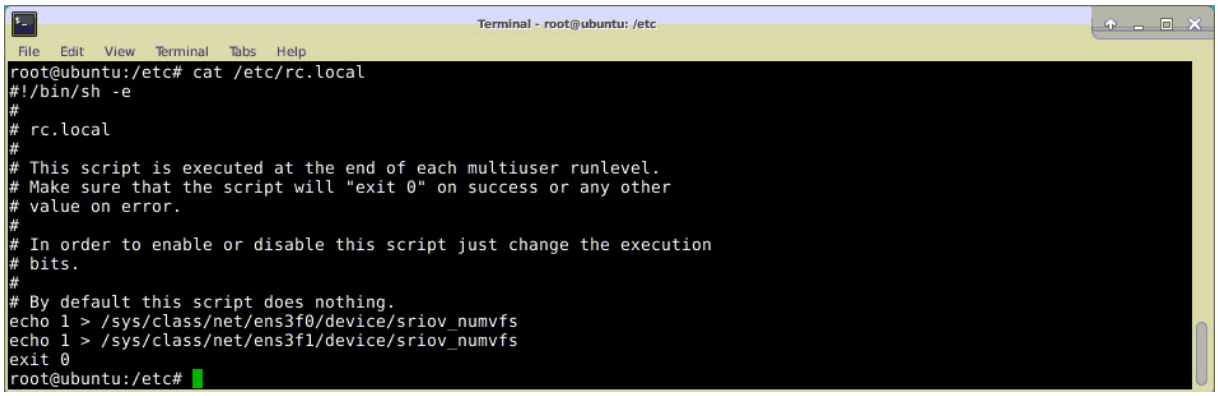
- number_of_VFs は、作成する Virtual Function の数です。
- device_name はインターフェイス名です。

> 重要:
>
> SR-IOV VF を作成する場合、MAC アドレスを VF に割り当てないようにしてください。

次に、作成している 4 つの VF の例を示します。



VF を永続的にし、VF の作成に使用したコマンドを **rc.local** ファイルに追加します。rc.local ファイルの内容を示す例を次に示します。



詳細については、このインテル SR-IOV 構成ガイドを参照してください。

OpenStack で VF を設定して利用できるようにする

以下のリンクに記載されている手順に従って、OpenStack で SR-IOV を設定します: https://wiki.openstack.org/wiki/SR-IOV-Passthrough-For-Networking。

OpenStack で Citrix ADC VPX インスタンスをプロビジョニングする

OpenStack CLI を使用して、OpenStack 環境で Citrix ADC VPX インスタンスをプロビジョニングできます。

VPX インスタンスをプロビジョニングします。オプションで、コンフィグドライブのデータを使用します。「構成ドライブ」とは、インスタンスの起動時にアタッチされる特殊な構成ドライブを指します。この構成ドライブを使用して、インスタンスのネットワーク設定を構成する前に、管理 IP アドレス、ネットワークマスク、デフォルトゲートウェイなどのネットワーク構成情報をインスタンスに渡すことができます。

OpenStack が VPX インスタンスをプロビジョニングする場合、まず OpenStack を示す特定の BIOS 文字列（OpenStack ファウンデーション）を読み取ることによって、インスタンスが OpenStack 環境で起動していることを検出します。Red Hat Linux ディストリビューションの場合、この文字列は/etc/nova/release に保存されます。これは、KVM ハイパーバイザープラットフォームに基づくすべての OpenStack 実装で使用できる標準メカニズムです。ドライブには特定の OpenStack ラベルが必要です。構成ドライブが検出されると、インスタンスは nova boot コマンドで指定されたファイル名から次の情報を読み取ろうとします。以下の手順では、このファイルを「userdata.txt」と呼びます。

- 管理 IP アドレス
- ネットワークマスク
- デフォルトゲートウェイ

パラメーターが正しく読み取られると、それらの値が NetScaler スタックに適用されます。これにより、インスタンスをリモートから管理できるようになります。パラメーターが読み取られない場合、または構成ドライブが存在しない場合は、インスタンスが以下のデフォルトの処理を実行します。

- DHCP から IP アドレス情報を取得する。
- DHCP から情報を取得できない場合は、デフォルトのネットワーク構成として 192.168.100.1/16 を使用する。

CLI を使用して OpenStack 上の NetScaler VPX インスタンスをプロビジョニングします

OpenStack 環境で VPX インスタンスをプロビジョニングするには、OpenStack の CLI を使用します。次に、OpenStack で Citrix ADC VPX インスタンスをプロビジョニングする手順の概要を示します。

1. .tgz ファイルから.qcow2ファイルを抽出する
2. qcow2 イメージから OpenStack イメージを作成する
3. VPX インスタンスのプロビジョニング

OpenStack 環境で VPX インスタンスをプロビジョニングするには、次の手順を実行します。

1. 抽出します。次のコマンドを入力して、.tqzファイルからqcow2ファイルを抽出します。

```
tar xvzf <TAR file>
tar xvzf NSVPX-KVM-12.0-26.2_nc.tgz
```



```
3   NSVPX-KVM.xml
4   NSVPX-KVM-12.0-26.2_nc.qcow2
```

2. 次のコマンドを入力して、手順 1 で抽出した.qcoz2ファイルを使用して OpenStack イメージをビルドします。

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

下図は、glance image-create コマンドの出力例です。

```
+------------------+--------------------------------------+
| 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. OpenStack イメージが作成されたら、NetScaler VPX インスタンスをプロビジョニングします。

```
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
```

前述のコマンドでは、userdata.txt は、VPX インスタンスの IP アドレス、ネットマスク、デフォルトゲ

ートウェイなどの詳細を含むファイルです。ユーザーデータファイルは、ユーザーカスタマイズ可能なファイルです。NSVPX-KVM-12.0-26.2 は、プロビジョニングする仮想アプライアンスの名前です。—NIC port-id=218ba819-9f55-4991-adb6-02086a6bdee2 は OpenStack VF です。

次の図に、nova boot コマンドの出力例を示します。

```
+--------------------------------------+--------------------------------------------------------------------+
| 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                                   |
+--------------------------------------+--------------------------------------------------------------------+
```

次の図は、userdata.txt ファイルのサンプルです。タグ内の値は、ユーザーが設定可能な値で、IP アドレス、ネットマスク、デフォルトゲートウェイなどの情報を保持します。

```
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  <Property oe:key="com.citrix.netscaler.platform" oe:value="vpx"
        />
15  citrix.com 4
16  <Property oe:key="com.citrix.netscaler.orch_env"
```



```
17    oe:value="openstack-orch-env"/>
18    <Property oe:key="com.citrix.netscaler.mgmt.ip"
19    oe:value="10.1.0.100"/>
20    <Property oe:key="com.citrix.netscaler.mgmt.netmask"
21    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>
```

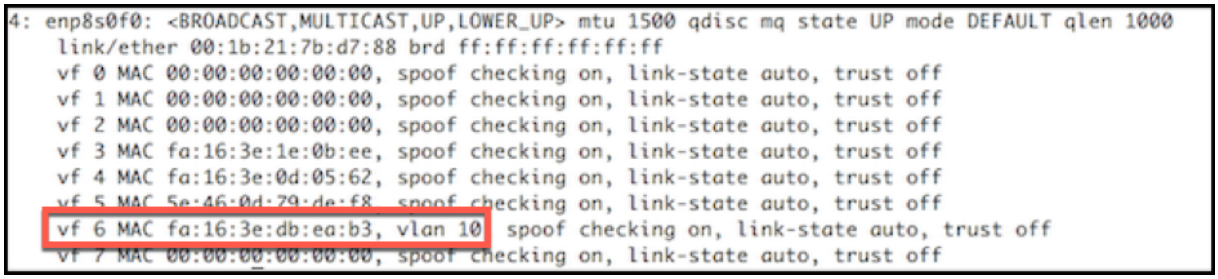
サポートされているその他の構成：ホストからの **SR-IOV VF** 上の **VLAN** の作成と削除

SR-IOV VF 上の VLAN を作成するには、次のコマンドを入力します。

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

前述のコマンドでは、「enp8s0f0」は物理機能の名前です。

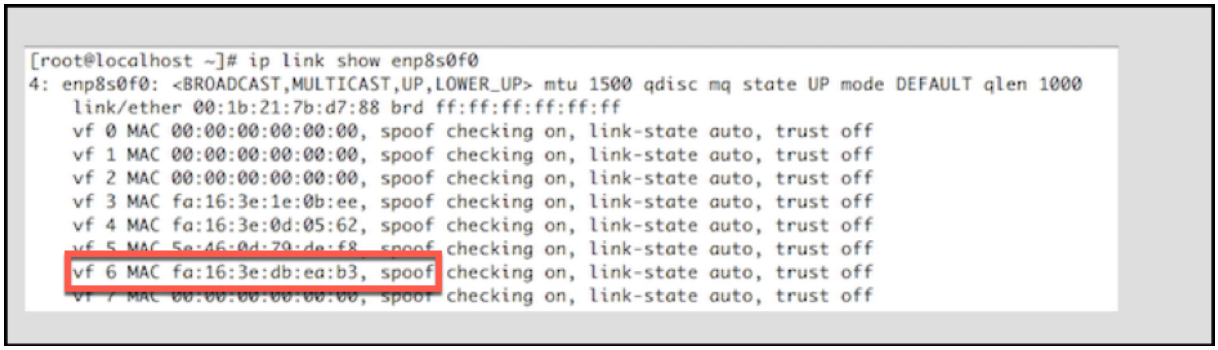
例：vf 6 で作成された VLAN 10



SR-IOV VF 上の VLAN を削除するには、次のコマンドを入力します。

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

例：VLAN 10、vf 6 から削除された



これらの手順により、SRIOV テクノロジを使用する NetScaler VPX インスタンスを OpenStack 上で展開する方法が完了します。

KVM 上の NetScaler VPX インスタンスが OVS DPDK ベースのホストインターフェイスを使用するように構成する

October 17, 2024

KVM (Fedora と RHOS) で実行されている NetScaler VPX インスタンスを Open vSwitch (OVS) と Data Plane Development Kit（DPDK）を使用するように構成して、ネットワークパフォーマンスを向上させることができます。このドキュメントでは、KVM ホスト上の OVS-DPDK によって公開される`vhost-user` ポートで動作するように NetScaler ADC VPX インスタンスを構成する方法について説明します。

OVS は、オープンソースの Apache 2.0 ライセンスでライセンスされている多層仮想スイッチです。DPDK は、高速パケット処理のためのライブラリとドライバのセットです。

以下のバージョンの Fedora、RHOS、OVS、および DPDK は、NetScaler VPX インスタンスを設定するために認定されています。

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

前提条件

DPDK をインストールする前に、ホストに 1GB の巨大なページがあることを確認してください。

詳細については、この DPDK システム要件ドキュメントを参照してください。OVS DPDK ベースのホストインターフェイスを使用するように KVM で NetScaler ADC VPX インスタンスを構成するために必要な手順の概要は次のとおりです。

- DPDK をインストールします。
- OVS を構築し、インストールします。
- OVS ブリッジを作成します。
- OVS ブリッジに物理インターフェイスを接続します。
- OVS データパスに`vhost-user`ポートを接続します。
- OVS-DPDK ベースの`vhost-user`ポートで KVM-VPX をプロビジョニングします

DPDK のインストール

DPDK をインストールするには、この Open vSwitch with DPDK ドキュメントに記載されている指示に従ってください。

OVS のビルドとインストール

OVS のダウンロードページから OVS をダウンロードします。次に、DPDK データパスを使用して OVS をビルドおよびインストールします。「Open vSwitch のインストール 」ドキュメントに記載されている手順に従います。

詳細については、「DPDK 入門ガイド for Linux」を参照してください。

OVS ブリッジの作成

必要に応じて、Fedora コマンドか RHOS コマンドを入力して、OVS ブリッジを作成します。

Fedora コマンド:

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

RHOS コマンド:

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

物理インターフェイスを OVS ブリッジに接続します

ポートを DPDK にバインドし、次の Fedora または RHOS コマンドを入力して OVS ブリッジにアタッチします。

Fedora コマンド:

```
> $OVS_DIR/utilities/ovs-vsctl add-port ovs-br0 dpdk0 -- set
   Interface dpdk0 type=dpdk  options:dpdk-devargs=0000:03:00.0

> $OVS_DIR/utilities/ovs-vsctl add-port ovs-br0 dpdk1 -- set
   Interface dpdk1 type=dpdk  options:dpdk-devargs=0000:03:00.1
```

RHOS コマンド:

```
ovs-vsctl add-port ovs-br0 dpdk0 -- set Interface dpdk0 type=dpdk
   options:dpdk-devargs=0000:03:00.0


ovs-vsctl add-port ovs-br0 dpdk1 -- set Interface dpdk1 type=dpdk
   options:dpdk-devargs=0000:03:00.1
```

オプションの一部として表示される `dpdk-devargs` は、それぞれの物理 NIC の PCI BDF を指定します。

OVS データパスに `vhost-user` ポートを接続する

OVS データパスに`vhost-user`ポートを接続するには、次の Fedora または RHOS コマンドを入力します。

Fedora コマンド:


```
> $OVS_DIR/utilities/ovs-vsctl add-port ovs-br0 vhost-user1 -- set
   Interface vhost-user1 type=dpdkvhostuser -- set Interface vhost-
   user1  mtu_request=9000

> $OVS_DIR/utilities/ovs-vsctl add-port ovs-br0 vhost-user2 -- set
   Interface vhost-user2 type=dpdkvhostuser -- set Interface vhost-
   user2  mtu_request=9000

chmod g+w  /usr/local/var/run/openvswitch/vhost*
```

RHOS コマンド:

```
ovs-vsctl add-port ovs-br0 vhost-user1 -- set Interface vhost-user1
   type=dpdkvhostuser -- set Interface vhost-user1  mtu_request=9000

ovs-vsctl add-port ovs-br0 vhost-user2 -- set Interface vhost-user2
   type=dpdkvhostuser -- set Interface vhost-user2  mtu_request=9000

chmod g+w /var/run/openvswitch/vhost*
```

OVS-DPDK ベースの `vhost-user` ポートを持つ KVM-VPX のプロビジョニング

次の QEMU コマンドを使用して、CLI からのみ、OVS-DPDK ベースの `vhost-user` ポートを持つ Fedora KVM 上の VPX インスタンスをプロビジョニングできます。**Fedora** コマンド:

```
qemu-system-x86_64 -name KVM-VPX -cpu host -enable-kvm -m 4096M \

-object memory-backend-file,id=mem,size=4096M,mem-path=/dev/hugepages
   ,share=on -numa node,memdev=mem \

-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> \

-device ide-drive,bus=ide.0,unit=0,drive=drive-ide0-0-0,id=ide0-0-0,
   bootindex=1 \

-netdev type=tap,id=hostnet0,script=no,downscript=no,vhost=on \

-device virtio-net-pci,netdev=hostnet0,id=net0,mac=52:54:00:3c:d1:ae,
   bus=pci.0,addr=0x3 \

-chardev socket,id=char0,path=</usr/local/var/run/openvswitch/vhost-
   user1> \

-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 \

-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
```

RHOS の場合は、次のサンプル XML ファイルを使用して、virshを使用して NetScaler ADC VPX インスタンスをプロビジョニングします。

```
1    <domain type='kvm'>
2
3      <name>dpdk-vpx1</name>
4
5      <uuid>aedb844b-f6bc-48e6-a4c6-36577f2d68d6</uuid>
6
7      <memory unit='KiB'>16777216</memory>
8
9      <currentMemory unit='KiB'>16777216</currentMemory>
10
11     <memoryBacking>
12
13       <hugepages>
14
15         <page size='1048576' unit='KiB'/>
16
17       </hugepages>
18
19     </memoryBacking>
20
21     <vcpu placement='static'>6</vcpu>
22
23     <cputune>
24
25       <shares>4096</shares>
26
27       <vcpupin vcpu='0' cpuset='0'/>
28
29       <vcpupin vcpu='1' cpuset='2'/>
30
31       <vcpupin vcpu='2' cpuset='4'/>
32
33       <vcpupin vcpu='3' cpuset='6'/>
34
35       <emulatorpin cpuset='0,2,4,6'/>
36
37     </cputune>
38
39     <numatune>
40
41       <memory mode='strict' nodeset='0'/>
42
```



```
43      </numatune>
44
45      <resource>
46
47        <partition>/machine</partition>
48
49      </resource>
50
51      <os>
52
53        <type arch='x86\_64' machine='pc-i440fx-rhel7.0.0'>hvm</type>
54
55        <boot dev='hd'/>
56
57      </os>
58
59      <features>
60
61        <acpi/>
62
63        <apic/>
64
65      </features>
66
67      <cpu mode='custom' match='minimum' check='full'>
68
69        <model fallback='allow'>Haswell-noTSX</model>
70
71        <vendor>Intel</vendor>
72
73        <topology sockets='1' cores='6' threads='1'/>
74
75        <feature policy='require' name='ss'/>
76
77        <feature policy='require' name='pcid'/>
78
79        <feature policy='require' name='hypervisor'/>
80
81        <feature policy='require' name='arat'/>
82
83    <domain type='kvm'>
84
85      <name>dpdk-vpx1</name>
86
87      <uuid>aedb844b-f6bc-48e6-a4c6-36577f2d68d6</uuid>
88
89      <memory unit='KiB'>16777216</memory>
90
91      <currentMemory unit='KiB'>16777216</currentMemory>
92
93      <memoryBacking>
94
95        <hugepages>
```



```
96
97          <page size='1048576' unit='KiB'/>
98
99        </hugepages>
100
101     </memoryBacking>
102
103     <vcpu placement='static'>6</vcpu>
104
105     <cputune>
106
107       <shares>4096</shares>
108
109       <vcpupin vcpu='0' cpuset='0'/>
110
111       <vcpupin vcpu='1' cpuset='2'/>
112
113       <vcpupin vcpu='2' cpuset='4'/>
114
115       <vcpupin vcpu='3' cpuset='6'/>
116
117       <emulatorpin cpuset='0,2,4,6'/>
118
119     </cputune>
120
121     <numatune>
122
123       <memory mode='strict' nodeset='0'/>
124
125     </numatune>
126
127     <resource>
128
129       <partition>/machine</partition>
130
131     </resource>
132
133     <os>
134
135       <type arch='x86\_64' machine='pc-i440fx-rhel7.0.0'>hvm</type>
136
137       <boot dev='hd'/>
138
139     </os>
140
141     <features>
142
143       <acpi/>
144
145       <apic/>
146
147     </features>
148
```



```
149     <cpu mode='custom' match='minimum' check='full'>
150
151       <model fallback='allow'>Haswell-noTSX</model>
152
153       <vendor>Intel</vendor>
154
155       <topology sockets='1' cores='6' threads='1'/>
156
157       <feature policy='require' name='ss'/>
158
159       <feature policy='require' name='pcid'/>
160
161       <feature policy='require' name='hypervisor'/>
162
163       <feature policy='require' name='arat'/>
164
165       <feature policy='require' name='tsc\_adjust'/>
166
167       <feature policy='require' name='xsaveopt'/>
168
169       <feature policy='require' name='pdpe1gb'/>
170
171       <numa>
172
173         <cell id='0' cpus='0-5' memory='16777216' unit='KiB' memAccess=
              'shared'/>
174
175       </numa>
176
177     </cpu>
178
179     <clock offset='utc'/>
180
181     <on\_poweroff>destroy</on\_poweroff>
182
183     <on\_reboot>restart</on\_reboot>
184
185     <on\_crash>destroy</on\_crash>
186
187     <devices>
188
189       <emulator>/usr/libexec/qemu-kvm</emulator>
190
191       <disk type='file' device='disk'>
192
193         <driver name='qemu' type='qcow2' cache='none'/>
194
195         <source file='/home/NSVPX-KVM-12.0-52.18\_nc.qcow2'/>
196
197         <target dev='vda' bus='virtio'/>
198
199         <address type='pci' domain='0x0000' bus='0x00' slot='0x07'
              function='0x0'/>
```



```
200
201         </disk>
202
203         <controller type='ide' index='0'>
204
205           <address type='pci' domain='0x0000' bus='0x00' slot='0x01'
                function='0x1'/>
206
207         </controller>
208
209         <controller type='usb' index='0' model='piix3-uhci'>
210
211           <address type='pci' domain='0x0000' bus='0x00' slot='0x01'
                function='0x2'/>
212
213         </controller>
214
215         <controller type='pci' index='0' model='pci-root'/>
216
217         <interface type='direct'>
218
219           <mac address='52:54:00:bb:ac:05'/>
220
221           <source dev='enp129s0f0' mode='bridge'/>
222
223           <model type='virtio'/>
224
225           <address type='pci' domain='0x0000' bus='0x00' slot='0x03'
                function='0x0'/>
226
227         </interface>
228
229         <interface type='vhostuser'>
230
231           <mac address='52:54:00:55:55:56'/>
232
233           <source type='unix' path='/var/run/openvswitch/vhost-user1'
                mode='client'/>
234
235           <model type='virtio'/>
236
237           <address type='pci' domain='0x0000' bus='0x00' slot='0x04'
                function='0x0'/>
238
239         </interface>
240
241         <interface type='vhostuser'>
242
243           <mac address='52:54:00:2a:32:64'/>
244
245           <source type='unix' path='/var/run/openvswitch/vhost-user2'
                mode='client'/>
246
```



```
247           <model type='virtio'/>
248
249           <address type='pci' domain='0x0000' bus='0x00' slot='0x05'
                 function='0x0'/>
250
251         </interface>
252
253         <interface type='vhostuser'>
254
255           <mac address='52:54:00:2a:32:74'/>
256
257           <source type='unix' path='/var/run/openvswitch/vhost-user3'
                 mode='client'/>
258
259           <model type='virtio'/>
260
261           <address type='pci' domain='0x0000' bus='0x00' slot='0x06'
                 function='0x0'/>
262
263         </interface>
264
265         <interface type='vhostuser'>
266
267           <mac address='52:54:00:2a:32:84'/>
268
269           <source type='unix' path='/var/run/openvswitch/vhost-user4'
                 mode='client'/>
270
271          <model type='virtio'/>
272
273           <address type='pci' domain='0x0000' bus='0x00' slot='0x09'
                 function='0x0'/>
274
275         </interface>
276
277         <serial type='pty'>
278
279           <target port='0'/>
280
281         </serial>
282
283         <console type='pty'>
284
285           <target type='serial' port='0'/>
286
287         </console>
288
289         <input type='mouse' bus='ps2'/>
290
291         <input type='keyboard' bus='ps2'/>
292
293         <graphics type='vnc' port='-1' autoport='yes'>
294
```



```
295           <listen type='address'/>
296
297         </graphics>
298
299         <video>
300
301           <model type='cirrus' vram='16384' heads='1' primary='yes'/>
302
303           <address type='pci' domain='0x0000' bus='0x00' slot='0x02'
                 function='0x0'/>
304
305         </video>
306
307         <memballoon model='virtio'>
308
309           <address type='pci' domain='0x0000' bus='0x00' slot='0x08'
                 function='0x0'/>
310
311         </memballoon>
312
313       </devices>
314
315     </domain
```

注意事項

XML ファイルでは、サンプルファイルに示されているように、hugepage サイズは 1GB である必要があります。

```
1   <memoryBacking>
2
3       <hugepages>
4
5         <page size='1048576' unit='KiB'/>
6
7       </hugepages>
```

また、サンプルファイルでは、vhost-user1 は ovs-br0 にバインドされたvhostユーザーポートです。

```
1   <interface type='vhostuser'>
2
3         <mac address='52:54:00:55:55:56'/>
4
5         <source type='unix' path='/var/run/openvswitch/vhost-user1'
              mode='client'/>
6
7         <model type='virtio'/>
8
9         <address type='pci' domain='0x0000' bus='0x00' slot='0x04'
              function='0x0'/>
10
11      </interface>
```


NetScaler VPX インスタンスを起動するには、`virsh` コマンドの使用を開始します。

KVM ハイパーバイザーでの NetScaler ADC アプライアンスの初回起動時に NetScaler ADC VPX の構成を適用する

October 17, 2024

NetScaler ADC アプライアンスの初回起動時に、KVM ハイパーバイザーに NetScaler ADC VPX 構成を適用できます。したがって、VPX インスタンスでのお客様のセットアップは、はるかに短時間で構成できます。

プリブート ユーザー データとその形式の詳細については、「クラウド内の NetScaler アプライアンスの最初の起動時に NetScaler VPX 構成を適用する」を参照してください。

> 注
>
> KVM Hypervisor でプレブートユーザーデータを使用してブートストラップするには、デフォルトのゲートウェイ設定を<NS-CONFIG>セクションに渡す必要があります。<NS-CONFIG> タグの内容について詳しくは、次の「サンプル」<NS-CONFIG> セクションを参照してください。

Sample <NS-CONFIG> section:

```
<NS-PRE-BOOT-CONFIG>

    <NS-CONFIG>
        add route 0.0.0.0 0.0.0.0  10.102.38.1
    </NS-CONFIG>

    <NS-BOOTSTRAP>
            <SKIP-DEFAULT-BOOTSTRAP>YES</SKIP-DEFAULT-BOOTSTRAP>
            <NEW-BOOTSTRAP-SEQUENCE>YES</NEW-BOOTSTRAP-SEQUENCE>

        <MGMT-INTERFACE-CONFIG>
                <INTERFACE-NUM> eth0 </INTERFACE-NUM>
                <IP> 10.102.38.216 </IP>
                <SUBNET-MASK> 255.255.255.0 </SUBNET-MASK>
        </MGMT-INTERFACE-CONFIG>
    </NS-BOOTSTRAP>

</NS-PRE-BOOT-CONFIG>
```

KVM ハイパーバイザーでプリブートユーザーデータを提供する方法

KVM ハイパーバイザー上のプリブートユーザーデータは、CD-ROM デバイスを使用して接続された ISO ファイルを介して提供できます。

CD-ROM ISO ファイルを使用したユーザーデータの提供

バーチャルマシンマネージャー (VMM) を使用すると、CDROM デバイスを使用して ISO イメージとしてバーチャルマシン (VM) にユーザーデータを挿入できます。KVM は、VM ホストサーバー上の物理ドライブに直接アクセスするか、ISO イメージにアクセスして VM ゲストの CD-ROM をサポートします。

CD-ROM ISO ファイルを使用してユーザーデータを指定するには、次の手順に従います。

1. プレブートユーザーデータコンテンツを含むファイル名`userdata`でファイルを作成します。

   > 注
   >
   > ファイル名は厳密に `userdata`として使用する必要があります。

2. Store the `userdata` file in a folder, and build an ISO image using the folder.

   You can build an ISO image with `userdata` file by the following two methods:

   - Using any image processing tool such as PowerISO.
   - Using `mkisofs` command in Linux.

   The following sample configuration shows how to generate an ISO image using the `mkisofs` command in Linux.

   ```
   root@ubuntu:~/sai/19oct# ls -lh
   total 4.0K
   -rw-r--r-- 1 root root 1.1K Oct 19 16:25 userdata
   root@ubuntu:~/sai/19oct#
   root@ubuntu:~/sai/19oct# mkisofs -o kvm-userdata.iso userdata
   I: -input-charset not specified, using utf-8 (detected in locale
       settings)
   Total translation table size: 0
   Total rockridge attributes bytes: 0
   Total directory bytes: 0
   Path table size(bytes): 10
   Max brk space used 0
   175 extents written (0 MB)
   root@ubuntu:~/sai/19oct#
   root@ubuntu:~/sai/19oct# ls -lh
   total 356K
   -rw-r--r-- 1 root root 350K Oct 19 16:25 kvm-userdata.iso
   -rw-r--r-- 1 root root 1.1K Oct 19 16:25 userdata
   ```

3. 標準の展開プロセスを使用して NetScaler ADC VPX インスタンスをプロビジョニングし、仮想マシンを作成します。But do not power on the VM automatically.

4. 仮想マシンマネージャーで CD-ROM デバイスを追加するには、次の手順に従います。

   a) 仮想マシンマネージャで VM ゲストエントリをダブルクリックしてコンソールを開き、[表示] **> [**詳細**]** の順に選択して **[**詳細**]** ビューに切り替えます。
   b) [ハードウェアの追加**] > [**ストレージ**] > [**デバイスの種類**] > [CDROM** デバイス] をクリックします。

c) [管理] をクリックして正しい ISO ファイルを選択し、[完了] をクリックします。NetScaler VPX インスタンスの「リソース」の下に新しい CDROM が作成されます。

5. Power on the VM.

AWS での NetScaler VPX

October 17, 2024

NetScaler VPX インスタンスは、Amazon Web Services (AWS) で起動できます。NetScaler VPX アプライアンスは、AWS マーケットプレイスで Amazon Machine Image (AMI) として利用できます。AWS 上の NetScaler VPX インスタンスを使用すると、AWS のクラウドコンピューティング機能を使用したり、NetScaler の負荷分散機能とトラフィック管理機能をビジネスニーズに合わせて使用したりできます。VPX インスタンスは、物理 NetScaler アプライアンスのすべてのトラフィック管理機能をサポートし、スタンドアロンインスタンスまたは HA ペアとして展開できます。VPX の機能の詳細については、VPX のデータシートを参照してください。

はじめに

VPX のデプロイを開始する前に、次の情報を理解しておく必要があります。

- AWS 用語
- AWS-VPX サポートマトリックス
- 制限事項と使用ガイドライン
- 前提条件
- AWS 上の NetScaler VPX インスタンスの仕組み

AWS で NetScaler ADC VPX インスタンスを展開する

AWS では、VPX インスタンスで次のデプロイタイプがサポートされています。

- Standalone
- 高可用性 (アクティブ-パッシブ)
  - 同一ゾーン内での高可用性
  - Elastic IP を使用した異なるゾーンでの高可用性
  - プライベート IP を使用して、異なるゾーン間で高可用性
- アクティブ-アクティブ GSLB
- ADM を使用した自動スケーリング (アクティブ-アクティブ)

ハイブリッド展開

- NetScaler を AWS アウトポストにデプロイ
- AWS の VMC に NetScaler をデプロイする

ライセンス

AWS 上の NetScaler VPX インスタンスにはライセンスが必要です。AWS で実行されている NetScaler VPX インスタンスでは、次のライセンスオプションを使用できます。

- 無料 (無制限)
- 毎時
- 年次
- BYOL
- 無料トライアル（AWS マーケットプレイスでは、すべての NetScaler VPX-AWS サブスクリプションを 21 日間無料で提供）

自動化

- NetScaler ADM: スマートな導入
- GitHub CFT: AWS デプロイ用の NetScaler テンプレートとスクリプト
- GitHub Ansible: AWS デプロイ用の NetScaler テンプレートとスクリプト
- GitHub Terraform: AWS デプロイ用の NetScaler テンプレートとスクリプト
- AWS パターンライブラリ (PL): NetScaler VPX

ブログ

- NetScaler on AWS が顧客によるアプリケーションの安全な配信をどのように支援するか
- NetScaler と AWS によるハイブリッドクラウドでのアプリケーション配信
- Citrix は AWS ネットワーキングコンピテンシーパートナーです
- NetScaler: いつでもパブリッククラウドに対応
- NetScaler を使用してパブリッククラウドで簡単にスケールアウトまたはスケールインできます
- Citrix、AWS Outposts で ADC のデプロイメントの選択肢を拡大
- NetScaler と Amazon VPC イングレスルーティングの使用
- Citrix は、AWS での選択肢、パフォーマンス、シンプルなデプロイメントを提供します

- NetScaler Web App Firewall のセキュリティ—現在 AWS Marketplace で公開中
- Aria Systems が AWS で NetScaler Web App Firewall を使用する方法

ビデオ

- ADM によるパブリッククラウドの NetScaler 導入の簡素化
- すぐに使用できるテラフォームスクリプトを使用して AWS で NetScaler VPX を Provisioning および構成する
- クラウドフォーメーションテンプレートを使用して NetScaler HA を AWS にデプロイ
- AWS クイックスタートを使用してアベイラビリティーゾーン全体に NetScaler HA
- ADM を使用した NetScaler オートスケール

お客様のケーススタディ

- テクノロジーソリューション-Xenit AB
- Citrix と AWS クラウドとのより良いビジネス方法—Aria
- NetScaler と AWS の優位性をご覧ください
- Rain for Rent-お客様事件

解決方法

- NetScaler を使用して AWS にデジタル広告プラットフォームをデプロイする
- NetScaler による AWS でのクリックストリーム分析の強化

サポート

- サポートケースを開く
- NetScaler サブスクリプション オファリングについては、「AWS 上の VPX インスタンスのトラブルシューティング」を参照してください。サポートケースを提出するには、AWS アカウント番号とサポート PIN コードを見つけて、NetScaler サポートに連絡してください。
- NetScaler カスタマーライセンスサービスまたは BYOL の場合は、有効なサポートおよびメンテナンス契約を結んでいることを確認してください。契約を結んでいない場合は、NetScaler の担当者にお問い合わせください。

その他の参考資料

- AWS オンデマンドウェビナー-AWS 上の NetScaler
- NetScaler VPX データシート
- AWS Marketplace の NetScaler
- NetScaler は、AWS ネットワーキングパートナーソリューション（ロードバランサー）の一部です。
- AWS に関するよくある質問

AWS 用語

October 17, 2024

このセクションでは、よく使用される AWS の用語と語句のリストについて説明します。詳細については、「AWS 用語集」を参照してください。

用語	定義
Amazon マシンイメージ（AMI）	マシンイメージ。クラウド内の仮想サーバーであるインスタンスを起動するのに必要な情報を提供します。
Elastic Block Store	AWS クラウドで Amazon EC2 インスタンスと一緒に使用される、永続ブロックストレージボリュームを提供します。
Simple Storage Service（S3）	Internet 用のストレージ。Web 規模のコンピューティングを開発者が簡単に実施できるように設計されています。
Elastic Compute Cloud（EC2）	クラウドで、安全でサイズ変更できる処理能力を提供する Web サービスです。Web 規模のクラウドコンピューティングを開発者が簡単に実施できるように設計されています。
Elastic Load Balancing（ELB）	複数のアベイラビリティゾーンで、複数の EC2 インスタンスにまたがる受信アプリケーショントラフィックを分散します。これによってアプリケーションのフォールトトレランスが増加します。
エラスティックネットワークインターフェイス (ENI)	仮想プライベートクラウド（VPC）内のインスタンスにアタッチできる仮想ネットワークインターフェイス。

用語	定義
Elastic IP（EIP）アドレス	Amazon EC2 または Amazon VPC で割り当てられ、インスタンスにアタッチされた、静的パブリック IPv4 アドレスです。Elastic IP アドレスは特定のインスタンスではなく、お使いのアカウントに関連しています。ニーズの変化に応じて、割り当て、アタッチ、デタッチ、および解放が簡単にできるため、Elastic（融通が利く）と呼ばれています。
インスタンスの種類	Amazon EC2 では、さまざまなユースケースに対応できるよう最適化された幅広い種類のインスタンスを提供してます。インスタンスタイプを構成する CPU、メモリ、ストレージ、およびネットワーク機能の組み合わせはさまざまで、アプリケーションに合わせて最適なリソースの組み合わせを柔軟に選択できます。
Identity and Access Management（IAM）	AWS で ID が実行できること、または実行できないことを決定する許可ポリシーを持つ AWS の ID。IAM ロールを使うことで EC2 インスタンス上で実行されるアプリケーションが、AWS リソースに安全にアクセスできるようになります。高可用性セットアップで VPX インスタンスを展開する場合、IAM ロールは必須です。
インターネットゲートウェイ	ネットワークをインターネットに接続します。VPC 外部の IP アドレスのトラフィックをインターネットゲートウェイにルーティングできます。
キーペア	身元を電子的に証明するために使用する一連の資格情報。キーペアはプライベートキーとパブリックキーで構成されます。
ルートテーブル	関連付けられているサブネットからのトラフィックを制御するための一連のルーティング規則。1 つのルートテーブルに対して複数のサブネットを関連付けることができますが、各サブネットは一度に 1 つのルートテーブルにしか関連付けることができません。
セキュリティグループ	あるインスタンスに対して許可されている、名前が付けられた一連の受信方向のネットワーク接続。
サブネット	EC2 インスタンスをアタッチできる VPC の IP アドレス範囲の一部分。セキュリティと運用上の必要に応じて、サブネットを作成し、インスタンスをグループ分けできます。

用語	定義
Virtual Private Cloud（VPC）	定義した仮想ネットワーク内で AWS リソースを起動できる、AWS クラウドの論理的に隔離されたセクションをプロビジョニングする Web サービス。
Auto Scaling	ユーザー定義のポリシー、スケジュール、ヘルスチェックに基づいて Amazon EC2 インスタンスを自動的に起動または終了するウェブサービス。
クラウドの形成	関連する AWS リソースを 1 つの単位として一緒に作成および削除するテンプレートを書き込んだり変更したりするサービス。

AWS-VPX サポートマトリックス

October 17, 2024

次の表に、サポートされている VPX モデルと AWS リージョン、インスタンスタイプ、およびサービスを示します。

表 1: AWS でサポートされている VPX モデル

サポートされている VPX モデル
NetScaler VPX アドバンスド-200 Mbps
NetScaler VPX プレミアム - 1 Gbps
NetScaler VPX プレミアム - 5 Gbps
NetScaler VPX エクスプレス - 20 Mbps
NetScaler VPX - 顧客ライセンス
NetScaler VPX FIPS - 顧客ライセンス
NetScaler VPX FIPS ENA - 顧客ライセンス

表 2: サポートされている AWS リージョン

サポートされている AWS リージョン
米国西部 (オレゴン)
米国東部 (バージニア北部) 米国西部 (北カリフォルニア)
米国東部 (オハイオ)

| 米国東部 (北アメリカ) バージニア州 |
| アジアパシフィック (ムンバイ) |
| アジアパシフィック (ソウル) |
| アジアパシフィック (シンガポール) |
| アジアパシフィック (シドニー) |
| アジアパシフィック (東京) |
| アジアパシフィック (香港) |
| アジアパシフィック (大阪) |
| アジアパシフィック (ジャカルタ) |
| アジアパシフィック (ハイデラバード) |
| カナダ (中部) |
| EU (フランクフルト) |
| 欧州 (アイルランド) |
| 欧州 (ロンドン) |
| 欧州 (パリ) |
| 欧州 (ミラノ) |
| 南米 (サンパウロ) |
| AWS GovCloud (米国東部) |
| AWS GovCloud (米国西部) |
| AWS トップシークレット (C2S) |
| 中東 (バーレーン) |
| アフリカ (ケープタウン) |
| C2S |

> 注
>
> AWS 香港リージョンでは、NetScaler VPX サポートは BYOL ライセンスでのみ利用可能です。

表 3: サポートされている AWS インスタンスタイプ

| サポートされる AWS インスタンスタイプ |
| ———————————————————————————–|
| c4.large、c4.xlarge、c4.2xlarge、c4.4xlarge、c4.8xlarge |
| c5.large、c5.xlarge、c5.2xlarge、c5.4xlarge、c5.9xlarge、c5.18xlarge、c5.24xlarge |
| c5n.large、c5n.xlarge、c5n.2xlarge、c5n.4xlarge、c5n.9xlarge、c5n.18xlarge |
| 奥行 2.XL サイズ、D2.2xL サイズ、D2.4xL サイズ、D2.8xL サイズ |
| m3.large, m3.xlarge, m3.2xlarge |
| m4.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m4.10xlarge, m4.16xlarge |
| m5.large, m5.xlarge, m5.2xlarge, m5.4xlarge, m5.8xlarge, m5.12xlarge, m5.16xlarge, m5.24xlarge |
| m5a.large、m5a.xlarge、m5a.2xlarge、m5a.4xlarge、m5a.8xlarge、m5a.12xlarge、m5a.16xlarge、m5a.24xlarge |
| m5n.large、m5n.xlarge、m5n.2xlarge、m5n.4xlarge、m5n.8xlarge、m5n.12xlarge、m5n.16xlarge、

m5n.24xlarge |
| m6i.large, m6i.xlarge, m6i.2xlarge, m6i.4xlarge, m6i.8xlarge, m6i.12xlarge, m6i.16xlarge, m6i.24xlarge, m6i.32xlarge |
| r7iz.large、r7iz.xlarge、r7iz.2xlarge、r7iz.4xlarge、r7iz.8xlarge、r7iz.12xlarge、r7iz.16xlarge、r7iz.32xlarge |
| t2.medium, t2.large, t2.xlarge, t2.2xlarge |
| t3a.medium, t3a.large, t3a.xlarge, t3a.2xlarge |

> 注
>
> AWS m6i および r7iz インスタンスタイプでプロビジョニングされた NetScaler VPX は、ENA 低レイテンシーキュー（LLQ）機能をサポートしていません。

表 4: サポートされる AWS サービス

サポートされている AWS サービス
EC2: ADC インスタンスを起動します。
ラムダ： CFT からの NetScaler VPX インスタンスのプロビジョニング中に、NetScaler VPX NITRO API を呼び出します。
VPC と **VPC** イングレスルーティング**:** VPC は、ADC を起動できる分離されたネットワークを作成します。VPC 入力ルーティング
Route53: NetScaler Autoscale e ソリューション内のすべての NetScaler VPX ノードにトラフィックを分散します。
ELB: NetScaler Autoscale e ソリューション内のすべての NetScaler VPX ノードにトラフィックを分散します。
Cloudwatch： NetScaler VPX インスタンスのパフォーマンスとシステムパラメーターを監視します。
AWS Autoscaling: バックエンドサーバーの自動スケーリングに使用されます。
クラウドの形成： CloudFormation テンプレートは、NetScaler VPX インスタンスをデプロイするために使用されます。
Simple Queue Service (SQS): バックエンド自動スケーリングでスケールアップおよびスケールダウンイベントを監視します。
簡易通知サービス **(SNS):** バックエンド自動スケーリングでスケールアップおよびスケールダウンイベントを監視します。
ID とアクセス管理 **(IAM):** AWS のサービスとリソースへのアクセスを提供します。
AWS Outposts: AWS Outposts で NetScaler VPX インスタンスをプロビジョニングします。

NetScaler では、以下の AWS インスタンスタイプを推奨しています：

- マーケットプレイスエディションまたは帯域幅ベースのプールライセンス用の M5 および C5n シリーズ。
- vCPU ベースのプールライセンス用の C5n シリーズ

AWS マーケットプレイスでの VPX オファリング	AWS インスタンスの推奨事項
VPX Express 20、VPX 200	M5.xLarge
VPX 1G, VPX 5G	M5.2xLarge

NetScaler は、スループットに基づいて次の AWS インスタンスタイプを推奨します。

プールライセンス付き VPX（帯域幅ライセンス）	AWS インスタンスの推奨事項
VPX 8G	C5n.4xLarge
VPX 10G、VPX 15G、VPX 25G	C5n.9xLarge

> 注
>
> VPX 25G サービスでは、AWS で期待される 25G のスループットは得られませんが、SSL トランザクションレートは高くなります。

5G を超えるスループットを実現するには、次の手順を実行します:

- **AWS** マーケットプレイスで提供されている **NetScaler VPX-**カスタマーライセンス（**BYOL**）サービスを選択してください。
- NetScaler GUI または CLI で［プールライセンス（帯域幅ライセンス)］を選択します。

1 秒あたりのパケット数、SSL トランザクションレートなどのさまざまなメトリックに基づいてインスタンスを決定するには、NetScaler の担当者に連絡してガイダンスを受けてください。vCPU ベースのプールライセンスとサイジングのガイダンスについては、NetScaler サポートにお問い合わせください。

制限事項と使用ガイドライン

October 17, 2024

NetScaler VPX インスタンスを AWS にデプロイする際には、以下の制限事項と使用上のガイドラインが適用されます。

- 始める前に、「AWS に NetScaler VPX インスタンスをデプロイする」の AWS 用語のセクションをお読みください。
- クラスタリング機能は、VPX ではサポートされていません。

- 高可用性セットアップを効果的に機能させるには、専用の NAT デバイスを管理インターフェイスに関連付けるか、EIP を NSIP に関連付けます。NAT について詳しくは、AWS ドキュメントの「NAT Instances」を参照してください。
- データトラフィックおよび管理トラフィックは、異なるサブネットに属する ENI で分離する必要があります。
- 管理 ENI には NSIP アドレスのみが必要です。
- セキュリティ上の理由により、EIP を NSIP に関連付ける代わりに NAT インスタンスを使用する場合は、VPC レベルでルーティングを適切に変更する必要があります。VPC レベルのルーティングの変更手順については、AWS ドキュメントの「シナリオ 2: パブリックサブネットとプライベートサブネットを持つ VPC」を参照してください。
- VPX インスタンスは、ある EC2 インスタンスタイプから別のインスタンスタイプへ（たとえば、m3.large から m3.xlarge へ）移動できます。
- AWS 上の VPX のストレージオプションについては、EBS は耐久性があり、インスタンスからデタッチした後でもデータが利用可能になるため、EBS をお勧めします。
- VPX への ENI の動的追加はサポートされていません。VPX インスタンスを再起動して更新を適用します。スタンドアロンインスタンスまたは HA インスタンスを停止し、新しい ENI を接続してからインスタンスを再起動することをお勧めします。
- 1 つの ENI に複数の IP アドレスを割り当てることができます。ENI あたりの IP アドレスの最大数は EC2 インスタンスタイプによって決まります。Elastic Network Interfacesの「インスタンスタイプごとのネットワークインターフェイスごとの IP アドレス」のセクションを参照してください。IP アドレスを ENI に割り当てる前に、AWS で割り当てる必要があります。詳細については、「Elastic ネットワークインターフェイス」を参照してください。
- NetScaler VPX インターフェイスでは、インターフェイスの有効化および無効化コマンドは使用しないことをお勧めします。
- NetScaler`set ha node \\<NODE\\_ID\\> -haStatus STAYPRIMARY` と`set ha node \\<NODE\\_ID\\> -haStatus STAYSECONDARY` コマンドはデフォルトで無効になっています。
- IPv6 は VPX ではサポートされていません。
- AWS の制限により、次の機能はサポートされていません。
  - GARP（Gratuitous ARP）
  - L2 モード
  - タグ付き VLAN
  - 動的ルーティング
  - 仮想 MAC
- RNAT が機能するには、送信元/送信先チェックが無効になっていることを確認してください。詳細については、Elastic Network Interfacesの「ソース/デスティネーションチェックの変更」を参照してください。

- AWS での NetScaler VPX デプロイメントでは、一部の AWS リージョンで AWS インフラストラクチャが AWS API 呼び出しを解決できない場合があります。これは、API 呼び出しが NetScaler VPX インスタンスの非管理インターフェイスを介して発行された場合に発生します。回避策として、API 呼び出しを管理インターフェイスにのみ制限してください。回避策として、API 呼び出しを管理インターフェースのみに制限します。これを行うには、VPX インスタンス上に NSVLAN を作成し、適切なコマンドを使用して管理インターフェイスを NSVLAN にバインドします。例えば: `ns config -nsvlan を設定します <vlan id> -ifnum 1/1 -tagged NO 設定を保存` プロンプトで VPX インスタンスを再起動します。`nsvlan`の設定の詳細については、NSVLAN の設定を参照してください。
- AWS コンソールでは、実際の使用量がはるかに低い場合でも、監視タブに表示される **VPX** インスタンスの **vCPU** 使用率が高い場合があります（最大 100%）。実際の vCPU 使用率を確認するには、「すべての **CloudWatch** メトリックスを表示」に移動します。詳細については、「Amazon CloudWatch を使用してインスタンスを監視する」を参照してください。
- ホットアドは、AWS 上の NetScaler を使用する PV および SRIOV インターフェイスでのみサポートされます。ENA インターフェイスを持つ VPX インスタンスはホットプラグをサポートしていないため、ホットプラグを試みるとインスタンスの動作が予測できない場合があります。
- AWS ウェブコンソールまたは AWS CLI インターフェイスを介したホット削除は、NetScaler の PV、SRIOV、および ENA インターフェイスではサポートされていません。ホット削除を試みると、インスタンスの動作が予測できなくなる可能性があります。

前提条件

October 17, 2024

AWS で VPX インスタンスを作成する前に、次のものがあることを確認してください。

- **AWS** アカウント:AWS 仮想プライベートクラウド (VPC) で NetScaler VPX AMI を起動します。AWS アカウントは www.aws.amazon.comで無料で作成できます。
- **AWS ID** およびアクセス管理（**IAM**）ユーザーアカウント: ユーザーの AWS サービスおよびリソースへのアクセスを安全にコントロールします。IAM ユーザーアカウントの作成方法の詳細については、「IAM ユーザーの作成 (コンソール)」を参照してください。IAM ロールは、スタンドアロンデプロイと高可用性デプロイの両方で必須です。

  AWS アカウントに関連付けられた IAM ロールには、さまざまなシナリオで次の IAM アクセス権限が必要です。

  同じ **AWS** ゾーン内の **IPv4** アドレスと **HA** ペア**:**

```
1     "ec2:DescribeInstances",
2     "ec2:AssignPrivateIpAddresses",
3     "iam:SimulatePrincipalPolicy",
```



```
    "iam:GetRole",
    "ec2:CreateTags"
```

同じ **AWS** ゾーン内の **IPv6** アドレスと **HA** ペア**:**

```
    "ec2:DescribeInstances",
    "ec2:AssignIpv6Addresses",
    "ec2:UnassignIpv6Addresses",
    "iam:SimulatePrincipalPolicy",
    "iam:GetRole",
    "ec2:CreateTags"
```

同じ **AWS** ゾーン内の **IPv4** と **IPv6** の両方のアドレスとの **HA** ペア**:**

```
    "ec2:DescribeInstances",
    "ec2:AssignPrivateIpAddresses",
    "ec2:AssignIpv6Addresses",
    "ec2:UnassignIpv6Addresses",
    "iam:SimulatePrincipalPolicy",
    "iam:GetRole",
    "ec2:CreateTags"
```

異なる **AWS** ゾーンにまたがる **Elastic IP** アドレスを持つ **HA**

```
    "ec2:DescribeInstances",
    "ec2:DescribeAddresses",
    "ec2:AssociateAddress",
    "ec2:DisassociateAddress",
    "iam:SimulatePrincipalPolicy",
    "iam:GetRole",
    "ec2:CreateTags"
```

異なる **AWS** ゾーンのプライベート **IP** アドレスを持つ **HA** ペア**:**

```
    "ec2:DescribeInstances",
    "ec2:DescribeRouteTables",
    "ec2:DeleteRoute",
    "ec2:CreateRoute",
    "ec2:ModifyNetworkInterfaceAttribute",
    "iam:SimulatePrincipalPolicy",
    "iam:GetRole",
    "ec2:CreateTags"
```

異なる **AWS** ゾーンのプライベート **IP** アドレスと **Elastic IP** アドレスの両方を持つ **HA** ペア**:**

```
    "ec2:DescribeInstances",
    "ec2:DescribeAddresses",
    "ec2:AssociateAddress",
    "ec2:DisassociateAddress",
    "ec2:DescribeRouteTables",
    "ec2:DeleteRoute",
    "ec2:CreateRoute",
    "ec2:ModifyNetworkInterfaceAttribute",
```



```
9     "iam:SimulatePrincipalPolicy",
10    "iam:GetRole",
11    "ec2:CreateTags"
```

AWS バックエンドの自動スケーリング**:**

```
1     "ec2:DescribeInstances",
2     "autoscaling:*",
3     "sns:CreateTopic",
4     "sns:DeleteTopic",
5     "sns:ListTopics",
6     "sns:Subscribe",
7     "sqs:CreateQueue",
8     "sqs:ListQueues",
9     "sqs:DeleteMessage",
10    "sqs:GetQueueAttributes",
11    "sqs:SetQueueAttributes",
12    "iam:SimulatePrincipalPolicy",
13    "iam:GetRole",
14    "ec2:CreateTags"
```

> 注
>
> - 前述の機能を組み合わせて使用する場合は、各機能に IAM アクセス権限を組み合わせて使用します。
> - Citrix CloudFormation テンプレートを使用すると、IAM ロールが自動的に作成されます。このテンプレートでは、作成済みの IAM ロールを選択することはできません。
> - GUI から VPX インスタンスにログオンすると、IAM ロールに必要な権限を設定するよう求めるプロンプトが表示されます。権限をすでに構成している場合は、このプロンプトを無視してください。

- **AWS CLI**: ターミナルプログラムから AWS マネジメントコンソールが提供するすべての機能を使用する。詳細については、AWS CLI ユーザーガイドを参照してください。また、ネットワークインターフェイスの種類を SR-IOV に変更するには、AWS CLI も必要です。
- **Elastic Network Adapter (ENA)**: M5、C5 インスタンスなどの ENA ドライバー対応インスタンスタイプの場合、ファームウェアバージョンは 13.0 以降である必要があります。
- NetScaler VPX の EC2 インスタンスでインスタンスメタデータサービス（IMDS）を構成する必要があります。IMDSv1 と IMDSv2 は、実行中の AWS EC2 インスタンスからインスタンスメタデータにアクセスするための 2 つのモードです。IMDSv2 は IMDSv1 よりも安全です。インスタンスを両方の方法（デフォルトオプション）を使用するように構成することも、IMDSv2 モードのみを使用するように構成することもできます（IMDSv1 を無効にする）。Citrix ADC VPX は、NetScaler VPX リリース 13.1.48.x 以降の IMDSv2 専用モードをサポートしています。

NetScaler VPX インスタンスで AWS IAM ロールを設定します

October 17, 2024

Amazon EC2 インスタンスで実行されるアプリケーションには、AWS API リクエストに AWS 認証情報を含める必要があります。AWS 認証情報を Amazon EC2 インスタンス内に直接保存し、そのインスタンス内のアプリケーションがそれらの認証情報を使用できるようにすることができます。ただし、認証情報を管理し、認証情報が各インスタンスに安全に渡されるようにし、認証情報をローテーションするときに各 Amazon EC2 インスタンスを更新する必要があります。それは多くの追加作業です。

代わりに、Amazon EC2 インスタンスで実行されるアプリケーションの一時的な認証情報を管理するには、ID とアクセス管理 (IAM) ロールを使用することができ、また使用する必要があります。ロールを使用すると、長期にわたる認証情報 (ユーザー名、パスワード、アクセスキーなど) を Amazon EC2 インスタンスに配布する必要はありません。代わりに、ロールはアプリケーションが他の AWS リソースを呼び出すときに使用できる一時的なアクセス権限を提供します。Amazon EC2 インスタンスを起動するときに、インスタンスに関連付ける IAM ロールを指定します。インスタンスで実行されるアプリケーションは、ロールが提供した一時的な認証情報を使用して API リクエストに署名できます。

AWS アカウントに関連付けられた IAM ロールには、さまざまなシナリオで次の IAM アクセス権限が必要です。

同じ **AWS** ゾーン内の **IPv4** アドレスと **HA** ペア**:**

```
"ec2:DescribeInstances",
"ec2:AssignPrivateIpAddresses",
"iam:SimulatePrincipalPolicy",
"iam:GetRole"
```

同じ **AWS** ゾーン内の **IPv6** アドレスと **HA** ペア**:**

```
"ec2:DescribeInstances",
"ec2:AssignIpv6Addresses",
"ec2:UnassignIpv6Addresses",
"iam:SimulatePrincipalPolicy",
"iam:GetRole"
```

同じ **AWS** ゾーン内の **IPv4** と **IPv6** の両方のアドレスとの **HA** ペア**:**

```
"ec2:DescribeInstances",
"ec2:AssignPrivateIpAddresses",
"ec2:AssignIpv6Addresses",
"ec2:UnassignIpv6Addresses",
"iam:SimulatePrincipalPolicy",
"iam:GetRole"
```

異なる **AWS** ゾーンにまたがる **Elastic IP** アドレスを持つ **HA**

```
"ec2:DescribeInstances",
"ec2:DescribeAddresses",
```



```
    "ec2:AssociateAddress",
    "ec2:DisassociateAddress",
    "iam:SimulatePrincipalPolicy",
    "iam:GetRole"
```

異なる **AWS** ゾーンのプライベート **IP** アドレスを持つ **HA** ペア**:**

```
    "ec2:DescribeInstances",
    "ec2:DescribeRouteTables",
    "ec2:DeleteRoute",
    "ec2:CreateRoute",
    "ec2:ModifyNetworkInterfaceAttribute",
    "iam:SimulatePrincipalPolicy",
    "iam:GetRole"
```

異なる **AWS** ゾーンのプライベート **IP** アドレスと **Elastic IP** アドレスの両方を持つ **HA** ペア**:**

```
    "ec2:DescribeInstances",
    "ec2:DescribeAddresses",
    "ec2:AssociateAddress",
    "ec2:DisassociateAddress",
    "ec2:DescribeRouteTables",
    "ec2:DeleteRoute",
    "ec2:CreateRoute",
    "ec2:ModifyNetworkInterfaceAttribute",
    "iam:SimulatePrincipalPolicy",
    "iam:GetRole"
```

AWS バックエンドの自動スケーリング**:**

```
    "ec2:DescribeInstances",
    "autoscaling:*",
    "sns:CreateTopic",
    "sns:DeleteTopic",
    "sns:ListTopics",
    "sns:Subscribe",
    "sqs:CreateQueue",
    "sqs:ListQueues",
    "sqs:DeleteMessage",
    "sqs:GetQueueAttributes",
    "sqs:SetQueueAttributes",
    "iam:SimulatePrincipalPolicy",
    "iam:GetRole"
```

注意事項:

- 前述の機能を組み合わせて使用する場合は、各機能に IAM アクセス権限を組み合わせて使用します。
- Citrix CloudFormation テンプレートを使用すると、IAM ロールが自動的に作成されます。このテンプレートでは、作成済みの IAM ロールを選択することはできません。
- GUI から VPX インスタンスにログオンすると、IAM ロールに必要な権限を設定するよう求めるプロンプトが表示されます。権限をすでに構成している場合は、このプロンプトを無視してください。

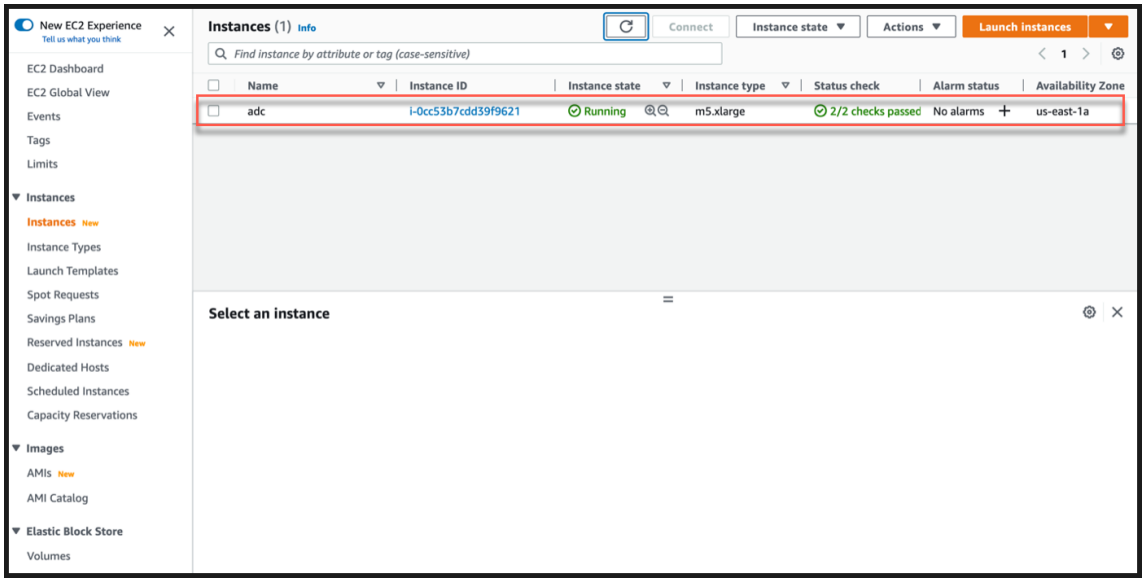
- IAM ロールは、スタンドアロンデプロイと高可用性デプロイの両方で必須です。

IAM 役割を作成する

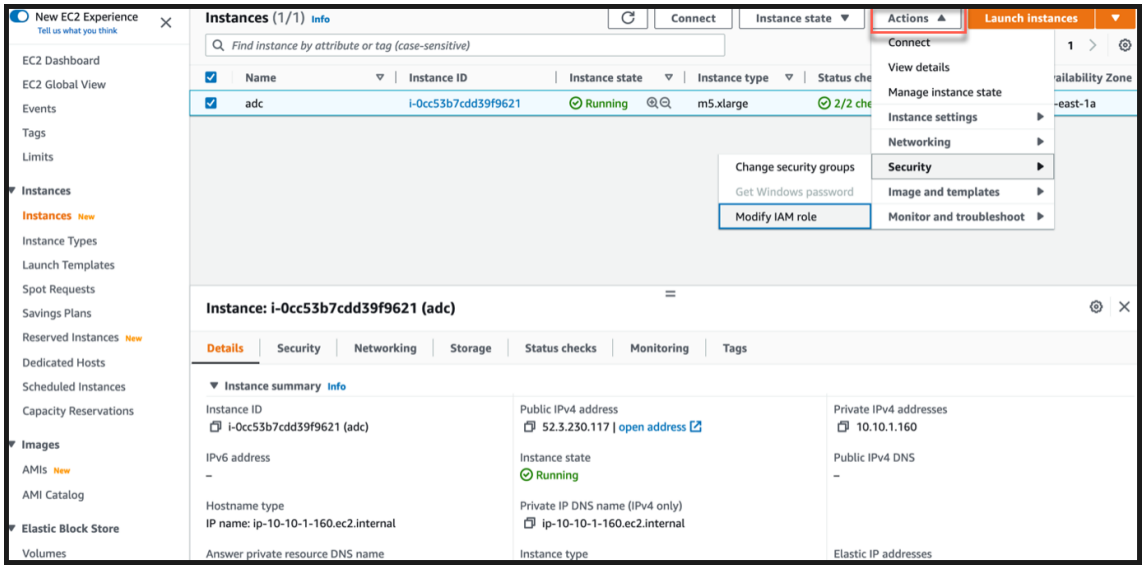
この手順では、AWS バックエンド自動スケーリング機能の IAM ロールを作成する方法について説明します。

> 注
>
> 同じ手順に従って、他の機能に対応する任意の IAM ロールを作成できます。

1. EC2 用 AWS マネジメントコンソールにログインします。
2. EC2 インスタンスページに移動し、ADC インスタンスを選択します。

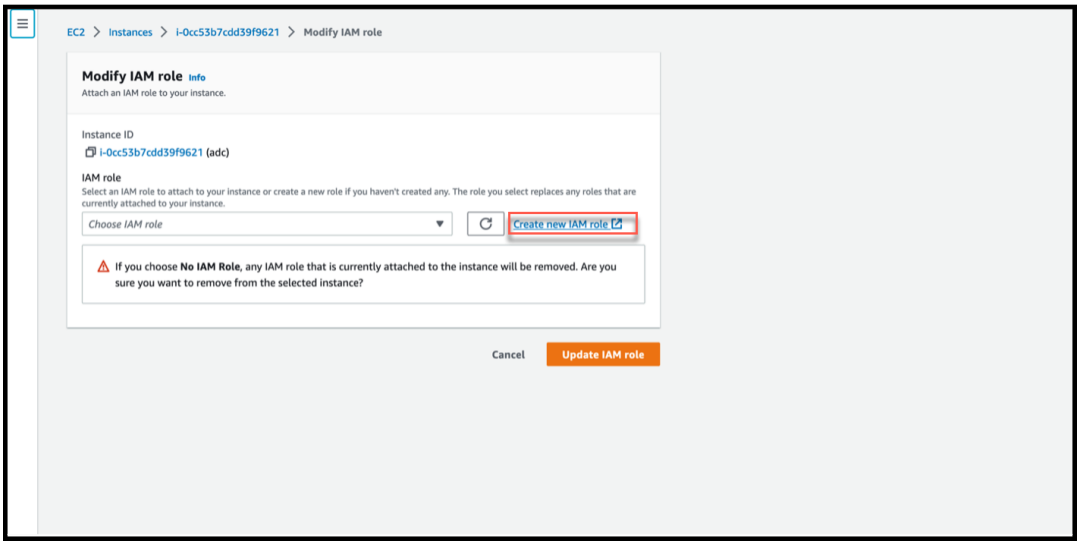


3. [アクション] **>** **[**セキュリティ**] > [IAM** ロールの変更**]** に移動します。

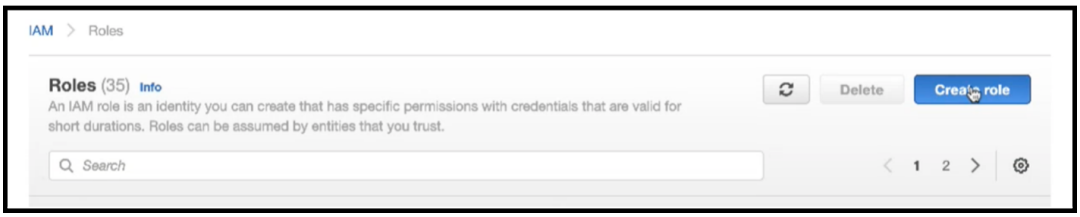


4. **IAM** ロールの変更ページでは、既存の IAM ロールを選択するか、IAM ロールを作成できます。
5. IAM ロールを作成するには、次の手順に従います。

a)「**IAM** ロールの変更」ページで、「新しい **IAM** ロールを作成」をクリックします。



b)「ロール」ページで、「ロールを作成」をクリックします。



c) [信頼できるエンティティタイプ] で [**AWS service**] を選択し、[一般的な使用例] で [**EC2**] を選択し、[次へ] をクリックします。

d)「権限の追加」ページで、「ポリシーの作成」をクリックします。

e) **JSON** タブをクリックして JSON エディターを開きます。

f) JSON エディターで、すべてを削除し、使用したい機能の IAM 権限を貼り付けます。

たとえば、AWS バックエンド自動スケーリング機能用の次の IAM アクセス権限を貼り付けます。

```
{

    "Version": "2012-10-17",
    "Statement": [
        {

            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "autoscaling:*",
                "sns:CreateTopic",
                "sns:DeleteTopic",
                "sns:ListTopics",
                "sns:Subscribe",
                "sqs:CreateQueue",
                "sqs:ListQueues",
                "sqs:DeleteMessage",
                "sqs:GetQueueAttributes",
                "sqs:SetQueueAttributes",
                "iam:SimulatePrincipalPolicy",
                "iam:GetRole"
            ],
```

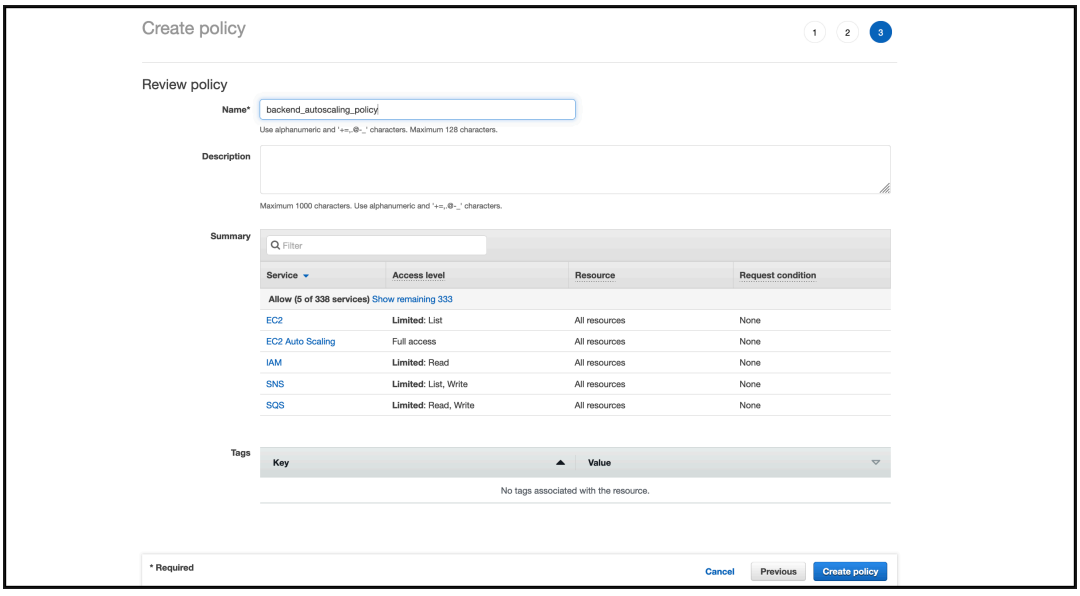


```
24                  "Resource": "*"
25              }
26
27          ]
28      }
```

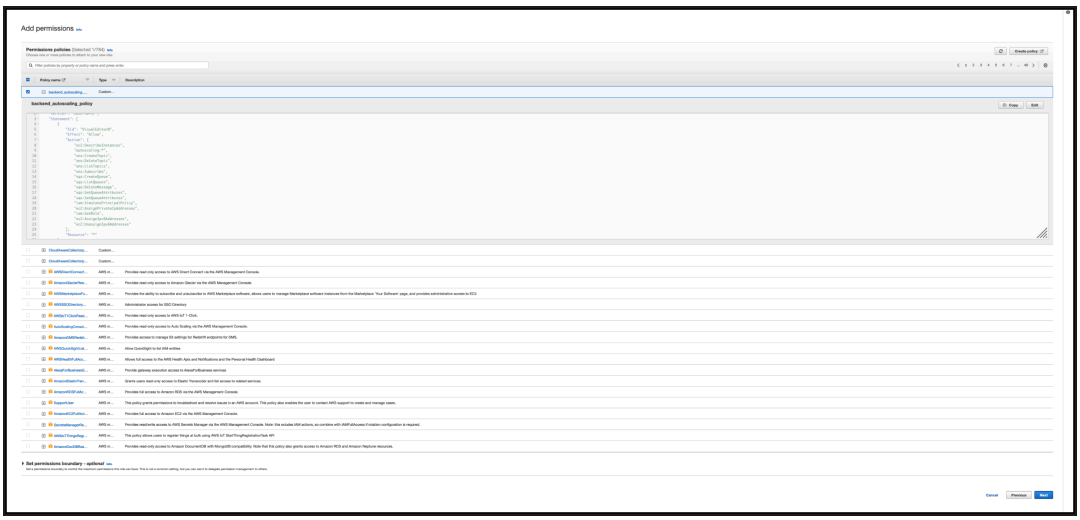
指定する「バージョン」キーと値のペアが、AWS によって自動的に生成されるものと同じであることを確認してください。

g) [次へ**:** 確認] をクリックします。

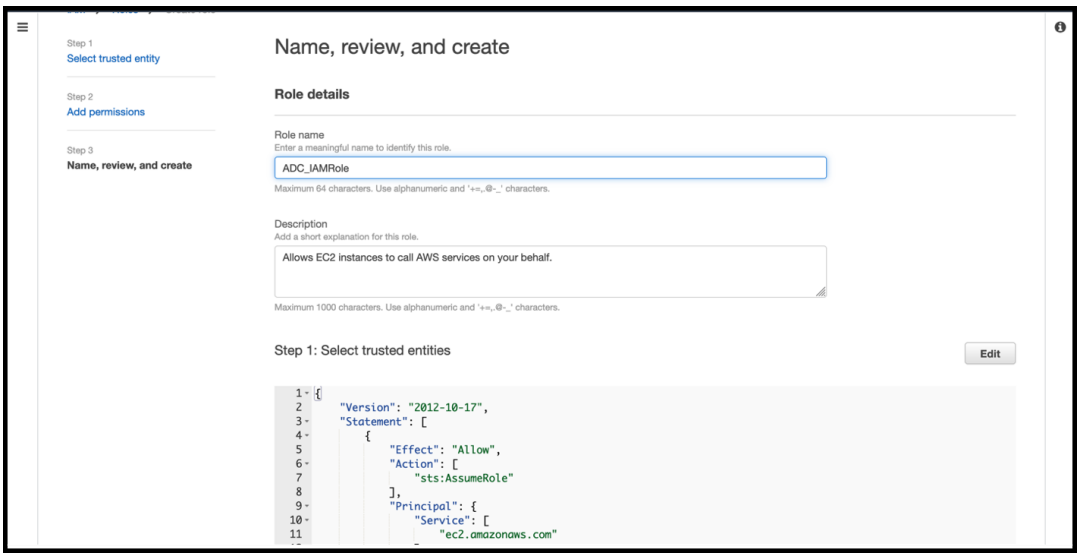
h) [ポリシーの確認] タブで、ポリシーに有効な名前を付けて、[ポリシーの作成] をクリックします。



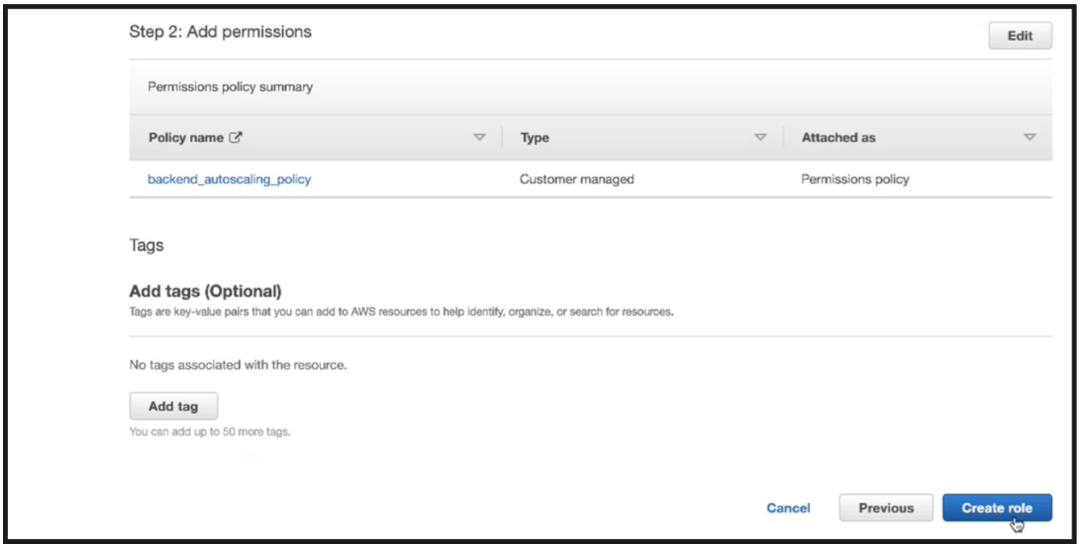
i) **ID** アクセス管理ページで、作成したポリシー名をクリックします。ポリシーを展開して JSON 全体を確認し、[次へ] をクリックします。



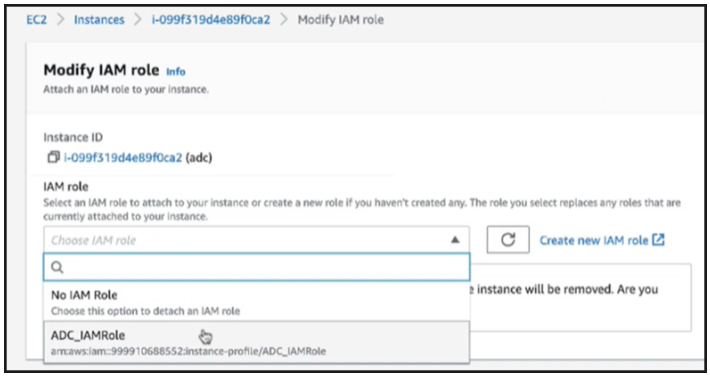
j)「名前、レビュー、作成」ページで、ロールに有効な名前を付けます。



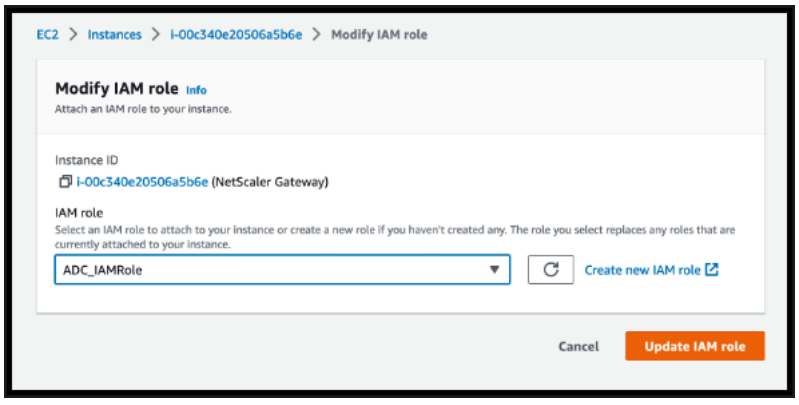
k)「ロールを作成」をクリックします。



6. 手順 1、2、3 を繰り返します。[更新] ボタンを選択し、ドロップダウンメニューを選択すると、作成したロールが表示されます。



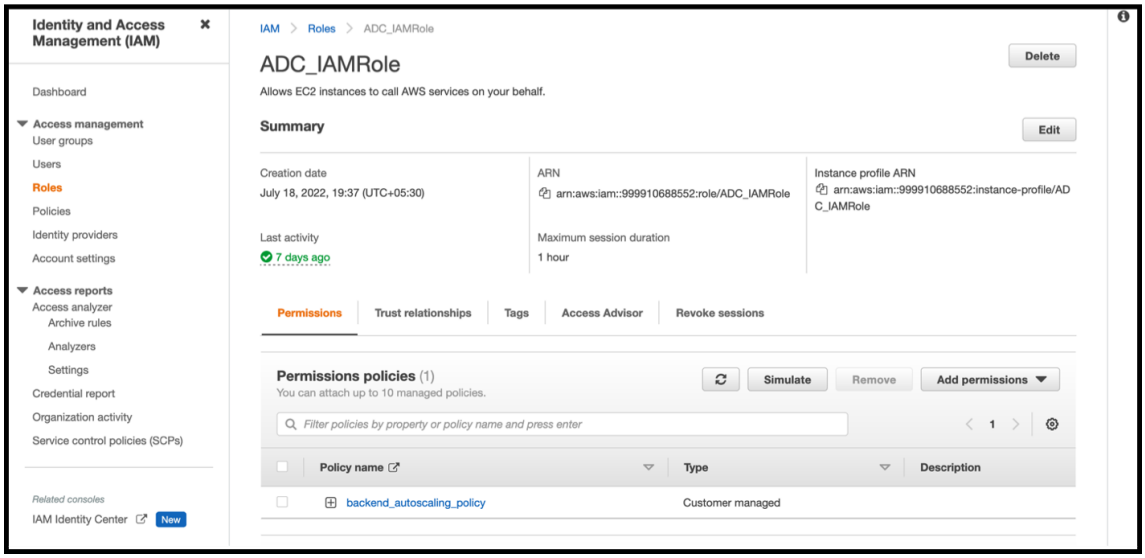
7.「**IAM** ロールを更新」をクリックします。



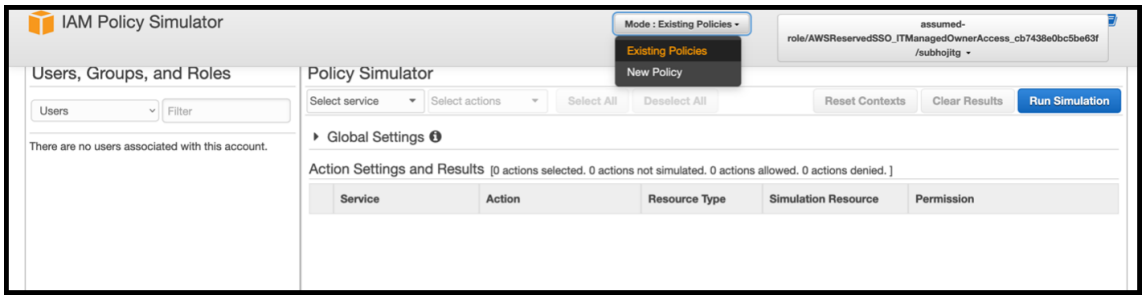
IAM ポリシーシミュレーターで **IAM** ポリシーをテストする

IAM ポリシーシミュレーターは、IAM アクセスコントロールポリシーを実稼働環境に導入する前にその効果をテストできるツールです。権限の確認とトラブルシューティングが簡単になります。

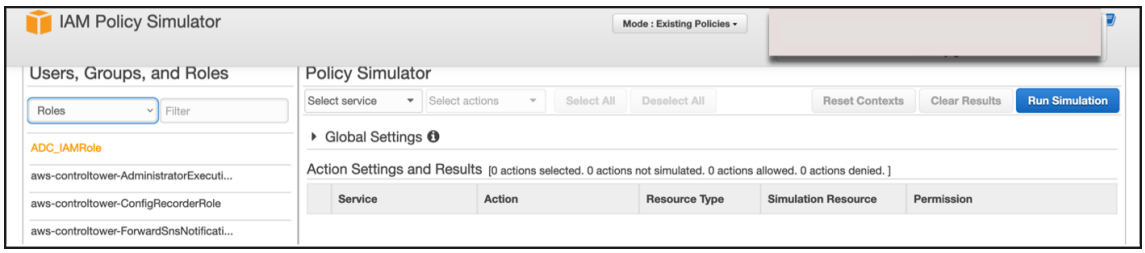
1. **IAM** ページで、テストする IAM ロールを選択し、「**Simulate**」をクリックします。次の例では、「ADC_IAMRole」が IAM ロールです。



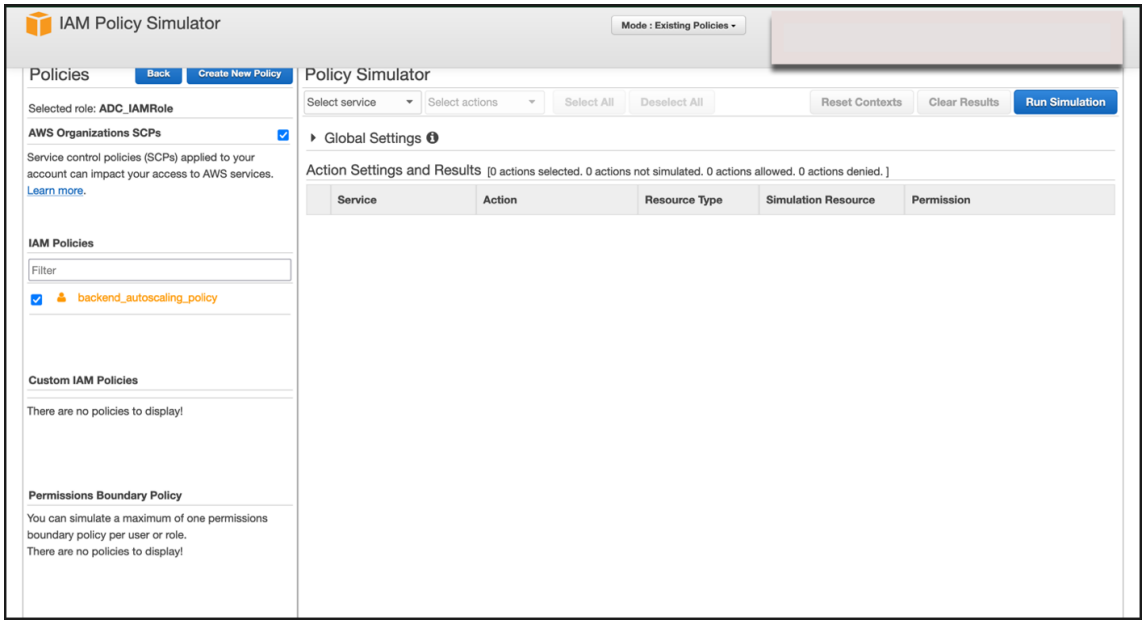
2. **IAM** ポリシーシミュレーターコンソールで、「モード」として「既存のポリシー」を選択します。



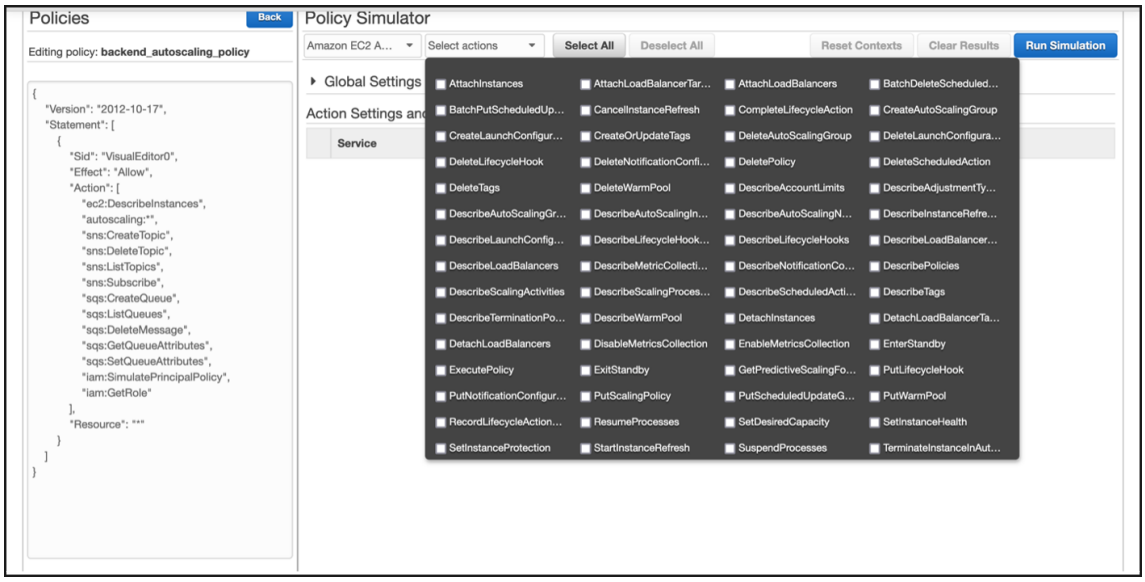
3. [ユーザー、グループ、ロール] タブで、ドロップダウンメニューから [ロール] を選択し、既存のロールを選択します。



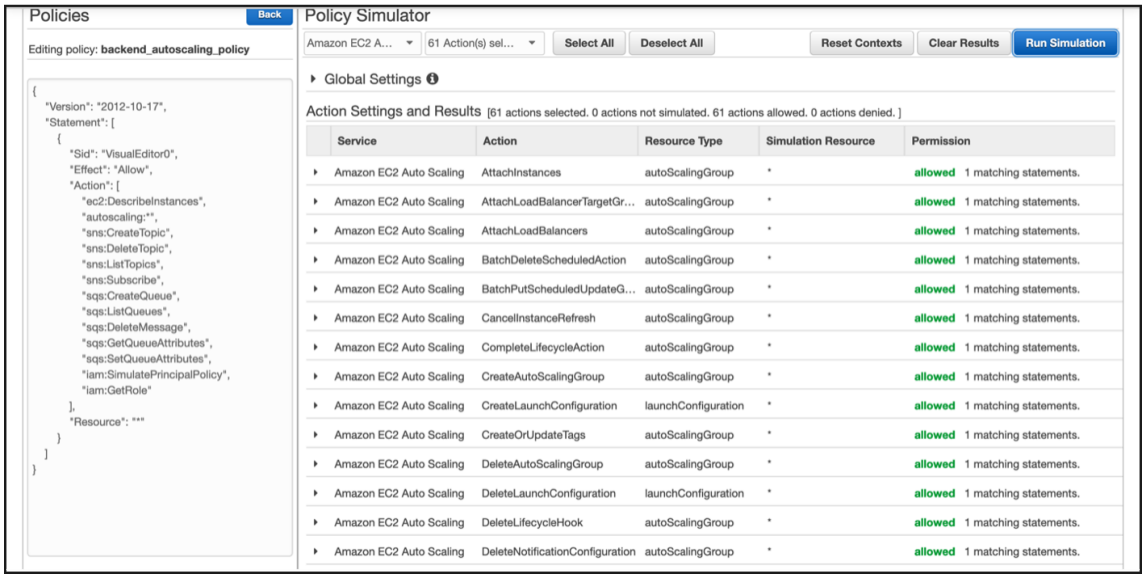
4. 既存のロールを選択したら、その下にある既存のポリシーを選択します。



5. ポリシーを選択すると、画面の左側に正確な JSON が表示されます。[アクションの選択] ドロップダウンメニューで目的のアクションを選択します。



6. [シミュレーションを実行] をクリックします。



詳細については、AWS IAM ドキュメントを参照してください。

その他の参考資料

IAM ロールを使用して Amazon EC2 インスタンスで実行されているアプリケーションにアクセス権限を付与する

AWS 上の NetScaler VPX インスタンスの仕組み

October 17, 2024

NetScaler VPX インスタンスは AWS マーケットプレイスで AMI として入手でき、AWS VPC 内で EC2 インスタンスとして起動することもできます。NetScaler VPX AMI インスタンスには、少なくとも 2 つの仮想 CPU と 2 GB のメモリが必要です。また、AWS VPC 内で起動される EC2 インスタンスは、複数のインターフェイス、インターフェイスごとに複数の IP アドレス、VPX 構成に必要なパブリックおよびプライベート IP アドレスも提供できます。各 VPX インスタンスには、少なくとも 3 つの IP サブネットが必要です。

- 管理サブネット
- クライアント側サブネット（VIP）
- バックエンド向けサブネット (SNIP、MIP など)

AWS での標準の VPX インスタンスのインストールには、3 つのネットワークインターフェイスをお勧めします。

現在、AWS では、AWS VPC 内で実行しているインスタンスでのみ、マルチ IP 機能を使用できます。VPC 内の VPX インスタンスを使用して、EC2 インスタンスで実行しているサーバーの負荷を分散できます。Amazon VPC を使用すれば、独自の IP アドレス範囲、サブネット、ルートテーブル、ネットワークゲートウェイなどを含めて、仮想ネットワーク環境を作成および管理できます。

> 注
>
> デフォルトでは、各 AWS アカウントの AWS リージョンごとに最大 5 つの VPC インスタンスを作成できます。Amazon のリクエストフォームを送信することで、より高い VPC 制限をリクエストhttp://aws.amazon.com/contact-us/vpc-requestできます。

図 **1.** AWS アーキテクチャ上の NetScaler VPX インスタンスのサンプル展開

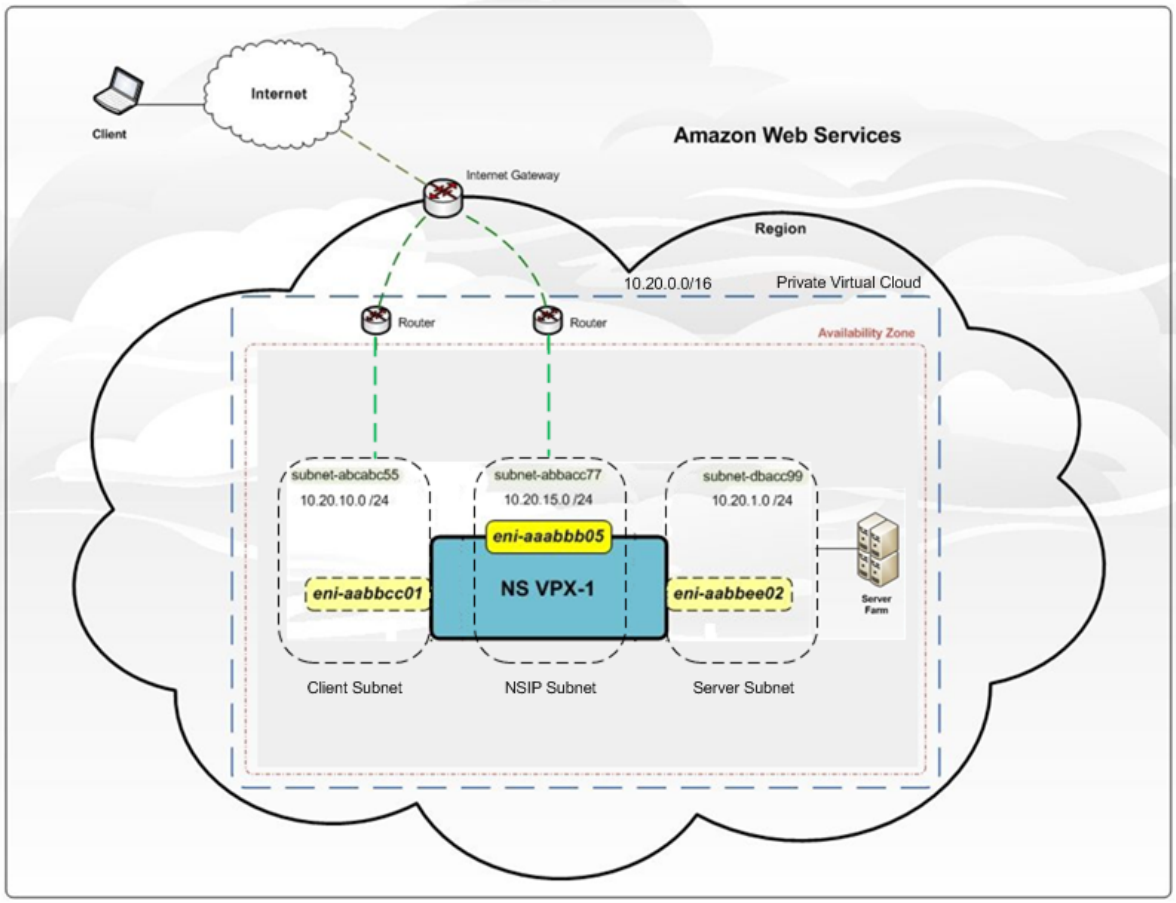


図 1 は、AWS VPC のシンプルなトポロジを示しています。NetScaler VPX の展開。AWS VPC は、以下の要素で構成されています。

1. VPC からの送受信トラフィックをルーティングするための単一のインターネットゲートウェイ。
2. インターネットゲートウェイとインターネット間のネットワーク接続。
3. 3 つのサブネット（管理、クライアント、サーバー用に 1 つずつ）。
4. インターネットゲートウェイと 2 つのサブネット（管理用とクライアント用）間のネットワーク接続。
5. VPC 内にデプロイされたスタンドアロンの NetScaler VPX インスタンス。VPX インスタンスには、各サブネットに 1 つずつ接続された ENI が 3 つあります。

NetScaler VPX スタンドアロンインスタンスを AWS にデプロイする

October 17, 2024

NetScaler VPX スタンドアロンインスタンスは、次のオプションを使用して AWS にデプロイできます。

- AWS ウェブコンソール
- Citrix が作成した CloudFormation テンプレート
- AWS CLI

このトピックでは、NetScaler VPX インスタンスを AWS にデプロイする手順について説明します。

展開を開始する前に、以下のトピックをお読みください。

- 前提条件
- 制限事項と使用上のガイドライン

AWS ウェブコンソールを使用して NetScaler VPX インスタンスを AWS にデプロイします

AWS Web コンソールを使用して、AWS で NetScaler VPX インスタンスを展開できます。展開のプロセスには、次の手順が含まれます。

1. キーペアの作成
2. 仮想プライベートクラウド（VPC）の作成
3. サブネットをさらに追加する
4. セキュリティグループとセキュリティルールの作成
5. ルートテーブルの追加
6. インターネットゲートウェイを作成する
7. NetScaler VPX インスタンスを作成する
8. ネットワークインターフェースをさらに作成してアタッチする
9. エラスティック IP の管理 NIC へのアタッチ
10. VPX インスタンスに接続する

ステップ **1:** キーペアを作成します。

Amazon EC2 は、キーペアを使用してログオン情報を暗号化および復号します。インスタンスにログオンするには、キーペアを作成し、インスタンスを起動するときにキーペアの名前を指定し、インスタンスに接続するときにプライベートキーを指定する必要があります。

AWS Launch Instance ウィザードを使用してインスタンスを確認し、起動すると、既存のキーペアを使用するか、新しいキーペアを作成するように求められます。キーペアの作成方法の詳細については、「Amazon EC2 キーペア」を参照してください。

ステップ **2: VPC** を作成します。

NetScaler VPC インスタンスは AWS VPC 内で展開されます。VPC では、AWS アカウント専用の仮想ネットワークを定義できます。AWS VPC の詳細については、「Amazon VPC の使用開始」を参照してください。

NetScaler VPX インスタンスに対する VPC の作成中は、次の点に留意してください。

- AWS アベイラビリティーゾーンに AWS VPC を作成するには、単一のパブリックサブネットのみのオプションで VPC を使用します。
- Citrix では、以下のタイプのサブネットを少なくとも **3** つ作成することをお勧めします。
  - 管理トラフィック用の 1 つのサブネット。このサブネットに管理 IP (NSIP) を配置します。デフォルトでは、エラスティックネットワークインターフェース (ENI) eth0 が管理 IP に使用されます。
  - クライアントアクセス（ユーザーから NetScaler ADC VPX）トラフィック用の 1 つ以上のサブネット。クライアントが NetScaler ADC 負荷分散仮想サーバーに割り当てられた 1 つ以上の仮想 IP（VIP）アドレスに接続します。
  - サーバーアクセス（VPX からサーバーへ）トラフィック用の 1 つ以上のサブネット。サーバーはこのサブネットを介して VPX 所有のサブネット IP（SNIP）アドレスに接続します。NetScaler 負荷分散と仮想サーバー、仮想 IP アドレス（VIP）、サブネット IP アドレス（SNIP）の詳細については、以下を参照してください。
  - すべてのサブネットは、同じアベイラビリティーゾーンに存在する必要があります。

ステップ **3:** サブネットを追加します。

VPC ウィザードを使った場合、作成されたサブネットは 1 つのみです。要件に応じて、さらにサブネットを作成することもできます。サブネットをさらに作成する方法について詳しくは、「VPC へのサブネットの追加」を参照してください。

ステップ **4:** セキュリティグループとセキュリティルールを作成します。

受信トラフィックと送信トラフィックを制御するには、セキュリティグループを作成し、そのグループに規則を追加します。グループを作成してルールを追加する方法の詳細については、「VPC のセキュリティグループ」を参照してください。

NetScaler VPX インスタンスの場合、EC2 ウィザードはデフォルトのセキュリティグループを提供します。このセキュリティグループは、AWS マーケットプレイスによって生成され、Citrix が推奨する設定に基づいています。ただし、要件に応じてさらにセキュリティグループを作成できます。

> 注
>
> ポート 22、80、443 をセキュリティグループでそれぞれ SSH、HTTP、HTTPS アクセス用に開きます。

ステップ **5:** ルートテーブルを追加します。

ルートテーブルには、ネットワークトラフィックの経路を判断する際に使用される、ルートと呼ばれる一連のルールが含まれます。VPC の各サブネットはルートテーブルに関連付ける必要があります。ルートテーブルの作成方法について詳しくは、「ルートテーブル」を参照してください。

ステップ **6**: インターネット **Gateway** を作成します。

インターネットゲートウェイには 2 つの目的があります。1 つは、インターネットでルーティング可能なトラフィックのターゲットを VPC ルートテーブルに提供すること、もう 1 つはパブリック IPv4 アドレスが割り当てられたインスタンスに対してネットワークアドレス変換 (NAT) を実行することです。

インターネットトラフィックに対して、インターネットゲートウェイを作成します。インターネットゲートウェイの作成方法の詳細については、「インターネットゲートウェイをアタッチする」を参照してください。

ステップ **7**: **AWS EC2** サービスを使用して **NetScaler ADC VPX** インスタンスを作成します。

AWS EC2 サービスを使って NetScaler VPX インスタンスを作成するには、次の手順に従います。

1. AWS ダッシュボードから、**[**コンピューティング**] > [EC2] > [**インスタンスの起動**] > [AWS** マーケットプレイス] に移動します。

   Launch Instance をクリックする前に、**Launch Instance** の下に表示される注記を確認して、リージョンが正しいことを確認してください。

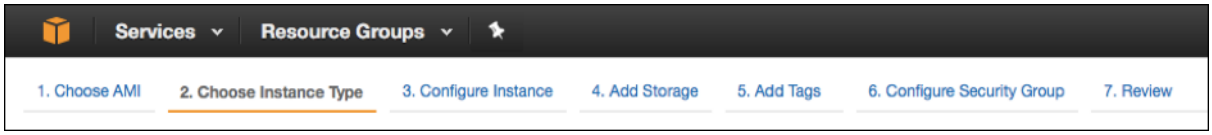
   

2. ［Search AWS Marketplace］バーで、「NetScaler VPX」と入力して検索します。
3. 展開するバージョンを選択し、[**Select**] をクリックします。NetScaler VPX バージョンでは、次のオプションがあります。

   - ライセンスバージョン
   - NetScaler VPX Express アプライアンス（これは無料の仮想アプライアンスで、NetScaler 12.0 56.20 から入手できます。）
   - 自分のデバイスを持参

Launch Instance ウィザードが起動します。ウィザードに従って、インスタンスを作成します。このウィザードでは、次のことを求められます。

- インスタンスの種類の選択
- インスタンスの構成
- ストレージの追加
- タグの追加
- セキュリティグループの構成
- レビュー



ステップ **8:** ネットワークインターフェースをさらに作成してアタッチします。

VIP と SNIP 用に 2 つのネットワークインターフェイスを作成します。ネットワークインターフェイスの作成方法の詳細については、「ネットワークインターフェイスの作成 」を参照してください。

ネットワークインターフェイスを作成したら、VPX インスタンスにアタッチする必要があります。インターフェイスを接続する前に、VPX インスタンスをシャットダウンし、インターフェイスを接続し、インスタンスの電源をオンにします。ネットワークインターフェイスの接続方法の詳細については、「インスタンスの起動時にネットワークインターフェイスをアタッチする 」セクションを参照してください。

ステップ **9: Elastic IP** を割り当てて関連付けます。

EC2 インスタンスにパブリック IP アドレスを割り当てた場合、そのアドレスはインスタンスが停止されるまで割り当てられたままになります。その後、アドレスはプールに解放されます。インスタンスを再起動すると、新しいパブリック IP アドレスが割り当てられます。

対照的に、Elastic IP (EIP) アドレスは、インスタンスとの関連付けが解除されるまで割り当てられたままになります。

管理 NIC のエラスティック IP を割り当てて、関連付けます。Elastic IP アドレスを割り当てて関連付ける方法の詳細については、以下のトピックを参照してください。

- Elastic IP アドレスの割り当て
- Elastic IP アドレスを実行中のインスタンスに関連付ける

これらのステップで、AWS に NetScaler VPX インスタンスを作成する手順が完了します。インスタンスの準備が完了するまで数分かかる場合があります。インスタンスがステータスチェックに合格したことを確認します。この情報は、「インスタンス」ページの「ステータスチェック」列で確認できます。

ステップ **10: VPX** インスタンスに接続します。

VPX インスタンスを作成したら、GUI と SSH クライアントを使用してインスタンスを接続します。

- GUI

NetScaler VPX インスタンスにアクセスするためのデフォルト管理者の資格情報は以下のとおりです。

ユーザー名: `nsroot`

パスワード:ns root アカウントのデフォルトパスワードは、NetScaler VPX インスタンスの AWS インスタンス ID に設定されます。最初のログオン時に、セキュリティ上の理由からパスワードを変更するように求められます。パスワードを変更した後、構成を保存する必要があります。構成が保存されずにインスタンスが再起動する場合は、デフォルトのパスワードでログオンする必要があります。プロンプトでパスワードを再度変更します。

- SSH クライアント

AWS マネジメントコンソールから、**NetScaler VPX** インスタンスを選択して **[接続]** をクリックします。「インスタンスへの接続」ページの指示に従ってください。

AWS Web コンソールを使用して AWS に NetScaler VPX スタンドアロンインスタンスを展開する方法の詳細については、「シナリオ: スタンドアロンインスタンス」を参照してください。

Citrix の **CloudFormation** テンプレートを使用して **NetScaler VPX** インスタンスを構成する

Citrix が提供する CloudFormation テンプレートを使用して、VPX インスタンスの起動を自動化できます。このテンプレートには、単一の NetScaler VPX インスタンスを起動したり、NetScaler VPX インスタンスのペアを使用して高可用性環境を作成したりする機能があります。

テンプレートは AWS Marketplace または GitHub から起動できます。

CloudFormation テンプレートには既存の VPC 環境が必要で、3 つのエラスティックネットワークインターフェース（ENI）を備えた VPX インスタンスを起動します。CloudFormation テンプレートを開始する前に、以下の要件を満たしていることを確認してください。

- AWS 仮想プライベートクラウド（VPC）
- VPC 内の 3 つのサブネット (1 つは管理用、1 つはクライアントトラフィック用、もう 1 つはバックエンドサーバー用)
- インスタンスへの SSH アクセスを有効にする EC2 キーペア
- UDP 3003、TCP 3009—3010、HTTP、SSH ポートが開いているセキュリティグループ

前提条件を満たす方法の詳細については、「AWS ウェブコンソールを使用して AWS に NetScaler ADC VPX インスタンスをデプロイする」セクションまたは AWS のドキュメントを参照してください。

このビデオでは 、AWS Marketplace で利用可能な Citrix CloudFormation テンプレートを使用して、NetScaler VPX スタンドアロンインスタンスを構成して起動する方法について説明します。

https://github.com/citrix/citrix-adc-aws-cloudformation/tree/master/templates/standalone/

IAM ロールはスタンドアロンデプロイでは必須ではありません。ただし、Citrix では、将来の必要に備えて、必要な権限を持つ IAM ロールを作成してインスタンスにアタッチすることを推奨しています。IAM ロールにより、スタンドアロンインスタンスは、必要に応じて SR-IOV を使用して高可用性ノードに簡単に変換されます。

必要な権限の詳細については、「SR-IOV ネットワーク インターフェイスを使用するための NetScaler VPX インスタンスの構成」を参照してください。

> 注
>
> AWS ウェブコンソールを使用して AWS に NetScaler VPX インスタンスをデプロイすると、CloudWatch サービスはデフォルトで有効になります。Citrix CloudFormation テンプレートを使用して NetScaler VPX インスタンスを展開する場合、デフォルトのオプションは「はい」です。CloudWatch サービスを無効にする場

> 合は、「いいえ」を選択します。詳細については、「Amazon CloudWatch を使用してインスタンスを監視する」を参照してください。

AWS CLI を使用して **NetScaler ADC VPX** インスタンスを構成する

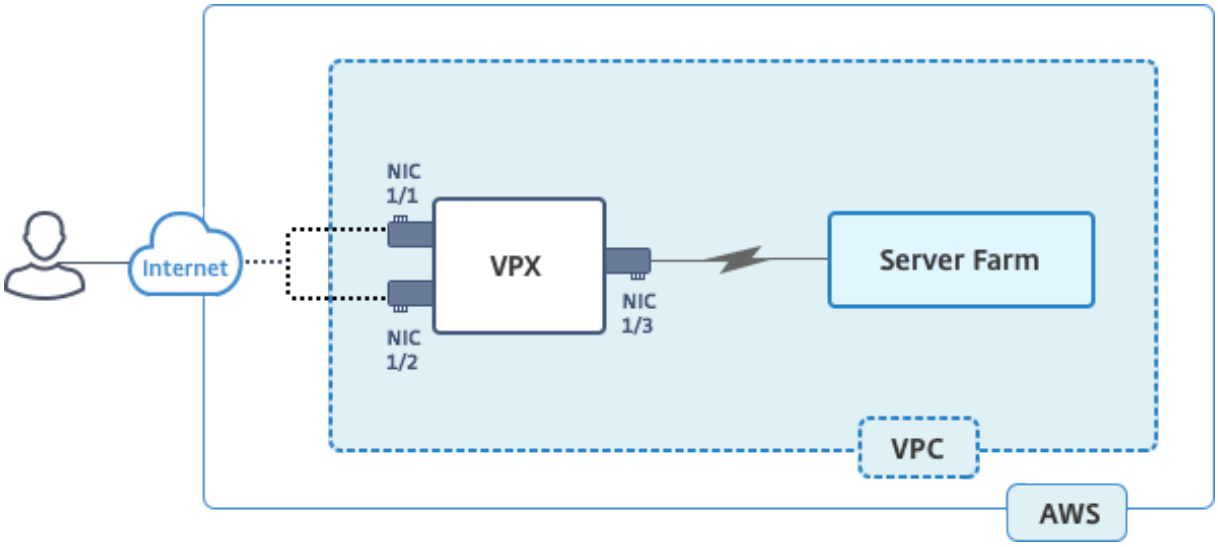
AWS CLI を使用してインスタンスを起動できます。詳細については、AWS コマンドラインインターフェイスのドキュメントを参照してください。

シナリオ: スタンドアロンインスタンス

October 17, 2024

このシナリオでは、AWS GUI を使用して NetScaler VPX スタンドアロン EC2 インスタンスを AWS にデプロイする方法を示しています。3 つの NIC を持つスタンドアロン VPX インスタンスを作成します。インスタンスは、負荷分散仮想サーバーとして構成されており、バックエンドサーバー（サーバーファーム）と通信します。この設定では、インスタンスとバックエンドサーバー間、およびパブリックインターネット上のインスタンスと外部ホスト間の必要な通信ルートを設定します。

VPX インスタンスを展開する手順の詳細については、「AWS に NetScaler VPX スタンドアロンインスタンスを展開する」を参照してください。

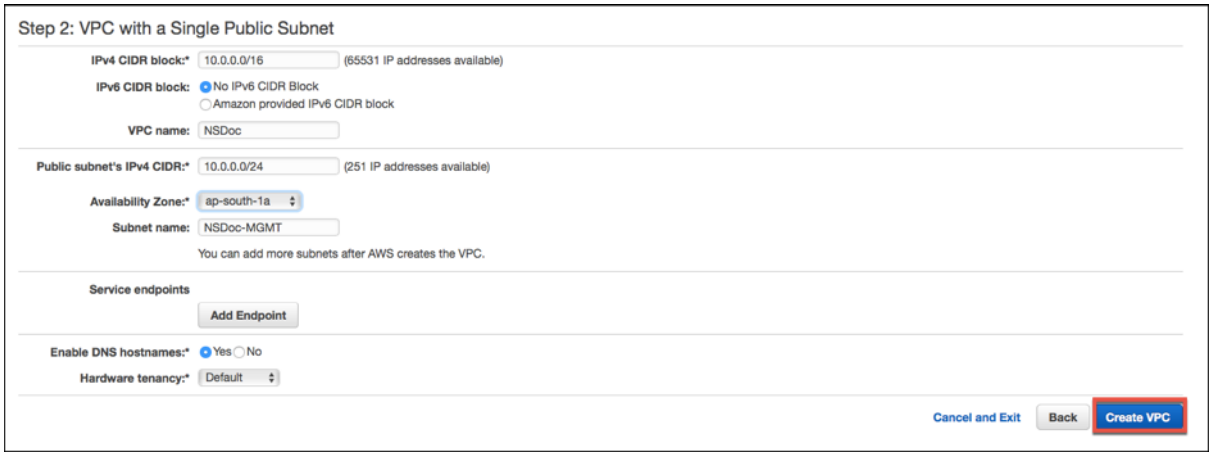


3 つの NIC を作成します。各 NIC は、IP アドレスのペア（パブリックとプライベート）を使用して構成できます。NIC は、次の目的に役立ちます。

NIC	目的	関連付けられている
eth0	NSIP（管理トラフィックを処理する）	パブリック IP アドレスとプライベート IP アドレス
eth1	クライアント側のトラフィック（VIP）をサービスする	パブリック IP アドレスとプライベート IP アドレス
eth2	バックエンド・サーバ（SNIP）との通信	パブリック IP アドレス (プライベート IP アドレスは必須ではありません)

ステップ 1: VPC を作成します。

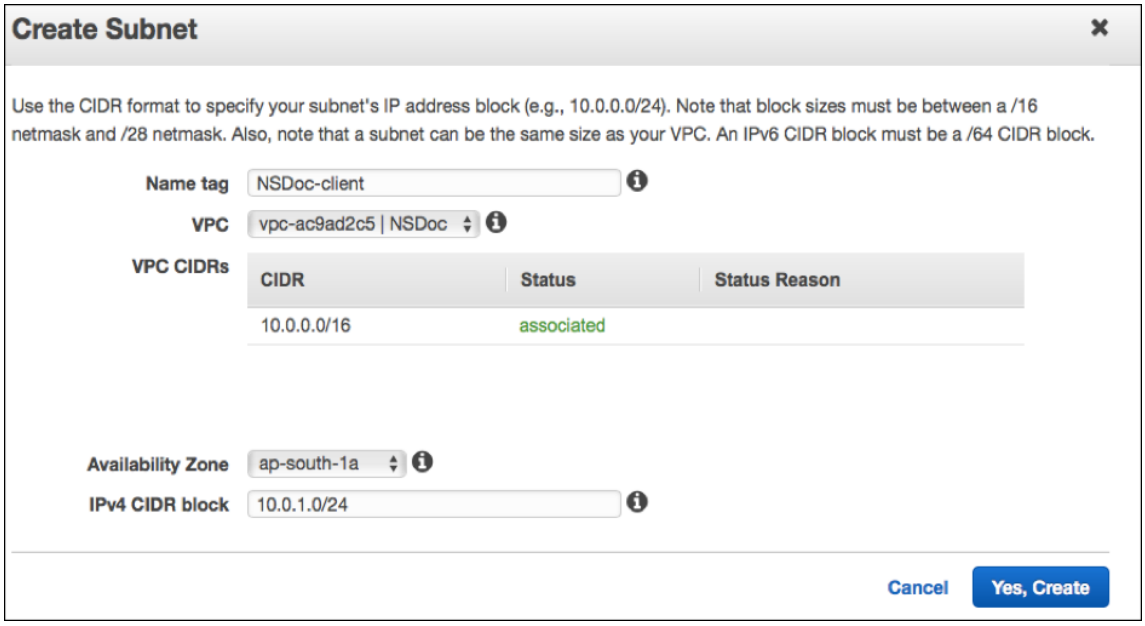
1. AWS ウェブコンソールにログオンし、[ネットワークとコンテンツ配信] > [**VPC**] に移動します。[**VPC** ウィザードの開始**]** をクリックします。
2. 単一のパブリックサブネットを持つ **VPC** を選択し、[**Select**] をクリックします。
3. このシナリオでは、IP CIDR ブロックを 10.0.0.0/16 に設定します。
4. VPC の名前を指定します。
5. パブリックサブネットを 10.0.0.0/24 に設定します。（これは管理ネットワークです）。
6. アベイラビリティ ゾーンを選択してください。
7. サブネットの名前を付けます。
8. [**VPC** の作成] をクリックします。



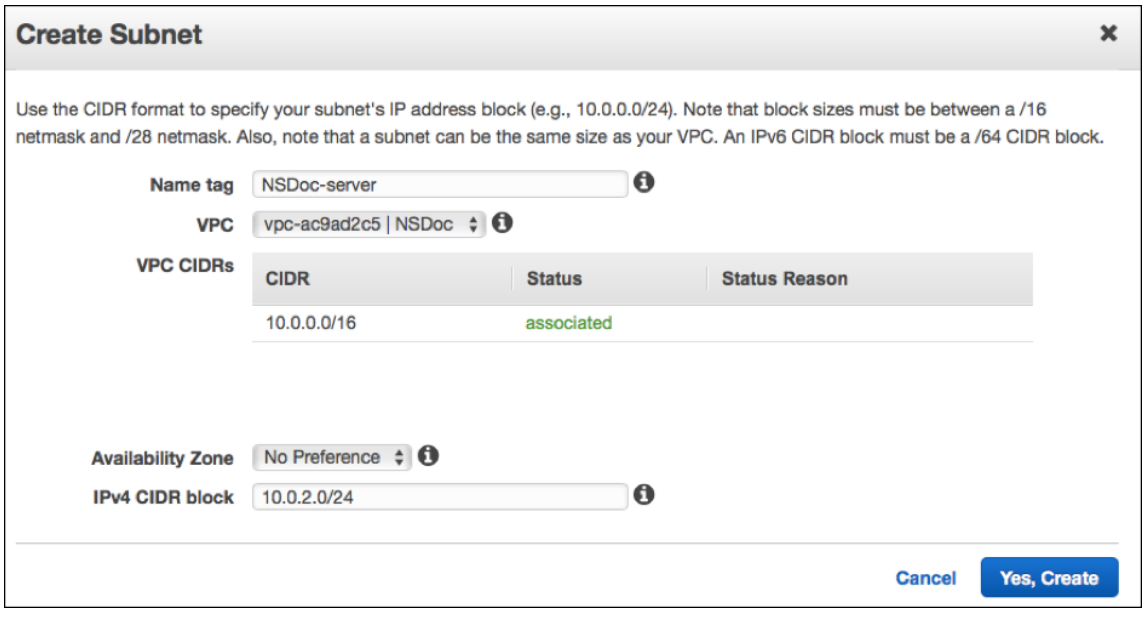
ステップ 2: 追加のサブネットを作成します。

1. https://console.aws.amazon.com/vpc/で Amazon VPC コンソールを開きます。
2. ナビゲーションペインで、次の詳細を入力した後、[Subnets]、[Create Subnet] の順に選択します。
   - 名前タグ: サブネットの名前を指定します。

- VPC: サブネットを作成する VPC を選択します。
- アベイラビリティーゾーン: ステップ 1 で VPC を作成したアベイラビリティーゾーンを選択します。
- IPv4 CIDR ブロック: サブネットの IPv4 CIDR ブロックを指定します。このシナリオでは、10.0.1.0/24 を選択します。



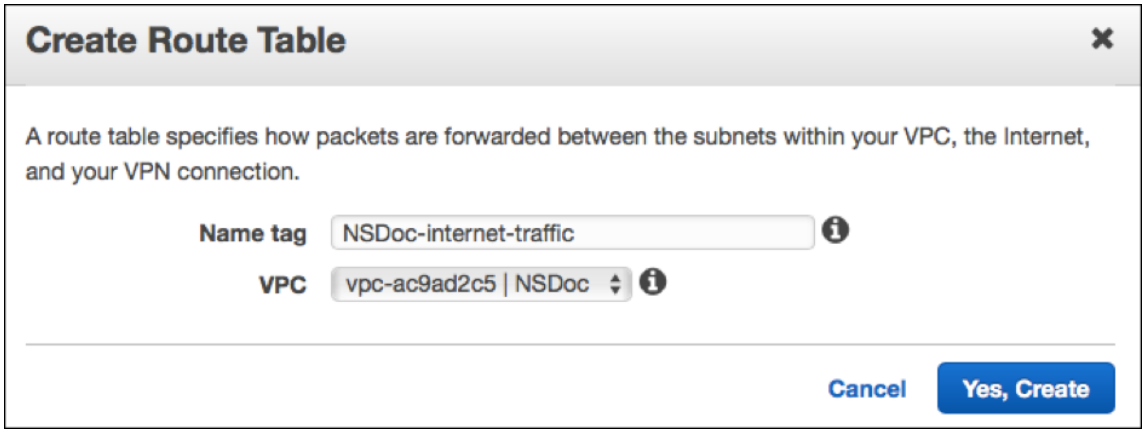
3. この手順を繰り返して、バックエンドサーバー用のサブネットをもう 1 つ作成します。



ステップ 3: ルートテーブルを作成します。

1. https://console.aws.amazon.com/vpc/で Amazon VPC コンソールを開きます。
2. ナビゲーションペインで、「ルートテーブル」>「ルートテーブル ** を作成」を選択します。**
3. [Create Route Table] ウィンドウで、名前を追加し、ステップ 1 で作成した VPC を選択します。

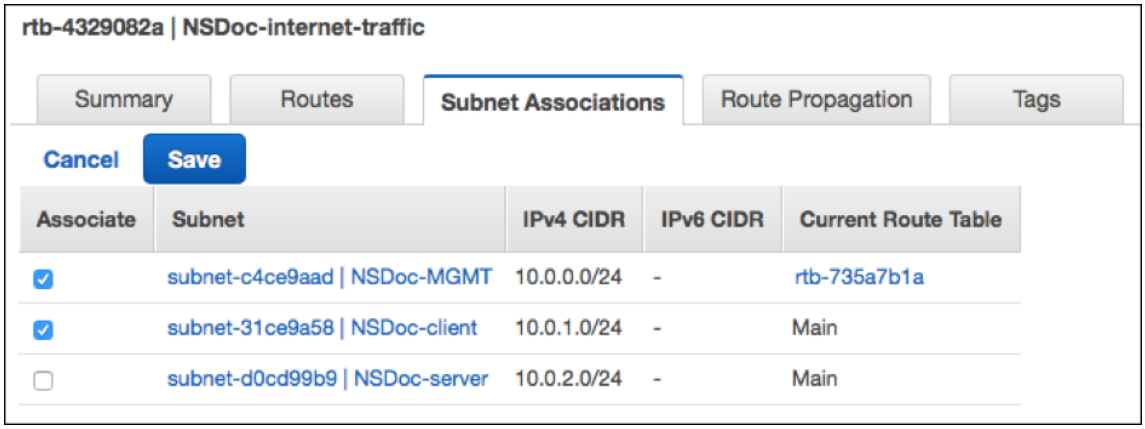
4. [**Yes, Create**] をクリックします。



ルートテーブルは、この VPC 用に作成したすべてのサブネットに割り当てられます。これにより、あるサブネット内のインスタンスからのトラフィックのルーティングが別のサブネットのインスタンスに到達できるようになります。

5. サブネットの関連付けをクリックし、次に 編集をクリックします。

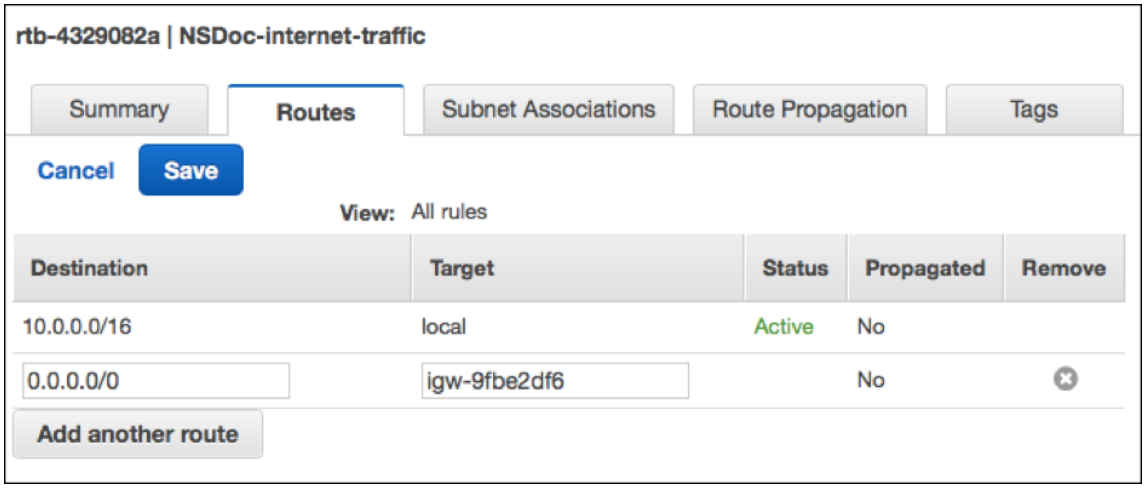
6. 管理サブネットとクライアントサブネットをクリックし、[Save] をクリックします。これにより、インターネットトラフィック専用のルートテーブルが作成されます。



7. **[**ルート**] > [**編集**] > [**別のルートを追加**]** をクリックします。

8. [Destination] フィールドに 0.0.0.0/0 を追加し、[Target] フィールドをクリックして [igw] (<xxxx> VPC ウィザードによって自動的に作成されたインターネットゲートウェイ) を選択します。

9. [保存] をクリックします。

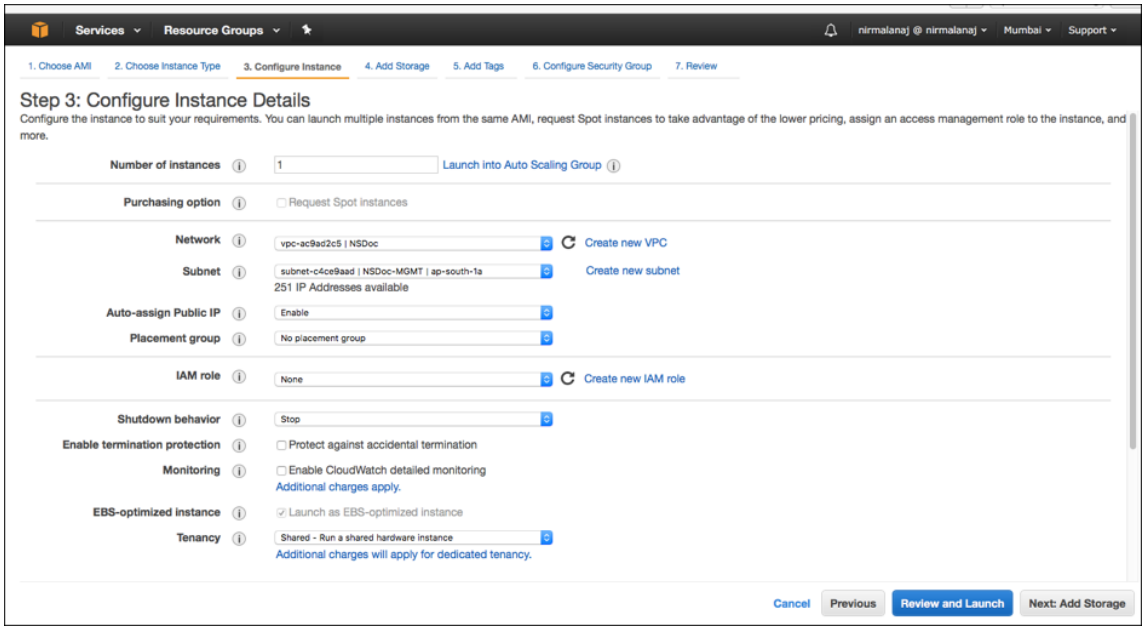


10. サーバー側のトラフィックのルートテーブルを作成する手順に従います。

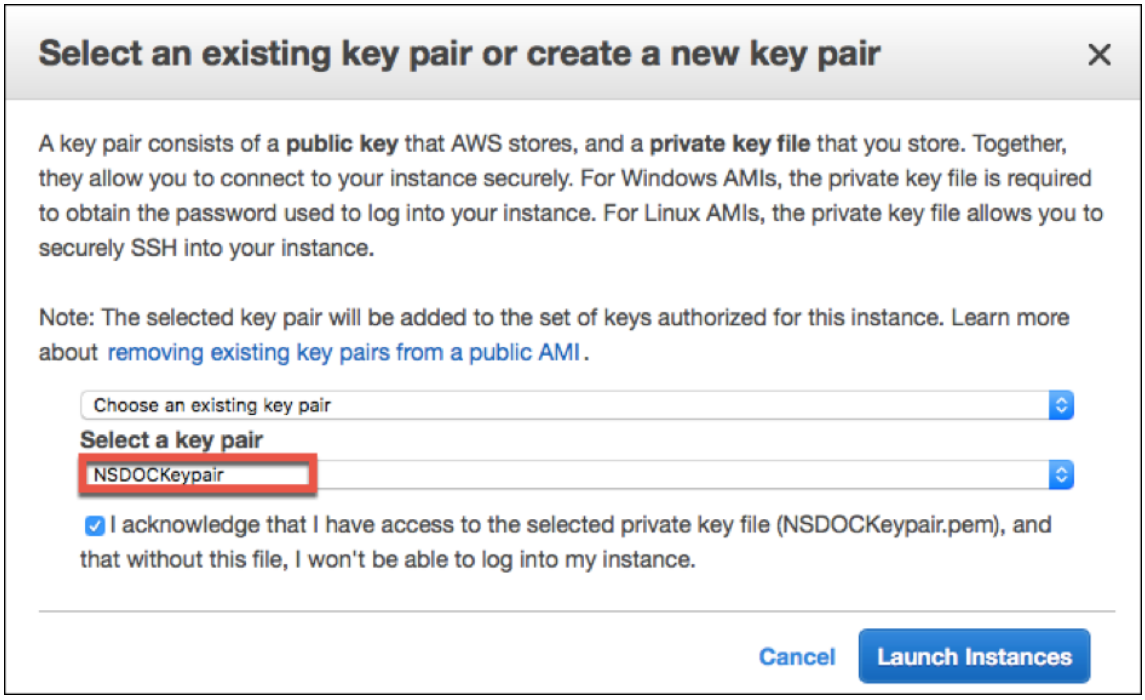
ステップ 4: NetScaler VPX インスタンスを作成します。

1. AWS マネジメントコンソールにログオンし、[**Compute**] の下の [**EC2**] をクリックします。
2. [AWS マーケットプレイス] をクリックします。AWS Marketplace 検索バーに「NetScaler VPX」と入力し、Enter キーを押します。使用可能な NetScaler VPX エディションが表示されます。
3. 「選択」をクリックして、目的の NetScaler VPX エディションを選択します。EC2 インスタンスウィザードが起動します。
4. [インスタンスタイプの選択**]** ページで、[**m4]** を選択します。**Xlarge** (推奨) をクリックし、[次へ**:** インスタンスの詳細を設定] をクリックします。
5. 「インスタンスの詳細の構成」ページで、以下を選択し、「次へ**:** ストレージの追加」をクリックします。

    - インスタンス数:1
    - ネットワーク: ステップ 1 で作成した VPC
    - サブネット: 管理サブネット
    - パブリック IP の自動割り当て: 有効



6. [ストレージの追加] ページで、デフォルトのオプションを選択し、[次へ**:** タグの追加] をクリックします。
7. 「タグの追加」ページでインスタンスの名前を追加し、「次へ**:** セキュリティ グループの構成」をクリックします。
8. [セキュリティグループの設定] ページで、デフォルトのオプション（AWS Marketplace によって生成され、Citrix Systems の推奨設定に基づいています）を選択し、[レビューして起動] をクリックします > 起動します。
9. 既存のキーペアを選択するか、新しいキーペアを作成して新しいキーペアを選択するように求められます。[Select a key pair] ドロップダウンリストから、前提条件として作成したキーペアを選択します（「前提条件」セクションを参照）。
10. キーペアを確認するにはチェックボックスをオンにして、[インスタンスの起動] をクリックします。



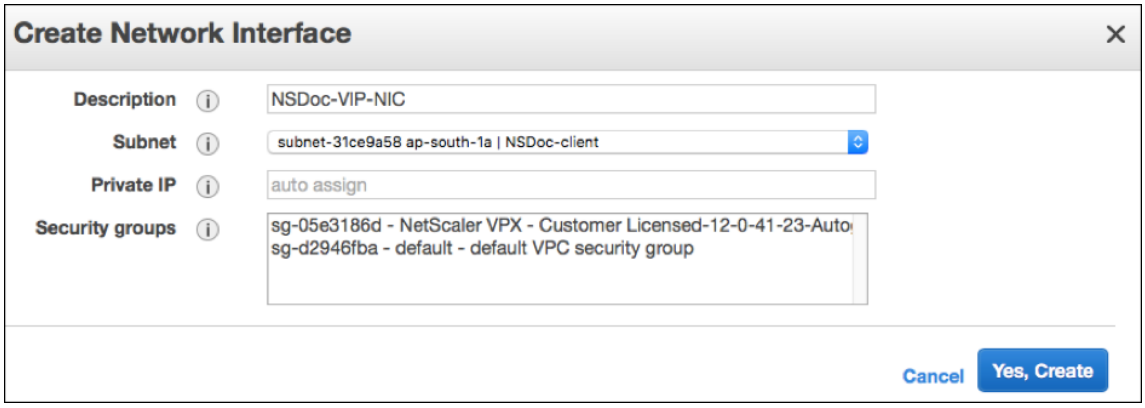
[Launch Instance Wizard] に [Launch Status] が表示され、インスタンスが完全に起動されるとインスタンスのリストに表示されます。

インスタンスを確認するには、AWS コンソールに移動し、**EC2 >** 実行中のインスタンスをクリックします。インスタンスを選択し、名前を追加します。インスタンスの状態が実行中で、ステータスチェックが完了していることを確認します。

ステップ 5: より多くのネットワークインターフェイスを作成し、アタッチします。

VPC を作成したとき、それに関連付けられたネットワークインターフェイスは 1 つだけです。ここで、VIP と SNIP 用に、VPC にさらに 2 つのネットワーク インターフェイスを追加します。

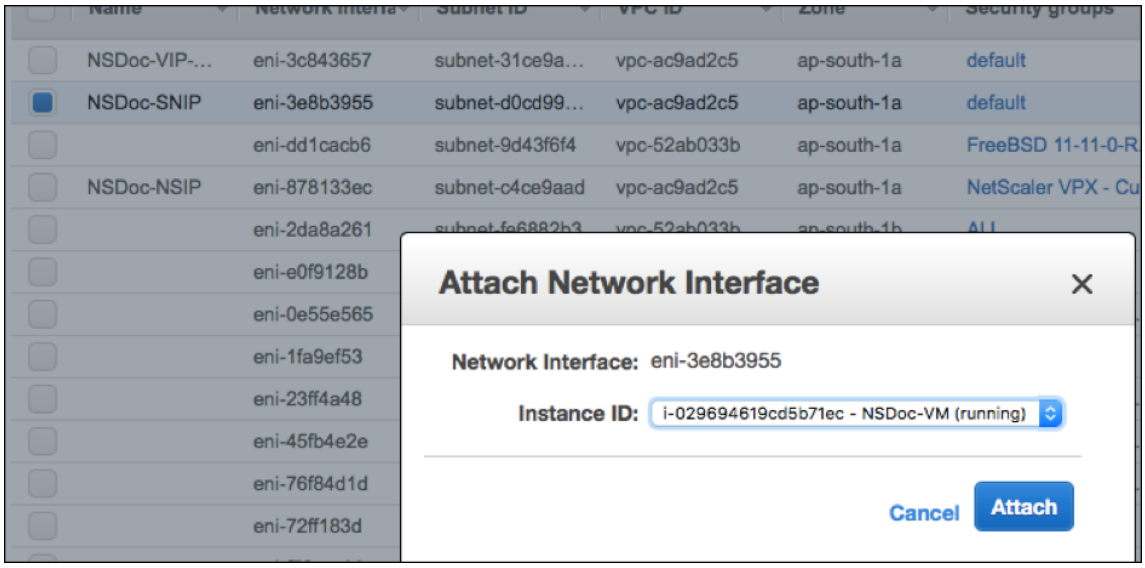
1. https://console.aws.amazon.com/ec2/で Amazon EC2 コンソールを開きます。
2. ナビゲーションペインで、[ネットワークインターフェイス] を選択します。
3. 「ネットワークインタフェースの作成」を選択します。
4. 説明には、わかりやすい名前を入力します。
5. サブネットには、以前に VIP 用に作成したサブネットを選択します。
6. プライベート **IP** については、デフォルトオプションのままにします。
7. セキュリティ グループの場合は、グループを選択します。
8. [**Yes, Create**] をクリックします。



9. ネットワークインターフェースが作成されたら、インターフェースに名前を追加します。
10. この手順を繰り返して、サーバー側のトラフィック用のネットワークインターフェイスを作成します。

ネットワークインターフェイスをアタッチします。

1. ナビゲーションペインで、[ネットワークインターフェイス] を選択します。
2. ネットワークインターフェースを選択し、アタッチをクリックします。
3. [ネットワーク インターフェイスの接続] ダイアログ ボックスでインスタンスを選択し、[接続] をクリックします。



ステップ 6: エラスティック IP を NSIP に接続します。

1. AWS マネジメントコンソールから、[ネットワークとセキュリティ] **> [Elastic IP**] に移動します。
2. アタッチする無料の EIP がないか確認してください。存在しない場合は、[新しいアドレスの割り当て] をクリックします。
3. 新しく割り当てられた IP アドレスを選択し、[アクション] **> [**アドレスの関連付け**]** を選択します。

4. [ネットワークインターフェイス] オプションボタンをクリックします。
5. [Network Interface] ドロップダウンリストから、管理 NIC を選択します。
6. [プライベート **IP**] ドロップダウンメニューから、AWS によって生成された IP アドレスを選択します。
7. [再関連付け] チェックボックスをオンにします。
8. [関連付け] をクリックします。

VPX インスタンスにアクセスします。

スタンドアロンの NetScaler VPX インスタンスを 3 つの NIC で構成したら、VPX インスタンスにログオンして NetScaler 側の構成を完了します。次のオプションを使用します。

- GUI: ブラウザに管理 NIC のパブリック IP を入力します。ユーザー名として `nsroot` を使用し、パスワードとしてインスタンス ID（i-0c1ffe1d987817522）を使用してログオンします。

> 注
>
> 最初のログオン時に、セキュリティ上の理由からパスワードを変更するように求められます。パスワードを変更した後、構成を保存する必要があります。構成が保存されずにインスタンスが再起動する場合は、デフォルトのパスワードでログオンする必要があります。プロンプトでパスワードを再度変更し、構成を保存します。

- SSH: SSH クライアントを開き、次のように入力します。

```
ssh -i \\&#060;location of your private key\\&#062; ns root@\\&#060;
public DNS of the instance\\&#062;
```

パブリック DNS を見つけるには、インスタンスをクリックし、[接続] をクリックします。

関連情報:

- Citrix ADC が所有する IP アドレス（NSIP、VIP、および SNIP）を構成するには、Citrix ADC 所有の IP アドレスの構成を参照してください。
- NetScaler VPX アプライアンスの BYOL バージョンを構成しました。詳細については、VPX ライセンスガイド（http://support.citrix.com/article/CTX122426）

NetScaler VPX ライセンスをダウンロードする

October 17, 2024

AWS マーケットプレイスから NetScaler ADC VPX-カスタマーライセンスインスタンスを起動した後、ライセンスが必要です。VPX ライセンスの詳細については、ライセンスの概要を参照してください。

次の操作を実行する必要があります:

1. Citrix Web サイト内のライセンスポータルを使用して、有効なライセンスを生成します。
2. ライセンスをインスタンスにアップロードします。

有料 マーケットプレイスインスタンスの場合は、ライセンスをインストールする必要はありません。正しい機能セットとパフォーマンスが自動的にアクティブ化されます。

モデル番号が VPX5000 より大きい NetScaler VPX インスタンスを使う場合は、ネットワークのスループットはインスタンスのライセンスに規定されているのと同じではないことがあります。ただし、SSL スループットや 1 秒あたりの SSL トランザクションといった他の機能は改善されている場合があります。

`c4.8xlarge`インスタンスタイプでは 5 Gbps のネットワーク帯域幅が観測されます。

AWS サブスクリプションを BYOL に移行する方法

このセクションでは、AWS サブスクリプションから独自のライセンス (BYOL) に移行する手順、およびその逆について説明します。

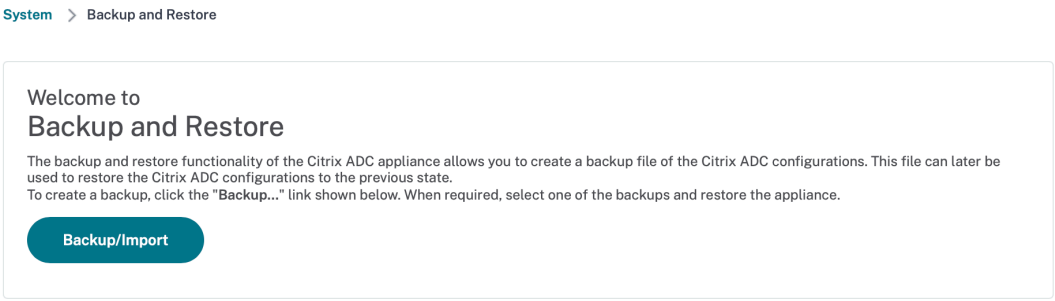
AWS サブスクリプションを BYOL に移行するには、次の手順を実行します。

> 注
>
> ステップ **2** と ステップ **3** は NetScaler VPX インスタンスで実行され、その他のすべての手順は AWS ポータルで実行されます。

1. NetScaler VPX-同じセキュリティグループ、IAM ロール、サブネットを持つ古い EC2 インスタンスと同じアベイラビリティーゾーンで、カスタマーライセンスを使用して BYOL EC2 インスタンスを作成します。新しい EC2 インスタンスには ENI インターフェイスが 1 つだけ必要です。
2. NetScaler GUI を使用して古い EC2 インスタンスのデータをバックアップするには、次の手順に従います。

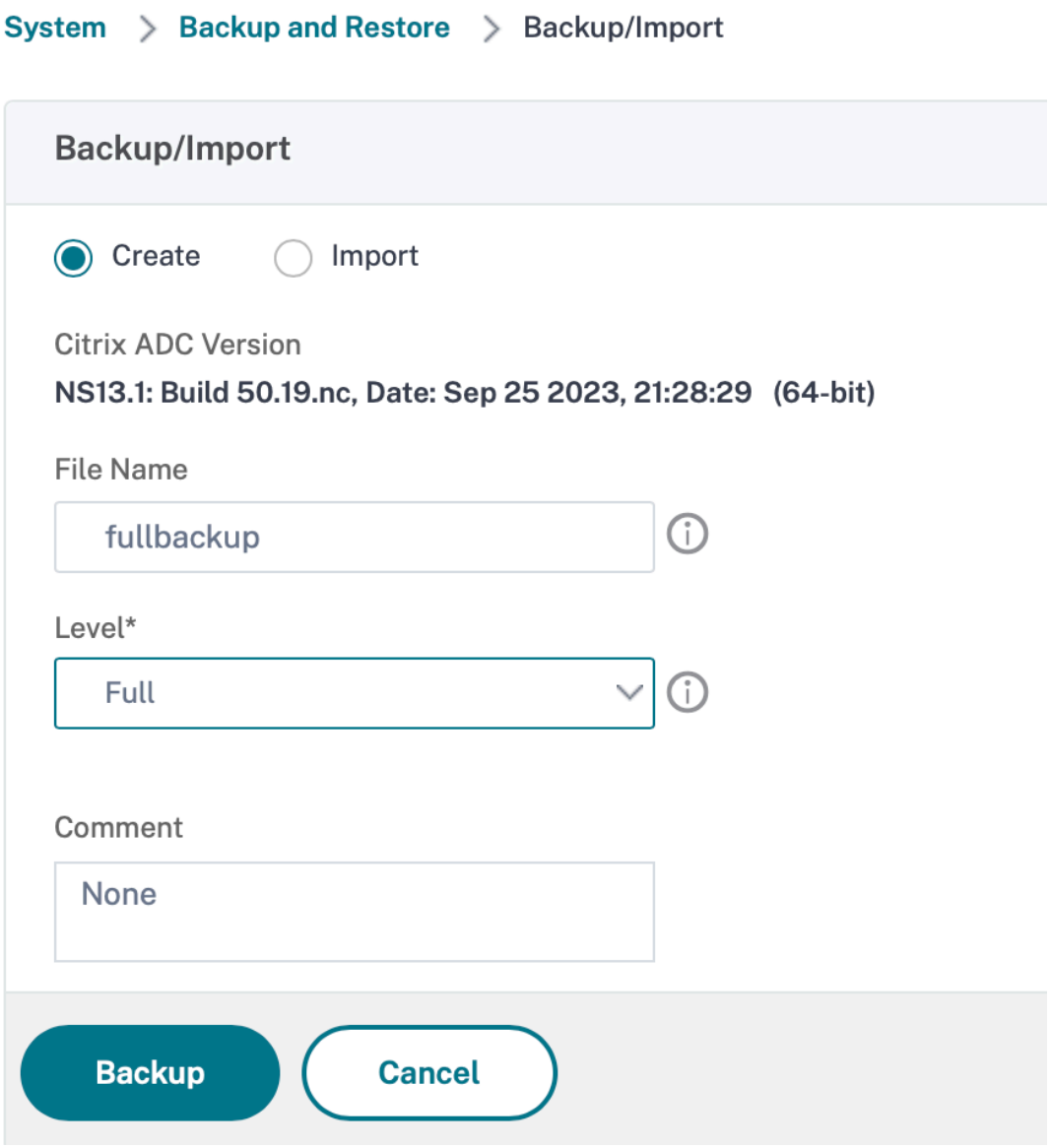
   a) [システム] **>** [バックアップと復元] に移動します。

   b) [ようこそ] ページで、[バックアップ/インポート] をクリックしてプロセスを開始します。

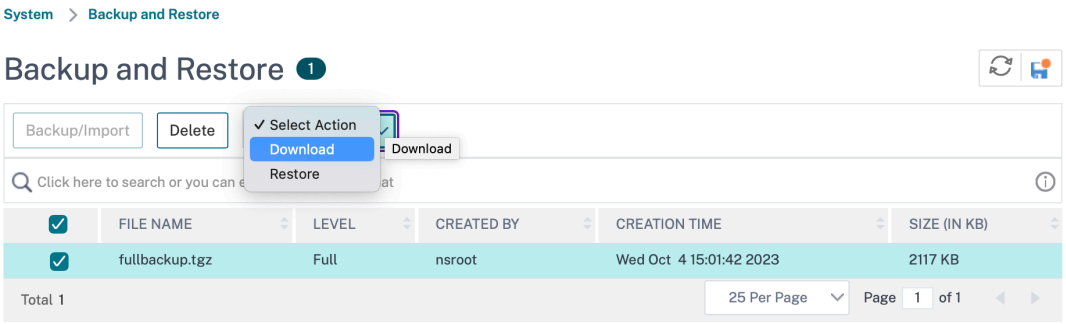


c) [バックアップ/インポート] ページで、次の詳細を入力します。

- **Name**： バックアップファイルの名前。
- **Level**： バックアップレベルを「フル」として選択します。
- [コメント]: バックアップの簡単な説明を入力します。



d) [バックアップ] をクリックします。バックアップが完了したら、ファイルを選択してローカルマシンにダウンロードできます。

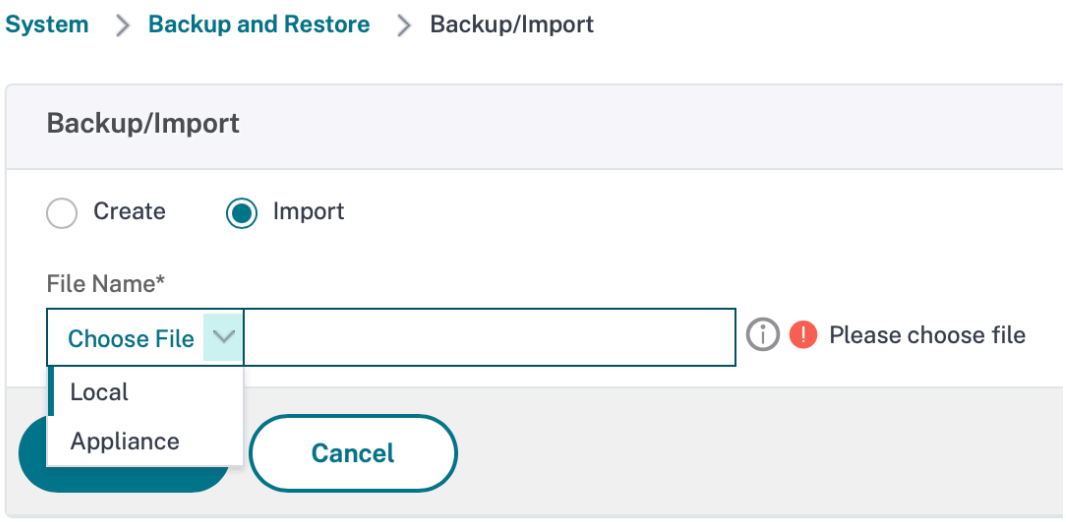


3. NetScaler GUI を使用して新しい EC2 インスタンスにデータを復元するには、次の手順に従います。

a) [システム] **>** [バックアップと復元] に移動します。

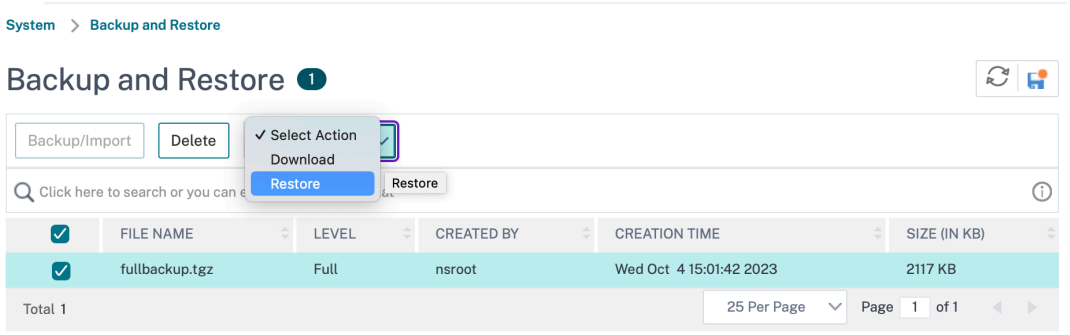
b) [バックアップ/インポート] をクリックして、プロセスを開始します。

c) [インポート] オプションを選択し、バックアップファイルをアップロードします。

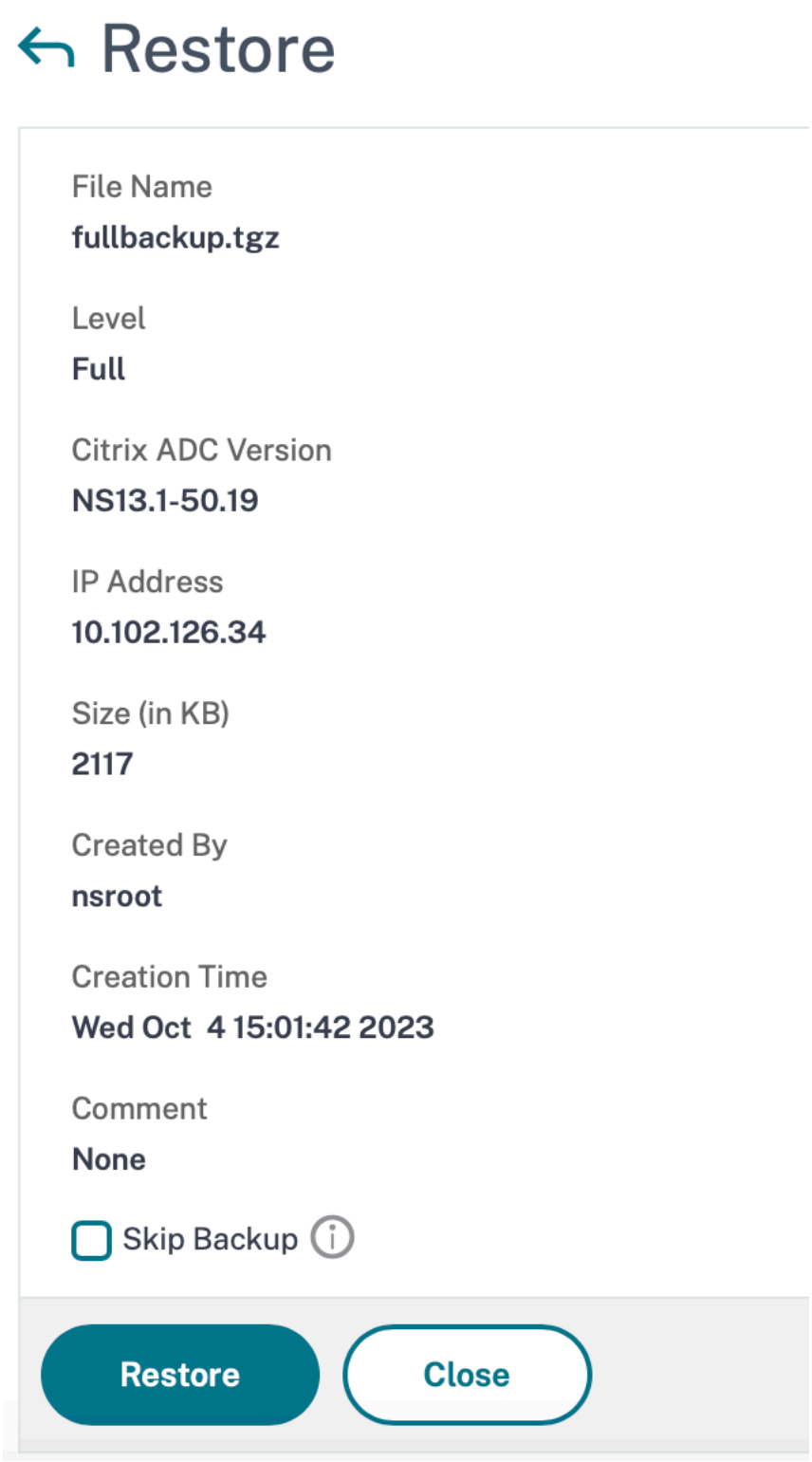


d) ファイルを選択します。

e) [アクションの選択] ドロップダウンメニューから、[復元] を選択します。



f) [復元] ページで、ファイルの詳細を確認し、[復元] をクリックします。



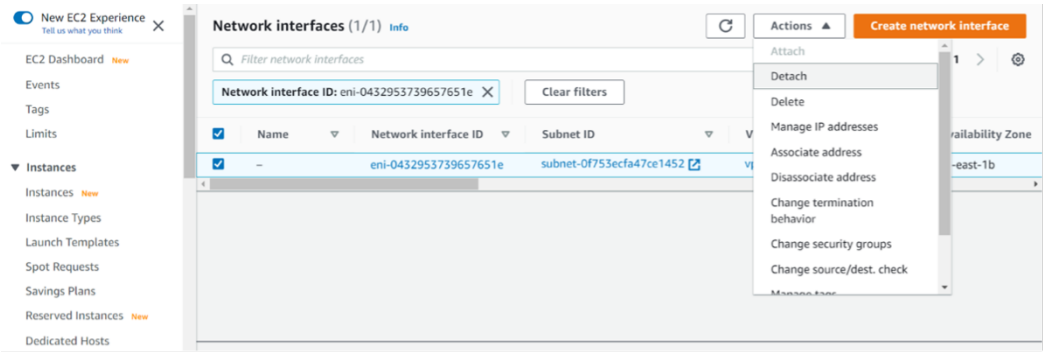
g) 復元後、EC2 インスタンスを再起動します。

4. 古い EC2 インスタンスから新しい EC2 インスタンスに、すべてのインターフェイス（NSIP アドレスがバインドされている管理インターフェイスを除く）を移動します。ネットワークインターフェイスを別の EC2 インスタンスに移動するには、次の手順を実行します。

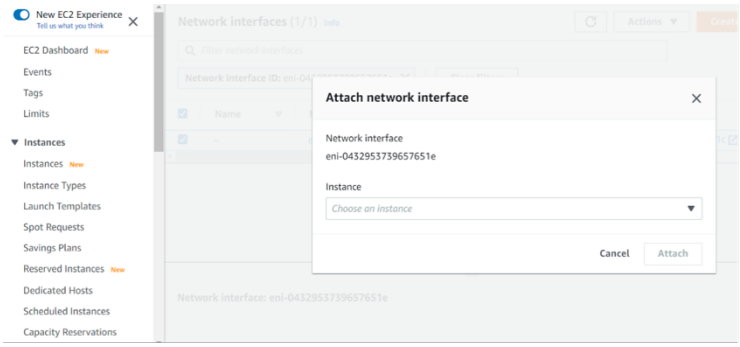
   a) **AWS** ポータルで、古い EC2 インスタンスと新しい EC2 インスタンスの両方を停止します。

   b) [ネットワークインターフェイス] に移動し、古い EC2 インスタンスにアタッチされたネットワークインターフェイスを選択します。

   c) [アクション] **> [**デタッチ**]** をクリックして **EC2** インスタンスをデタッチします。



   d) [アクション] **> [Attach]** の順にクリックして、ネットワークインターフェイスを新しい **EC2** インスタンスにアタッチします。ネットワークインターフェイスをアタッチする必要がある EC2 インスタンス名を入力します。



   e) 接続されている他のすべてのインターフェイスについて、ステップ **1** からステップ **4** を実行します。シーケンスに従い、インターフェイスの順序を維持するようにしてください。つまり、まずインターフェイス 2 をデタッチして接続し、次にインターフェイス 3 をデタッチして接続します。

5. 古い EC2 インスタンスから管理インターフェイスをデタッチすることはできません。したがって、古い EC2 インスタンスの管理インターフェイス（プライマリネットワークインターフェイス）上のすべてのセカンダリ IP アドレス（存在する場合）を新しい EC2 インスタンスに移動します。IP アドレスをあるインターフェイスから別のインターフェイスに移動するには、次の手順を実行します。

   a) **AWS** ポータルで、古い EC2 インスタンスと新しい EC2 インスタンスの両方が **Stop** 状態であることを確認します。

b) [ネットワークインターフェイス] に移動し、古い EC2 インスタンスにアタッチされた管理ネットワークインターフェイスを選択します。

c) [アクション**] > [IP** アドレスの管理] の順にクリックし、割り当てられているすべてのセカンダリ IP アドレス（存在する場合）を書き留めます。

d) 新しい EC2 インスタンスの管理ネットワークインターフェイスまたはプライマリインターフェイスに移動します。

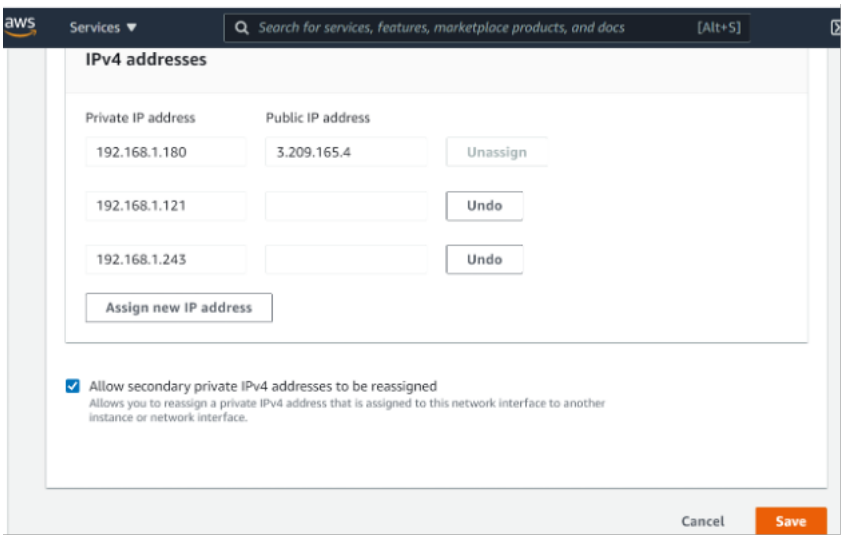
e) [アクション**] > [IP** アドレスの管理] の順にクリックします。

f) [**IPv4** アドレス] で、[新しい **IP** アドレスを割り当て] をクリックします。

g) ステップ **3** で説明する IP アドレスを入力します。

h) [セカンダリプライベート **IP** アドレスの再割り当てを許可する] チェックボックスをオンにします。

i) [保存] をクリックします。



6. 新しい EC2 インスタンスを起動し、設定を確認します。すべての設定が移動されたら、要件に従って古い EC2 インスタンスを削除または保持できます。
7. 古い EC2 インスタンスの NSIP アドレスに EIP アドレスがアタッチされている場合は、古いインスタンスの NSIP アドレスを新しいインスタンスの NSIP アドレスに移動します。
8. 古いインスタンスに戻す場合は、古いインスタンスと新しいインスタンスの逆の方法で同じ手順を実行します。
9. サブスクリプションインスタンスから BYOL インスタンスに移行した後、ライセンスが必要です。ライセンスをインストールするには、次の手順に従います。

   - Citrix Web サイトのライセンスポータルを使用して、有効なライセンスを生成します。
   - ライセンスをインスタンスにアップロードします。

> 注
>
> BYOL インスタンスをサブスクリプションインスタンス (有料マーケットプレイスインスタンス) に移動する場合、ライセンスをインストールする必要はありません。正しい機能セットとパフォーマンスが自動的にアクティブ化されます。

制限事項

管理インターフェイスを新しい EC2 インスタンスに移動することはできません。したがって、管理インターフェイスを手動で構成することをお勧めします。詳細については、前の手順の手順 **5** を参照してください。新しい EC2 インスタンスは、古い EC2 インスタンスの正確なレプリカで作成されますが、新しい IP アドレスは NSIP アドレスだけです。

異なる可用性ゾーンでの負荷分散サーバー

October 17, 2024

VPX インスタンスを使用して、同じアベイラビリティーゾーンまたは以下の場所で稼働しているサーバーの負荷分散を行うことができます。

- 同じ AWS VPC 内の異なるアベイラビリティーゾーン (AZ)
- 別の AWS リージョン
- VPC 内の AWS EC2

VPX インスタンスが AWS VPC 外で稼働するサーバーの負荷を分散できるようにするには、VPX インスタンスが存在する場合は、次のように、EIP を使用してインターネット ゲートウェイ経由でトラフィックをルーティングするようにインスタンスを構成します。

1. NetScaler CLI または GUI を使用して、NetScaler VPX インスタンスで SNIP を構成します。
2. サーバー側のトラフィック用にパブリックサブネットを作成することで、トラフィックが AZ からルーティングされるようにします。
3. AWS GUI コンソールを使用して、インターネット Gateway ルートをルーティングテーブルに追加します。
4. 更新したルーティングテーブルをサーバー側のサブネットに関連付けます。
5. NetScaler SNIP アドレスにマップされているサーバー側のプライベート IP アドレスに EIP を関連付けます。

AWS での高可用性の機能

October 17, 2024

AWS 上の 2 つの NetScaler ADC VPX インスタンスを高可用性（HA）アクティブ/パッシブペアとして構成できます。1 つのインスタンスをプライマリノードとして構成し、もう 1 つをセカンダリノードとして設定すると、プライマリノードは接続を受け入れ、サーバーを管理します。セカンダリノードはプライマリを監視します。何らかの理由で 1 次ノードが接続を受け入れることができない場合は、2 次ノードが引き継ぎます。

AWS では、VPX インスタンスで次のデプロイタイプがサポートされています。

- 同一ゾーン内での高可用性
- 異なるゾーン間の高可用性

> 注
>
> 高可用性を機能させるには、両方の NetScaler ADC VPX インスタンスに IAM ロールがアタッチされ、Elastic IP（EIP）アドレスが NSIP に割り当てられていることを確認してください。NSIP が NAT インスタンスを介してインターネットにアクセスできる場合は、NSIP に EIP を割り当てる必要はありません。

同じゾーン内の高可用性

同じゾーン内の高可用性展開では、両方の VPX インスタンスのネットワーク構成が類似している必要があります。

次の 2 つのルールに従います。

ルール 1. 1 つの VPX インスタンスの NIC は、他の VPX の対応する NIC と同じサブネットにある必要があります。どちらのインスタンスにも次のものが必要です。

- 同じサブネット (管理サブネットと呼ばれる) 上の管理インターフェイス
- 同じサブネット (クライアントサブネットと呼ばれる) 上のクライアントインターフェイス
- 同じサブネット (サーバーサブネットと呼ばれる) 上のサーバーインターフェイス

ルール 2。両方のインスタンスの管理 NIC、クライアント NIC、およびサーバ NIC のシーケンスが同じである必要があります。たとえば、次のシナリオはサポートされていません。たとえば、次のシナリオはサポートされていません。

VPX インスタンス 1

NIC 0: 管理 NIC 1: クライアント NIC 2: サーバー

VPX インスタンス 2

NIC 0: 管理

NIC 1: サーバ

NIC 2: クライアント

このシナリオでは、インスタンス 1 の NIC 1 はクライアントサブネットにあり、インスタンス 2 の NIC 1 はサーバーサブネットにあります。HA が機能するには、両方のインスタンスの NIC 1 がクライアントサブネットまたはサーバーサブネット内にある必要があります。

13.0 41.xx から、フェールオーバー後にプライマリ HA ノードの NIC（クライアント側およびサーバ側の NIC）に接続されたセカンダリプライベート IP アドレスをセカンダリの HA ノードに移行することで、高可用性を実現できます。この展開は、以下のように管理されます。

- 両方の VPX インスタンスは、NIC 列挙に従って NIC の数とサブネットマッピングが同じです。
- 各 VPX NIC には、管理 IP アドレスに対応する最初の NIC を除き、追加のプライベート IP アドレスが 1 つあります。追加のプライベート IP アドレスは、AWS ウェブコンソールでプライマリプライベート IP アドレスとして表示されます。このドキュメントでは、この余分な IP アドレスをダミー IP アドレスと呼んでいます)。
- ダミー IP アドレスは、NetScaler インスタンスで VIP および SNIP として構成しないでください。
- 必要に応じて、その他のセカンダリプライベート IP アドレスを作成し、VIP および SNIP として設定する必要があります。
- フェールオーバー時に、新しいプライマリノードは設定された SNIP および VIP を検索し、前のプライマリに接続されている NIC から新しいプライマリ上の対応する NIC に移動します。
- NetScaler インスタンスでは、HA が機能するためには IAM アクセス許可が必要です。各インスタンスに追加された IAM ポリシーに次の IAM 権限を追加します。

`"iam:GetRole"` 「`ec2:インスタンスの説明`」 「`ec2:ネットワークインターフェースの説明`」 「`ec2:プライベートIPアドレスの割り当て`」

> 注
>
> `unassignPrivateIpAddress` は必要ありません。

この方法は従来の方法よりも高速です。古い方法では、HA はプライマリノードの AWS Elastic ネットワークインターフェイスからセカンダリノードへの移行に依存します。

従来の方法では、次のポリシーが必要です。

`"iam:GetRole"` 「`ec2:インスタンスの説明`」 「`ec2:アドレスの説明`」 `"ec2:アソシエイトアドレス"` 「`ec2:アドレスの関連付けを解除`」

詳細については、「AWS に高可用性ペアをデプロイする」を参照してください。

異なるゾーン間の高可用性

独立ネットワーク構成（INC）モードでは、2 つの異なるサブネットまたは 2 つの異なる AWS アベイラビリティーゾーンに 2 つの NetScaler ADC VPX インスタンスを高可用性アクティブ/パッシブのペアとして構成できます。フェイルオーバー時に、プライマリインスタンスの VIP の EIP (Elastic IP) がセカンダリに移行し、セカンダリが新しいプライマリとして引き継がれます。フェイルオーバープロセスでは、AWS API は以下を実行します。

- `IPSets`が接続されている仮想サーバーをチェックします。

- 仮想サーバーがリッスンしている 2 つの IP アドレスから、パブリック IP が関連付けられている IP アドレスを検索します。1 つは仮想サーバに直接接続され、もう 1 つは IP セットを介して接続されます。
- パブリック IP (EIP) を、新しいプライマリ VIP に属するプライベート IP に再関連付けします。

異なるゾーン間の HA には、次のポリシーが必要です。

`"iam:GetRole"` 「`ec2:インスタンスの説明`」 「`ec2:アドレスの説明`」 `"ec2:アソシエイトアドレス"` 「`ec2:アドレスの関連付けを解除`」

詳細については、「AWS アベイラビリティーゾーン全体の高可用性」を参照してください。

展開を開始する前に

AWS で HA のデプロイを開始する前に、次のドキュメントをお読みください。

- 前提条件
- 制限事項と使用ガイドライン
- AWS で NetScaler ADC VPX インスタンスを展開する
- 高可用性

トラブルシューティング

AWS クラウド上の NetScaler ADC VPX インスタンスの HA フェイルオーバー中の障害をトラブルシューティングするには、`/var/log/`の場所に保存されている`cloud-ha-daemon.log`ファイルを確認してください。

同じ **AWS** 可用性ゾーンに **VPX HA** ペアを展開する

October 17, 2024

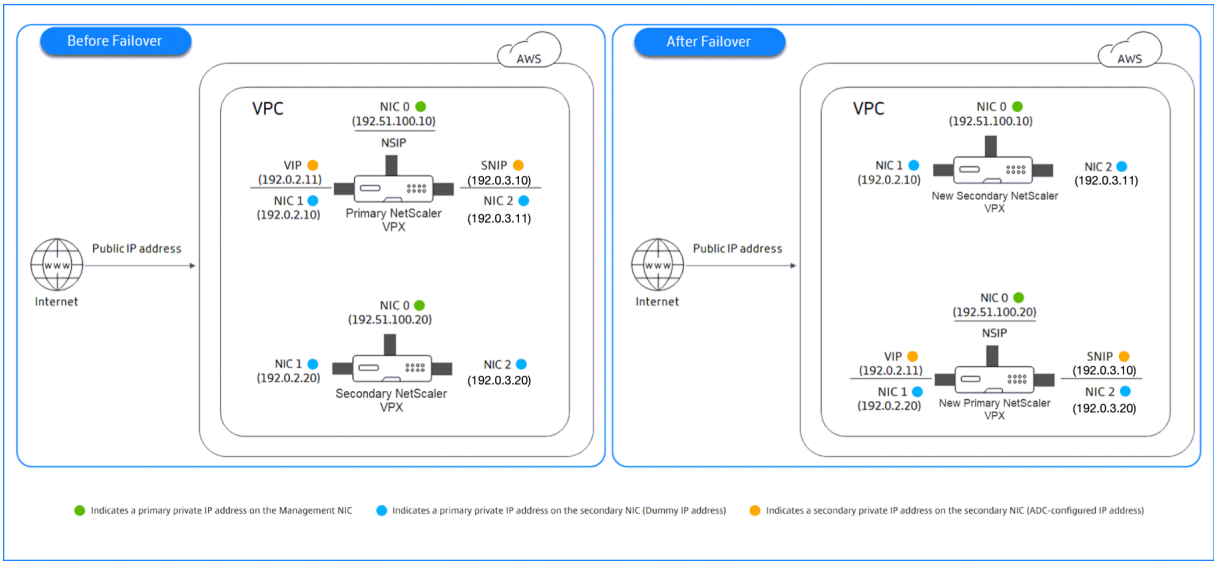
> 注
>
> NetScaler リリース 13.1 ビルド 27.x 以降、同じ AWS アベイラビリティーゾーン内の VPX HA ペアは IPv6 アドレスをサポートします。

両方の VPX インスタンスが同じサブネット上にある同じ AWS ゾーンで、AWS 上の 2 つの NetScaler VPX インスタンスを HA ペアとして構成できます。HA は、フェイルオーバー後に、プライマリ HA ノードの NIC（クライアント側およびサーバ側 NIC）に接続されているセカンダリプライベート IP アドレスをセカンダリ HA ノードに移行することで実現されます。セカンダリプライベート IP アドレスに関連付けられているすべての Elastic IP アドレスも移行されます。

NetScaler VPX HA ペアは、同じ AWS アベイラビリティーゾーンで IPv4 アドレスと IPv6 アドレスの両方をサポートします。

次の図は、セカンダリプライベート IP アドレスを移行する HA フェールオーバーのシナリオを示しています。

図 1：インライン展開 図 1：プライベート IP マイグレーションを使用した、AWS 上の NetScaler VPX HA ペア



ドキュメントを開始する前に、次のドキュメントをお読みください。

- 前提条件
- 制限事項と使用ガイドライン
- AWS で NetScaler ADC VPX インスタンスを展開する
- 高可用性

VPX HA ペアを同じゾーンにデプロイする方法

VPX HA ペアを同じゾーンにデプロイする手順の概要を次に示します。

1. 手順 1：同じ VPC を使用して、それぞれ 3 つの NIC（イーサネット 0、イーサネット 1、イーサネット 2）を持つ 2 つの VPX インスタンス（プライマリノードとセカンダリノード）を作成します
2. AWS セカンダリプライベート IP アドレスをプライマリノードの VIP と SNIP に割り当てます。
3. AWS セカンダリプライベート IP アドレスを使用して、プライマリノードで VIP と SNIP を設定します。
4. 両方のノードで HA を設定します。

手順 **1. AWS** 上に **2** つの **VPX** インスタンスを作成します。各インスタンスには **3** つの **NIC** があります

AWS ウェブコンソールを使用して、NetScaler VPX インスタンスを AWS にデプロイするに記載されている手順に従います。

手順 **3.** 手順 **2**: プライマリノードで、イーサネット **1 (**クライアント **IP** または **VIP)** とイーサネット **2 (**バックエンドサーバー **IP** または **SNIP)** にプライベート **IP** アドレスを割り当てます

AWS コンソールは、設定された NIC にプライマリプライベート IP アドレスを自動的に割り当てます。VIP と SNIP には、セカンダリプライベート IP アドレスと呼ばれる、より多くのプライベート IP アドレスを割り当てます。

プライベート IP アドレスをネットワークインターフェイスに割り当てるには、次の手順に従います。

1. https://console.aws.amazon.com/ec2/で Amazon EC2 コンソールを開きます。
2. ナビゲーションペインで、[**Network Interfaces**] を選択し、インスタンスにアタッチされているネットワークインターフェイスを選択します。
3. アクション **> IP** アドレスの管理を選択します。
4. 要件に基づいて [**IPv4** アドレス] または [**IPv6** アドレス] を選択します。
5. IPv4 アドレスの場合:

    a) [新しい **IP** を割り当てる] を選択します。
    b) インスタンスのサブネット範囲内にある特定の IPv4 アドレスを入力するか、フィールドを空白のままにして Amazon が IP アドレスを選択するようにします。
    c) (オプション) セカンダリプライベート IP アドレスが既に別のネットワークインターフェイスに割り当てられている場合に、そのアドレスを再割り当てできるようにするには、[Allow reassign] を選択します。

6. IPv6 アドレスの場合:

    a) [新しい **IP** を割り当てる] を選択します。
    b) インスタンスのサブネット範囲内にある特定の IPv6 アドレスを入力するか、フィールドを空白のままにして Amazon が IP アドレスを選択できるようにします。
    c) (オプション) プライマリまたはセカンダリプライベート IP アドレスが既に別のネットワークインターフェイスに割り当てられている場合に、そのアドレスを再割り当てできるようにするには、[Allow reassign] を選択します。

7. はい **>** 更新を選択します。

インスタンスの説明の下に、割り当てられたプライベート IP アドレスが表示されます。

> 注
>
> IPv4 HA ペア展開では、インターフェイスにセカンダリ IPv4 アドレスのみを割り当て、それらを VIP アドレスおよび SNIP アドレスとして使用できます。ただし、IPv6 HA ペア展開では、インターフェイスでプライマリ IPv6 アドレスまたはセカンダリ IPv6 アドレスを割り当て、それらを VIP アドレスおよび SNIP アドレスとして使用できます。

手順 **3.** 手順 **3**: セカンダリプライベート **IP** アドレスを使用して、プライマリノードで **VIP** と **SNIP** を構成します

SSH を使用してプライマリノードにアクセスします。ssh クライアントを開き、次のように入力します。


```
ssh -i <location of your private key> nsroot@<public DNS of the instance>
```

次に、VIP と SNIP を設定します。

VIP の場合は、次のように入力します。

```
add ns ip <IPAddress> <netmask> -type <type>
```

SNIP の場合は次のように入力します。

```
add ns ip <IPAddress> <netmask> -type SNIP
```

`save config`を入力して保存します。

設定された IP アドレスを表示するには、次のコマンドを入力します。

```
show ns ip
```

詳しくは、次のトピックを参照してください:

- 仮想 IP（VIP）アドレスの構成と管理
- NSIP アドレスの構成

ステップ **4:** 両方のインスタンスで **HA** を設定する

プライマリノードでシェルクライアントを開き、次のコマンドを入力します。

```
add ha node <id> <private IP address of the management NIC of the secondary node>
```

セカンダリノードで、次のコマンドを入力します。

```
add ha node <id> <private IP address of the management NIC of the primary node>
```

`save config` と入力して、設定を保存します。

構成された HA ノードを表示するには、`show ha node`と入力します。

フェイルオーバー時に、前のプライマリノードで VIP および SNIP として構成されたセカンダリプライベート IP アドレスは、新しいプライマリノードに移行されます。

ノードでフェイルオーバーを強制するには、`force HA`failover と入力します。

セカンダリプライベート **IP** 移行に基づいて、レガシー **HA** ペアを新しい **HA** ペアに移行

> 注
>
> ENI 移行に基づいて機能する VPX HA ペアをデプロイする従来の方法は廃止されました。そのため、セカンダリプライベート IP 移行に基づく HA ペア展開を使用することをお勧めします。

セカンダリプライベート IP の移行に基づいてレガシー HA ペアから新しい HA ペアへのシームレスな移行を可能にするには、次の点を確認してください:

1. プライマリノードとセカンダリノードの両方に同じ数のインターフェイスが必要で、これらのインターフェイスは同じサブネット内にある必要があります。
2. 従来の方法でプライマリプライベート IP アドレスとして設定された VIP と SNIP は、新しい方法ではセカンダリプライベート IP アドレスに移行する必要があります。
3. 新しい HA 展開に必要な IAM 権限を、プライマリおよびセカンダリの NetScaler インスタンスに追加する必要があります。
4. プライマリとセカンダリの NetScaler インスタンスの両方を再起動します。

詳細については、「同じゾーン内の高可用性」を参照してください。

Citrix CloudFormation テンプレートを使用して高可用性ペアをデプロイする

CloudFormation テンプレートを開始する前に、次の要件を満たしていることを確認してください。

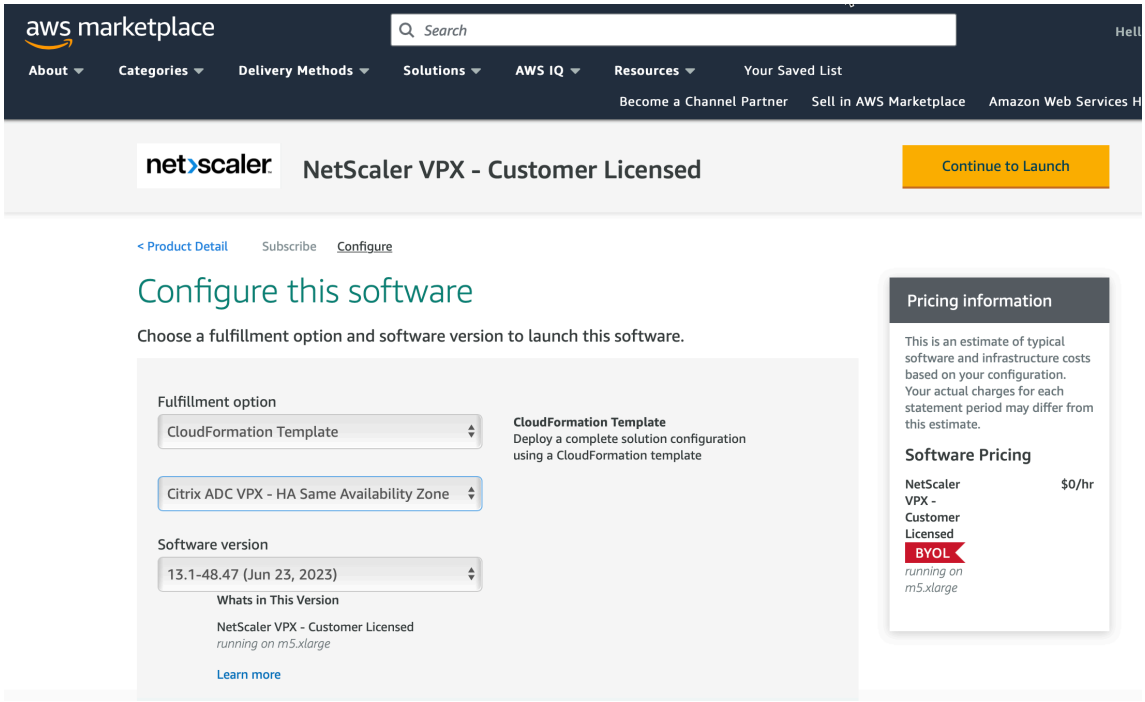
- VPC
- VPC 内の 3 つのサブネット
- UDP 3003、TCP 3009—3010、HTTP、SSH ポートが開いているセキュリティグループ
- キーペア
- インターネットゲートウェイを作成する
- クライアントネットワークと管理ネットワークのルートテーブルを編集して、インターネットゲートウェイを指すようにする

> 注
>
> Citrix CloudFormation テンプレートは、IAM ロールを自動的に作成します。既存の IAM ロールはテンプレートには表示されません。

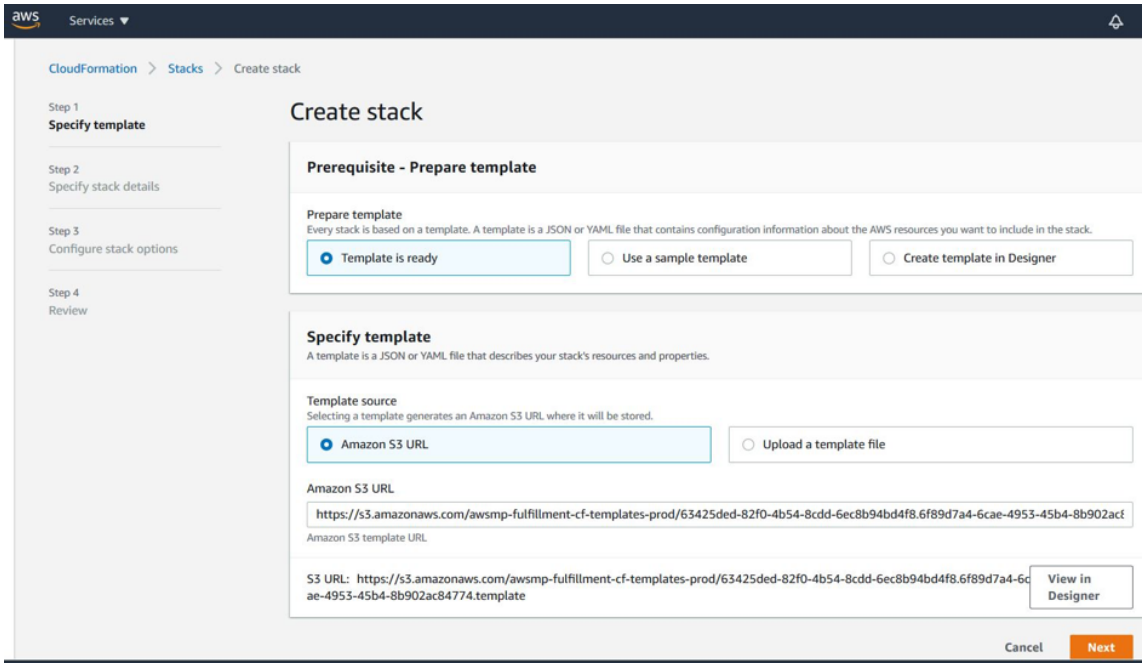
Citrix CloudFormation テンプレートを起動するには、次の手順に従います。

1. AWS 認証情報を使用して AWS マーケットプレイスにログオンします 。
2. 検索フィールドに「**NetScaler VPX**」と入力して NetScaler AMI を検索し、[**Go**] をクリックします。
3. 検索結果ページで、目的の NetScaler VPX 製品をクリックします。
4. 「価格設定」タブをクリックして、「価格情報」に移動します。
5. リージョンとフルフィルメントオプションを「**NetScaler VPX-**カスタマーライセンス」として選択します。

6. [続行] をクリックして購読します。
7. [購読] ページで詳細を確認し、[構成に進む] をクリックします。
8. **CloudFormation** テンプレートとして [配信方法] を選択します。
9. 必要な CloudFormation テンプレートを選択します。
10. [****** ソフトウェアのバージョンとリージョン] を選択し、**[**** 続行] をクリックして起動します。

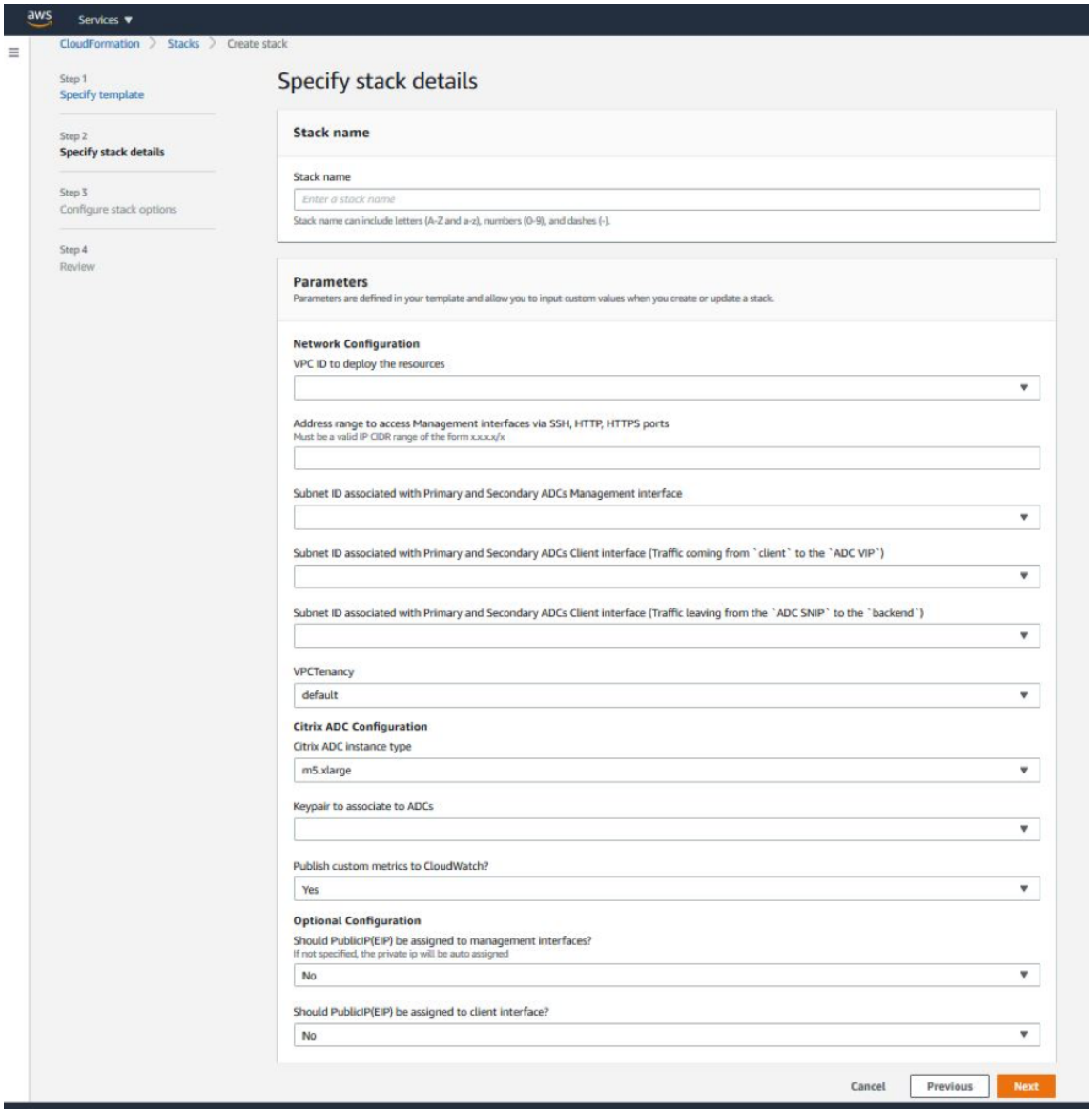


11. [**AWS CloudFormation** が **IAM** リソースを作成する可能性があることを承認します] を選択します。****** チェックボックスをオンにし、**[**** スタックを作成] をクリックします。
12. [次へ] をクリックします。

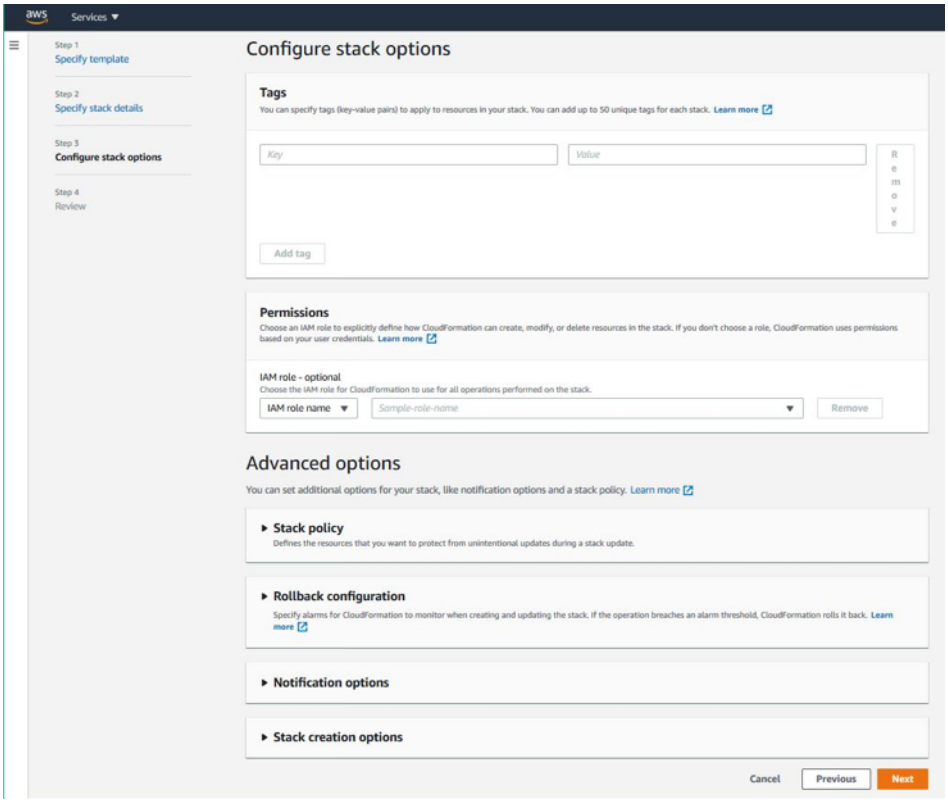


13. **[**スタックの詳細を指定] ページが表示されます。次の詳細を入力します。

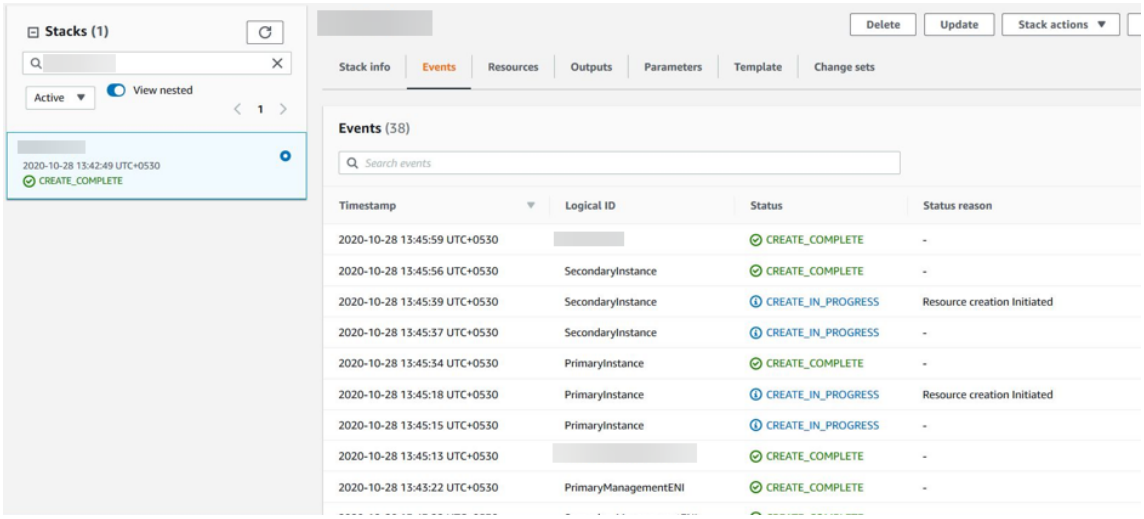
    - スタック名を入力します。名前は 25 文字以内である必要があります。
    - [ネットワーク構成] で、次の操作を実行します。
        - [管理サブネットワーク]、[クライアントサブネットワーク**]**、および **[**サーバーサブネットワーク**]** を選択します。[VPC ID] で選択した VPC 内で作成した正しいサブネットワークを選択していることを確認します。
        - プライマリ管理 **IP**、セカンダリ管理 **IP**、クライアント **IP**、およびサーバ **IP** を追加します。IP アドレスは、それぞれのサブネットワークの同じサブネットに属している必要があります。または、テンプレートに IP アドレスが自動的に割り当てられるようにすることもできます。
        - **vpcTenancy** で [デフォルト] を選択します。
    - **NetScaler** 構成で、以下を実行します。
        - [インスタンスタイプ] で [**m5.xlarge**] を選択します。
        - [Key Pair] のメニューから、作成済みのキーペアを選択します。
        - デフォルトでは、**CloudWatch** にカスタム メトリックを公開しますか**?** オプションは はいに設定されています。グラフをカスタマイズするには、**[**グラフ**]** オプションを使用します。CloudWatch メトリクスの詳細については、[Amazon CloudWatch を使用してインスタンスを監視する] (#monitor-your-instances-using-amazon-cloudWatch) を参照してください。
    - [オプション構成] で、次の操作を行います。
        - デフォルトでは、管理インターフェースにパブリック **IP(EIP)** を割り当てる必要がありますか**?** オプションは いいえに設定されています。
        - デフォルトでは、パブリック **IP(EIP)** をクライアントインターフェースに割り当てる必要がありますか**?** オプションは いいえに設定されています。



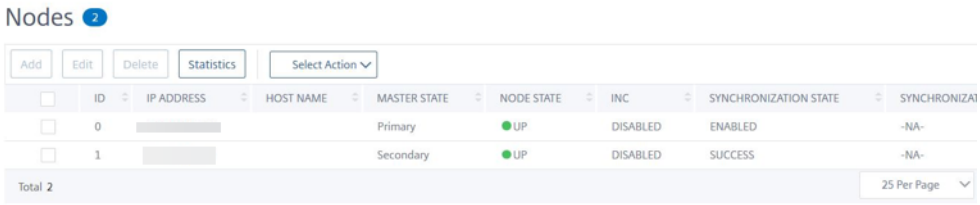
14. ［次へ］をクリックします。
15. **［スタックオプションの設定］** ページが表示されます。これはオプションのページです。



16. ［次へ］をクリックします。
17. [オプション] ページが表示されます。(これはオプションのページです)。［次へ］をクリックします。
18. [**Review**] ページが表示されます。しばらくして、設定を確認し、必要に応じて変更を加えます。
19. **AWS CloudFormation** が **IAM** リソースを作成する可能性があることを承認します。を選択します。チェックボックスをオンにして、[スタックの作成] をクリックします。
20. **CREATE-IN-PROGRESS** が表示されます。ステータスが **CREATE-COMPLETE** になるまで待ちます。ステータスが **COMPLETE** に変更されない場合は、[**Events**] タブで失敗の原因を確認し、適切な構成でインスタンスを再作成します。



21. IAM リソースが作成されたら、[**EC2** マネジメントコンソール] **>** [インスタンス] に移動します。IAM ロールで作成された 2 つの VPX インスタンスがあります。プライマリノードとセカンダリノードは、それぞれ 3 つのプライベート IP アドレスと 3 つのネットワークインターフェイスを使用して作成されます。
22. ユーザー名`nsroot`とインスタンス ID をパスワードとしてプライマリノードにログオンします。GUI から、[システム] **>** [高可用性] **>** [ノード] に移動します。NetScaler VPX は、CloudFormation テンプレートによって HA ペアで既に構成されています。
23. NetScaler VPX HA ペアが表示されます。



Amazon CloudWatch を使用してインスタンスをモニタリングする

Amazon CloudWatch サービスを使用して、CPU とメモリの使用率、スループットなど、一連の NetScaler VPX メトリクスを監視できます。CloudWatch は AWS で実行されるリソースとアプリケーションをリアルタイムでモニタリングします。AWS マネジメントコンソールを使用して、Amazon CloudWatch ダッシュボードにアクセスできます。詳細については、「Amazon CloudWatch」を参照してください。

注意事項

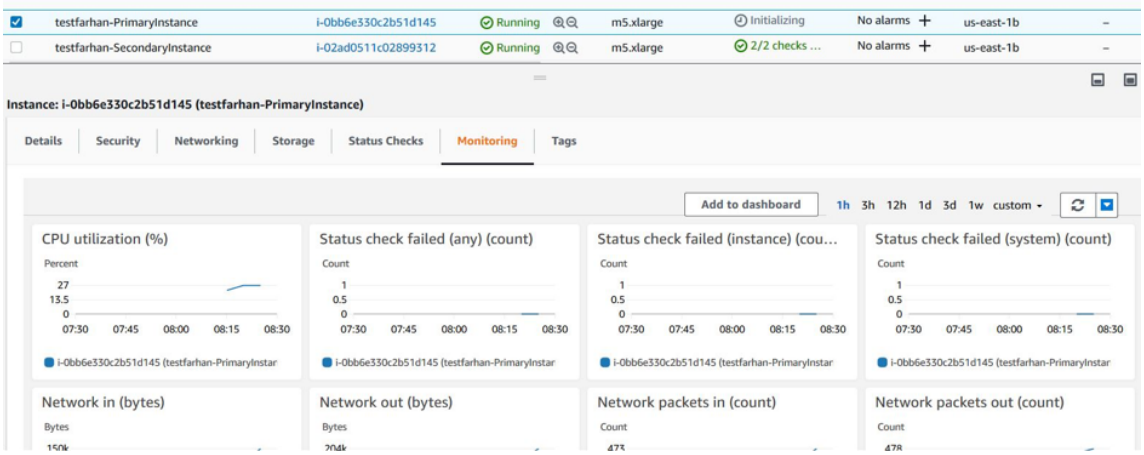
- AWS ウェブコンソールを使用して AWS に NetScaler VPX インスタンスをデプロイすると、CloudWatch サービスはデフォルトで有効になります。

- Citrix CloudFormation テンプレートを使用して NetScaler VPX インスタンスをデプロイする場合、デフォルトのオプションは「はい」です。CloudWatch サービスを無効にする場合は、「いいえ」を選択します。
- メトリクスは、CPU (管理およびパケット CPU 使用率)、メモリ、およびスループット (インバウンドとアウトバウンド) で使用できます。

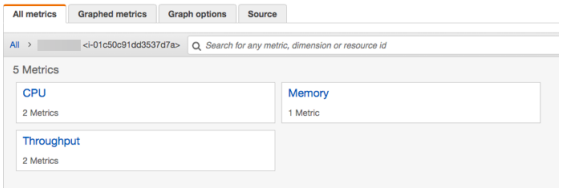
CloudWatch メトリクスの表示方法

インスタンスの CloudWatch メトリックスを表示するには、次の手順に従います。

1. **AWS** マネジメントコンソール **> EC2 >** インスタンスにログオンします。
2. インスタンスを選択します。
3. [監視] をクリックします。
4. [**CloudWatch** メトリックスをすべて表示] をクリックします。

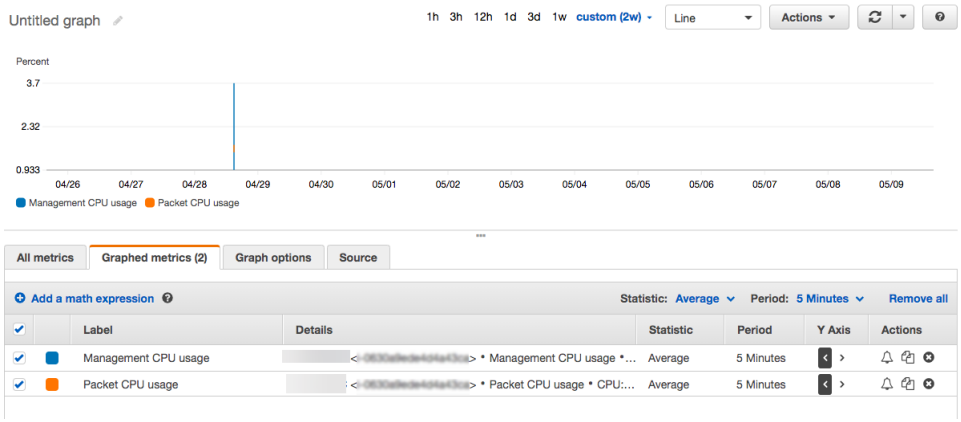


5. [すべてのメトリックス] で、インスタンス ID をクリックします。



6. 表示するメトリクスをクリックし、期間（分、時間、日、週、月）を設定します。
7. グラフ化されたメトリクスをクリックして、使用量の統計を表示します。 オプションが [はい] に設定されています。

図。CPU 使用率に関するグラフ化されたメトリック



高可用性セットアップでの **SR-IOV** の設定

高可用性セットアップでの SR-IOV インターフェイスのサポートは、NetScaler リリース 12.0 57.19 以降から利用できます。SR-IOV を構成する方法の詳細については、「SR-IOV ネットワーク インターフェイスを使用するための NetScaler VPX インスタンスの構成」を参照してください。

関連情報

AWS での高可用性の機能

異なる **AWS** アベイラビリティーゾーンでの高可用

October 17, 2024

独立ネットワーク構成（INC）モードでは、2 つの異なるサブネットまたは 2 つの異なる AWS アベイラビリティーゾーンに 2 つの NetScaler ADC VPX インスタンスを高可用性アクティブ/パッシブのペアとして構成できます。何らかの理由で 1 次ノードが接続を受け入れることができない場合は、2 次ノードが引き継ぎます。

高可用性の詳細については、「高可用性」を参照してください。INC の詳細については、「異なるサブネットでの高可用性ノードの設定」を参照してください。

注意事項

- 配置を開始する前に、次のドキュメントをお読みください。
  - AWS 用語
  - 前提条件
  - 制限事項と使用ガイドライン

- VPX 高可用性ペアは、異なるサブネットの同じアベイラビリティーゾーンに存在することも、2 つの異なる AWS アベイラビリティーゾーンに存在することもできます。
- 管理（NSIP）、クライアントトラフィック（VIP）、バックエンドサーバー（SNIP）には異なるサブネットを使用することをお勧めします。
- フェイルオーバーを機能させるには、独立ネットワーク構成 (INC) モードで高可用性を設定する必要があります。
- 2 つのインスタンスでは、ハートビートに使用される UDP トラフィック用にポート 3003 が開いている必要があります。
- 残りの API が機能するように、両方のノードの管理サブネットは、内部 NAT を介してインターネットまたは AWS API サーバーにアクセスできる必要があります。
- IAM ロールには、パブリック IP または Elastic IP (EIP) 移行用の E2 アクセス権限と、プライベート IP 移行用の EC2 ルートテーブルのアクセス許可が必要です。

次の方法で AWS アベイラビリティーゾーン間で高可用性をデプロイできます。

- エラスティック IP アドレスの使用
- プライベート IP アドレスの使用

その他の参考資料

AWS 向け NetScaler Application Delivery Management (ADM) の詳細については、「AWS に NetScaler ADM エージェントをインストールする」を参照してください。

異なる **AWS** ゾーンに **Elastic IP** アドレスを使用した **VPX** 高可用性ペアをデプロイする

October 17, 2024

INC モードで Elastic IP（EIP）アドレスを使用して、2 つの異なるサブネットまたは 2 つの異なる AWS アベイラビリティーゾーンに 2 つの NetScaler VPX インスタンスを設定できます。

高可用性の詳細については、「高可用性」を参照してください。INC の詳細については、「異なるサブネットでの高可用性ノードの設定」を参照してください。

異なる **AWS** ゾーンにわたる **EIP** アドレスを持つ **HA** のしくみ

フェールオーバー時には、プライマリインスタンスの VIP の EIP がセカンダリに移行し、セカンダリが新しいプライマリとして引き継がれます。フェイルオーバープロセスでは、AWS API は以下を行います。

1. `IPSets`が接続されている仮想サーバーをチェックします。

2. 仮想サーバーがリッスンしている 2 つの IP アドレスから、パブリック IP が関連付けられている IP アドレスを検索します。1 つは仮想サーバに直接接続され、もう 1 つは IP セットを介して接続されます。
3. パブリック IP (EIP) を、新しいプライマリ VIP に属するプライベート IP に再関連付けします。

> 注
>
> EIP を使用する際に、サービス拒否 (DoS) などの攻撃からネットワークを保護するために、AWS でセキュリティグループを作成して IP アクセスを制限できます。高可用性を実現するために、展開に従って EIP からプライベート IP 移動ソリューションに切り替えることができます。

異なる **AWS** ゾーン間でエラスティック **IP** アドレスを使用して **VPX** 高可用性ペアをデプロイする方法

VPX ペアを 2 つの異なるサブネットまたは 2 つの異なる AWS アベイラビリティーゾーンにデプロイする手順の概要を以下に示します。

1. Amazon 仮想プライベートクラウドを作成します。
2. 2 つの VPX インスタンスを、2 つの異なるアベイラビリティーゾーン、または同じゾーンで異なるサブネットにデプロイします。
3. 高可用性の構成

    a) 両方のインスタンスで INC モードで高可用性をセットアップします。
    b) 両方のインスタンスに IP セット を追加します。
    c) 両方のインスタンスの IP セットを VIP にバインドします。
    d) プライマリ・インスタンスに仮想サーバを追加します。

ステップ 1 と 2 では、AWS コンソールを使用します。手順 3 では、NetScaler VPX GUI または CLI を使用します。

ステップ **1**。 Amazon 仮想プライベートクラウド (VPC) を作成します。

ステップ **2**。2 つの異なるアベイラビリティーゾーン、または同じゾーンだが異なるサブネットに 2 つの VPX インスタンスをデプロイします。プライマリ VPX の VIP に EIP を接続します。

VPC を作成し、AWS に VPX インスタンスを展開する方法の詳細については、「AWS に NetScaler VPX スタンドアロン インスタンスを展開する 」および「シナリオ: スタンドアロン インスタンス」を参照してください。

ステップ **3**。高可用性の構成. NetScaler VPX CLI または GUI を使用して、高可用性をセットアップできます。

CLI を使用した高可用性の設定

1. 両方のインスタンスで INC モードで高可用性をセットアップします。

    プライマリノードで、次の操作を行います。

    ```
    add ha node 1 <sec_ip> -inc ENABLED
    ```


セカンダリノード:

```
add ha node 1 <prim_ip> -inc ENABLED
```

<sec_ip>はセカンダリノードの管理 NIC のプライベート IP アドレスを指します。

<prim_ip>はプライマリノードの管理 NIC のプライベート IP アドレスを指します

2. 両方のインスタンスに IP セットを追加します。

   両方のインスタンスで以下のコマンドを入力します。

   ```
   add ipset <ipsetname>
   ```

3. IP セットを両方のインスタンスの VIP セットにバインドします。

   両方のインスタンスで次のコマンドを入力します。

   ```
   add ns ip <secondary vip> <subnet> -type VIP
   bind ipset <ipsetname> <secondary VIP>
   ```

   > 注
   >
   > IP セットは、プライマリ VIP またはセカンダリ VIP にバインドできます。ただし、IP セットをプライマリ VIP にバインドする場合は、セカンダリ VIP を使用して仮想サーバに追加し、逆にセカンダリ VIP を使用します。

4. プライマリ・インスタンスに仮想サーバを追加します。

   次のコマンドを入力します:

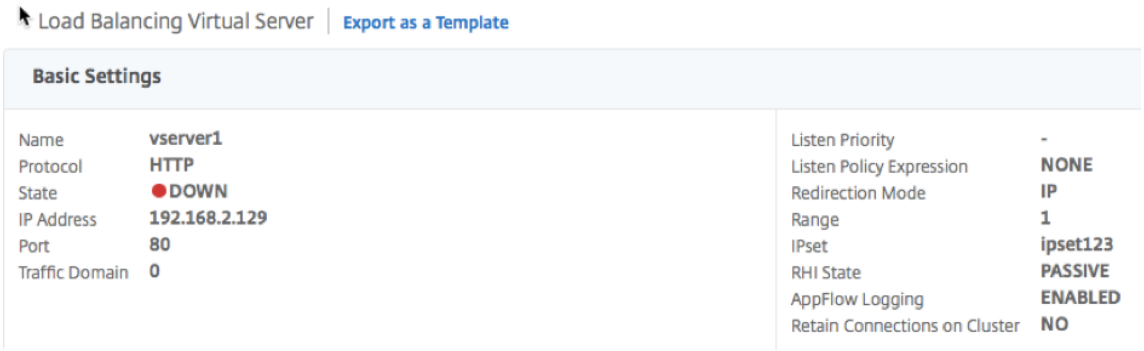
   ```
   add &#060;server_type&#062; vserver &#060;vserver_name&#062;
   &#060;protocol&#062; &#060;primary_vip&#062; &#060;port&#062; -
   ipset \\&#060;ipset_name&#062;
   ```

GUI を使用した高可用性の構成

1. 両方のインスタンスで INC モードで高可用性をセットアップする
2. ユーザー名nsrootとインスタンス ID をパスワードとしてプライマリノードにログオンします。
3. GUI から、[設定] **>** [システム]> [ハイアベイラビリティ] に移動します。[追加] をクリックします。
4. リモートノード **IP** アドレスフィールドに、2 次ノードの管理 NIC のプライベート IP アドレスを追加します。
5. [セルフノードで **NIC (**独立ネットワーク構成**)** モードをオンにする] を選択します。
6. [リモートシステムログイン認証情報] で、セカンダリノードのユーザー名とパスワードを追加し、[作成] をクリックします。
7. セカンダリノードで手順を繰り返します。

8. IP セット名を追加し、[Insert] をクリックします。
9. GUI から、[システム] > [ネットワーク] > [**IP**] > **[**追加**]** に移動します。
10. [IP アドレス]、[ネットマスク]、[IP タイプ (仮想 IP)] に必要な値を追加し、[作成] をクリックします。
11. システム > ネットワーク > **IP** セット > **Add** にナビゲートして下さい。IP セット名を追加し、[**Insert**] をクリックします。
12. [IPv4] ページで、仮想 IP を選択し、[挿入] をクリックします。[**Create**] をクリックして IP セットを作成します。
13. プライマリ・インスタンスに仮想サーバを追加する

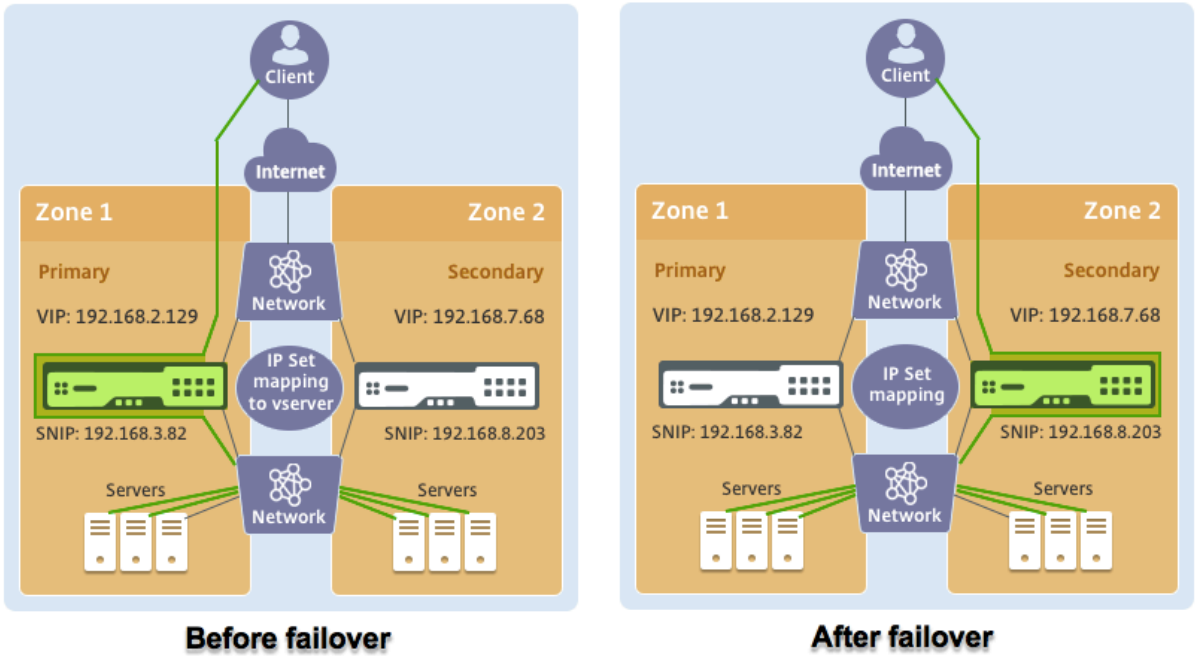
    GUI から、[構成**]**> [トラフィック管理] > [仮想サーバ**] > [**追加**]** に移動します。



シナリオ

このシナリオでは、1 つの VPC が作成されます。その VPC では、2 つのアベイラビリティーゾーンに 2 つの VPX インスタンスが作成されます。各インスタンスには、管理用、クライアント用、バックエンドサーバー用の 3 つのサブネットがあります。EIP はプライマリノードの VIP に接続されます。

図: この図は、AWS での INC モードでの Citrix ADC VPX の高可用性セットアップを示しています



このシナリオでは、CLI を使用して高可用性を設定します。

1. 両方のインスタンスで INC モードで高可用性をセットアップします。

   プライマリノードとセカンダリノードで次のコマンドを入力します。

   プライマリ:

   ```
   add ha node 1 192.168.6.82 -inc enabled
   ```

   ここで、192.168.6.82 は、セカンダリノードの管理 NIC のプライベート IP アドレスを指します。

   セカンダリ:

   ```
   add ha node 1 192.168.1.108 -inc enabled
   ```

   ここで、192.168.1.108 は、プライマリノードの管理 NIC のプライベート IP アドレスを指します。

2. IP セットを追加し、IP セットを両方のインスタンスの VIP にバインドします。

   プライマリ:

   ```
   add ipset ipset123
   add ns ip 192.168.7.68 255.255.255.0 -type VIP
   bindipset ipset123 192.168.7.68
   ```

   セカンダリ:

   ```
   add ipset ipset123
   add ns ip 192.168.7.68 255.255.255.0 -type VIP
   bind ipset ipset123 192.168.7.68
   ```

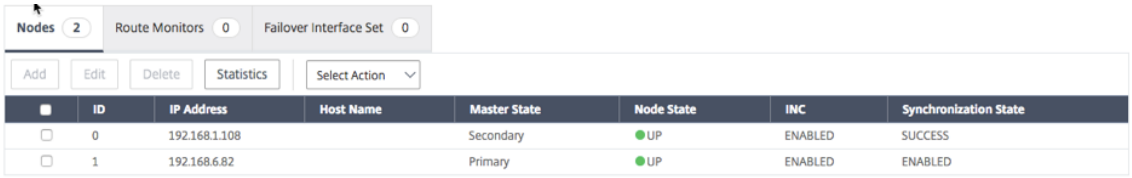

3. プライマリ・インスタンスに仮想サーバを追加します。

   以下のコマンドを実行します。

   ```
   add lbvserver vserver1 http 192.168.2.129 80 -ipset ipset123
   ```

4. 構成を保存します。

5. 強制フェールオーバーの後、セカンダリは新しいプライマリになります。



異なる **AWS** ゾーンにプライベート **IP** アドレスを使用して **VPX** 高可用性ペアを展開する

October 17, 2024

INC モードのプライベート IP アドレスを使用して、2 つの異なるサブネットまたは 2 つの異なる AWS アベイラビリティーゾーンで 2 つの NetScaler ADC VPX インスタンスを設定できます。このソリューションは、弾性 IP アドレス を備えた既存のマルチゾーン
VPX 高可用性ペアと簡単に統合できます。したがって、両方のソリューションを一緒に使用できます。</p>

高可用性の詳細については、「高可用性」を参照してください。INC の詳細については、「異なるサブネットでの高可用性ノードの設定」を参照してください。

> 注
>
> このデプロイは、NetScaler リリース 13.0 ビルド 67.39 以降でサポートされています。このデプロイは AWS Transit Gateway と互換性があります。

AWS 非共有 VPC を使用したプライベート IP アドレスを使用した高可用性ペア

前提条件

AWS アカウントに関連付けられた IAM ロールに次の IAM アクセス権限があることを確認します。

```
{

    "Version": "2012-10-17",
    "Statement": [
        {

            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeAddresses",
                "ec2:AssociateAddress",
                "ec2:DisassociateAddress",
                "ec2:DescribeRouteTables",
                "ec2:DeleteRoute",
                "ec2:CreateRoute",
                "ec2:ModifyNetworkInterfaceAttribute",
                "iam:SimulatePrincipalPolicy",
                "iam:GetRole"
            ],
            "Resource": "*",
            "Effect": "Allow"
         }

    ]
 }
```

AWS 非共有 VPC を使用して、プライベート IP アドレスを持つ VPX HA ペアをデプロイする

次に、プライベート IP アドレスを使用して 2 つの異なるサブネットまたは 2 つの異なる AWS アベイラビリティーゾーンに VPX ペアをデプロイする手順の概要を示します。

1. Amazon 仮想プライベートクラウドを作成します。
2. 2 つの異なるアベイラビリティーゾーンに 2 つの VPX インスタンスをデプロイします。
3. 高可用性の構成

    a) 両方のインスタンスで INC モードで高可用性をセットアップします。
    b) クライアントインターフェイスを指すそれぞれのルートテーブルを VPC に追加します。
    c) プライマリ・インスタンスに仮想サーバを追加します。

ステップ 1、2、および 3b では、AWS コンソールを使用します。ステップ 3a と 3c では、NetScaler VPX GUI または CLI を使用します。

ステップ **1**。 Amazon 仮想プライベートクラウド (VPC) を作成します。

ステップ **2**。同じ数の ENI (ネットワーク インターフェイス) を持つ 2 つの異なるアベイラビリティ ゾーンに 2 つの VPX インスタンスをデプロイします。

VPC を作成し、AWS に VPX インスタンスを展開する方法の詳細については、「AWS に NetScaler VPX スタンドアロン インスタンスを展開する 」および「シナリオ: スタンドアロン インスタンス」を参照してください。

ステップ **3**。 Amazon VPC サブネットと重複しないサブネットを選択して、ADC VIP アドレスを設定します。VPC が 192.168.0.0/16 の場合、ADC VIP アドレスを設定するには、次の IP アドレス範囲から任意のサブネットを選択できます。

- 0.0.0.0-192.167.0.0
- 192.169.0.0-254.255.255.0

この例では、10.10.10.0/24 サブネットを選択し、このサブネットに VIP を作成しました。VPC サブネット以外の任意のサブネットを選択できます (192.168.0.0/16).

ステップ **4**。 VPC ルート テーブルからプライマリ ノードのクライアント インターフェイス (VIP) を指すルートを追加します。

AWS CLI から、次のコマンドを入力します。

```
aws ec2 create-route --route-table-id rtb-2272532 --destination-cidr-block 10.10.10.0/24 --gateway-id <eni-client-primary>
```

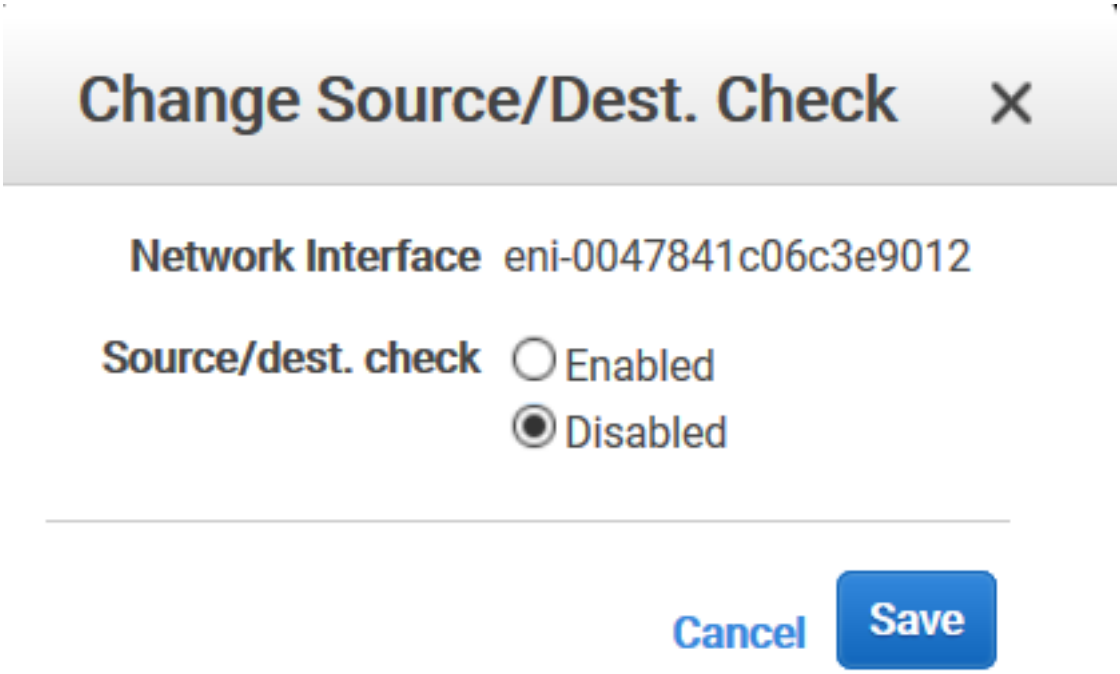
AWS GUI から、次の手順を実行してルートを追加します。

1. Amazon EC2 コンソールを開きます。
2. ナビゲーションペインで、ルートテーブルを選択し、ルートテーブルを選択します。
3. [アクション] を選択し、[ルートの編集] をクリックします。
4. ルートを追加するには、[**Add route**] を選択します。[宛先] に、宛先 CIDR ブロック、単一の IP アドレス、またはプレフィックスリストの ID を入力します。ゲートウェイ ID には、プライマリノードのクライアントインターフェイスの ENI を選択します。

> 注
>
> プライマリ・インスタンスのクライアント ENI で **Source/Dest Check** を無効にする必要があります。

コンソールを使用してネットワークインターフェイスの source/destination チェックを無効にするには、次の手順を実行します。

1. Amazon EC2 コンソールを開きます。
2. ナビゲーションペインで、[ネットワークインターフェイス] を選択します。
3. プライマリクライアントインターフェイスのネットワークインターフェイスを選択し、[アクション] を選択し、[** ソース/デストを変更] をクリックします。** を確認してください。
4. ダイアログボックスで、[無効] を選択し、[保存] をクリックします。



ステップ **5**。高可用性の構成. NetScaler VPX CLI または GUI を使用して、高可用性をセットアップできます。

CLI を使用した高可用性の設定

1. 両方のインスタンスで INC モードで高可用性をセットアップします。

プライマリノードで、次の操作を行います。

```
```
  add ha node 1 \<sec_ip\> -inc ENABLED
```
```

セカンダリノード:


```
```
  add ha node 1 \<prim_ip\> -inc ENABLED
```
```

<sec_ip> セカンダリノードの管理 NIC のプライベート IP アドレスを参照します。

<prim_ip> プライマリノードの管理 NIC のプライベート IP アドレスを参照します。

1. プライマリ・インスタンスに仮想サーバを追加します。選択したサブネット（10.10.10.0/24 など）から追加する必要があります。

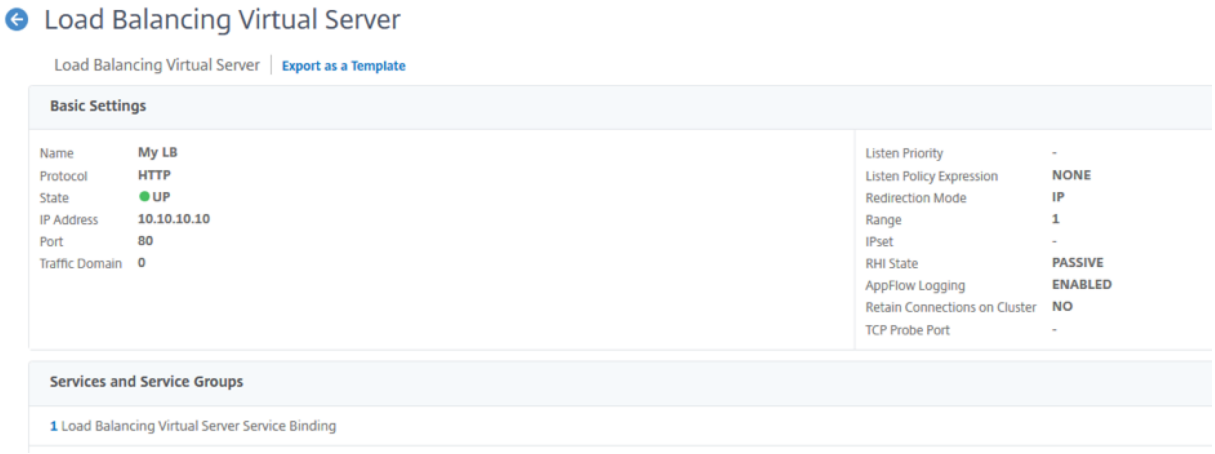
次のコマンドを入力します:

```
```
  add \<server_type\> vserver \<vserver_name\> \<protocol\> \<primary
     _vip\> \<port\>
```
```

GUI を使用した高可用性の構成

1. 両方のインスタンスで INC モードで高可用性をセットアップする
2. ユーザー名nsrootとインスタンス ID をパスワードとしてプライマリノードにログオンします。
3. [構成] > [システム] > [高可用性] に移動し、[追加] をクリックします。
4. リモートノード **IP** アドレスフィールドに、2 次ノードの管理 NIC のプライベート IP アドレスを追加します。
5. [セルフノードで **NIC (**独立ネットワーク構成**)** モードをオンにする] を選択します。
6. [リモートシステムログイン認証情報] で、セカンダリノードのユーザー名とパスワードを追加し、[作成] をクリックします。
7. セカンダリノードで手順を繰り返します。
8. プライマリ・インスタンスに仮想サーバを追加する

[設定] > [トラフィック管理] > [仮想サーバー**] > [**追加**]** に移動します。



AWS 共有 VPC を使用して、プライベート IP アドレスを持つ VPX HA ペアをデプロイする

AWS 共有 VPC モデルでは、VPC を所有するアカウント (所有者) が 1 つ以上のサブネットを他のアカウント (参加者) と共有します。そのため、VPC 所有者アカウントと参加者アカウントがあります。サブネットが共有されると、参加者は共有されたサブネット内のアプリケーションリソースを表示、作成、変更、および削除できます。参加者は、他の参加者または VPC 所有者に属するリソースを表示、変更、削除することはできません。

AWS 共有 VPC の詳細については、AWS ドキュメントを参照してください。

> 注
>
> AWS 共有 VPC を使用してプライベート IP アドレスで VPX HA ペアをデプロイする設定手順は、AWS 非共有 VPC を使用してプライベート IP アドレスで VPX HA ペアをデプロイする手順と同じです 。
>
> - クライアントインターフェイスを指す VPC 内のルートテーブルは、*VPC* 所有者アカウントから追加する必要があります。

前提条件

- AWS 参加者アカウントの NetScaler VPX インスタンスに関連付けられている IAM ロールに次の IAM 権限があることを確認してください。

```
{
    "Version": "2012-10-17",
        "Statement": [
            {

                "Sid": "VisualEditor0",
                "Effect": "Allow",
                "Action": [
                    "ec2:DisassociateAddress",
                    "iam:GetRole",
                    "iam:SimulatePrincipalPolicy",
                    "ec2:DescribeInstances",
```



```
                    "ec2:DescribeAddresses",
                    "ec2:ModifyNetworkInterfaceAttribute",
                     “ec2:AssociateAddress” ,
                    "sts:AssumeRole"
            ],
                "Resource": "*"
             }

        ]
    }
```

注

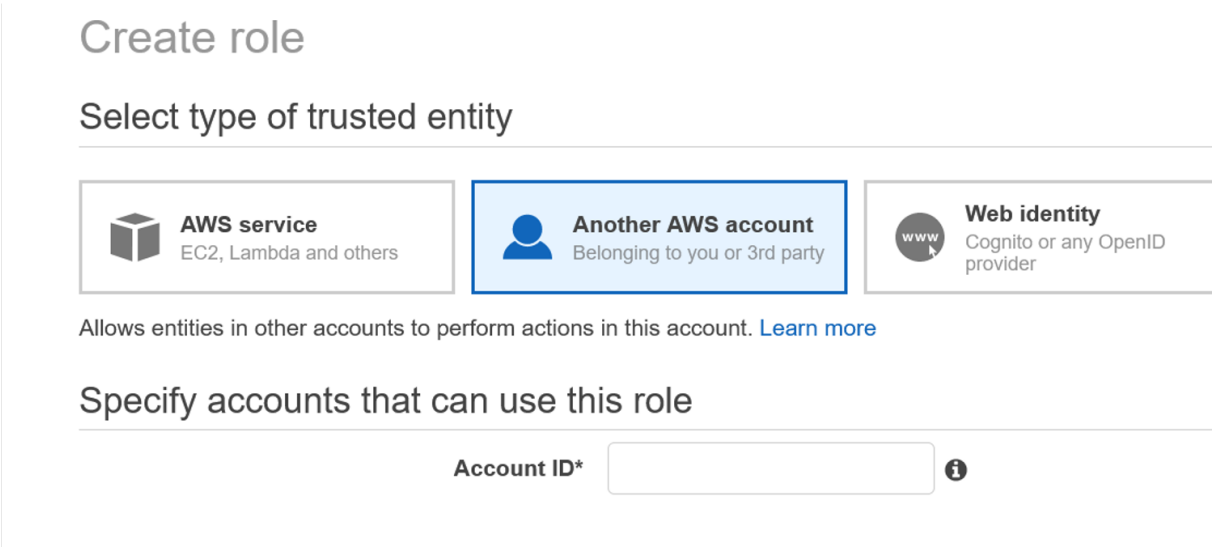
AssumeRole を使用すると、NetScaler VPX インスタンスは、VPC 所有者アカウントによって作成されたクロスアカウント IAM ロールを引き継ぐことができます。

- VPC 所有者アカウントが、クロスアカウント IAM ロールを使用して、参加者アカウントに次の IAM アクセス権限を付与していることを確認します。

```
    {

        "Version": "2012-10-17",
        "Statement": [
            {

                "Sid": "VisualEditor0",
                "Effect": "Allow",
                "Action": [
                    "ec2:CreateRoute",
                    "ec2:DeleteRoute",
                    "ec2:DescribeRouteTables"
                ],
                "Resource": "*"
             }

        ]
    }
```

クロスアカウント **IAM** ロールの作成

1. AWS ウェブコンソールにログインします。
2. [**IAM**] タブで [ロール] に移動し、[****** ロールの作成 ****] を選択します。
3. [別の **AWS** アカウント] を選択します。



1. 管理者アクセス権を付与する参加者アカウントの 12 桁のアカウント ID 番号を入力します。

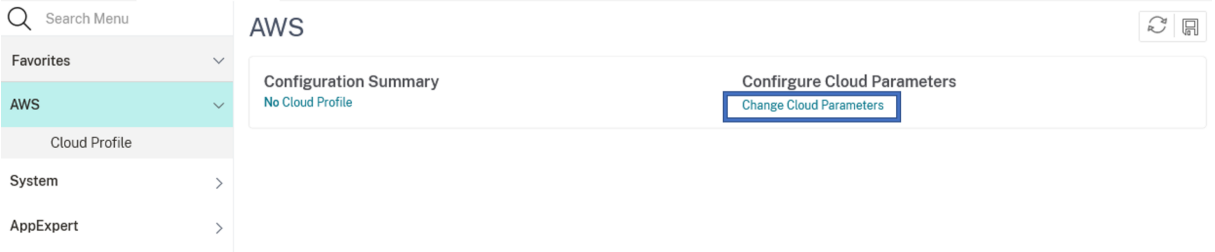
NetScaler CLI を使用してクロスアカウント **IAM** ロールを設定する

次のコマンドを実行すると、NetScaler VPX インスタンスが VPC 所有者アカウントに存在するクロスアカウント IAM ロールを引き継ぐことができます。

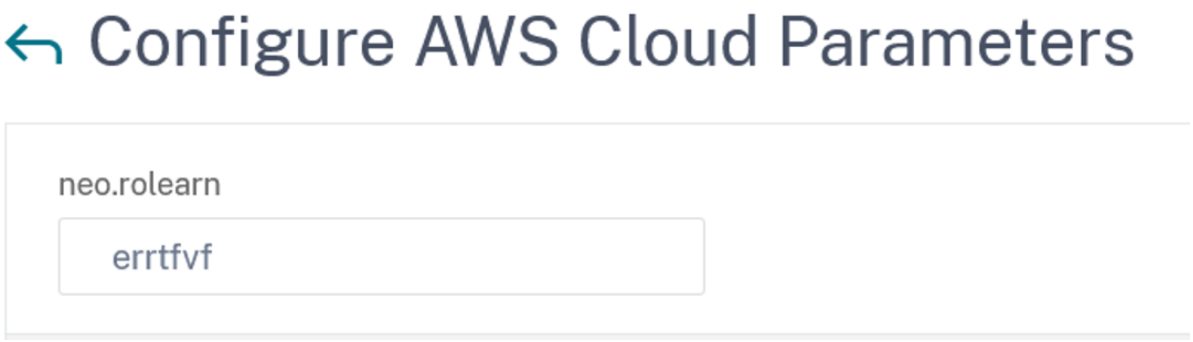
```
set cloud awsParam -roleARN <string>
```

NetScaler GUI を使用してクロスアカウント **IAM** ロールを設定

1. NetScaler アプライアンスにサインインし、[構成] **> [AWS] > [**クラウドパラメータの変更] に移動します。



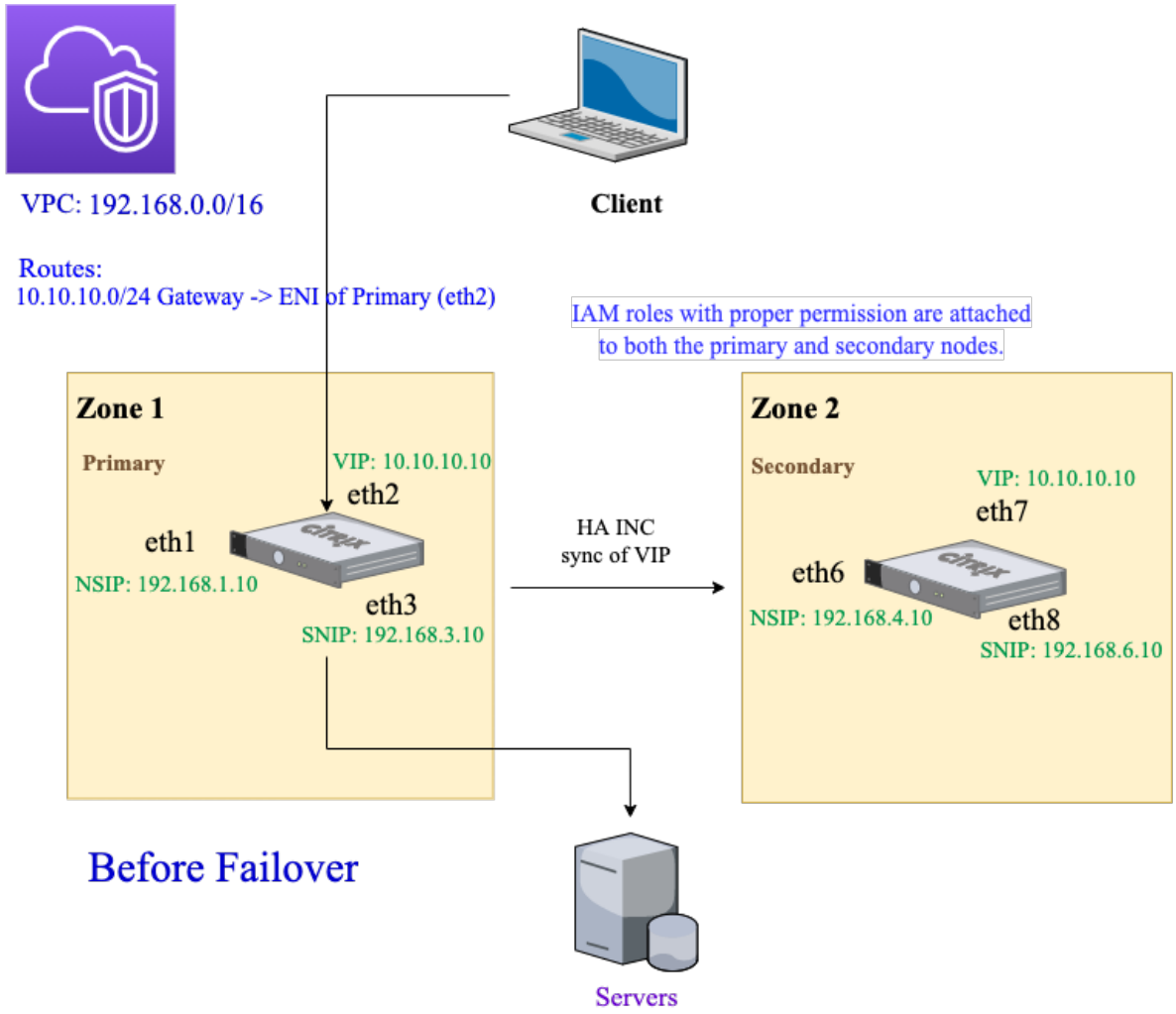
1. **AWS** クラウドパラメータの設定ページで、**RolLearn** フィールドの値を入力します。

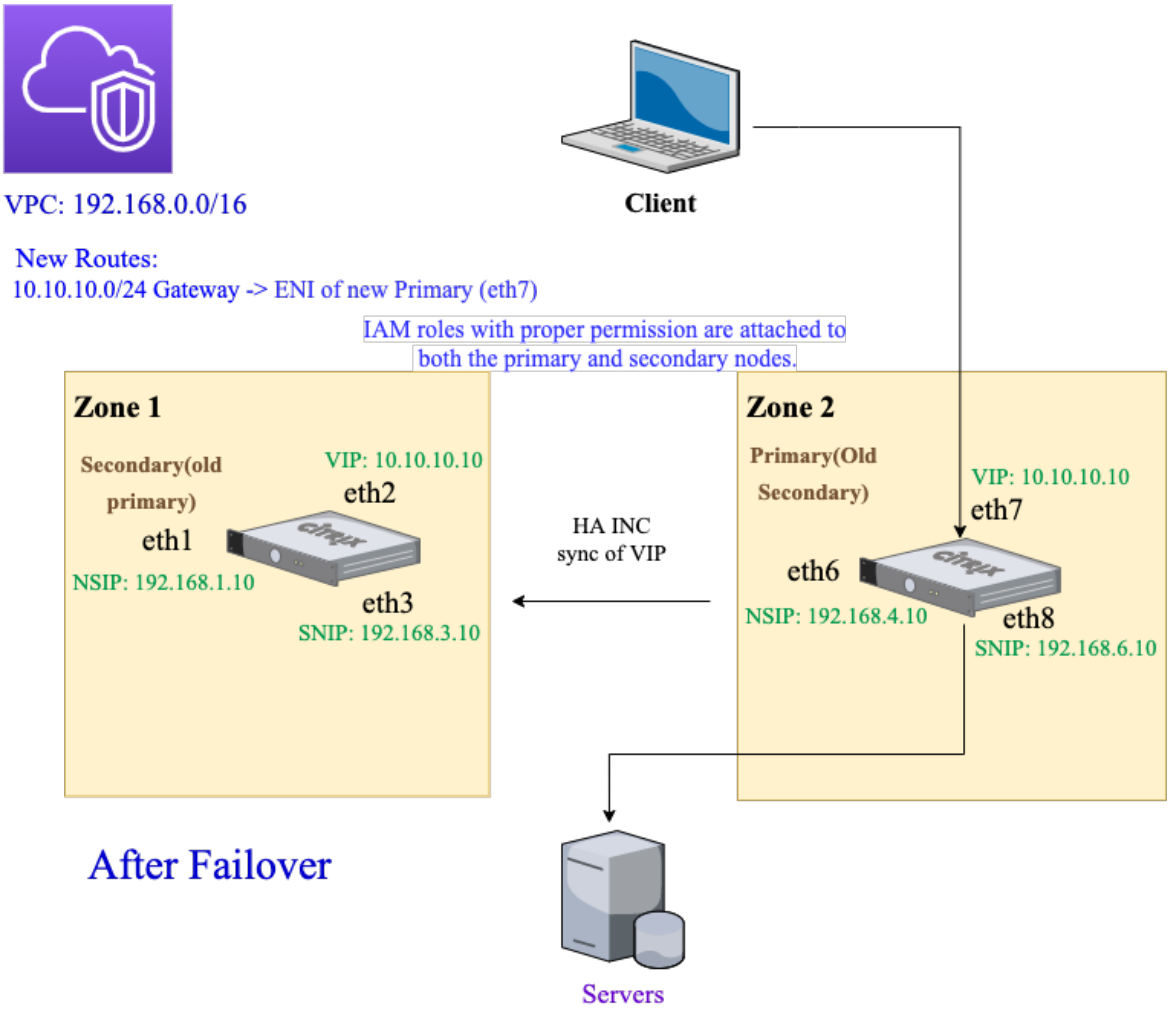


シナリオ

このシナリオでは、1 つの VPC が作成されます。その VPC では、2 つのアベイラビリティーゾーンに 2 つの VPX インスタンスが作成されます。各インスタンスには、管理用、クライアント用、バックエンドサーバー用の 3 つのサブネットがあります。

次の図は、AWS 上の INC モードでの NetScaler VPX 高可用性セットアップを示しています。VPC の一部ではないカスタムサブネット 10.10.10.10 が VIP として使用されます。したがって、10.10.10.10 サブネットはアベイラビリティーゾーン全体で使用できます。





このシナリオでは、CLI を使用して高可用性を設定します。

1. 両方のインスタンスで INC モードで高可用性をセットアップします。

プライマリノードとセカンダリノードで次のコマンドを入力します。

プライマリノードで、次の操作を行います。

```
```
  add ha node 1 192.168.4.10 -inc enabled
```
```

ここで、192.168.4.10 は、セカンダリノードの管理 NIC のプライベート IP アドレスを指します。

セカンダリノード:

```
```
  add ha node 1 192.168.1.10 -inc enabled
```
```

ここで、192.168.1.10 は、プライマリノードの管理 NIC のプライベート IP アドレスを指します。

1. プライマリ・インスタンスに仮想サーバを追加します。

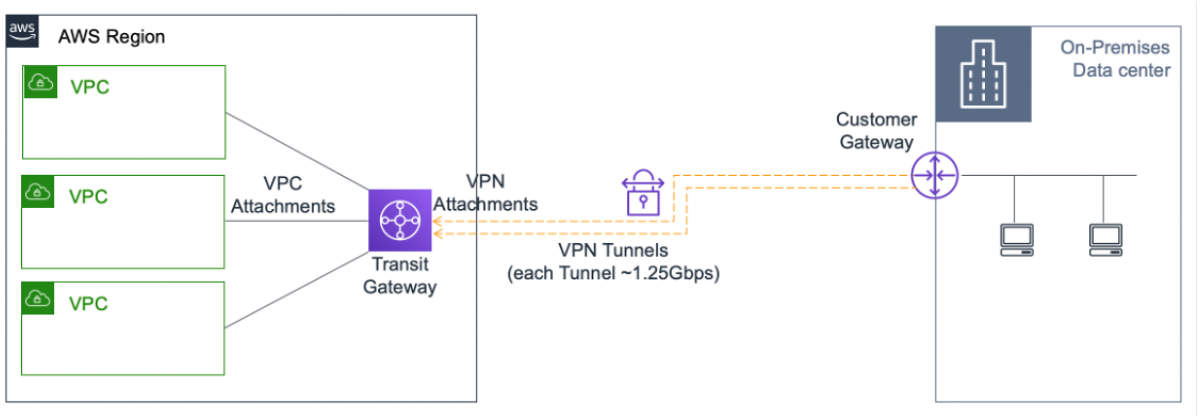
次のコマンドを入力します:

```
  add lbvserver vserver1 http 10.10.10.10 80
```

1. 構成を保存します。
2. 強制フェールオーバーの後:
   - セカンダリインスタンスが新しいプライマリインスタンスになります。
   - プライマリ ENI を指す VPC ルートは、セカンダリクライアント ENI に移行します。
   - クライアントトラフィックは、新しいプライマリインスタンスに再開されます。

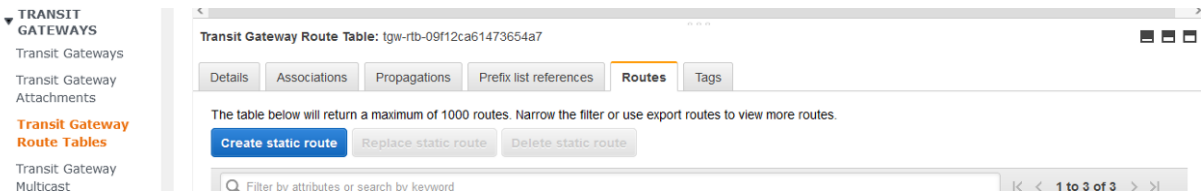
HA プライベート IP ソリューションの AWS Transit Gateway の設定

AWS Transit Gateway は、AWS VPC、リージョン、およびオンプレミスネットワーク全体で、内部ネットワーク内でプライベート VIP サブネットをルーティング可能にする必要があります。VPC は AWS Transit Gateway に接続する必要があります。AWS Transit Gateway ルートテーブル内の VIP サブネットまたは IP プールの静的ルートが作成され、VPC をポイントします。

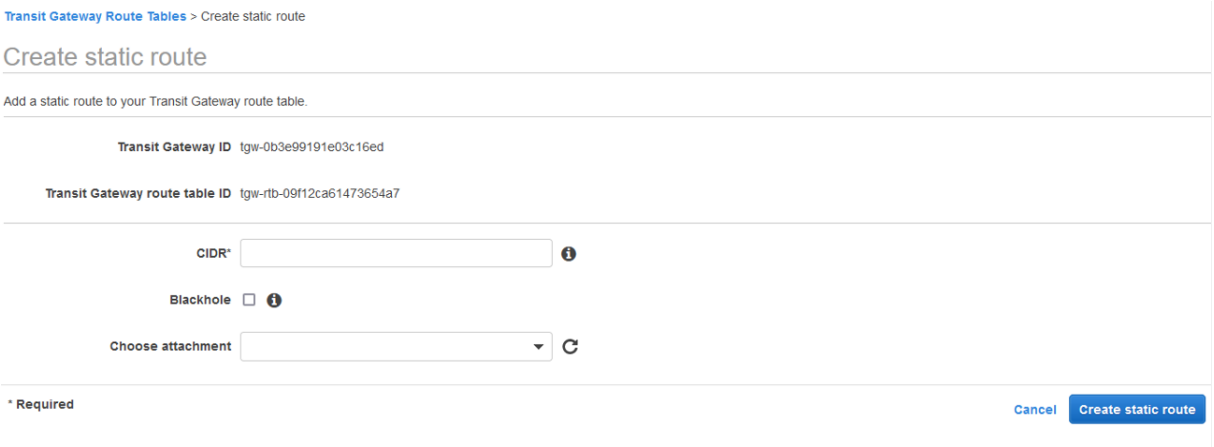


AWS Transit Gateway を設定するには、次の手順に従います。

1. Amazon VPC コンソールを開きます。
2. ナビゲーションペインで、[**Transit Gateway** ートテーブル] を選択します。
3. [ルート] タブを選択し、[静的ルートの作成] をクリックします。



1. CIDR がプライベート VIPS サブネットを指し、アタッチメントが NetScaler VPX のある VPC を指す静的ルートを作成します。



1. [スタティックルートの作成] をクリックし、[閉じる] を選択します。

トラブルシューティング

マルチゾーン HA で HA プライベート IP ソリューションを設定する際に問題が発生した場合は、トラブルシューティングのために次の重要なポイントを確認してください。

- プライマリノードとセカンダリノードの両方に同じ IAM 権限セットがあります。
- INC モードは、プライマリノードとセカンダリノードの両方で有効になっています。
- プライマリノードとセカンダリノードの両方に同じ数のインターフェイスがあります。
- インスタンスを作成するときは、デバイスインデックス番号に基づいてプライマリノードとセカンダリノードの両方にインターフェイスをアタッチする同じ手順に従います。プライマリノードで、クライアントインターフェイスが最初に接続され、サーバーインターフェイスが 2 番目に接続されているとします。セカンダリノードでも同じ手順に従います。不一致がある場合は、正しい順序でインターフェイスを取り外して再接続します。
- インターフェイスの順序を確認するには、[**AWS** コンソール] **>** [ネットワークとセキュリティ] **>** [**ENI**] **>** [デバイスインデックス番号] のナビゲーションパスをたどります。デフォルトでは、これらのインターフェイスには次のデバイスインデックス番号が割り当てられます。- 管理インターフェイス—0 - クライアントインターフェイス—1 - サーバーインターフェイス—2 デフォルトでは、これらのインターフェイスには次のデバイスインデックス番号が割り当てられます。
  - 管理インターフェース-0
  - クライアントインターフェース-1
  - サーバーインターフェース-2
- プライマリ ENI のデバイスインデックス番号の順序が 0、1、2 の場合。セカンダリ ENI も、デバイスインデックス番号と同じシーケンス（0、1、2）に従う必要があります。

デバイスインデックス番号の順序に不一致がある場合、ルートが失われないように、一致しないすべてのルートが管理インターフェイスであるインデックス 0 に転送されます。ただし、管理インターフェイスへのルートの移動は、トラフィックの輻輳を引き起こす可能性があるため、インターフェイスを切り離してから正しい順序で接続し直す必要があります。

- トラフィックが流れない場合は、「送信元/宛先」を確認してください。プライマリノードのクライアントインターフェイスで「Check」が初めて無効になります。
- `cloudhadaemon` コマンド (`ps -aux | grep cloudha`) がシェルで実行されていることを確認します。
- NetScaler ファームウェアのバージョンが 13.0 ビルド 70.x 以降であることを確認してください。
- フェイルオーバープロセスに関する問題については、次の場所にあるログファイルを確認してください: `/var/log/cloud-ha-daemon.log`

AWS Outpost で NetScaler VPX インスタンスを展開する

October 17, 2024

AWS Outposts は、お客様のサイトにデプロイされている AWS のコンピューティングおよびストレージ容量のプールです。Outposts は、オンプレミスの場所に AWS のインフラストラクチャとサービスを提供します。AWS は AWS リージョンの一部としてこの容量を運用、監視、管理します。オンプレミスと AWS クラウドで同じ NetScaler VPX インスタンス、AWS API、ツール、およびインフラストラクチャを使用して、一貫したハイブリッドエクスペリエンスを実現できます。

Outposts にサブネットを作成し、EC2 インスタンス、EBS ボリューム、ECS クラスター、RDS インスタンスなどの AWS リソースを作成するときに指定できます。Outposts サブネット内のインスタンスは、プライベート IP アドレスを使用して AWS リージョンの他のインスタンスと通信します。これらはすべて、同じ Amazon 仮想プライベートクラウド（VPC）内にあります。

詳細については、AWS Outposts ユーザーガイドを参照してください。

AWS アウトポストの仕組み

AWS Outposts は、お客様のアウトポストと AWS リージョンの間で常時接続された状態で稼働するように設計されています。リージョンとオンプレミス環境のローカルワークロードにこの接続を実現するには、Outpost をオンプレミスネットワークに接続する必要があります。オンプレミスネットワークは、リージョンとインターネットへの WAN アクセスを提供する必要があります。また、インターネットは、オンプレミスのワークロードやアプリケーションが存在するローカルネットワークへの LAN または WAN アクセスを提供する必要があります。

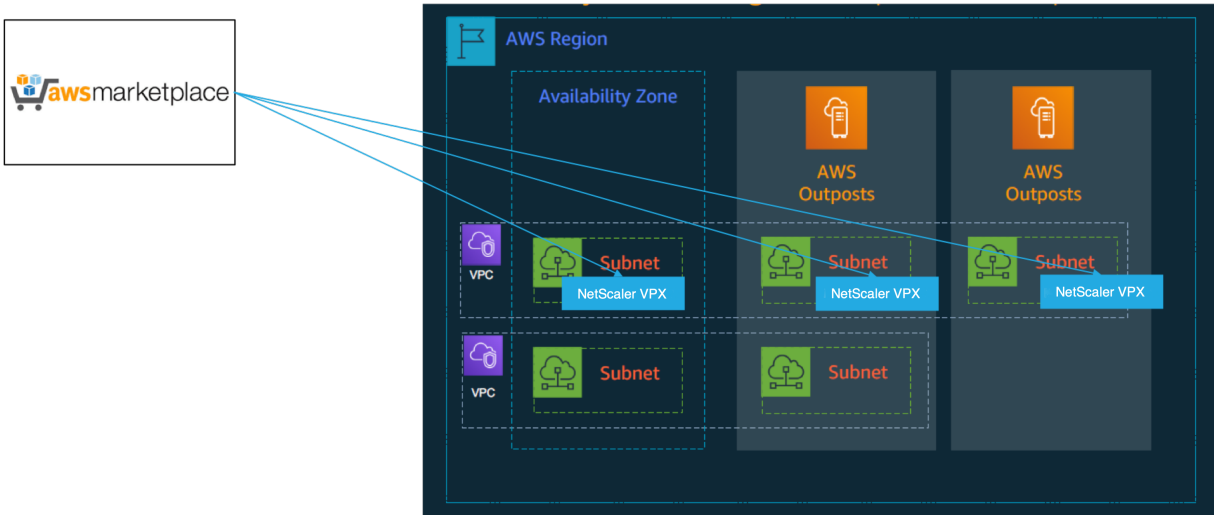
前提要件

- サイトに AWS Outposts をインストールする必要があります。
- AWS Outposts のコンピューティングおよびストレージ容量が使用可能である必要があります。

AWS Outposts の注文方法の詳細については、次の AWS ドキュメントを参照してください。https://aws.amazon.com/blogs/aws/aws-outposts-now-available-order-your-racks-today/

AWS ウェブコンソールを使用して **NetScaler VPX** インスタンスを **AWS** アウトポストにデプロイする

次の図は、アウトポストへの NetScaler VPX インスタンスの簡単な展開を示しています。AWS Marketplace にある NetScaler AMI は、アウトポストにもデプロイされています。



AWS ウェブコンソールにログインし、次の手順を実行して、NetScaler VPX EC2 インスタンスを AWS アウトポストにデプロイします。

1. キーペアを作成します。
2. 仮想プライベートクラウド (VPC) を作成します。
3. サブネットをさらに追加します。
4. セキュリティグループとセキュリティルールを作成します。
5. ルートテーブルを追加します。
6. インターネットゲートウェイを作成します。
7. AWS EC2 サービスを使用して NetScaler VPX インスタンスを作成します。AWS EC2 サービスを使用して NetScaler VPX インスタンスを作成します。AWS ダッシュボードから、[コンピュート**] > [EC2] > [**インスタンスの起動**] > [AWS Marketplace**] に移動します。
8. ネットワークインターフェースをさらに作成して接続します。
9. エラスティック IP を管理用 NIC に接続します。
10. VPX インスタンスに接続します。

各手順の詳細な手順については、「AWS Web コンソールを使用して AWS に NetScaler VPX インスタンスをデプロイする」を参照してください。

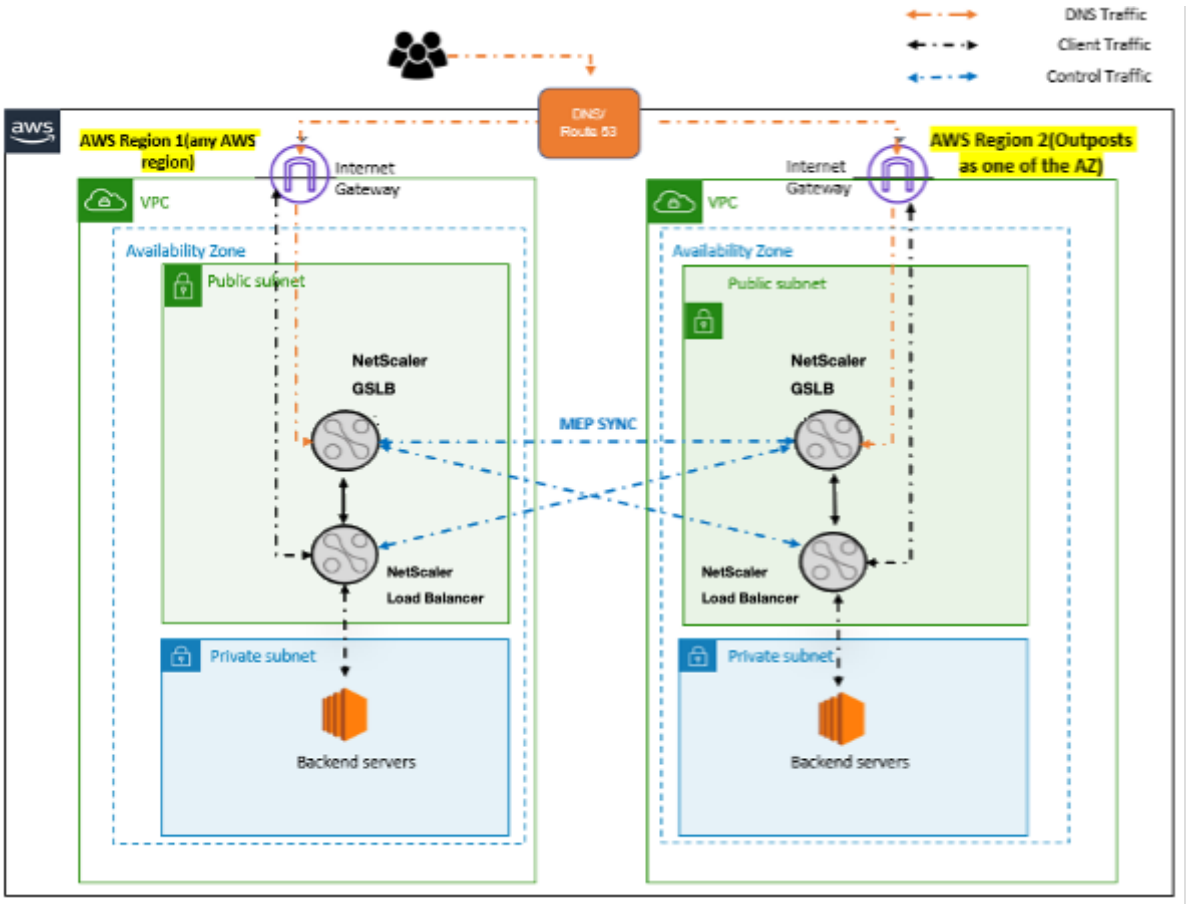
同じアベイラビリティーゾーン内での高可用性の展開については、「AWS に高可用性ペアを展開する」を参照してください。

AWS アウトポストを使用してハイブリッドクラウドに NetScaler VPX インスタンスをデプロイする

NetScaler VPX インスタンスは、AWS のアウトポストを含む AWS 環境のハイブリッドクラウドにデプロイできます。NetScaler グローバルサーバー負荷分散（GSLB）ソリューションを使用すると、アプリ配信メカニズムを簡素化できます。GSLB ソリューションは、AWS リージョンと AWS Outposts インフラストラクチャを使用して構築されたハイブリッドクラウド内の複数のデータセンターにアプリケーショントラフィックを分散します。

NetScaler GSLB は、さまざまなユースケースに対応するために、アクティブ-アクティブとアクティブ-パッシブの両方の展開タイプをサポートしています。これらの柔軟な導入オプションとアプリケーション配信メカニズムに加えて、NetScaler は、アプリケーションが AWS Cloud にネイティブにデプロイされているか、AWS Outposts にネイティブにデプロイされているかに関係なく、ネットワークとアプリケーションのポートフォリオ全体を保護します。

次の図は、AWS とのハイブリッドクラウドにおける NetScaler アプライアンスによるアプリケーション配信を示しています。



アクティブ-アクティブ展開では、NetScaler は分散環境全体でトラフィックをグローバルに誘導します。環境内のすべてのサイトは、メトリクス交換プロトコル (MEP) を通じて、リソースの可用性と状態に関するメトリックを交換します。NetScaler アプライアンスは、この情報を使用してサイト間のトラフィックを負荷分散し、GSLB 構成で指定された定義された方法（ラウンドロビン、最小接続、および静的近接性）によって決定された最も適切な GSLB サイトにクライアント要求を送信します。

アクティブ-アクティブ GSLB デプロイメントは次の目的で使用できます。

- すべてのノードがアクティブな状態で、リソース使用率を最適化します。
- リクエストを個々のユーザーに最も近いサイトに誘導することで、ユーザーエクスペリエンスを向上させます。
- ユーザーが定義したペースでアプリケーションをクラウドに移行します。

アクティブ/パッシブ GSLB デプロイメントは次の用途に使用できます。

- 障害回復
- クラウドバースト

参照ドキュメント

- AWS で NetScaler ADC VPX インスタンスを展開する

- AWS ウェブコンソールを使用して NetScaler VPX インスタンスを AWS アウトポストにデプロイする
- NetScaler VPX インスタンスで GSLB を構成する

NetScaler Web App Firewall を使用して **AWS API** ゲートウェイを保護

October 17, 2024

NetScaler アプライアンスを AWS API Gateway の前にデプロイし、API ゲートウェイを外部の脅威から保護できます。NetScaler Web App Firewall（WAF）は、OWASP の上位 10 件の脅威とゼロデイ攻撃から API を保護できます。NetScaler Web App Firewall は、すべての ADC フォームファクターで単一のコードベースを使用します。そのため、あらゆる環境にわたってセキュリティポリシーを一貫して適用し、適用することができます。NetScaler Web App Firewall は導入が簡単で、単一のライセンスとして利用できます。NetScaler Web App Firewall には次の機能があります。

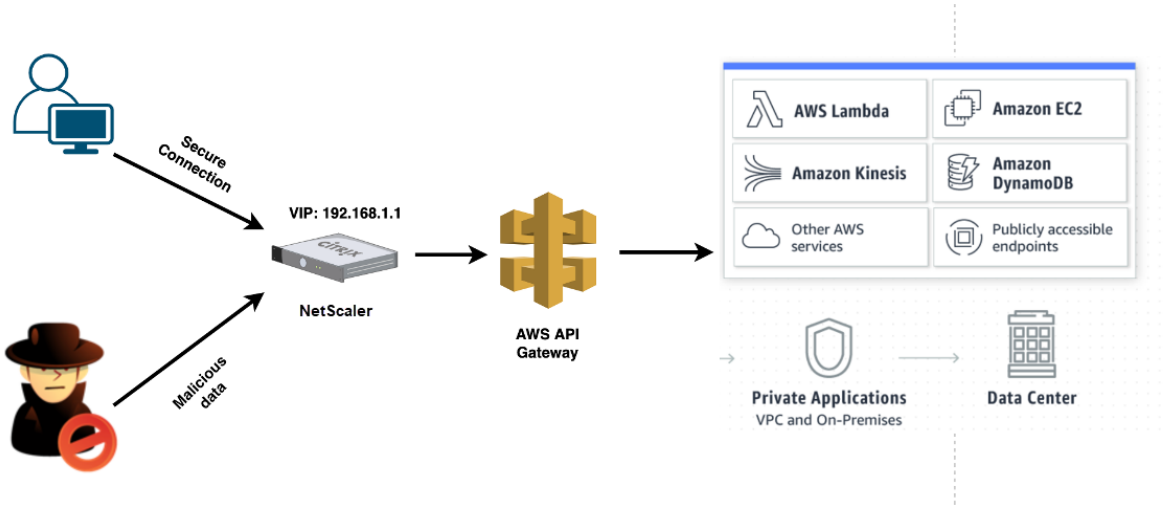
- 構成の簡素化
- ボットの管理
- 総合的な可視性
- 複数のソースからのデータを照合し、統一された画面にデータを表示する

API ゲートウェイ保護に加えて、他の Citrix ADC 機能も使用できます。詳しくは、NetScaler のドキュメントを参照してください。データセンターのフェイルオーバーを回避し、シャットダウン時間を最小限に抑えるだけでなく、アベイラビリティーゾーン内またはアベイラビリティーゾーン間で ADC を高可用性に設定できます。Autoscale 機能でクラスタリングを使用または構成することもできます。

以前、AWS API Gateway は、その背後にあるアプリケーションを保護するために必要な保護をサポートしていませんでした。Web アプリケーションファイアウォール (WAF) 保護がなければ、API はセキュリティ上の脅威にさらされがちでした。

AWS API ゲートウェイの前に **Citrix ADC** アプライアンスをデプロイする

次の例では、NetScaler アプライアンスが AWS API ゲートウェイの前にデプロイされています。



AWS Lambda サービスに対する本物の API リクエストがあるとします。このリクエストは、Amazon API Gateway のドキュメントに記載されているどの APIサービスにも適用できます。上の図に示すように、トラフィックフローは次のようになります。

1. クライアントが AWS Lambda 関数 (XYZ) にリクエストを送信します。このクライアント要求は、Citrix ADC 仮想サーバー（192.168.1.1）に送信されます。
2. 仮想サーバはパケットを検査し、悪意のあるコンテンツがないかチェックします。
3. Citrix ADC アプライアンスは、書き換えポリシーをトリガーして、クライアント要求のホスト名と URL を変更します。たとえば、`https://restapi.citrix.com/default/LamdaFunctionXYZ`を`https://citrix.execute-api.<region>.amazonaws.com/default/LambdaFunctionXYZ`に変更するとします。
4. NetScaler アプライアンスは、このリクエストを AWS API ゲートウェイに転送します。
5. AWS API Gateway はさらに Lambda サービスにリクエストを送信し、Lambda 関数「XYZ」を呼び出します。
6. 同時に、攻撃者が悪意のあるコンテンツを含む API リクエストを送信すると、その悪意のあるリクエストは Citrix ADC アプライアンスに到達します。
7. NetScaler ADC アプライアンスはパケットを検査し、構成されたアクションに基づいてパケットをドロップします。

WAF を有効にして NetScaler ADC アプライアンスを構成する

NetScaler ADC アプライアンスで WAF を有効にするには、次の手順を実行します。

1. コンテンツスイッチまたは負荷分散仮想サーバーを追加します。仮想サーバーの IP アドレスが 192.168.1.1 で、ドメイン名（restapi.citrix.com）に解決されるとします。
2. NetScaler 仮想サーバーで WAF ポリシーを有効にします。詳細については、「Web App Firewall の構成」を参照してください。

3. 書き換えポリシーを有効にして、ドメイン名を変更します。たとえば、「restapi.citrix.com」ドメイン名のロードバランサーへの受信リクエストを、「citrix.execute-api」のバックエンド AWS API ゲートウェイに書き換えるように変更するとします。<region>.amazonaws” ドメイン名。
4. Citrix ADC アプライアンスで L3 モードを有効にして、プロキシとして機能させます。次のコマンドを使用します:

```
enable ns mode L3
```

前の例のステップ 3 で、Web サイト管理者が Citrix ADC アプライアンスで「restapi.citrix.com」ドメイン名を「citrix.execute-api」に置き換えることを望んでいるとします。<region>.amazonaws.com” と入力し、URL に「デフォルト/ラムダ/XYZ」を付けます。

次の手順では、書き換え機能を使用してクライアント要求のホスト名と URL を変更する方法について説明します。

1. SSH を使用して NetScaler ADC アプライアンスにログオンします。
2. 書き換えアクションを追加する。

```
add rewrite action rewrite_host_hdr_act replace "HTTP.REQ.HEADER
    (\"Host\")" "\"citrix.execute-api.<region>.amazonaws.com\""

add rewrite action rewrite_url_act replace HTTP.REQ.URL.
    PATH_AND_QUERY "\"/default/lambda/XYZ\""
```

3. 書き換えアクションの書き換えポリシーを追加します。

```
add rewrite policy rewrite_host_hdr_pol "HTTP.REQ.HEADER(\"Host
    \").CONTAINS(\"restapi.citrix.com\") "rewrite_host_hdr_act

add rewrite policy rewrite_url_pol "HTTP.REQ.HEADER(\"Host\").
    CONTAINS(\"restapi.citrix.com\") "rewrite_url_act
```

4. 書き換えポリシーを仮想サーバにバインドします。

```
bind lb vserver LB_API_Gateway -policyName rewrite_host_hdr_pol
    -priority 10 -gotoPriorityExpression 20 -type REQUEST

bind lb vserver LB_API_Gateway -policyName rewrite_url_pol -
    priority 20 -gotoPriorityExpression END -type REQUEST
```

詳しくは、「Citrix ADC アプライアンスのクライアント要求でホスト名と URL を変更するように書き換えを構成する」を参照してください。

NetScaler の機能と機能

NetScaler ADC アプライアンスは、展開を保護するだけでなく、ユーザーの要件に基づいて要求を強化することもできます。NetScaler ADC アプライアンスには、次の主要な機能があります。

- **API** ゲートウェイの負荷分散：複数の API ゲートウェイがある場合は、Citrix ADC アプライアンスを使用して複数の API ゲートウェイを負荷分散し、API リクエストの動作を定義できます。
  - さまざまな負荷分散方式を使用できます。たとえば、Least 接続メソッドは API Gateway 制限のオーバーロードを回避し、Custom load メソッドは特定の API ゲートウェイの特定の負荷を維持するなどです。詳細については、負荷分散アルゴリズムを参照してください。
  - SSL オフロードは、トラフィックを中断することなく設定されます。
  - [送信元 IP (USIP) を使用] モードを有効にすると、クライアント IP アドレスが保持されます。
  - ユーザー定義の SSL 設定: 独自の署名証明書とアルゴリズムを使用して、独自の SSL 仮想サーバーを作成できます。
  - バックアップ仮想サーバー:API ゲートウェイにアクセスできない場合は、追加のアクションのためにリクエストをバックアップ仮想サーバーに送信できます。
  - 他にも多くの負荷分散機能を使用できます。詳しくは、「Citrix ADC アプライアンスのトラフィックの負荷分散」を参照してください。
- 認証、承認、監査**:** LDAP、SAML、RADIUS などの独自の認証方法を定義し、API リクエストの承認と監査を行うことができます。
- レスポンダー**:** シャットダウン時に API リクエストを他の API Gateway にリダイレクトできます。
- レート制限**:** レート制限機能を設定して、API ゲートウェイの過負荷を回避できます。
- 可用性の向上： NetScaler アプライアンスを高可用性セットアップまたはクラスターセットアップで構成して、AWS API トラフィックの可用性を高めることができます。
- **REST API:** REST API をサポートします。REST API は、クラウド本番環境での作業の自動化に使用できます。
- データの監視**:** 参照用にデータを監視し、ログに記録します。

NetScaler アプライアンスにはさらに多くの機能があり、AWS API ゲートウェイと統合できます。詳しくは、NetScaler のドキュメントを参照してください。

バックエンドの **AWS Autoscaling** サービスを追加する

October 17, 2024

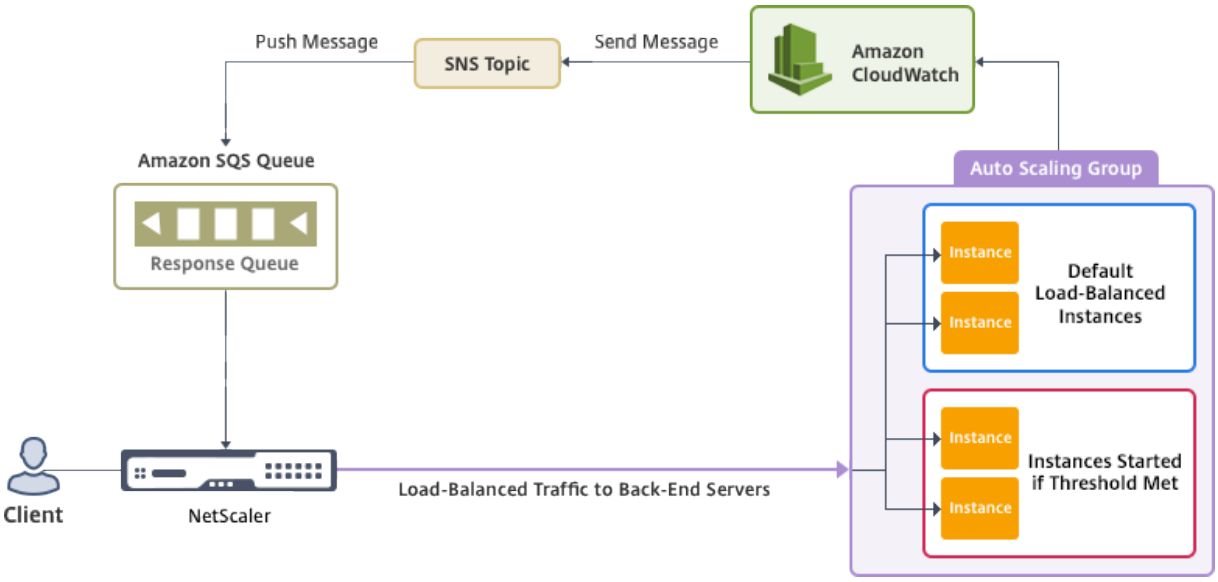
クラウドでアプリケーションを効率的にホストすると、アプリケーションの必要に応じて、リソースを簡単にコスト効率よく管理できます。需要の増大に対応するには、ネットワークリソースをスケールアップする必要があります。需要が収まったら、アイドル状態のリソースによる不要なコストを避けるためにスケールダウンする必要があります。常に必要な数のインスタンスのみをデプロイすることで、アプリケーションの実行コストを最小限に抑えることができます。これを実現するには、トラフィック、メモリ、CPU 使用率などを常に監視する必要があります。しかし、ト

ラフィックを手動で監視するのは煩雑です。アプリケーション環境を動的にスケールアップまたはスケールダウンするには、トラフィックを監視し、必要に応じてリソースをスケールアップまたはスケールダウンするプロセスを自動化する必要があります。

AWS Auto Scaling サービスと統合され、NetScaler VPX インスタンスには次の利点があります。

- 負荷分散と管理: 需要に応じてサーバーをスケールアップおよびスケールダウンするように自動構成します。VPX インスタンスは、バックエンドサブネット内の Autoscale グループを自動検出し、ユーザーが Autoscale グループを選択して負荷を分散できるようにします。これはすべて、VPX インスタンスの仮想 IP アドレスとサブネット IP アドレスを自動構成することによって行われます。
- 高可用性: 複数のアベイラビリティーゾーンにまたがる Autoscale e グループを検出し、サーバーの負荷を分散します。
- ネットワークの可用性の向上:VPX インスタンスは以下をサポートします。
  - VPC ピアを使用した異なる VPC のバックエンドサーバー
  - 同じプレイスメントグループのバックエンドサーバー
  - 異なるアベイラビリティゾーンのバックエンドサーバー
- 正常な接続終了**:** グレースフルタイムアウト機能を使用して、スケールダウンアクティビティが発生してもクライアント接続が失われないように、Autoscale サーバーを正常に削除します。
- スタンバイサーバーの接続ドレイニング: スタンバイ状態のサーバーに新しいクライアント接続を送信しないようにします。ただし、スタンバイサーバーはまだオートスケーリンググループの一部であり、閉じられるまで既存のクライアント接続を処理し続けます。サーバーが InService 状態に戻ると、サーバーは新しい接続の処理を再開します。スタンバイ状態を使用して、サーバーを更新、変更、またはトラブルシューティングしたり、要件に基づいてスケールダウンしたりできます。スタンバイ状態を使用して、サーバーの更新、変更、トラブルシューティングを行ったり、要件に基づいてスケールダウンしたりできます。詳細については、AWS のドキュメントを参照してください。

図:NetScaler VPX インスタンスによる AWS オートスケーリングサービス



この図は、AWS オートスケーリングサービスが NetScaler VPX インスタンス（負荷分散仮想サーバー）とどのように互換性があるかを示しています。詳しくは、次の AWS のトピックを参照してください。

- オートスケーリンググループ
- CloudWatch
- Simple Notification Service（SNS）
- シンプルキューサービス (Amazon SQS)

はじめに

NetScaler VPX インスタンスで自動スケーリングの使用を開始する前に、次のタスクを完了する必要があります。

- 次のトピックをお読みください。
  - 前提条件
  - 制限事項と使用上のガイドライン
- 要件に応じて、AWS で NetScaler VPX インスタンスを作成します。
  - NetScaler VPX スタンドアロン インスタンスの作成方法の詳細については、「AWS に NetScaler VPX スタンドアロン インスタンスをデプロイする 」および「シナリオ: スタンドアロン インスタンス」を参照してください。
  - VPX インスタンスを HA モードでデプロイする方法の詳細については、「AWS に高可用性ペアをデプロイする」を参照してください。

  > 注
  >
  > 以下をお勧めします:
  >
  > - AWS 上で NetScaler VPX インスタンスを作成するには、クラウドフォーメーションテンプレートを使用してください。
  > - 3 つの個別のインターフェイスを作成します。1 つは管理 (NSIP) 用、もう 1 つはクライアント側の LB 仮想サーバー (VIP) 用、もう 1 つはサブネット IP (NSIP) 用です。

- AWS Autoscale グループを作成します。既存の自動スケーリング設定がない場合は、次のことを行う必要があります。

  1. 起動設定を作成する
  2. Autoscaling グループの作成
  3. Autoscaling グループの検証

  詳しくは、http://docs.aws.amazon.com/autoscaling/latest/userguide/GettingStartedTutorial.htmlを参照してください。

- NetScaler リリース 14.1-12.x 以降、AWS Autoscale グループでは、グレースフルオプションを有効にしている場合にのみスケールダウンポリシーを指定する必要があります。NetScaler 14.1-12.x より前のリリースでは、Graceful オプションが有効かどうかに関係なく、少なくとも 1 つのスケールダウン ポリシーを指定する必要がありました。

  NetScaler VPX インスタンスは、ステップスケーリングポリシーのみをサポートしています。簡易スケーリングポリシーとターゲット追跡スケーリングポリシーは、Autoscale グループではサポートされていません。

- AWS アカウントに次の IAM 権限があることを確認してください:

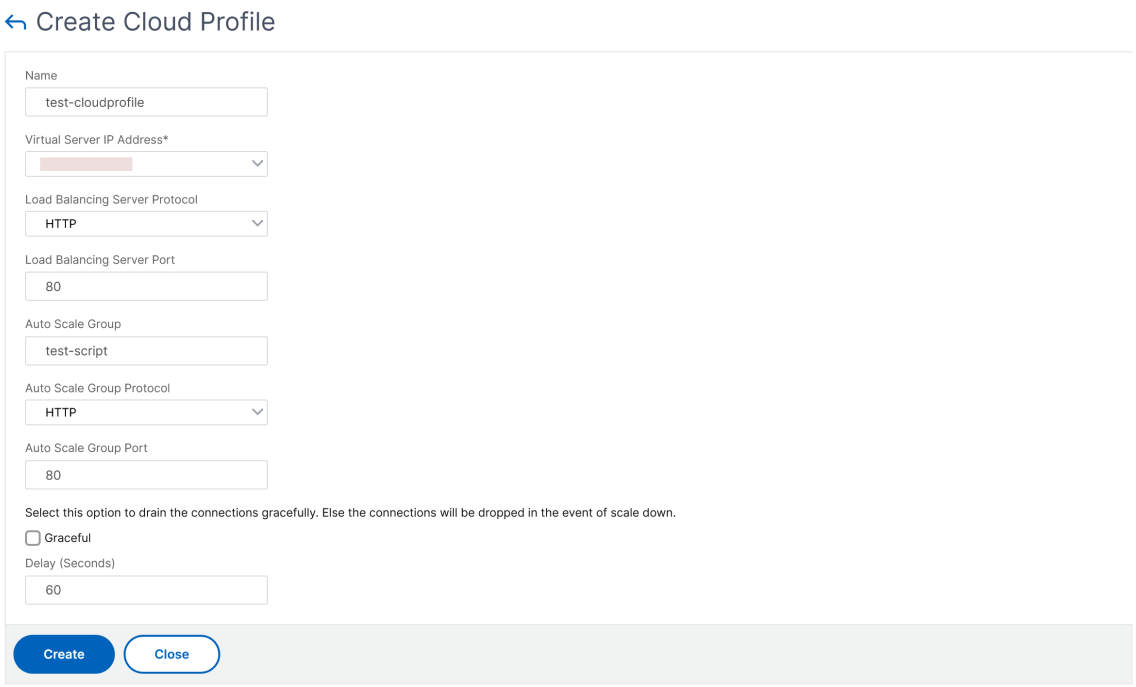
```
{

    "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"
           }

    \]
 }
```

AWS 自動スケーリングサービスを NetScaler VPX インスタンスに追加する

次の手順を実行して、自動スケーリングサービスを VPX インスタンスに追加します:

1. nsrootの認証情報を使用して VPX インスタンスにログオンします。
2. [システム] **>** **[AWS]** **>** **[**クラウドプロファイル] に移動し、[追加] をクリックします。

   クラウドプロファイルの作成設定ページが表示されます。



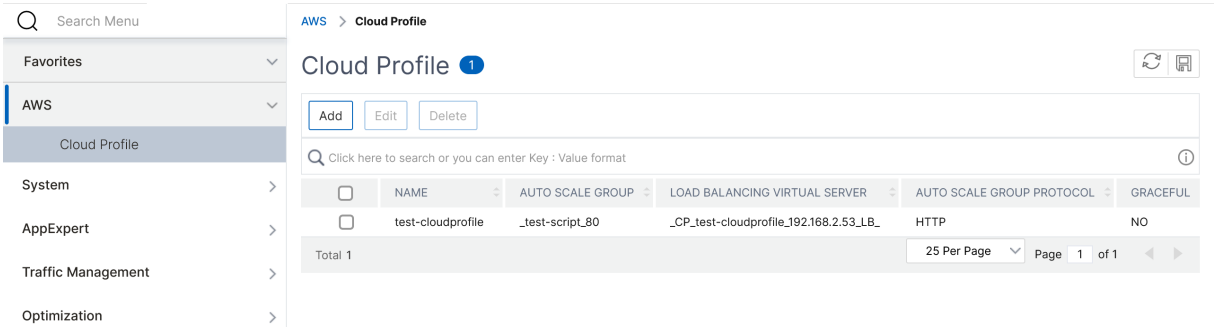
クラウドプロファイルを作成する際の注意点:

- 仮想サーバーの IP アドレスは、VPX インスタンスで使用可能な空き IP アドレスから自動入力されます。詳細については、「複数の IP アドレスの管理」を参照してください。
- AWS アカウントで設定した Autoscale グループの正確な名前を入力します。詳細については、「AWS Auto Scaling グループ」を参照してください。
- オートスケーリンググループのプロトコルとポートを選択する際には、サーバーがそれらのプロトコルとポートでリッスンしていることを確認し、サービスグループに正しいモニターをバインドしてください。デフォルトでは、TCP モニターが使用されます。
- SSL プロトコルタイプの自動スケーリングでは、クラウドプロファイルを作成した後、証明書がないために負荷分散仮想サーバーまたはサービスグループがダウンしているように見えます。証明書は、仮想サーバまたはサービスグループに手動でバインドできます。
- Autoscale e サーバーを正常に削除するには、「**Graceful** 」を選択し、「**Delay** 」フィールドにタイムアウト値を指定します。このオプションはスケールダウンイベントを開始します。VPX インスタンスはサーバーをすぐには削除しませんが、サーバーの 1 つを正常に削除するようにマークします。この間、VPX インスタンスはこのサーバーへの新規接続を許可しません。既存の接続は、タイムアウトが発生するまで処理されます。タイムアウト後、VPX インスタンスはサーバーを削除します。

  Graceful オプションを選択しない場合、負荷が下がった直後に Autoscale グループのサーバーが削除されます。これにより、接続されている既存のクライアントのサービスが中断される可能性があります。

クラウドプロファイルを作成すると、NetScaler 負荷分散仮想サーバーと、自動スケーリンググループのサーバーとしてメンバーを含むサービスグループが作成されます。バックエンドサーバーは、VPX インスタンスで構成された

SNIP を介して到達可能である必要があります。



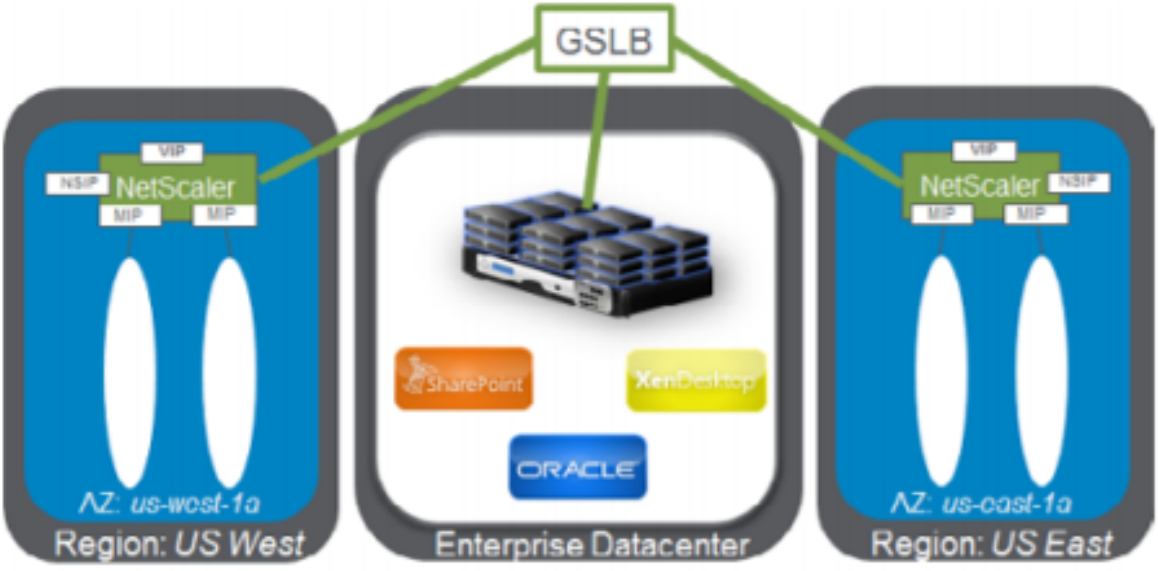
注

- **AWS** コンソールでオートスケール関連の情報を表示するには、**[**EC2**] > [ダッシュボード] > [**自動スケーリング**]> [Auto Scaling** Group] に移動します。
- AWS の同じ AutoScaling Group (ASG) を使用して、サービスごとに (異なるポートを使用して) 異なるクラウドプロファイルを作成できます。したがって、NetScaler VPX インスタンスは、パブリッククラウド内の同じ自動スケーリンググループを持つ複数のサービスをサポートします。

NetScaler GSLB を AWS に展開

October 17, 2024

GSLB for NetScaler on AWS を設定するには、基本的に、NetScaler が属する VPC の外部にあるサーバー（別のアベイラビリティリージョンの別の VPC 内やオンプレミスのデータセンターなど）にトラフィックを負荷分散するように NetScaler を構成する必要があります。



DBS の概要

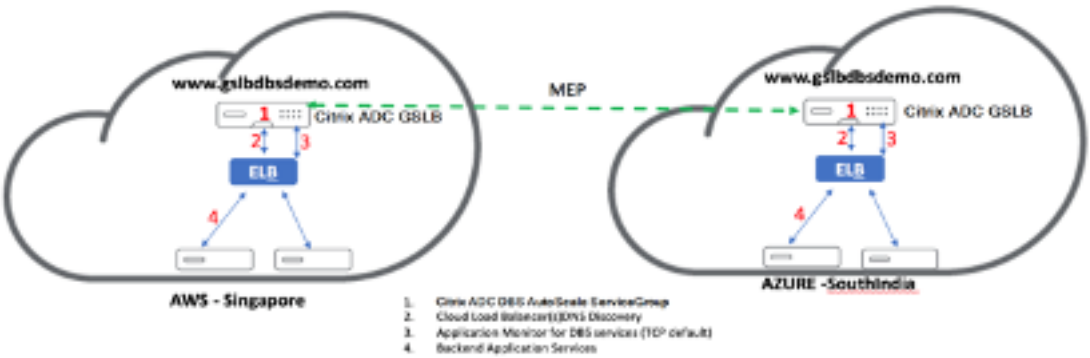
クラウド ロード バランサー用のドメイン名ベース サービス (DBS) を使用した NetScaler GSLB サポートにより、クラウド ロード バランサー ソリューションを使用して動的なクラウド サービスを自動検出できるようになります。この構成により、NetScaler はアクティブ/アクティブ環境でグローバル サーバー負荷分散ドメイン名ベース サービス (GSLB DBS) を実装できます。DBS では、DNS 検出から AWS 環境のバックエンドリソースを拡張できます。

このセクションでは、AWS AutoScaling 環境における NetScaler 間の統合について説明します。このドキュメントの最後のセクションでは、AWS リージョンに固有の 2 つの異なる可用性ゾーン（AZ）にまたがる NetScaler ADC の HA ペアを設定する機能について詳しく説明します。

DBS と ELB

GSLB DBS は、ユーザー Elastic Load Balancer (ELB) の FQDN を利用して、AWS 内で作成および削除されるバックエンドサーバーを含むように GSLB サービスグループを動的に更新します。AWS のバックエンドサーバーまたはインスタンスは、ネットワーク需要または CPU 使用率に基づいてスケーリングするように設定できます。この機能を構成するには、NetScaler を ELB にポイントして、AWS 内でインスタンスが作成および削除されるたびに NetScaler を手動で更新しなくても、AWS 内のさまざまなサーバーに動的にルーティングできます。GSLB サービス グループの NetScaler DBS 機能は、DNS 対応のサービス検出を使用して、Autoscale グループで識別された DBS 名前空間のメンバー サービス リソースを決定します。

クラウド ロード バランサーを備えた NetScaler GSLB DBS Autoscale コンポーネント:



AWS コンポーネントの設定

セキュリティグループ

> 注
>
> ELB、NetScaler GSLB インスタンス、Linux インスタンスには、それぞれ必要なルールセットが異なるため、

> 異なるセキュリティグループを作成することをお勧めします。この例では、簡潔にするために、統合セキュリティグループ設定があります。

仮想ファイアウォールが適切に構成されていることを確認するには、「VPC のセキュリティ グループ」を参照してください。

1. ユーザー **AWS** リソースグループにログインし、[**EC2] > [**ネットワークとセキュリティ**] > [**セキュリティグループ] に移動します。
2. [セキュリティグループの作成] をクリックし、名前と説明を入力します。このセキュリティグループには、NetScaler と Linux のバックエンド Web サーバーが含まれます。
3. 次のスクリーンショットから受信ポートルールを追加します。

> 注
>
> きめ細かなセキュリティ強化には、ソース IP アクセスを制限することが推奨されます。詳細については、「Web サーバー ルール」を参照してください。

1. Amazon Linux バックエンドウェブサービス

   a) ユーザー **AWS** リソースグループにログインし、[**EC2] > [**インスタンス] に移動します。

   b) 以下の詳細を使用して [インスタンスを起動] をクリック し、**Amazon Linux** インスタンスを設定し ます。

   このインスタンスでの Web サーバーまたはバックエンド サービスの設定に関する詳細を入力します。

2. NetScaler の構成

   a) ユーザー **AWS** リソースグループにログインし、[**EC2] > [**インスタンス] に移動します。

   b) [**Launch Instance**] をクリックし、次の詳細を使用して **Amazon AMI** インスタンスを設定します。

3. エラスティック IP 設定

> 注
>
> NetScaler は、NSIP 用のパブリック IP を持たないことで、コストを削減するために必要に応じて単一の Elastic IP で実行することもできます。代わりに、GSLB サイト IP と ADNS IP に加えて、ボックスへの管理アクセスをカバーできる Elastic IP を SNIP に添付します。

```
1. ユーザー**AWS リソースグループにログインし**、[**EC2] > [ネットワークとセキュリティ] > [Elastic IP**] に移動します。

1. Elastic IP アドレスを作成するには、[**新しいアドレスの割り当て**] をクリックします。

1. AWS内でNetScalerインスタンスを実行しているユーザーを指すように Elastic IPを設定します。
```



```
6
7  1.  2つ目のElastic IPを構成し、NetScalerインスタンスを実行しているユーザーに再度割り当てます。
```

1. エラスティックロードバランサー

   a) ユーザー **AWS** リソースグループにログインし、[**EC2**] **>** [負荷分散] **>** [ロードバランサー] に移動します。

   b) [**Create Load Balancer**] をクリックして、クラシックロードバランサーを設定します。

   ユーザー Elastic Load Balancers を使用すると、ユーザーはバックエンド Amazon Linux インスタンスの負荷を分散できると同時に、需要に基づいてスピンアップされる他のインスタンスの負荷を分散することもできます。

グローバルサーバー負荷分散ドメイン名ベースのサービスの設定

トラフィック管理の構成については、「NetScaler GSLB ドメインベース サービスの構成」を参照してください。

デプロイメントの種類

3 つの NIC の展開

- 典型的な展開
  - GSLB StyleBook
  - ADM と
  - GSLB (ドメイン登録ありの Route53)
  - ライセンス-プール/マーケットプレイス
- 使用例
  - 3 つの NIC の展開は、データと管理トラフィックの実際の分離を実現するために使用されます。
  - 3 つの NIC を導入すると、ADC のスケールとパフォーマンスも向上します。
  - 3 つの NIC の展開は、スループットが通常 1 Gbps 以上であるネットワーク アプリケーションで使用され、3 つの NIC の展開が推奨されます。

CFT デプロイメント

お客様は、デプロイをカスタマイズする場合や、デプロイを自動化する場合、CloudFormation テンプレートを使用してデプロイします。

展開手順

展開手順は次のとおりです。

1. GSLB 用の 3 つの NIC デプロイメント
2. ライセンス
3. 展開オプション

GSLB 用の **3** つの **NIC** デプロイメント　NetScaler VPX インスタンスは、AWS Marketplace では Amazon マシンイメージ（AMI）として入手でき、AWS VPC 内のエラスティックコンピューティングクラウド（EC2）インスタンスとして起動できます。NetScaler VPX でサポートされる AMI として許可されている最小 EC2 インスタンスタイプは m4.large です。NetScaler VPX AMI インスタンスには、最低 2 つの仮想 CPU と 2 GB のメモリが必要です。また、AWS VPC 内で起動される EC2 インスタンスは、複数のインターフェイス、インターフェイスごとに複数の IP アドレス、VPX 構成に必要なパブリックおよびプライベート IP アドレスも提供できます。各 VPX インスタンスには、少なくとも 3 つの IP サブネットが必要です。

- 管理サブネット
- クライアント側サブネット（VIP）
- バックエンド向けサブネット (SNIP)

NetScaler では、AWS に標準 VPX インスタンスをインストールする場合、3 つのネットワークインターフェイスを推奨しています。

現在、AWS では、AWS VPC 内で実行しているインスタンスでのみ、マルチ IP 機能を使用できます。VPC 内の VPX インスタンスを使用して、EC2 インスタンスで実行しているサーバーの負荷を分散できます。Amazon VPC を使用すると、ユーザーは、独自の IP アドレス範囲、サブネット、ルートテーブル、ネットワークゲートウェイなどの仮想ネットワーク環境を作成および制御できます。

> 注
>
> デフォルトでは、ユーザーは AWS アカウントごとに AWS リージョンごとに最大 5 つの VPC インスタンスを作成できます。ユーザーは、Amazon のリクエストフォーム「Amazon VPC リクエスト」を送信することで、VPC 制限の引き上げをリクエストできます。

ライセンス　AWS 上の NetScaler VPX インスタンスにはライセンスが必要です。AWS で実行されている NetScaler VPX インスタンスでは、次のライセンスオプションを使用できます。

- 無料 (無制限)
- 毎時
- 年次
- 自分のライセンスを持参する
- 無料トライアル（すべての NetScaler VPX-AWS サブスクリプションは、AWS Marketplace で 21 日間無料）。

展開オプション　ユーザーは、AWS 上に NetScaler VPX スタンドアロンインスタンスを展開できます。ユーザーは NetScaler VPX スタンドアロンインスタンスを AWS にデプロイできます。詳しくは、「NetScaler VPX スタンドアロンインスタンスを AWS にデプロイする」を参照してください

ハイブリッドおよびマルチクラウド展開向けの **NetScaler** グローバル サーバー負荷分散

NetScaler ハイブリッドおよびマルチクラウドのグローバルサーバー負荷分散（GSLB）ソリューションにより、ユーザーはハイブリッドクラウド、複数のクラウド、およびオンプレミス展開の複数のデータセンターにアプリケーショントラフィックを分散できます。NetScaler のハイブリッドおよびマルチクラウド GSLB ソリューションにより、ユーザーは既存の設定を変更することなく、ハイブリッドまたはマルチクラウド環境で負荷分散設定を管理できます。また、ユーザーがオンプレミス環境を使用している場合は、クラウドに完全に移行する前に、NetScaler ハイブリッドおよびマルチクラウドの GSLB ソリューションを使用して一部のサービスをクラウドでテストできます。たとえば、ユーザーはトラフィックのごく一部しかクラウドにルーティングできず、トラフィックのほとんどをオンプレミスで処理できます。また、NetScaler のハイブリッドおよびマルチクラウド GSLB ソリューションにより、ユーザーは地理的に離れた場所にある NetScaler インスタンスを単一の統合コンソールから管理および監視できます。

ハイブリッドおよびマルチクラウドアーキテクチャは、「ベンダーロックイン」を回避し、さまざまなインフラストラクチャを使用してユーザーパートナーや顧客のニーズを満たすことで、企業全体のパフォーマンスを向上させることもできます。マルチクラウド アーキテクチャにより、ユーザーは使用した分だけ支払うことになるため、インフラストラクチャ コストをより適切に管理できます。また、オンデマンドでインフラストラクチャを使用するようになったため、ユーザーはアプリケーションをより適切に拡張できます。また、クラウド間ですばやく切り替えて、各プロバイダーの最高のサービスを活用することもできます。

NetScaler GSLB ノードは DNS 名の解決を処理します。これらの GSLB ノードはいずれも、任意のクライアントロケーションから DNS リクエストを受信できます。DNS リクエストを受信する GSLB ノードは、設定された負荷分散方法で選択されたロードバランサー仮想サーバーの IP アドレスを返します。メトリクス（サイト、ネットワーク、およびパーシスタンスメトリック）は、独自の NetScaler プロトコルであるメトリック交換プロトコル（MEP）を使用して GSLB ノード間で交換されます。MEP プロトコルの詳細については、「メトリック交換プロトコルの構成」を参照してください。

GSLB ノードに設定されたモニターは、同じデータセンター内の負荷分散仮想サーバーのヘルスステータスを監視します。親子トポロジでは、GSLB ノードと NetScaler ノード間のメトリックは MEP を使用して交換されます。ただし、親子トポロジーでは、GSLB と NetScaler LB ノード間のモニタープローブの構成はオプションです。

NetScaler エージェントは、NetScaler ADM とユーザー データセンター内の管理対象インスタンス間の通信を可能にします。NetScaler エージェントとそのインストール方法の詳細については、「はじめに」を参照してください。

> 注
>
> このドキュメントでは、次の前提条件を定めています。
>
> - ユーザーが既存の負荷分散設定を持っている場合は、起動して実行中です。
> - SNIP アドレスまたは GSLB サイトの IP アドレスは、NetScaler GSLB ノードごとに構成されていま

> す。この IP アドレスは、他のデータセンターとメトリックスを交換するときに、データセンターのソース IP アドレスとして使用されます。
>
> - 各 NetScaler GSLB インスタンスには、DNS トラフィックを受信するように ADNS または ADNS-TCP サービスが構成されています。
> - 必要なファイアウォールとセキュリティグループは、クラウドサービスプロバイダーで設定されます。

セキュリティグループの設定

ユーザーは、クラウドサービスプロバイダーで必要なファイアウォール/セキュリティグループ構成を設定する必要があります。AWS セキュリティ機能の詳細については、AWS/Documentation/Amazon VPC/User Guide/Securityを参照してください。

また、GSLB ノードでは、ユーザーは MEP トラフィック交換用の ADNS サービス/DNS サーバーの IP アドレス用にポート 53 を開き、GSLB サイトの IP アドレス用にポート 3009 を開く必要があります。負荷分散ノードでは、ユーザーはアプリケーショントラフィックを受信するために適切なポートを開く必要があります。たとえば、ユーザーは HTTP トラフィックを受信するためにポート 80 を開き、HTTPS トラフィックを受信するためにポート 443 を開く必要があります。NetScaler エージェントと NetScaler ADM 間の NITRO 通信用にポート 443 を開きます。

動的ラウンドトリップタイム GSLB 方式では、ユーザーはポート 53 を開いて、設定されている LDNS プローブタイプに応じて UDP および TCP プローブを許可する必要があります。UDP または TCP プローブは SNIP の 1 つを使用して開始されるため、この設定はサーバー側のサブネットにバインドされたセキュリティグループに対して行う必要があります。

NetScaler ハイブリッドおよびマルチクラウド **GSLB** ソリューションの機能

このセクションでは、NetScaler ハイブリッドおよびマルチクラウド GSLB ソリューションの機能の一部について説明します。

他の負荷分散ソリューションとの互換性

NetScaler のハイブリッドおよびマルチクラウド GSLB ソリューションは、NetScaler ロードバランサー、NGINX、HAProxy、その他のサードパーティ製ロードバランサーなど、さまざまな負荷分散ソリューションをサポートしています。

> 注
>
> NetScaler 以外の負荷分散ソリューションは、近接ベースおよび非メトリックベースの GSLB メソッドが使用され、親子トポロジが構成されていない場合にのみサポートされます。

GSLB の方式

NetScaler のハイブリッドおよびマルチクラウド GSLB ソリューションは、以下の GSLB メソッドをサポートしています。

- メトリックベースの GSLB メソッド。メトリックベースの GSLB メソッドは、メトリック交換プロトコルを介して他の NetScaler ノードからメトリックを収集します。
  - 最小接続: クライアント要求は、アクティブな接続数が最も少ないロードバランサーにルーティングされます。
  - 最小帯域幅: クライアント要求は、現在最も少ない量のトラフィックを処理しているロードバランサーにルーティングされます。
  -
- 非メトリックベースの GSLB メソッド
  - ラウンドロビン: クライアントリクエストは、ロードバランサーのリストの上部にあるロードバランサーの IP アドレスにルーティングされます。その後、そのロードバランサーはリストの一番下に移動します。
  - ソース IP ハッシュ: このメソッドは、クライアント IP アドレスのハッシュ値を使用してロードバランサーを選択します。
- 近接ベースの GSLB メソッド
  - 静的近接: クライアントリクエストは、クライアント IP アドレスに最も近いロードバランサーにルーティングされます。
  - ラウンドトリップ時間 (RTT): この方法では、RTT 値 (クライアントのローカル DNS サーバーとデータセンター間の接続における遅延時間) を使用して、最もパフォーマンスの高いロードバランサーの IP アドレスを選択します。

負荷分散方法の詳細については、「負荷分散アルゴリズム」を参照してください。

GSLB トポロジ

NetScaler のハイブリッドおよびマルチクラウド GSLB ソリューションは、アクティブ/パッシブトポロジーと親子トポロジーをサポートします。

- アクティブ/パッシブトポロジ: 障害点からの保護により、災害復旧を実現し、アプリケーションの継続的な可用性を確保します。プライマリデータセンターがダウンすると、パッシブデータセンターは運用可能になります。GSLB アクティブ/パッシブ トポロジの詳細については、「災害復旧用の GSLB の構成」を参照してください。

- 親子トポロジ–顧客がメトリックベースの GSLB 方式を使用して GSLB および負荷分散ノードを構成しており、負荷分散ノードが別の NetScaler インスタンスに展開されている場合に使用できます。親子トポロジでは、LB ノード（子サイト）は NetScaler アプライアンスである必要があります。親サイトと子サイト間のメトリックの交換はメトリック交換プロトコル（MEP）を介して行われます。

親子トポロジの詳細については、「MEP プロトコルを使用した親子トポロジの展開」を参照してください。

IPv6 サポート

NetScaler のハイブリッドおよびマルチクラウド GSLB ソリューションは IPv6 もサポートしています。

監視

NetScaler のハイブリッドおよびマルチクラウド GSLB ソリューションは、安全な接続を有効にするオプションを備えた組み込みモニターをサポートしています。ただし、LB 構成と GSLB 構成が同じ NetScaler インスタンス上にある場合、または親子トポロジーが使用されている場合、モニターの構成は任意です。

永続性

NetScaler のハイブリッドおよびマルチクラウド GSLB ソリューションは以下をサポートします:

- ソース IP ベースの永続性セッション。これにより、設定されたタイムアウトウィンドウ内に到達した場合に、同じクライアントからの複数の要求が同じサービスに送信されます。クライアントが別の要求を送信する前にタイムアウト値が期限切れになると、セッションは破棄され、構成された負荷分散アルゴリズムを使用して、クライアントの次の要求に対して新しいサーバーが選択されます。
- プライマリへの負荷がしきい値を下回った後も、バックアップ仮想サーバは受信した要求を処理し続けます。詳細については、「スピルオーバーの構成」を参照してください。
- サイトパーシステンスにより、GSLB ノードがクライアントリクエストを処理するデータセンターを選択し、選択したデータセンターの IP アドレスを以降のすべての DNS リクエストに転送します。構成された永続性がダウンしているサイトに適用される場合、GSLB ノードは GSLB メソッドを使用して新しいサイトを選択し、新しいサイトはクライアントからのその後の要求に対して永続的になります。

NetScaler ADM スタイルブックを使用した構成

お客様は、NetScaler ADM 上のデフォルトのマルチクラウド GSLB スタイルブックを使用して、ハイブリッドおよびマルチクラウド GSLB 構成で NetScaler インスタンスを構成できます。

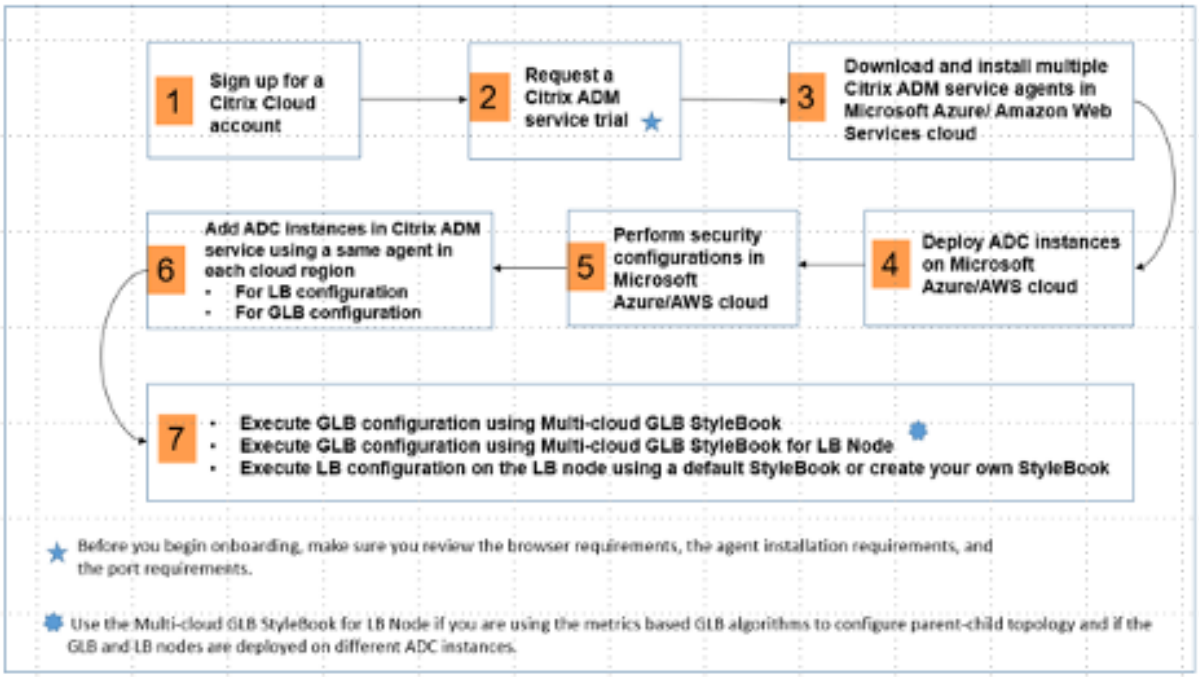
お客様は、ロード バランシング ノード スタイルブックのデフォルトのマルチクラウド GSLB スタイルブックを使用して、アプリケーション トラフィックを処理する親子トポロジ内の子サイトである NetScaler ロード バランシング

ノードを構成できます。このスタイルブックは、ユーザーが親子トポロジで負荷分散ノードを構成する場合にのみ使用してください。ただし、各 LB ノードは、この StyleBook を使用して個別に設定する必要があります。

NetScaler ハイブリッドおよびマルチクラウド **GSLB** ソリューション構成のワークフロー

お客様は、NetScaler ADM に同梱されているマルチクラウド GSLB スタイルブックを使用して、ハイブリッドおよびマルチクラウド GSLB 構成で NetScaler インスタンスを構成できます。

次の図は、NetScaler ハイブリッドおよびマルチクラウド GSLB ソリューションを構成するためのワークフローを示しています。ワークフロー図の手順については、図の後で詳しく説明します。



クラウド管理者として次のタスクを実行します:

1. NetScaler ラウドアカウントにサインアップしてください。

   NetScaler ADM の使用を開始するには、NetScaler Cloud 企業アカウントを作成するか、社内の誰かが作成した既存のアカウントに参加します。

2. ユーザーが NetScaler Cloud にログオンした後、**NetScaler** アプリケーション配信管理 タイルの 「管理**」をクリック して、ADM サービスを初めて設定します。

3. 複数の NetScaler ADM サービス エージェントをダウンロードしてインストールします。

   NetScaler ADM とデータセンターまたはクラウド内の管理対象インスタンス間の通信を可能にするには、ユーザーはネットワーク環境に NetScaler ADM サービス エージェントをインストールして構成する必要があります。各リージョンにエージェントをインストールして、管理対象インスタンスで LB と GSLB の設定を構成できるようにします。LB 構成と GSLB 構成では、1 つのエージェントを共有できます。上記の 3 つのタスクの詳細については、「はじめに」を参照してください。

4. Microsoft AWS クラウド/オンプレミスのデータセンターにロードバランサーをデプロイします。

   ユーザーがクラウドとオンプレミスにデプロイするロードバランサーのタイプに応じて、それに応じてプロビジョニングします。たとえば、ユーザーは Amazon Web Services（AWS）の仮想プライベートクラウドとオンプレミスのデータセンターに NetScaler VPX インスタンスをプロビジョニングできます。仮想マシンを作成して他のリソースを構成することにより、NetScaler インスタンスがスタンドアロンモードで LB または GSLB ノードとして機能するように構成します。NetScaler VPX インスタンスを展開する方法の詳細については、次のドキュメントを参照してください：

   - AWS 上の NetScaler VPX。
   - NetScaler VPX スタンドアロンインスタンスを構成します。

5. セキュリティ設定を実行します。

   ARM と AWS でネットワークセキュリティグループとネットワーク ACL を設定し、ユーザーインスタンスとサブネットのインバウンドトラフィックとアウトバウンドトラフィックを制御します。

6. NetScaler ADM に NetScaler インスタンスを追加します。

   NetScaler インスタンスは、ユーザーが NetScaler ADM から検出、管理、監視するネットワーク アプライアンスまたは仮想アプライアンスです。これらのインスタンスを管理および監視するには、ユーザーはインスタンスをサービスに追加し、LB（ユーザーが NetScaler for LB を使用している場合）と GSLB インスタンスの両方を登録する必要があります。NetScaler ADM に NetScaler インスタンスを追加する方法の詳細については、「はじめに」を参照してください。

7. デフォルトの NetScaler ADM スタイルブックを使用して、GSLB および LB 構成を実装します。

   - マルチクラウド GSLB StyleBook を使用して、選択した GSLB NetScaler インスタンスで GSLB 構成を実行します。
   - 負荷分散設定を実装します。(管理対象インスタンスにすでに LB 構成がある場合は、この手順をスキップできます。) ユーザーは、次の 2 つの方法のいずれかで NetScaler インスタンスにロード バランサーを構成できます。
   - アプリケーションの負荷分散のためにインスタンスを手動で設定します。インスタンスを手動で構成する方法の詳細については、「基本的な負荷分散の設定」を参照してください。
   - StyleBook を使用してください。ユーザーは、NetScaler ADM スタイルブック (HTTP/SSL ロード バランシング スタイルブックまたは HTTP/SSL ロード バランシング (モニター付き) スタイルブック) のいずれかを使用して、選択した NetScaler インスタンスにロード バランサー構成を作成できます。ユーザーは独自の StyleBook を作成することもできます。StyleBook について詳しくは、「StyleBook」を参照してください。

8. 次のいずれかの場合に GSLB 親子トポロジを構成するには、LB ノード用のマルチクラウド GSLB スタイルブックを使用します。

- ユーザーがメトリックベースの GSLB アルゴリズム (最小パケット、最小接続、最小帯域幅) を使用して GSLB および負荷分散ノードを構成しており、負荷分散ノードが別の NetScaler インスタンスに展開されている場合。
- サイトの永続性が必要な場合。

StyleBooks を使用して **NetScaler** 負荷分散ノードで **GSLB** を構成する

メトリックベースの GSLB アルゴリズム (最小パケット、最小接続、最小帯域幅) を使用して GSLB および負荷分散ノードを構成しており、負荷分散ノードが別の NetScaler インスタンスに展開されている場合、お客様は **LB** ノード用のマルチクラウド **GSLB** スタイルブックを使用できます。

ユーザーはこの StyleBook を使用して、既存の親サイトに対してさらに多くの子サイトを構成することもできます。この StyleBook は、一度に 1 つの子サイトを構成します。したがって、この StyleBook から子サイトと同じ数の構成（構成パック）を作成します。StyleBook は子サイトに GSLB 設定を適用します。ユーザーは最大 1024 の子サイトを構成できます。

> 注
>
> マルチクラウド GSLB StyleBook を使用して親サイトを構成します。

この StyleBook では、次の前提条件があります:

- SNIP アドレスまたは GSLB サイトの IP アドレスが設定されています。
- 必要なファイアウォールとセキュリティグループは、クラウドサービスプロバイダーで設定されます。

LB ノード用のマルチクラウド **GSLB** スタイルブックを使用して親子トポロジで子サイトを構成する

1. アプリケーション **>** 構成 **>** 新規作成に移動します。
2. [アプリケーション] > [構成] に移動し 、[新規作成] をクリック します。

   StyleBook は、この StyleBook で定義されているすべてのパラメータの値を入力できるユーザー・インタフェース・ページとして表示されます。

> 注
>
> このドキュメントでは、データセンターとサイトという用語は同じ意味で使用されています。

1. 次のパラメーターを設定します:
   - アプリケーション名。子サイトを作成する GSLB サイトにデプロイされている GSLB アプリケーションの名前を入力します。
   - プロトコル。ドロップダウンリストボックスから、デプロイされたアプリケーションのアプリケーションプロトコルを選択します。

- **LB** ヘルスチェック (オプション)
- ヘルスチェックの種類。ドロップダウンリストボックスから、サイト上のアプリケーションを表すロードバランサー VIP アドレスの正常性のチェックに使用するプローブのタイプを選択します。
- セキュアモード。(オプション) SSL ベースのヘルスチェックが必要な場合は、[はい] を選択してこのパラメーターを有効にします。
- **HTTP** リクエスト。(オプション) ユーザがヘルスチェックタイプとして HTTP を選択した場合は、VIP アドレスのプローブに使用される完全な HTTP 要求を入力します。
- **HTTP** ステータス応答コードのリスト。(オプション) ユーザがヘルスチェックタイプとして HTTP を選択した場合は、VIP が正常であるときに HTTP 要求への応答で予想される HTTP ステータスコードのリストを入力します。

2. 親サイトを構成します。
   - 子サイト (LB ノード) を作成する親サイト (GSLB ノード) の詳細を指定します。
     - サイト名。サイトの名前を入力します。
     - サイト **IP** アドレス。親サイトが他のサイトとメトリックを交換するときにソース IP アドレスとして使用する IP アドレスを入力します。この IP アドレスは、各サイトの GSLB ノードですでに設定されていることを前提としています。
     - サイトのパブリック **IP** アドレス。(オプション) メトリックの交換に使用される子サイトのパブリック IP アドレスを入力します (そのサイトの IP アドレスが NAT の場合)。
3. 子サイトを構成します。
   - 子サイトの詳細を入力します。
     - サイト名。親サイトの名前を入力します。
     - サイト **IP** アドレス。子サイトの IP アドレスを入力します。ここでは、子サイトとして構成されている NetScaler ノードのプライベート IP アドレスまたは SNIP を使用します。
     - サイトのパブリック **IP** アドレス。(オプション) メトリックの交換に使用される親サイトのパブリック IP アドレスを入力します (そのサイトの IP アドレスが NAT の場合)。
4. アクティブな GSLB サービスの設定 (オプション)
   - LB 仮想サーバーの IP アドレスがパブリック IP アドレスでない場合にのみ、アクティブな GSLB サービスを構成します。このセクションでは、ユーザーがアプリケーションがデプロイされているサイトのローカル GSLB サービスのリストを設定できます。
     - サービス **IP**。このサイトの負荷分散仮想サーバーの IP アドレスを入力します。
     - サービスのパブリック **IP** アドレス。仮想 IP アドレスがプライベートで、パブリック IP アドレスが NAT に設定されている場合は、パブリック IP アドレスを指定します。

- サービスポート。このサイトの GSLB サービスのポートを入力します。
- サイト名。GSLB サービスがあるサイトの名前を入力します。

5. 「ターゲットインスタンス 」をクリック し、GSLB 構成を展開する各サイトの GSLB インスタンスとして構成された NetScaler インスタンスを選択します。
6. 「作成 」をクリック して、選択した NetScaler インスタンス（LB ノード）に LB 構成を作成します。ユーザーは、[ドライラン] をクリックして、ターゲットインスタンスに作成されるオブジェクトを確認することもできます。ユーザーが作成した StyleBook 構成は、構成ページの構成リストに表示されます。ユーザーは、NetScaler ADM GUI を使用してこの構成を確認、更新、または削除できます。

CloudFormation テンプレートの展開

NetScaler VPX は、AWS Marketplace で Amazon マシンイメージ（AMI）として入手できます。CloudFormation テンプレートを使用して AWS で NetScaler VPX をプロビジョニングする前に、AWS ユーザーは条件に同意し、AWS Marketplace 製品に登録する必要があります。マーケットプレイスで販売されている NetScaler VPX の各エディションでは、この手順が必要です。

CloudFormation リポジトリ内の各テンプレートには、テンプレートの使用法とアーキテクチャを説明するドキュメントが併置されています。テンプレートは、NetScaler VPX の推奨される展開アーキテクチャを体系化したり、ユーザーに NetScaler を紹介したり、特定の機能、エディション、またはオプションをデモンストレーションしたりすることを目的としています。ユーザーは、特定の制作およびテストのニーズに合わせてテンプレートを再利用、変更、または拡張できます。ほとんどのテンプレートには、IAM ロールを作成する権限に加えて、完全な EC2 権限が必要です。

CloudFormation テンプレートには、NetScaler VPX の特定のリリース（リリース 12.0-56.20 など）とエディション（たとえば、NetScaler VPX プラチナエディション-10 Mbps）または NetScaler BYOL に固有の AMI ID が含まれています。CloudFormation テンプレートで別のバージョン/エディションの NetScaler VPX を使用するには、ユーザーがテンプレートを編集して AMI ID を置き換える必要があります。

最新の NetScaler AWS-AMI-ID はここにあります：NetScaler AWS CloudFormation マスター。

CFT スリー **NIC** デプロイメント

このテンプレートは、2 つの可用性ゾーンに 3 つのサブネット (管理、クライアント、サーバー) を持つ VPC をデプロイします。パブリックサブネットにデフォルトルートを持つインターネットゲートウェイをデプロイします。また、このテンプレートは、NetScaler の 2 つのインスタンスを持つ可用性ゾーン間で HA ペアを作成します。プライマリの 3 つの VPC サブネット（管理、クライアント、サーバー）に関連付けられた 3 つの ENI と、セカンダリの 3 つの VPC サブネット（管理、クライアント、サーバー）に関連付けられた 3 つの ENI です。この CFT によって作成されるすべてのリソース名には、スタック名の tagName が接頭辞として付けられます。

CloudFormation テンプレートの出力には以下が含まれます。

- primaryCitrixADCManagementURL-プライマリ VPX の管理 GUI への HTTPS URL（自己署名証明書を使用）
- PrimaryCitrixADCManagementUrl2-プライマリ VPX の管理 GUI への HTTP URL
- primaryCitrixADCInstanceId-新しく作成されたプライマリ VPX インスタンスのインスタンス ID
- primaryCitrixADCPublicVIP-VIP に関連付けられているプライマリ VPX インスタンスの Elastic IP アドレス
- PrimaryCitrixADCPrivateNSIP-プライマリ VPX の管理に使用されるプライベート IP (NS IP)
- PrimaryCitrixADCPublicNSIP-プライマリ VPX の管理に使用されるパブリック IP (NS IP)
- PrimaryCitrixADCPrivateVIP-VIP に関連付けられているプライマリ VPX インスタンスのプライベート IP アドレス
- PrimaryCitrixADCSnip-SNIP に関連付けられているプライマリ VPX インスタンスのプライベート IP アドレス
- SecondaryCitrixADCManagementURL-セカンダリ VPX の管理 GUI への HTTPS URL（自己署名証明書を使用）
- セカンダリ CitrixADCManagementUrl2-セカンダリ VPX の管理 GUI への HTTP URL
- secondaryCitrixADCInstanceId-新しく作成されたセカンダリ VPX インスタンスのインスタンス ID
- SecondaryCitrixADCPrivateNSIP-セカンダリ VPX の管理に使用されるプライベート IP (NS IP)
- SecondaryCitrixADCPublicNSIP-セカンダリ VPX の管理に使用されるパブリック IP (NS IP)
- secondaryCitrixADCPrivateVIP-VIP に関連付けられているセカンダリ VPX インスタンスのプライベート IP アドレス
- SecondaryCitrixADCSnip-SNIP に関連付けられているセカンダリ VPX インスタンスのプライベート IP アドレス
- SecurityGroup-VPX が属するセキュリティグループ ID

CFT に入力を提供する場合、CFT のあらゆる パラメーターに対して*は、それが必須フィールドであることを意味します。たとえば、`VPC ID*` は必須フィールドです。

次の前提条件が満たされている必要があります。CloudFormation テンプレートには、通常の EC2 の完全な権限を超えて、IAM ロールを作成するための十分な権限が必要です。また、このテンプレートのユーザーは、この CloudFormation テンプレートを使用する前に、条件に同意して AWS Marketplace 製品に登録する必要があります。

以下のものも存在している必要があります。

- キー ペア
- 3 つの未割り当て EIP

- 一次管理
- クライアント VIP
- 二次管理

AWS での NetScaler VPX インスタンスのプロビジョニングの詳細については、「AWS での NetScaler VPX インスタンスのプロビジョニング」をご覧ください。

StyleBooks を使用して GSLB を構成する方法については、StyleBooks を使用して GSLB を構成するをご覧ください。

災害復旧 **(DR)**

災害（さいがん）とは、自然の災害、または人為的な出来事によって引き起こされる事業機能の突然の混乱である。災害はデータセンターの運用に影響を及ぼします。その後、災害現場で失われたリソースとデータを完全に再構築して復元する必要があります。データ消失やデータセンターのダウンタイムは重要であり、ビジネス継続性が低下します。

お客様が今日直面している課題の 1 つは、DR サイトをどこに置くかを決めることです。企業は、基盤となるインフラストラクチャやネットワーク障害に関係なく、一貫性とパフォーマンスを求めています。

災害復旧のために GSLB を展開するには、AWS に NetScaler VPX スタンドアロンインスタンスを展開するを参照してください。

そのほかの参照先

ハイブリッドおよびマルチクラウド展開向けの NetScaler ADM GSLB。

SR-IOV ネットワークインターフェイスの使用を **NetScaler ADC VPX** インスタンスで構成する

October 17, 2024

> 注
>
> 高可用性セットアップでの SR-IOV インターフェイスのサポートは、NetScaler リリース 12.0 57.19 以降から利用できます。

AWS で NetScaler ADC VPX インスタンスを作成した後、AWS CLI を使用して、SR-IOV ネットワークインターフェイスを使用するように仮想アプライアンスを構成できます。

NetScaler VPX 3G および 5G の NetScaler ADC VPX AWS マーケットプレイスエディションを除き、すべての NetScaler ADC VPX モデルでは、ネットワークインターフェイスのデフォルト構成で SR-IOV が有効になっていません。

設定を開始する前に、次のトピックをお読みください。

- 前提条件
- 制限事項および使用上のガイドライン

このセクションでは、以下のトピックについて説明します。

- インターフェイスタイプを SR-IOV に変更
- 高可用性セットアップでの SR-IOV の設定

インターフェイスタイプを **SR-IOV** に変更

show interface summary コマンドを実行すると、ネットワークインターフェイスのデフォルト設定を確認できます。

例 **1**: 次の CLI スクリーンキャプチャは、NetScaler VPX AWS Marketplace エディションの 3G および 5G で SR-IOV がデフォルトで有効になっているネットワークインターフェイスの構成を示しています。

```
> 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
```

例 **2:** 次の CLI 画面キャプチャは、SR-IOV が有効になっていないネットワークインターフェイスのデフォルト設定を示しています。

```
 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
>
```

インターフェイスの種類を SR-IOV に変更する方法の詳細については、http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/sriov-networking.htmlを参照してください

インターフェイスタイプを **SR-IOV** に変更するには

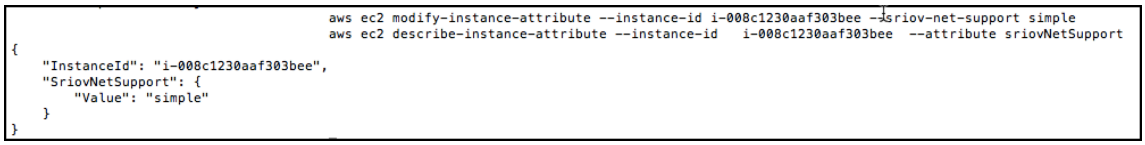
1. AWS の上で動作している NetScaler VPX インスタンスをシャットダウンします。
2. ネットワークインターフェイスで SR-IOV を有効にするには、次のコマンドを AWS CLI に入力します。

   ```
   $ aws ec2 modify-instance-attribute --instance-id \\&#060;instance
   \\_id\\&#062; --sriov-net-support simple
   ```

3. SR-IOV が有効にされたかどうか確認するには、次のコマンドを AWS CLI に入力します。

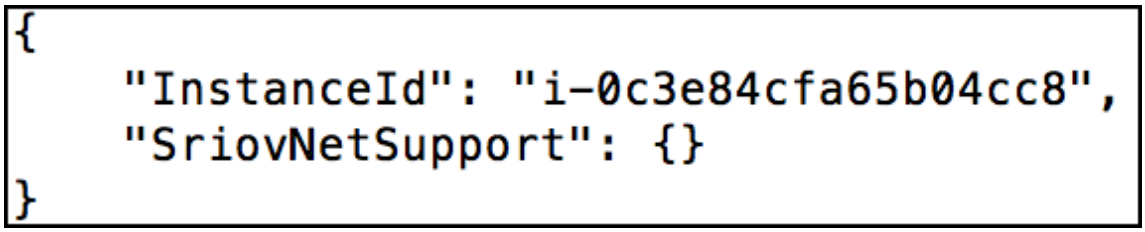
   ```
   $ aws ec2 describe-instance-attribute --instance-id \\&#060;
   instance\\_id\\&#062; --attribute sriovNetSupport
   ```

   例 **3:** AWS CLI を使用して、ネットワークインターフェイスの種類が SR-IOV に変更されました。

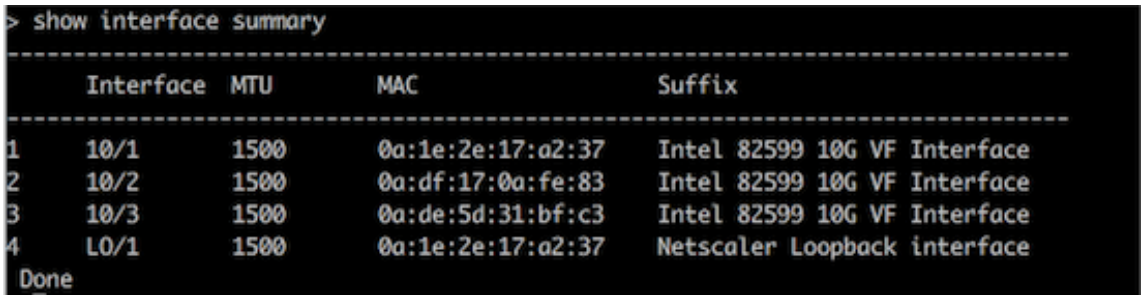
   SR-IOV が有効になっていない場合、SriovNetSupport の値は存在しません。

   例 **4:** 次の例では、SR-IOV サポートが有効になっていません。

4. VPX インスタンスの電源を入れます。ネットワークインターフェイスの変更されたステータスを確認するには、CLI で「show interface summary」と入力します。

   例 **5:** 次の画面キャプチャは、SR-IOV が有効になっているネットワークインターフェイスを示しています。インターフェイス 10/1、10/2、10/3 で SR-IOV が有効にされています。

これらの手順では、SR-IOV ネットワークインターフェイスを使用するように VPX インスタンスを構成する手順を完了します。

高可用性セットアップでの **SR-IOV** の設定

NetScaler リリース 12.0 ビルド 57.19 以降の SR-IOV インターフェイスでは、高可用性がサポートされています。

高可用性セットアップを手動で展開した場合、または NetScaler バージョン 12.0 56.20 以前の Citrix CloudFormation テンプレートを使用して展開した場合、高可用性セットアップにアタッチされた IAM ロールには次の権限が必要です。

- ec2:DescribeInstances
- ec2:DescribeNetworkInterfaces
- ec2:DetachNetworkInterface
- ec2:AttachNetworkInterface
- ec2:StartInstances
- ec2:StopInstances
- ec2:RebootInstances
- autoscaling:*
- sns: *
- sqs:*
- IAM: プリンシパルポリシーのシミュレーション
- IAM:GetRole

デフォルトでは、NetScaler ADC バージョン 12.0 57.19 用の Citrix CloudFormation テンプレートによって、必要な権限が IAM ロールに自動的に追加されます。

> 注
>
> SR-IOV インターフェイスを使用したハイアベイラビリティのセットアップには、約 100 秒のダウンタイムが発生します。

関連リソース:

IAM ロールの詳細については、AWS ドキュメントを参照してください。

AWS ENA での拡張ネットワークの使用を **Citrix ADC VPX** インスタンスで構成する

October 17, 2024

AWS で Citrix ADC VPX インスタンスを作成した後、AWS CLI を使用して、AWS Elastic Network Adapter (ENA) を使用した拡張ネットワーキングを使用するように仮想アプライアンスを構成できます。

AWS ENA と組み合わせると、拡張ネットワーキングは、より高い帯域幅、高いパケット/秒（PPS）パフォーマンス、一貫して低いインスタンス間レイテンシーを提供します。

設定を開始する前に、次のトピックをお読みください。

- 前提条件
- 制限事項および使用上のガイドライン

ENA 対応インスタンスでは、次の HA 構成がサポートされています。

- プライベート IP アドレスは同じアベイラビリティーゾーン内で移動できます。
- Elastic IP アドレスは、アベイラビリティーゾーン間で移動できます。

AWS 上の NetScaler VPX インスタンスのアップグレード

October 17, 2024

AWS 上で動作する NetScaler VPX の EC2 インスタンスの種類、スループット、ソフトウェアエディション、およびシステムソフトウェアをアップグレードすることができます。一部のアップグレード方法では、高可用性構成を使用してダウンタイムを最小限に抑えることができます。

> 注
>
> - NetScaler VPX AMI 用の NetScaler ソフトウェアの Release 10.1.e-124.1308.e 以降（ユーティリティライセンスおよびカスターマーライセンスを含む）では、M1 および M2 のインスタンスファミリをサポートしません。
> - VPX インスタンスのサポートが変更されたため、10.1.e-124 以降のリリースから 10.1.123.x 以前のリリースへのダウングレードはサポートされていません。
> - ほとんどのアップグレードでは新規の AMI を起動する必要はなく、現在の NetScaler AMI インスタンス上でアップグレードできます。新規の NetScaler AMI インスタンスへのアップグレードを行う場合は、高可用性構成を使用してください。

AWS 上の NetScaler VPX インスタンスの EC2 インスタンスタイプを変更する

Release 10.1.e-124.1308.e 以降が動作する NetScaler VPX インスタンスでは、AWS コンソールで EC2 インスタンスの種類を変更できます。次の手順に従います。

1. VPX インスタンスを停止します。
2. AWS コンソールで EC2 インスタンスの種類を変更します。
3. インスタンスを起動します。

ただし、EC2 インスタンスの種類を M3 に変更することはできません。その場合は、NetScaler ADC ソフトウェアを 10.1.e-124 以降のリリースにアップグレードするには、まず標準の NetScaler ADC アップグレード手順（）に従って、上記の手順を実行する必要があります。

AWS での NetScaler VPX インスタンスのスループットまたはソフトウェアエディションのアップグレード

ソフトウェアエディション（Standard エディションから Premium エディションへのアップグレードなど）またはスループット（たとえば、200 Mbps から 1000 Mbps へのアップグレードなど）をアップグレードするには、インスタンスのライセンスによって異なります。

カスタマーライセンスの使用 (自分のライセンスの持ち込み)

カスタマーライセンスを使用している場合は、Citrix Web サイトから新しいライセンスを購入してダウンロードし、VPX インスタンスにライセンスをインストールできます。Citrix Web サイトからライセンスをダウンロードおよびインストールする方法については、『VPX ライセンスガイド』を参照してください。

ユーティリティライセンスの使用（時間単位のユーティリティライセンス）

AWS では課金ベースのインスタンスの直接アップグレードがサポートされていません。課金ベースの NetScaler VPX インスタンスのソフトウェアエディションやスループットをアップグレードする場合は、適切なライセンスおよびキャパシティの新規の AMI を起動して、古いインスタンスから構成を移行します。これは、このページの [NetScaler 高可用性構成を使用して新しい NetScaler AMI インスタンスにアップグレードする] (#upgrade-to-a-new-citrix-adc-ami-instance-by-using-a-citrix-adc-high-availability-configuration) サブセクションで説明されているように、NetScaler 高可用性構成を使用することで実現できます。

AWS での NetScaler VPX インスタンスのシステムソフトウェアのアップグレード

10.1.e-124.1308.e 以降のリリースを実行している VPX インスタンスをアップグレードする必要がある場合は、「Citrix ADC アプライアンスのアップグレードとダウングレード」の標準の Citrix ADC アップグレード手順に従ってください。

10.1.e-124.1308.e より古いリリースを実行している VPX インスタンスを 10.1.e-124.1308.e 以降のリリースにアップグレードする必要がある場合は、まずシステムソフトウェアをアップグレードしてから、インスタンスタイプを次のように M3 に変更します。

1. VPX インスタンスを停止します。
2. AWS コンソールで EC2 インスタンスの種類を変更します。
3. インスタンスを起動します。

NetScaler の高可用性構成を使用して新しい NetScaler AMI インスタンスにアップグレードする

高可用性構成を使用して新しい NetScaler AMI インスタンスにアップグレードするには、以下のタスクを実行します。

- AWS Marketplace で、EC2 インスタンスの種類、ソフトウェアエディション、スループット、またはソフトウェアリリースを指定して新しいインスタンスを作成します。
- 古いインスタンス（アップグレード前）と新しいインスタンスとの間に高可用性を構成します。これにより、古いインスタンスの構成内容が新しいインスタンスに同期されます。
- 古いインスタンスから新しいインスタンスへの強制高可用性フェールオーバーを実行します。これにより、新しいインスタンスがプライマリノードとして設定され、新しいトラフィックを受信し始めます。
- 古いインスタンスを停止して、再構成するか AWS から削除します。

前提条件と考慮すべきポイント

- AWS 上の 2 つの Citrix ADC VPX インスタンス間で高可用性がどのように機能するかを理解してください。AWS 上の 2 つの NetScaler VPX インスタンス間の高可用性構成の詳細については、「AWS に高可用性ペアをデプロイする」を参照してください。
- 新しいインスタンスは、古いインスタンスと同じアベイラビリティゾーン内に作成し、同じセキュリティグループおよびサブネットが設定されている必要があります。
- 高可用性のセットアップでは、両インスタンスのユーザーの AWS IAM (Identity and Access Management) アカウントに関連付けられたアクセスキーと秘密キーが必要です。正しいキー情報を使用して VPX インスタンスを作成しないと、高可用性のセットアップに失敗します。VPX インスタンスの IAM アカウント作成の詳細については、「前提条件」を参照してください。
  - 新しいインスタンスを作成するには EC2 コンソールを使用する必要があります。AWS の 1-Click 起動は使用できません。これは、アクセスが許可されず、秘密キーを入力できないためです。
  - 新しいインスタンスには ENI インターフェイスが 1 つだけ必要です。

高可用性構成を使用して Citrix ADC VPX インスタンスをアップグレードするには、次の手順に従います。

1. 古いインスタンスと新しいインスタンスの間で高可用性を構成します。2 つの Citrix ADC VPX インスタンス間で高可用性を構成するには、各インスタンスのコマンドプロンプトで次のように入力します。

   - `add ha node <nodeID> <IPaddress of the node to be added>`
   - `save config`

   例:

   古いインスタンスのコマンドプロンプトで、次のように入力します。

   ```
   1    add ha node 30 192.0.2.30
   2    Done
   ```

   新しいインスタンスのコマンドプロンプトで、次のように入力します。

   ```
   1    add ha node 10 192.0.2.10
   2    Done
   ```

   以下の点に注意してください:

- この高可用性セットアップで、古いインスタンスがプライマリノードで新しいインスタンスがセカンダリノードになります。
- NSIP アドレスは古いインスタンスから新しいインスタンスにコピーされません。このため、アップグレード完了時に新しいインスタンスには異なる管理 IP アドレスが設定されます。
- 新しいインスタンスのnsrootアカウントパスワードは、HA 同期後に古いインスタンスのアカウントパスワードに設定されます。

AWS 上の 2 つの NetScaler VPX インスタンス間の高可用性構成の詳細については、「AWS に高可用性ペアをデプロイする」を参照してください。

2. HA フェールオーバーを強制します。高可用性構成でフェイルオーバーを強制するには、いずれかのインスタンスのコマンドプロンプトで次のように入力します。

```
1    force HA failover
```

強制フェールオーバーにより、古いインスタンスの ENI が新しいインスタンスに移行され、トラフィックが新しいインスタンス（新しいプライマリノード）に流れます。また、古いインスタンス（新しいセカンダリノード）が再起動します。

次の警告メッセージが表示された場合は、N を入力して操作を中止します。

```
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)?
```

この警告メッセージは、2 つの VPX インスタンスのシステムソフトウェアで高可用性がサポートされていない場合に表示されます。このため、強制フェールオーバー時に古いインスタンスの構成情報が新しいインスタンスに同期されません。

この問題に対する回避策を次に示します。

a) 古いインスタンスの NetScaler シェルプロンプトで、次のコマンドを入力して、構成ファイル (ns.conf) のバックアップを作成します。

```
copy /nsconfig/ns.conf to /nsconfig/ns.conf.bkp
```

b) バックアップ構成ファイル (ns.conf.bkp) から次の行を削除します。

- set ns config -IPAddress <IP> -netmask <MASK>

たとえば、set ns config -IPAddress 192.0.2.10 -netmask 255.255.255.0

c) 古いインスタンスのバックアップ構成ファイル (ns.conf.bkp) を新しいインスタンスの /nsconfig ディレクトリにコピーします。

d) 新しいインスタンスの NetScaler シェルプロンプトで、次のコマンドを入力して、古いインスタンスの構成ファイル (ns.conf.bkp) を新しいインスタンスにロードします。

- batch -f /nsconfig/ns.conf.bkp

e) 新しいインスタンスに設定を保存します。

- save conifg

f) いずれかのノードのコマンドプロンプトで、次のコマンドを入力してフェールオーバーを強制し、強制フェールオーバー操作を確認する警告メッセージに Y を入力します。

- force ha failover

例:

```
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
```

3. HA 設定を削除して、2 つのインスタンスが HA 設定に含まれないようにします。これを行うには、まずセカンダリノードの高可用性構成を削除して、次にプライマリノードの高可用性を削除します。

   2 つの NetScaler VPX インスタンス間の高可用性構成を削除するには、各インスタンスのコマンドプロンプトで以下のコマンドを実行します。

```
1        > remove ha node \<nodeID\>
2        >  save config
```

   AWS 上の 2 つの VPX インスタンス間の高可用性構成の詳細については、「AWS に高可用性ペアをデプロイする」を参照してください。

   例:

古いインスタンス (新しいセカンダリノード) のコマンドプロンプトで、次のように入力します。

```
1           > remove ha node 30
2             Done
3           > save config
4             Done
```

新しいインスタンス（新しいプライマリノード）のコマンドプロンプトで、次のように入力します。

```
1           > remove ha node 10
2             Done
3           > save config
4             Done
```

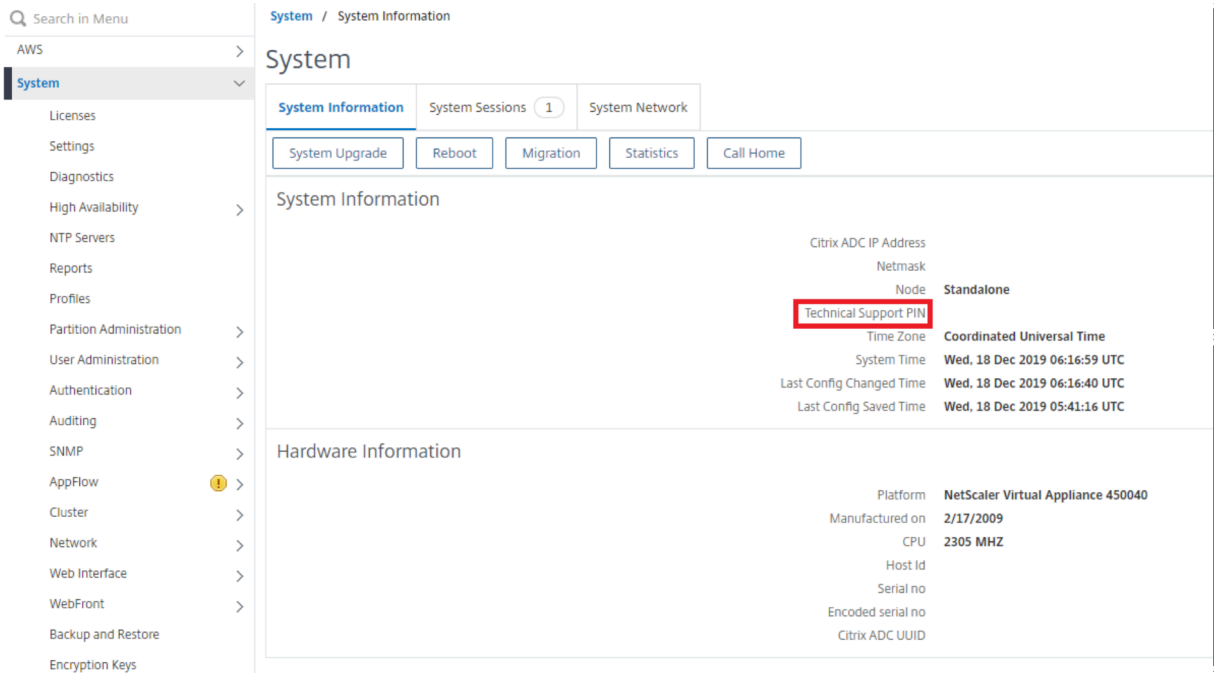

AWS での VPX インスタンスのトラブルシューティング

October 17, 2024

Amazon では、NetScaler VPX インスタンスへのコンソールアクセスを提供していません。トラブルシューティングを行うには、AWS GUI を使用してアクティビティログを表示する必要があります。デバッグできるのは、ネットワークが接続されている場合だけです。インスタンスのシステムログを表示するには、インスタンスを右クリックして [System Log] を選択します。

NetScaler は、AWS マーケットプレイスでライセンスされた NetScaler VPX インスタンス（時間単位の料金がかかるユーティリティライセンス）を AWS 上でサポートします。サポートケースを提出するには、AWS アカウント番号とサポート PIN コードを見つけて、NetScaler サポートに連絡してください。また、名前とメールアドレスの入力を求められます。サポート PIN を見つけるには、VPX GUI にログオンし、システムページに移動します。

サポート PIN を示すシステムページの例を次に示します。



AWS に関するよくある質問

October 17, 2024

- **NetScaler VPX** インスタンスは **AWS** の暗号化されたボリュームをサポートしていますか**?**

  暗号化と復号化はハイパーバイザーレベルで行われるため、どのインスタンスでもシームレスに機能します。暗号化されたボリュームの詳細については、次の AWS ドキュメントを参照してください。

https://docs.aws.amazon.com/kms/latest/developerguide/services-ebs.html

- **AWS** で **NetScaler VPX** インスタンスをプロビジョニングする最も良い方法は何ですか**?**

  NetScaler VPX インスタンスは、次のいずれかの方法で AWS にプロビジョニングできます。

  - AWS マーケットプレイスの AWS CloudFormation テンプレート (CFT)
  - NetScaler アドミニストレーター
  - AWS クイックスタート
  - GitHub の Citrix AWS CFT
  - GitHub の Citrix Terraform スクリプト
  - GitHub の Citrix Ansible プレイブック
  - AWS EC2 起動ワークフロー

  使用するオートメーションツールに基づいて、一覧表示されたオプションのいずれかを選択できます。

  オプションの詳細については、「NetScaler VPX on AWS」を参照してください。

- **AWS** で **NetScaler VPX** インスタンスをアップグレードするには**?**

  AWS で NetScaler VPX インスタンスをアップグレードするには、「AWS で NetScaler VPX インスタンスをアップグレードする」の手順に従って、システムソフトウェアをアップグレードするか、新しい NetScaler VPX Amazon Machine Image (AMI) にアップグレードします。

  NetScaler VPX インスタンスをアップグレードする推奨される方法は、ジョブを使用した NetScaler インスタンスのアップグレードの手順に従って、ADM サービスを使用することです。

- **AWS** での **NetScaler VPX** の **HA** フェイルオーバー時間はどれくらいですか**?**

  - AWS アベイラビリティーゾーン内での NetScaler VPX の高可用性フェイルオーバーには約 3 秒かかります。
  - AWS アベイラビリティーゾーン全体の NetScaler VPX の HA フェイルオーバーには、約 5 秒かかります。

- テクニカルサポート **PIN** を提供する **NetScaler VPX** マーケットプレイスのサブスクリプションをご利用のお客様には、どのレベルのサポートが提供されますか**?**

  デフォルトでは、テクニカルサポート PIN を提供するお客様には「ソフトウェアの選択」サービスが提供されます。

- **Elastic IP** デプロイメントを使用した異なるゾーンにわたる高可用性では、アプリケーションごとに複数の **IPSet** を作成する必要がありますか**?**

  はい。複数の VIP が複数の EIP にマッピングされた複数のアプリケーションがある場合は、複数の IPSet が必要です。したがって、HA フェールオーバー中に、EIP のすべてのプライマリ VIP マッピングがセカンダリ（新しいプライマリ）VIP に変更されます。

- 異なるゾーン展開で高可用性で **INC** モードが有効になるのはなぜですか。

アベイラビリティーゾーン全体の HA ペアは、異なるネットワークにあります。HA 同期の場合、ネットワーク構成を同期してはいけません。これは、HA ペアで INC モードを有効にすることによって実現されます。

- アベイラビリティーゾーンが同じ **VPC** 内にある場合、あるアベイラビリティーゾーンの **HA** ノードは、別のアベイラビリティーゾーンのバックエンドサーバーと通信できますか。

  はい。同じ VPC の異なるアベイラビリティーゾーンにあるサブネットには、SNIP 経由でバックエンドサーバーのサブネットを指す追加のルートを追加することで到達できます。たとえば、AZ1 の ADC の SNIP サブネットが 192.168.3.0/24、AZ2 のバックエンドサーバーのサブネットが 192.168.6.0/24 の場合、AZ1 に存在する NetScaler アプライアンスに 192.168.6.0 255.255.0 192.168.3.1 としてルートを追加する必要があります。

- **Elastic IP** を使用した異なるゾーン間の高可用性 と プライベート **IP** を使用した異なるゾーン間の高可用性 のデプロイメントは連携して機能しますか**?**

  はい、両方の設定を同じ HA ペアに適用できます。

- プライベート **IP** デプロイメントを使用した異なるゾーン間の高可用性において、**VPC** 内に複数のルート テーブルを持つ複数のサブネットがある場合、**HA** ペアのセカンダリ ノードは、**HA** フェイルオーバー中にチェックされるルート テーブルをどのように認識するのでしょうか。

  セカンダリノードはプライマリ NIC を認識し、VPC 内のすべてのルートテーブルを検索します。

- **AWS** で **VPX** のデフォルトイメージを使用する場合の**/var**パーティションのサイズはどれくらいですか? ディスク容量を増やすには?

  ディスクイメージを小さく保つために、ルートディスクのサイズは 20 GB に制限されています。

  /var/core/または/var/crash/ディレクトリ領域を増やす場合は、追加のディスクを接続します。/varサイズを大きくするには、現在のところ、重要なコンテンツを新しいディスクにコピーした後、追加のディスクを接続して/varにシンボリックリンクを作成する必要があります。

- **vCPU** にアクティブ化され、割り当てられるパケットエンジンは何台ありますか。

  パケットエンジン（PE）は、ライセンスされた vCPU の数によって制限されます。NetScaler デーモンは特定の vCPU に固定されず、非 PE vCPU で実行される可能性があります。AWS によると、C5.9xLarge は 72 GB のメモリを持つ 36vCPU インスタンスです。プールライセンスでは、NetScaler VPX インスタンスが最大数の PE でデプロイされます。この場合、19 PE がコア 1 ～19 で実行されます。ただし、ADC 管理プロセスは CPU 20～31 から実行されます。

- **ADC** の適切な **AWS** インスタンスを決定するには?

  1. スループット、PPS、SSL 要件、平均パケットサイズなどのユースケースと要件を理解します。
  2. VPX 帯域幅オファリングや vCPU ベースのライセンスなど、要件を満たす適切な ADC 製品とライセンスを選択します。
  3. 選択したオファリングに基づいて、AWS インスタンスを決定します。

  例

5 Gbps ライセンスでは、5 つのデータパケットエンジンが有効になります。したがって、vCPU 要件は 6（管理の場合は 5+1）です。ただし、6 つの vCPU インスタンスは利用できません。したがって、5 Gbps の帯域幅をサポートするネットワークを選択すれば、8 vCPU はそのスループットに到達するのに十分です。たとえば、5 Gbps のライセンスの最大 PE 割り当てを有効にするには、5 Gbps 帯域幅ライセンスに m5.2xlarge を選択する必要があります。ただし、スループットによって制限されない vCPU ライセンスを使用すると、m5.xlarge インスタンス自体を使用して 5 Gbps のスループットが得られる可能性があります。

Instance Size	vCPU	Memory (GiB)	Instance Storage (GiB)	Network Bandwidth (Gbps)	EBS Bandwidth (Mbps)
m5.large	2	8	EBS-Only	Up to 10	Up to 4,750
m5.xlarge	4	16	EBS-Only	Up to 10	Up to 4,750
m5.2xlarge	8	32	EBS-Only	Up to 10	Up to 4,750
m5.4xlarge	16	64	EBS-Only	Up to 10	4,750

- **AWS** の **ADC** には **3** つの **NIC 3** サブネットのデプロイメントが必須ですか**?**

  Three NICs-three subnets は、管理、クライアント、およびサーバーネットワーク用の推奨展開です。この展開により、トラフィックの分離と VPX パフォーマンスが向上します。2 つの NIC-2 サブネット、および 1 つの nic-One サブネットは、他の使用可能なオプションです。AWS で複数の NIC でサブネットを共有する (2 つの NIC と 1 つのサブネットの展開など) ことは推奨されません。このシナリオでは、非対称ルーティングなどのネットワークの問題が発生する可能性があります。詳細については、「AWS でネットワークインターフェイスを設定するためのベストプラクティス」を参照してください。

- インスタンスのネットワーク機能に関係なく、**AWS** の **ENA** ドライバーが常に **1Gbps (1/1)** のリンク速度を示すのはなぜですか**?**

  AWS Elastic Network Adapter (ENA) の報告された速度は、選択したインスタンスタイプに関係なく、1Gbps (1/1) と表示されることが多いです。これは、表示される速度が実際のネットワーク パフォーマンスを直接反映するものではないためです。従来のネットワーク インターフェイスとは異なり、ENA の速度はインスタンスの要件とワークロードに基づいて動的にスケーリングできます。実際のネットワーク パフォーマンスは、主にインスタンスのタイプとサイズによって決まります。したがって、実際のネットワーク スループットは、特定のインスタンスの種類と現在のネットワーク負荷によって大きく異なる可能性があります。

Microsoft Azure で NetScaler VPX インスタンスを展開する

October 17, 2024

NetScaler VPX インスタンスを Microsoft Azure Resource Manager（ARM）にデプロイすると、次の機能セットの両方を使用してビジネスニーズを満たすことができます。

- Azure クラウドコンピューティング機能

- NetScaler の負荷分散とトラフィック管理機能

NetScaler VPX インスタンスは、スタンドアロンインスタンスとして、またはアクティブ/スタンバイモードの高可用性ペアとして ARM にデプロイできます。

NetScaler VPX インスタンスを Microsoft Azure にデプロイするには、次の 2 つの方法があります。

- Azure マーケットプレイスを通じて。NetScaler VPX 仮想アプライアンスは、Microsoft Azure Marketplace でイメージとして使用することができます。
- GitHub で入手可能な NetScaler Azure Resource Manager (ARM) json テンプレートを使用する。詳細については、NetScaler ソリューションテンプレートの GitHub リポジトリを参照してください。

Microsoft Azure スタックは、ローカルデータセンターに Microsoft Azure パブリッククラウドサービスを提供し、組織がハイブリッドクラウドを構築できるようにするハードウェアとソフトウェアの統合プラットフォームです。NetScaler VPX インスタンスを Microsoft Azure スタックにデプロイできるようになりました。

> 注
>
> Azure は、Azure の外部から発信されるトラフィックへのアクセスを制限し、ブロックします。アクセスを提供するには、パブリック IP アドレスがアタッチされている VM の NIC に接続されているネットワークセキュリティグループにインバウンドルールを追加して、サービスまたはポートを有効にします。詳細については、インバウンド NATルールに関する Azure ドキュメントを参照してください。

前提要件

NetScaler VPX インスタンスを Azure にデプロイする前に、ある程度の前提知識が必要です。

- Azure の用語とネットワークの詳細に精通しています。詳細については、Azure の用語を参照してください。
- NetScaler ネットワークに関する知識。ネットワーキング トピックを参照してください。
- NetScaler アプライアンスに関する知識。ネットワーク トピックを参照してください。

NetScaler VPX インスタンスが Azure 上で動作する仕組み

オンプレミス展開では、NetScaler VPX インスタンスは、少なくとも次の 3 つの IP アドレスを必要とします。

- 管理 IP アドレス。NSIP アドレスと呼ばれます。
- サーバーファームとやり取りするためのサブネット IP（SNIP）アドレス
- クライアント要求を受け付ける仮想サーバー IP（VIP）アドレス

詳しくは、「Microsoft Azure 上の NetScaler VPX インスタンスのネットワークアーキテクチャ」を参照してください。

> 注
>
> NetScaler VPX インスタンスは Intel プロセッサと AMD プロセッサの両方をサポートします。VPX 仮想アプライアンスは、2 つ以上の仮想コアと 2 GB を超えるメモリを備えた任意のインスタンスタイプにデプロイできます。システム要件の詳細については、NetScaler VPX のデータシートを参照してください。

Azure 環境では、次の 3 つの方法で Azure 上に NetScaler VPX インスタンスをプロビジョニングできます。

- マルチ NIC マルチ IP アーキテクチャ
- シングル NIC マルチ IP アーキテクチャ
- 単一の NIC シングル IP

ニーズに応じて、これらのサポートされているアーキテクチャタイプのいずれかを使用できます。

マルチ **NIC** マルチ **IP** アーキテクチャ

このデプロイタイプでは、VPX インスタンスに複数のネットワークインターフェイス (NIC) をアタッチできます。NIC には、静的または動的パブリック IP アドレスとプライベート IP アドレスを 1 つ以上割り当てることができます。

詳細については、次のユースケースを参照してください:

- 複数の IP アドレスと NIC を使用して高可用性設定を構成する
- PowerShell コマンドを使用して複数の IP アドレスと NIC を使用して高可用性セットアップを構成する

> 注
>
> Azure 環境で MAC が移動したり、インターフェイスがミュートされたりしないように、NetScaler VPX インスタンスのデータインターフェイス（タグなし）ごとに VLAN を作成し、NIC のプライマリ IP を Azure にバインドすることを Citrix では推奨しています。詳細については、CTX224626 の記事を参照してください。

シングル **NIC** マルチ **IP** アーキテクチャ

この展開タイプでは、複数の IP 構成（静的または動的パブリック IP アドレスおよびプライベート IP アドレスに割り当てられている）に関連付けられた 1 つのネットワークインターフェイス（NIC）。詳細については、次のユースケースを参照してください：詳細については、次のユースケースを参照してください:

- NetScaler VPX スタンドアロンインスタンスに複数の IP アドレスを構成する
- PowerShell コマンドを使用して、NetScaler VPX スタンドアロンインスタンスに複数の IP アドレスを構成する

単一の **NIC** シングル **IP**

この展開タイプでは、単一の IP アドレスに関連付けられた 1 つのネットワークインターフェイス（NIC）で、NSIP、SNIP、および VIP の機能を実行するために使用されます。

詳細については、「NetScaler VPX スタンドアロン インスタンスの構成」を参照してください。

> 注
>
> 単一 IP モードは Azure 展開環境でのみ使用することができます。このモードは、オンプレミス、AWS、またはその他の種類の展開にある NetScaler VPX インスタンスでは使用できません。

NetScaler VPX ライセンス

Azure 上の NetScaler VPX インスタンスにはライセンスが必要です。Azure 上で実行される NetScaler VPX インスタンスでは、次のライセンスオプションを使用できます。

- サブスクリプションベースのライセンス：NetScaler VPX アプライアンスは、Azure Marketplace で有料インスタンスとして利用できます。サブスクリプションベースのライセンスは、従量課金制のオプションです。ユーザーは時間単位で課金されます。

  > 注
  >
  > サブスクリプションベースのライセンスインスタンスの場合、サブスクリプションの請求は、特定のライセンスモデルのライセンス期間を通じて適用されます。クラウドの制限により、Azure はサブスクリプションに適用されるライセンスモデルの変更または削除をサポートしていません。サブスクリプションライセンスを変更または削除するには、既存の ADC VM を削除し、必要なライセンスを使用して新しい ADC VM を再作成します。

  NetScaler は、サブスクリプションベースのライセンスインスタンスのテクニカルサポートを提供します。サポートケースを提出するには、「NetScaler on Azure のサポート—時間単位のサブスクリプションライセンス」を参照してください。

- 自分のライセンスを持参 **(BYOL)**: 自分のライセンス (BYOL) を持ち込む場合は、http://support.citrix.com/article/CTX122426にある VPX ライセンスガイドを参照してください。次の操作を実行する必要があります：

  - NetScaler Web サイトのライセンスポータルを使用して、有効なライセンスを生成します。
  - ライセンスをインスタンスにアップロードします。

  > 注
  >
  > Azure スタック環境では、**BYOL** が唯一のライセンスオプションです。

- **NetScaler VPX** チェックイン/チェックアウトライセンス：詳細については、「NetScalerVPX チェックイン/チェックアウトライセンス」を参照してください。

NetScaler リリース 12.0 56.20 以降、オンプレミスおよびクラウド展開用の NetScaler VPX Express にはライセンスファイルは必要ありません。NetScaler VPX Express の詳細については、Citrix ADC ライセンスの概要の「NetScaler VPX Express ライセンス」セクションを参照してください。

VPX のパフォーマンスと推奨される Azure インスタンスの種類

必要な VPX パフォーマンスを得るには、次の Azure インスタンス タイプが推奨されます。

VPX パフォーマンス	Azure インスタンス タイプ VPX 1 NIC/2 NIC	VPX3 NIC	VPX 最大 8 NIC
最大 200 Mbps	Standard_D2_v4 Standard_DS3_v2	Standard_D4_v4 Standard_DS3_v2	標準_DS3_v2 標準_DS4_v2
最大 1Gbps	Standard_D4_v4 Standard_DS3_v2	Standard_D8_v5 Standard_DS3_v2	標準_DS3_v2 標準_DS4_v2
最大 5Gbps	Standard_D8_v5 Standard_DS3_v2	Standard_D8_v5 Standard_DS4_v2	Standard_D8_v5 Standard_DS4_v2
最大 10Gbps	標準_D2_v5	標準_D8_v5	標準_D16_v5

注意事項

- 1 Gbps および 5 Gbps のスループットで NetScaler VPX インスタンスで最適なパフォーマンスを実現するには、Azure 高速ネットワークを有効にする必要があります。

  高速ネットワークの構成の詳細については、「Azure 高速ネットワークを使用するように NetScaler VPX インスタンスを構成する」を参照してください。

- Azure Marketplace から購入したサブスクリプションベースの時間単位ライセンスに関係なく、まれに、Azure にデプロイされた NetScaler VPX インスタンスにデフォルトの NetScaler ライセンスが付与されることがあります。これは、Azure インスタンスメタデータサービス (IMDS) の問題が原因で発生します。
- NetScaler VPX インスタンスの構成を変更する前に、ウォームリスタートを実行して、正しい NetScaler VPX ライセンスを有効にします。

Azure における NetScaler VPX インスタンスの IPv6 サポート

リリース 13.1-21.x 以降、NetScaler VPX スタンドアロンインスタンスは Azure の IPv6 アドレスをサポートします。IPv6 アドレスは、Azure クラウドの NetScaler VPX スタンドアロンインスタンスで VIP アドレスと SNIP アドレスとして構成できます。

Azure で IPv6 を有効にする方法については、次の Azure ドキュメントを参照してください。

- Azure 仮想ネットワークの IPv6 とは何ですか?
- Azure 仮想ネットワークの IPv4 アプリケーションに IPv6 を追加する-Azure CLI
- 住所の種類

NetScaler アプライアンスが IPv6 をサポートする方法については、「インターネット プロトコル バージョン 6」を参照してください。

IPv6 の制限事項**:**

- NetScaler の IPv6 環境では現在、Azure バックエンドの自動スケーリングはサポートされていません。
- IPv6 は NetScaler VPX HA 展開ではサポートされていません。

制限事項

NetScaler VPX 負荷分散ソリューションを ARM で実行すると、次の制限があります。

- Azure アーキテクチャでは、以下の NetScaler 機能をサポートしていません。
  - Gratuitous ARP (GARP)
  - L2 モード
  - タグ付き VLAN
  - 動的ルーティング
  - 仮想 MAC
  - USIP
  - クラスタリング

  > 注
  >
  > NetScaler Application Delivery Management (ADM) Autoscale 機能 (クラウド展開) では、ADC インスタンスはすべてのライセンスでクラスタリングをサポートします。詳細については、「NetScaler ADM を使用した Microsoft Azure での NetScaler VPX の自動スケーリング」を参照してください。

- NetScaler VPX 仮想マシンを任意のタイミングでシャットダウンし、一時的に割り当てを解除しなければならないことが予想される場合は、仮想マシンの作成時に静的内部 IP アドレスを割り当てます。静的内部 IP アドレスを割り当てないと、Azure が再起動のたびに異なる IP アドレスを仮想マシンに割り当てる可能性があり、仮想マシンにアクセスできなくなる場合があります。
- Azure は最大 10 Gbps の VPX スループットをサポートします。詳しくは、NetScaler VPX のデータシートを参照してください。
- スループットが 3 Gbps を超える NetScaler VPX インスタンスを使用する場合、実際のネットワーク スループットはインスタンスのライセンスで指定されたスループットと一致しない可能性があります。ただし、SSL スループットや SSL トランザクション/秒など、その他の機能が向上する可能性があります。

- 仮想マシンのプロビジョニング中に Azure によって生成されたデプロイ ID は、ARM のユーザーには表示されません。デプロイ ID を使用して NetScaler VPX アプライアンスを ARM にデプロイすることはできません。
- NetScaler VPX インスタンスは、初期化時に 20 Mbps のスループットと標準エディションの機能をサポートします。
- 高速ネットワークが有効になっている Azure 上の NetScaler VPX インスタンスは、パフォーマンスが向上します。Azure 高速ネットワークは、リリース 13.0 ビルド 76.x 以降の NetScaler VPX インスタンスでサポートされています。NetScaler VPX で高速ネットワークを有効にするには、高速ネットワークをサポートする Azure インスタンスタイプを使用することをお勧めします。
- Citrix Virtual Apps and Desktops の導入では、VPX インスタンス上の VPN 仮想サーバーを次のモードで構成できます:
  - 基本モード。`ICAOnly` VPN 仮想サーバーパラメーターが ON に設定されます。基本モードは、ライセンスされていない NetScaler VPX インスタンスでも完全に動作します。
  - SmartAccess モード。`ICAOnly` VPN 仮想サーバーパラメーターが OFF に設定されています。SmartAccess モードは、ライセンスのない NetScaler VPX インスタンス上の 5 人の NetScaler AAA セッションユーザーに対してのみ機能します。

> 注
>
> SmartControl 機能を設定するには、NetScaler VPX インスタンスにプレミアムライセンスを適用する必要があります。

Azure 用語集

October 17, 2024

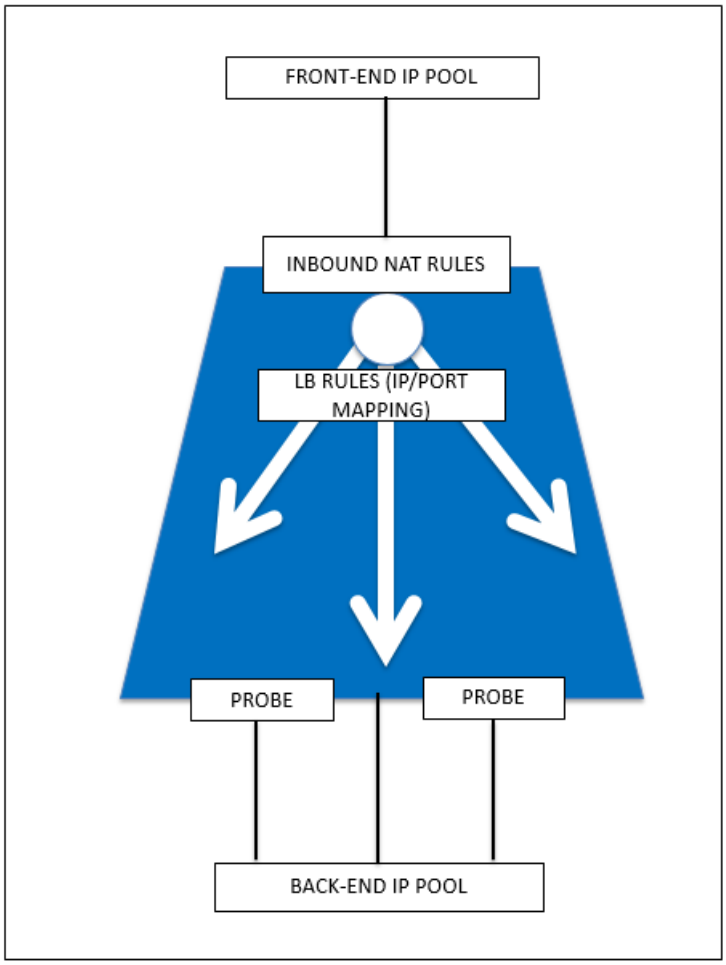
NetScaler VPX Azure のドキュメントで使用されている Azure 用語の一部を以下に示します。

1. Azure ロードバランサー—Azure ロードバランサーは、ネットワーク内のコンピューター間で着信トラフィックを分散するリソースです。トラフィックは、ロードバランサーセット内に定義された仮想マシンに分配されます。ロードバランサーには、外部ロードバランサー、インターネットに接続するロードバランサー、または内部ロードバランサーがあります。
2. Azure Resource Manager (ARM) —ARM は、Azure のサービスの新しい管理フレームワークです。Azure Load Balancer は、ARM ベースの API およびツールを使用して管理されます。
3. バックエンドアドレスプール—負荷が分散される仮想マシンの NIC (NIC) に関連付けられた IP アドレスです。
4. BLOB-バイナリラージオブジェクト—Azure ストレージに格納できるファイルまたはイメージのようなバイナリオブジェクト。

5. フロントエンド IP 構成—Azure ロードバランサーには、仮想 IP (VIP) とも呼ばれる 1 つ以上のフロントエンド IP アドレスを含めることができます。これらの IP アドレスがトラフィックの入口として使用されます。
6. インスタンスレベルのパブリック IP (ILPIP) —ILPIP は、仮想マシンまたはロールインスタンスが存在するクラウドサービスではなく、仮想マシンまたはロールインスタンスに直接割り当てることができるパブリック IP アドレスです。これは、クラウドサービスに割り当てられた VIP（仮想 IP）に代わるものではありません。これは、仮想マシンまたはロールインスタンスに直接接続するために使用できる追加の IP アドレスです。

   > 注
   >
   > 以前は、ILPIP は PIP (パブリック IP) と呼ばれていました。

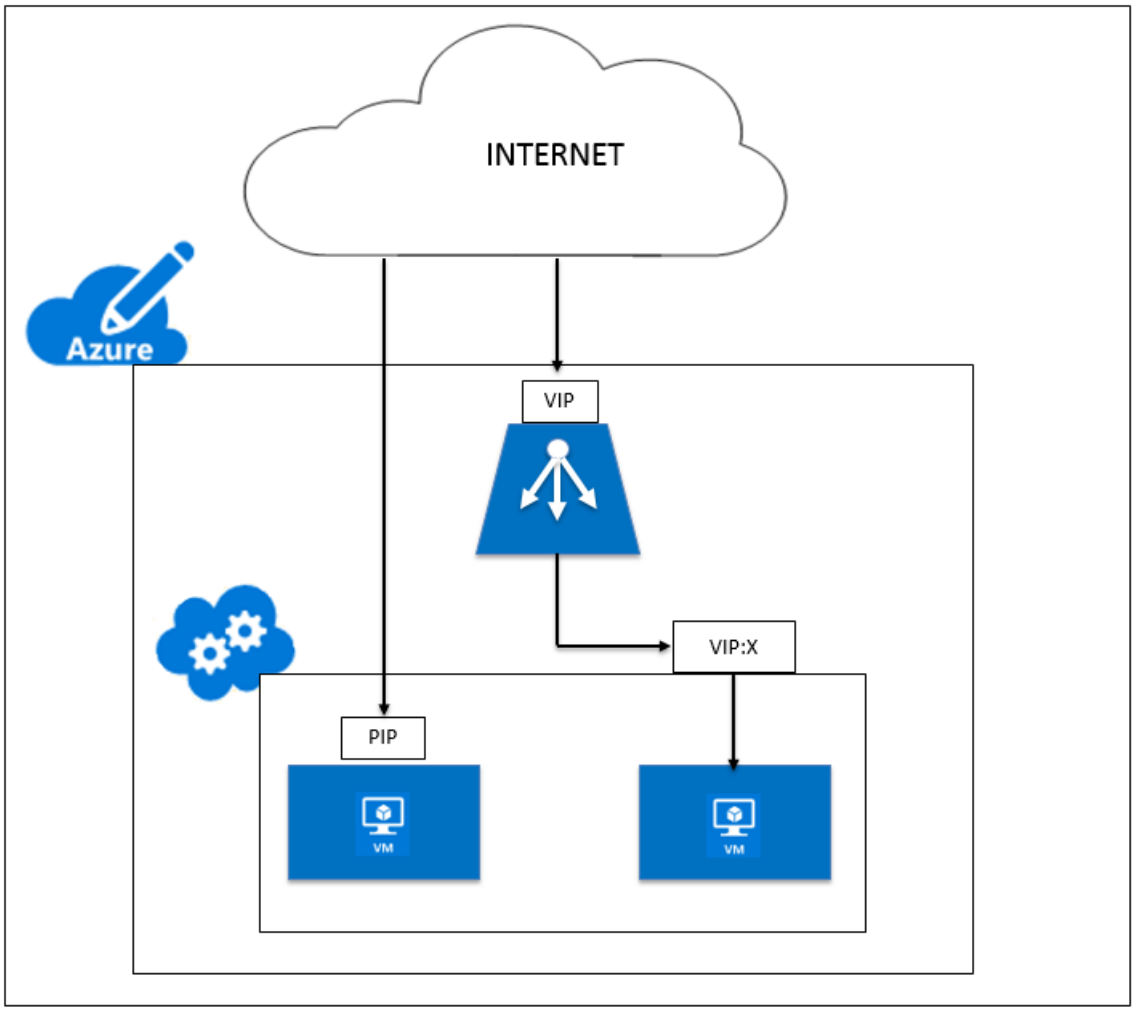
7. インバウンド NAT ルール—ロードバランサーのパブリックポートを、バックエンドアドレスプール内の特定の仮想マシンのポートにマッピングするルールが含まれます。
8. IP-Config：個々の NIC に関連付けられた IP アドレスのペア（パブリック IP とプライベート IP）として定義できます。IP-Config では、パブリック IP アドレスが NULL の場合があります。各 NIC には、最大 255 までの IP 構成を関連付けることができます。
9. 負荷分散ルール：特定のフロントエンド IP とポートの組み合わせを、バックエンド IP アドレスとポートの組み合わせのセットにマップする規則プロパティ。ロードバランサーリソースの単一の定義を使用して複数のロードバランサー規則を定義でき、その各規則は、フロントエンド IP およびポートと、仮想マシンに関連付けられたバックエンド IP およびポートの組み合わせを示します。



10. ネットワークセキュリティグループ—仮想ネットワーク内の仮想マシンインスタンスへのネットワークトラフィックを許可または拒否するアクセス制御リスト (ACL) ルールのリストが含まれます。NSG は、サブネット、またはそのサブネット内の個々の仮想マシンインスタンスに関連付けることができます。ネットワークセキュリティグループがサブネットに関連付けられている場合、ACL ルールはそのサブネット内のすべての仮想マシンインスタンスに適用されます。さらに、ネットワークセキュリティグループをその仮想マシンに直接関連付けることで、個々の仮想マシンへのトラフィックをさらに制限できます。
11. プライベート IP アドレス—Azure 仮想ネットワーク内の通信に使用され、VPN Gateway を使用してネットワークを Azure に拡張するときのオンプレミスネットワークで使用されます。プライベート IP アドレスを使用すると、Azure リソースは、VPN ゲートウェイまたは ExpressRoute 回路を経由して、インターネットで到達できる IP アドレスを使用せずに、仮想ネットワークまたはオンプレミスネットワーク内の他のリソースと通信できます。Azure Resource Manager 展開モデルでは、プライベート IP アドレスは次の種類の Azure リソースに関連付けられます - 仮想マシン、内部ロードバランサー（ILB)、およびアプリケーションゲートウェイ。
12. Probes —これには、バックエンドアドレスプール内の仮想マシンインスタンスの可用性をチェックするために使用されるヘルスプローブが含まれます。個別の仮想マシンが一定時間ヘルスプローブに応答しない場合、それはトラフィック供用から除外されます。プローブを使用すると、仮想インスタンスのヘルスを追跡できま

す。ヘルスプローブが失敗した場合、仮想インスタンスはローテーションから自動的に除外されます。

13. パブリック IP アドレス (PIP) —PIP は、Azure のパブリック向けサービスを含むインターネットとの通信に使用され、仮想マシン、インターネット向けロードバランサー、VPN ゲートウェイ、およびアプリケーションゲートウェイに関連付けられます。
14. リージョン-国境を越えず、1 つ以上のデータセンターを含む地理内のエリア。価格設定、地域サービスおよびタイプは、リージョンレベルで公開されます。リージョンは通常、（最大で数百マイル離れた）別のリージョンと対にされ、リージョンペアを形成します。障害回復シナリオおよび高可用性シナリオでは、リージョンペアをメカニズムとして使用できます。また、一般に場所とも呼ばれます。
15. リソースグループ-リソースマネージャのコンテナは、アプリケーションに関連するリソースを保持します。リソースグループには、アプリケーションのリソースをすべて含めることも、論理的にグループにまとめられたリソースだけを含めることもできます。
16. ストレージアカウント—Azure ストレージアカウントを使用すると、Azure Storage の Azure BLOB、キュー、テーブル、およびファイルサービスにアクセスできます。ストレージアカウントは、Azure ストレージデータオブジェクトに一意の名前空間を提供します。
17. 仮想マシン—オペレーティングシステムを実行する物理コンピュータのソフトウェア実装。同じハードウェア上で複数の仮想マシンを同時に実行できます。Azure には、いろいろなサイズの仮想マシンが用意されています。
18. 仮想ネットワーク-Azure 仮想ネットワークは、クラウド内の独自のネットワークを表現したものです。それはサブスクリプション専用の Azure クラウドの論理的隔離です。IP アドレスブロック、DNS 設定、セキュリティポリシー、およびこのネットワーク内のルートテーブルを全面的に制御できます。さらに VNet のサブネットに分割したり、Azure IaaS 仮想マシンおよびクラウドサービス（PaaS ロールインスタンス）を起動したりすることもできます。また、Azure で利用できる接続性オプションの 1 つを使用して、仮想ネットワークをオンプレミスネットワークに接続できます。本質的には、Azure が提供するエンタープライズスケールの利点を持つ IP アドレスブロック上の全面的なコントロールを使用して、ネットワークを Azure に拡張できます。



Microsoft Azure 上の NetScaler ADC VPX インスタンスのネットワークアーキテクチャ

October 17, 2024

Azure Resource Manager (ARM) では、NetScaler VPX 仮想マシン (VM) は仮想ネットワークに存在します。仮想ネットワークの特定のサブネットに単一のネットワークインターフェイスを作成でき、VPX インスタンスに接続できます。ネットワークセキュリティグループを使用して、Azure 仮想ネットワーク内の VPX インスタンスとの間のネットワークトラフィックをフィルタリングできます。ネットワークセキュリティグループには、VPX インスタンスへのインバウンドネットワークトラフィックまたは VPX インスタンスからのアウトバウンドネットワークトラフィックを許可または拒否するセキュリティルールが含まれています。詳細については、「セキュリティグループ」を参照してください。

ネットワークセキュリティグループは、NetScaler VPX インスタンスへの要求をフィルタリングし、VPX インスタ

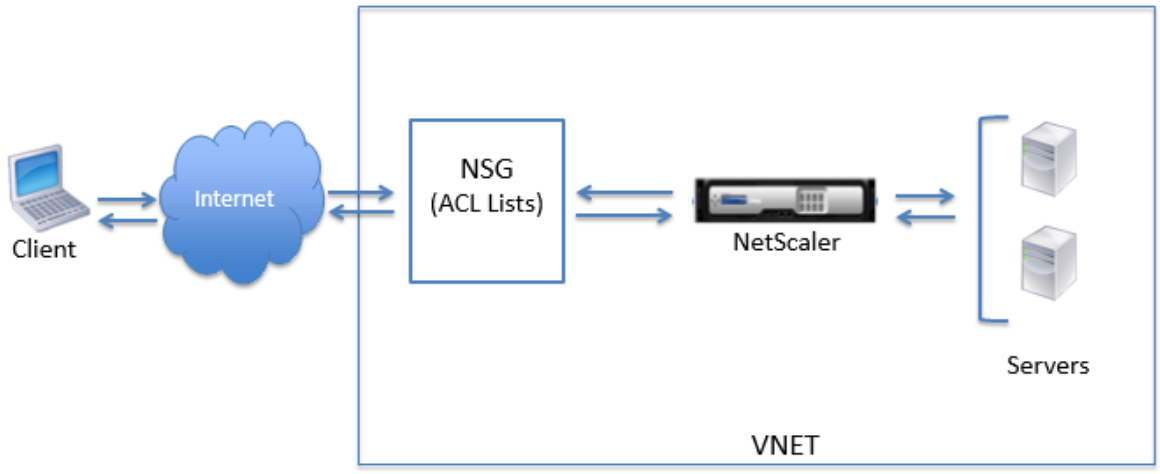
ンスはそれらをサーバーに送信します。サーバーからの応答は、逆の順序で同じパスをたどります。ネットワークセキュリティグループは、単一の VPX VM をフィルタリングするように構成することも、サブネットと仮想ネットワークを使用して、複数の VPX インスタンスを展開するトラフィックをフィルタリングすることもできます。

NIC には、ネットワーク構成の詳細（仮想ネットワーク、サブネット、内部 IP アドレス、パブリック IP アドレスなど）が含まれます。

ARM では、単一の NIC と 1 つの IP アドレスでデプロイされた仮想マシンにアクセスするために使用される、次の IP アドレスを知っておくとよいでしょう。

- パブリック IP（PIP）アドレスは、NetScaler VM の仮想 NIC 上で直接構成されるインターネット側 IP アドレスです。これにより、外部ネットワークから VM に直接アクセスできます。
- NetScaler IP（NSIP とも呼ばれる）アドレスは、仮想マシン上で構成された内部 IP アドレスです。これはルーティング不可能です。
- 仮想 IP アドレス（VIP）は、NSIP とポート番号を使用して構成されます。クライアントは PIP アドレスから NetScaler サービスにアクセスし、要求が NetScaler VPX VM または Azure ロードバランサーの NIC に到達すると、VIP が内部 IP（NSIP）および内部ポート番号に変換されます。
- 内部 IP アドレスは、仮想ネットワークのアドレス空間プールにある、VM のプライベート内部 IP アドレスです。この IP アドレスは、外部ネットワークから到達できません。この IP アドレスは、静的に設定しない限り、デフォルトで動的です。インターネットからのトラフィックは、ネットワークセキュリティグループで作成されたルールに従って、このアドレスにルーティングされます。ネットワークセキュリティグループは NIC と統合して、仮想マシンで設定されたサービスに応じて、適切なタイプのトラフィックを NIC の適切なポートに選択的に送信します。

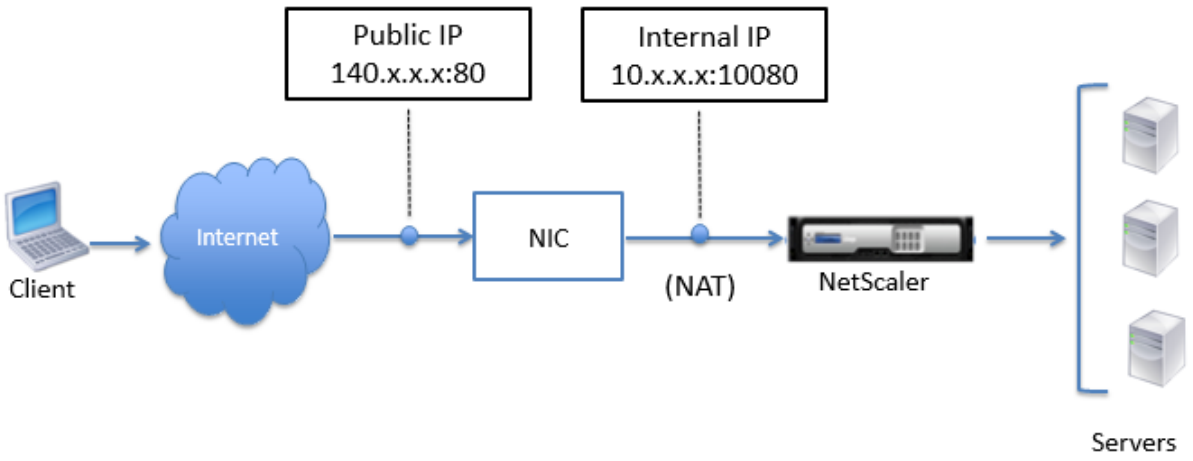
以下の図は、ARM でプロビジョニングされた NetScaler VPX インスタンスを介したクライアントからサーバーへのトラフィックフローを示しています。



ネットワークアドレス変換によるトラフィックフロー

NetScaler VPX インスタンス（インスタンスレベル）のパブリック IP（PIP）アドレスをリクエストすることもできます。この直接 PIP を VM レベルで使用する場合、ネットワークトラフィックを傍受する受信および送信規則を定義する必要はありません。インターネットからの着信要求が VM で直接受信されます。Azure はネットワークアドレス変換（NAT）を実行し、VPX インスタンスの内部 IP アドレスにトラフィックを転送します。

以下の図は、Azure がネットワークアドレス変換を実行し、NetScaler 内部 IP アドレスをマップする方法を示しています。



この例では、ネットワークセキュリティグループに割り当てられたパブリック IP は 140.x.x.x で、内部 IP アドレスは 10.x.x.x です。インバウンドルールとアウトバウンドルールが定義されている場合、パブリック HTTP ポート 80 はクライアントリクエストを受信するポートとして定義され、対応するプライベートポート 10080 は NetScaler ADC VPX インスタンスがリッスンするポートとして定義されます。クライアント要求はパブリック IP アドレス 140.x.x.x で受信されます。Azure がネットワークアドレス変換を実行して、PIP を内部 IP アドレス 10.x.x.x（ポート 10080）にマップし、クライアント要求を転送します。

> 注
>
> 高可用性における NetScaler VPX VM は、自身に定義された受信規則により負荷分散トラフィックを制御する、外部または内部のロードバランサーによって制御されます。外部トラフィックは最初にこれらのロードバランサによって代行受信され、トラフィックは設定されたロードバランシング規則に従って迂回されます。ロードバランサには、バックエンドプール、NAT ルール、および健全性プローブが定義されています。

ポートの使用に関する注意事項

NetScaler VPX インスタンスの作成中または仮想マシンのプロビジョニング後に、ネットワークセキュリティグループでより多くのインバウンドルールとアウトバウンドルールを構成できます。各受信および送信規則は、パブリックポートおよびプライベートポートに関連付けられています。

ネットワークセキュリティグループルールを設定する前に、使用できるポート番号に関する次のガイドラインに注意してください。

1. NetScaler VPX インスタンスは次のポートを予約します。インターネットからの要求にパブリック IP アドレスを使用する場合、これらをプライベートポートとして定義することはできません。

   ポート 21, 22, 80, 443, 8080, 67, 161, 179, 500, 520, 3003, 3008, 3009, 3010, 3011, 4001, 5061, 9000, 7000.

   ただし、VIP などのインターネットに直接接続するサービスで標準ポート（ポート 443 など）を使用する場合は、ネットワークセキュリティグループを使用してポートマッピングを作成する必要があります。これにより、標準ポートがこの VIP サービス用に NetScaler で構成された別のポートにマップされます。

   たとえば、VIP サービスが VPX インスタンスのポート 8443 で実行されているが、パブリックポート 443 にマッピングされているとします。したがって、ユーザーがパブリック IP を介してポート 443 にアクセスすると、要求はプライベートポート 8443 に送信されます。

2. パブリック IP アドレスでは、ポートマッピングが動的に解放される、パッシブ FTP や ALG のようなプロトコルをサポートしていません。

3. 高可用性は、Azure ロードバランサーで構成された PIP ではなく、VPX インスタンスに関連付けられたパブリック IP アドレス（PIP）を使用するトラフィックでは機能しません。

> 注
>
> Azure Resource Manager では、NetScaler VPX インスタンスにはパブリック IP アドレス（PIP）と内部 IP アドレスの 2 つの IP アドレスが関連付けられています。外部トラフィックは PIP に接続しますが、内部 IP アドレスまたは NSIP はルーティング不可能です。VPX で VIP を設定するには、内部 IP アドレスと使用可能な空きポートのいずれかを使用します。VIP の構成に PIP を使用してはいけません。

NetScaler VPX スタンドアロンインスタンスを構成する

October 17, 2024

仮想マシンを作成して他のリソースを構成することにより、スタンドアロンモードで Azure Resource Manager（ARM）ポータルで単一の NetScaler VPX インスタンスをプロビジョニングできます。

はじめに

お使いの環境が次の要件を満たしていることを確認してください：

- Microsoft Azure ユーザーアカウント
- Microsoft Azure Resource Manager へのアクセス

- Microsoft Azure SDK
- Microsoft Azure PowerShell

Microsoft Azure ポータルのページで 、ユーザー名とパスワードを指定して Azure Resource Manager ポータルにログオンします。

> 注
>
> ARM ポータルで、1 つのペインでオプションをクリックすると、右側に新しいペインが開きます。ペイン間を移動してデバイスを構成します。

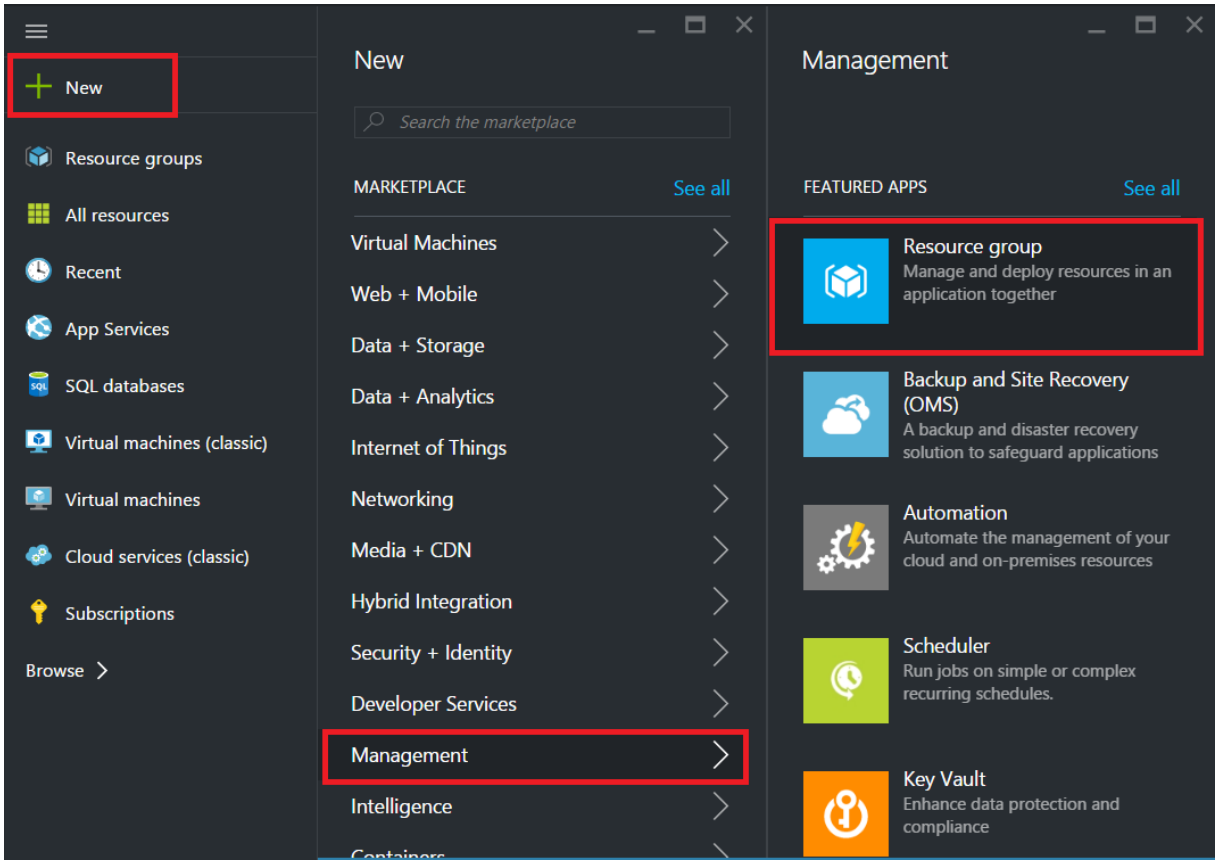
設定手順の概要

1. リソースグループの構成
2. ネットワークセキュリティグループの構成
3. 仮想ネットワークインターフェイスとそのサブネットの構成
4. ストレージアカウントの構成
5. 可用性セットの構成
6. NetScaler VPX インスタンスを構成します。

リソースグループの構成

すべてのリソースのコンテナとなる新しいリソースグループを作成します。リソースグループを使用して、リソースをグループとして展開、管理、および監視します。

1. 新規 > 管理 > リソースグループをクリックします。
2. リソースグループペインで、次の詳細を入力します。
    - リソースグループ名
    - リソースグループの場所
3. [**Create**] をクリックします。



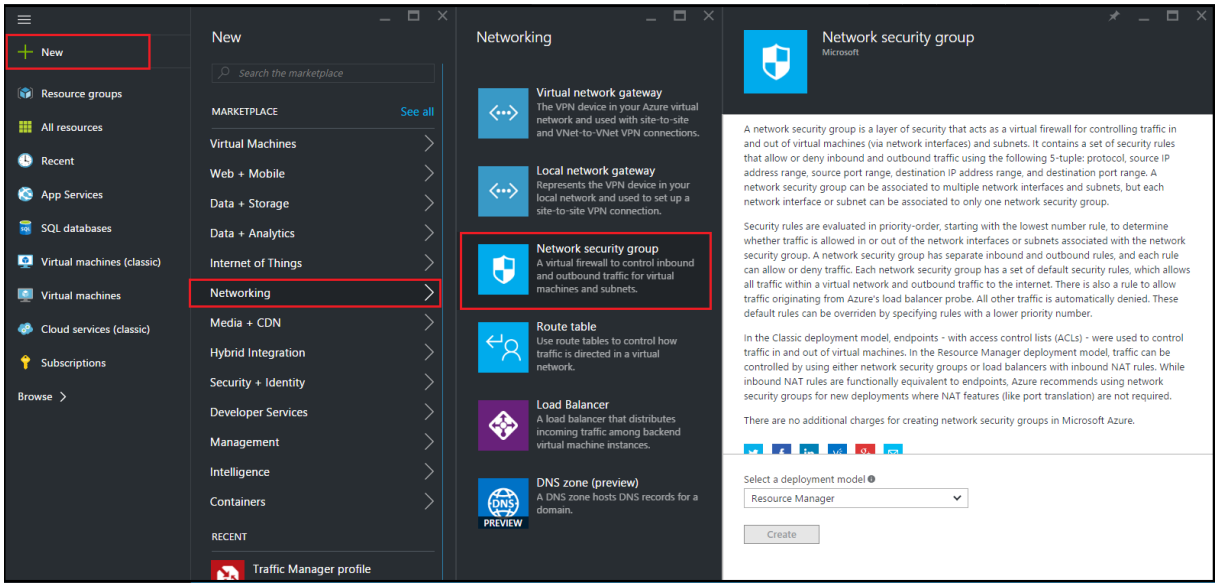
ネットワークセキュリティグループの構成

仮想ネットワーク内の着信トラフィックと発信トラフィックを制御するインバウンドルールとアウトバウンドルールを割り当てるネットワークセキュリティグループを作成します。ネットワークセキュリティグループを使用すると、単一の仮想マシンのセキュリティルールを定義したり、仮想ネットワークサブネットのセキュリティルールを定義したりできます。

1. [新規] > [ネットワーク] > [ネットワークセキュリティグループ] をクリックします。
2. [ネットワークセキュリティグループの作成] ペインで、次の詳細を入力し、[作成] をクリックします。

    - Name - セキュリティグループの名前を入力します
    - Resource group - ボックスの一覧からリソースグループを選択します

> 注
>
> 正しい場所を選択していることを確認します。場所が異なれば、ボックスの一覧に表示されるリソースの一覧も異なります。

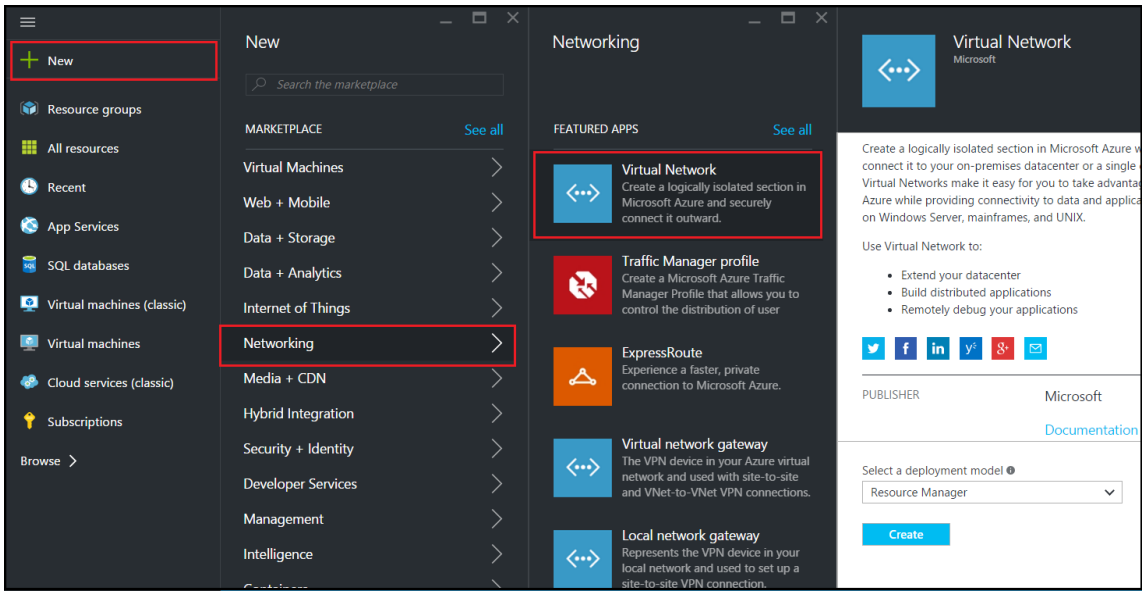


仮想ネットワークとサブネットを構成する

ARM の仮想ネットワークは、サービスのセキュリティを強化し、隔離するものです。同じ仮想ネットワークに属する VM およびサービスは、互いにアクセスできます。

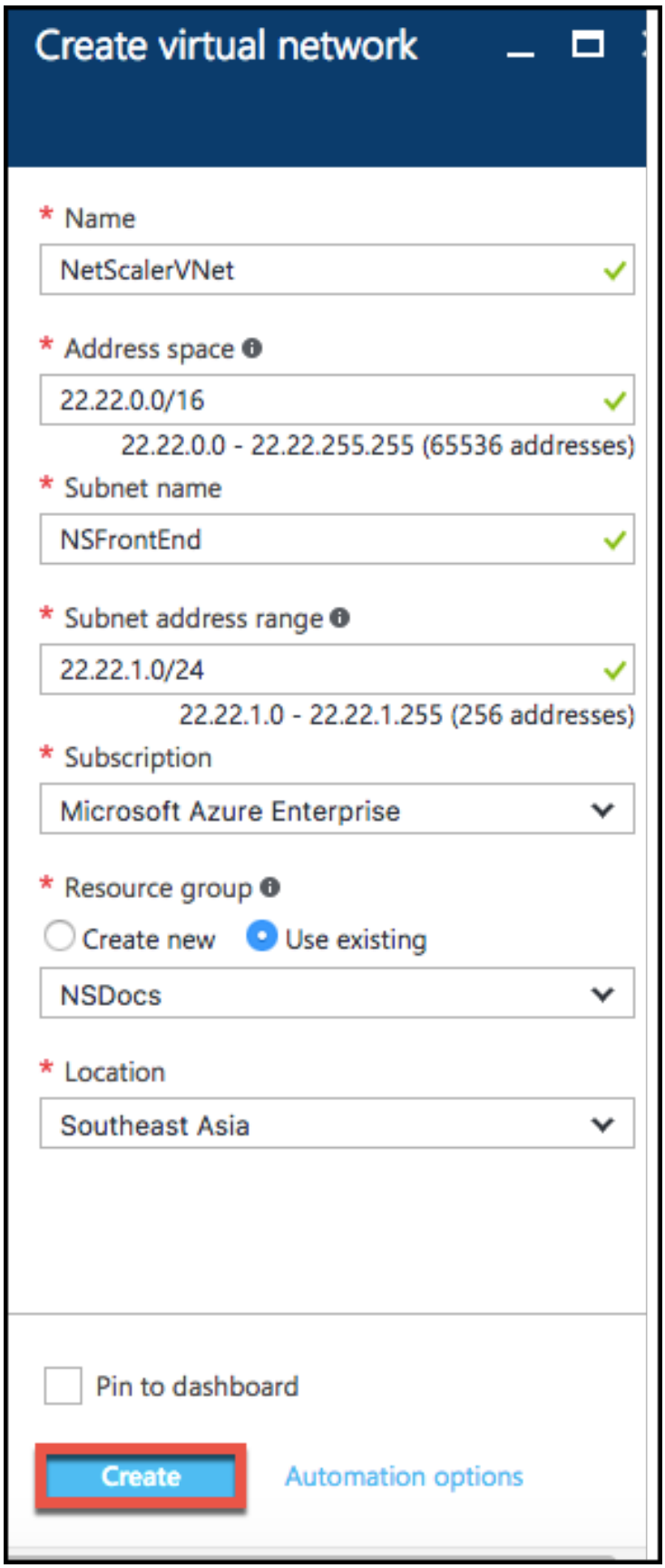
仮想ネットワークとサブネットを作成する手順は次のとおりです。

1. 新規 > ネットワーク > 仮想ネットワークをクリックします。
2. [仮想ネットワーク] ペインで、展開モードが [リソースマネージャ] であることを確認し、[作成] をクリックします。



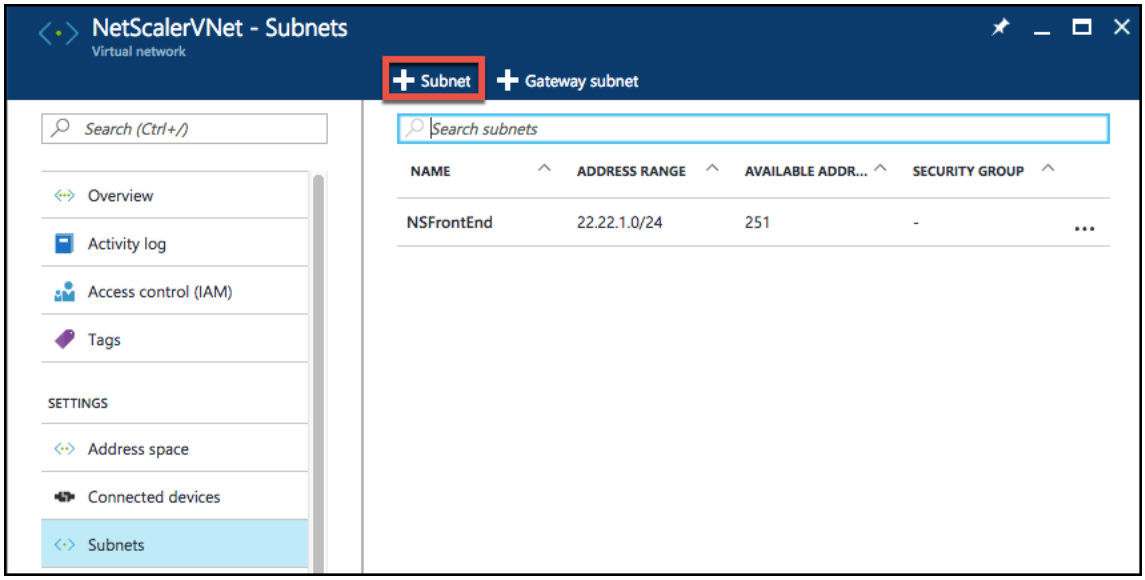
3. 仮想ネットワークの作成 ウィンドウで、次の値を入力し、作成 をクリックします。

- 仮想ネットワークの名前
- Address space - 仮想ネットワークの予約済 IP アドレスブロックを入力します
- サブネット-最初のサブネットの名前を入力します (2 つ目のサブネットはこのステップの後半で作成します)
- Subnet address range - サブネットの予約済 IP アドレスブロックを入力します
- Resource group - ボックスの一覧から以前に作成したリソースグループを選択します。



2 番目のサブネットを設定する

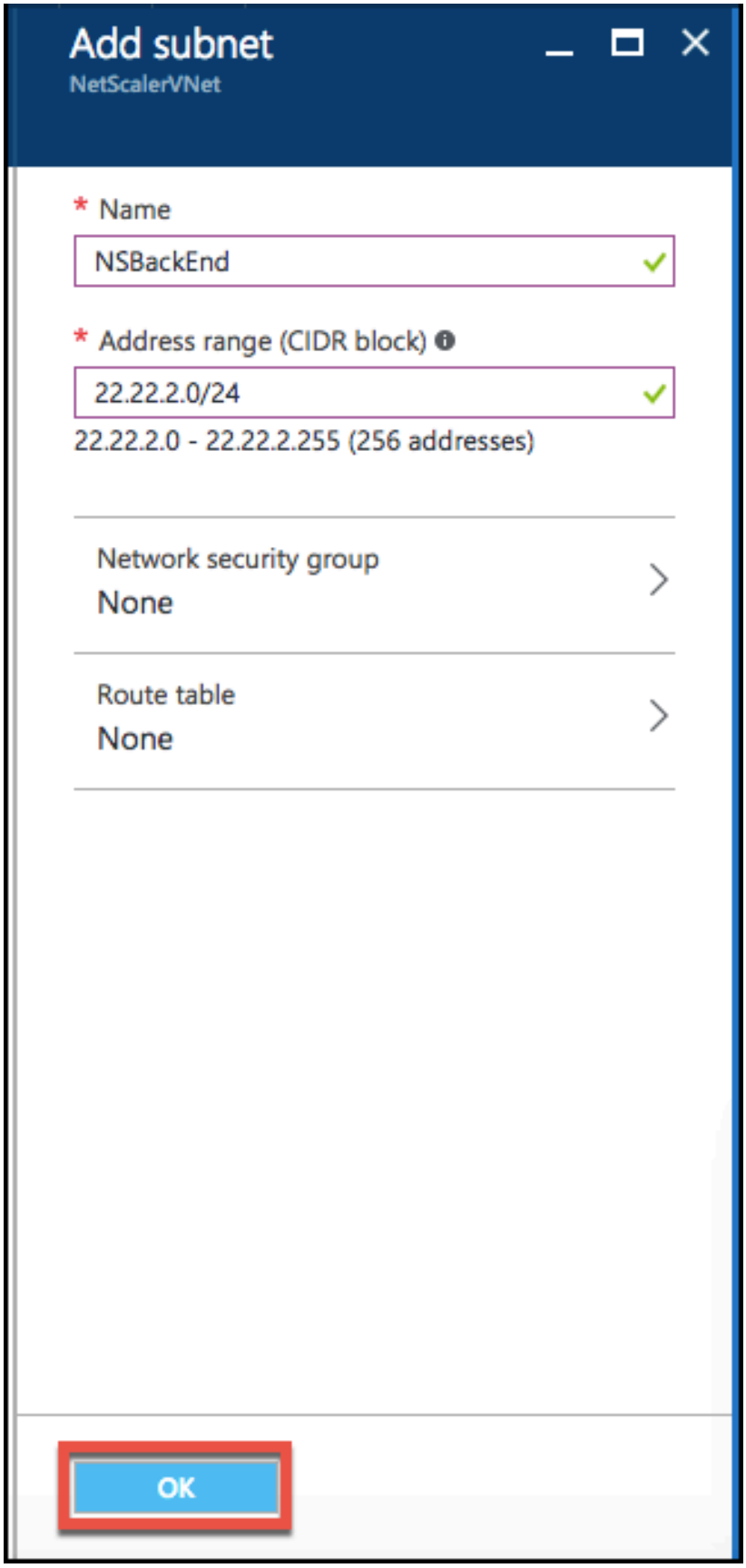
1. [すべてのリソース] ペインから新しく作成した仮想ネットワークを選択し、[設定] ペインで [サブネット] をクリックします。



2. **+Subnet** をクリックし、次の詳細を入力して 2 番目のサブネットを作成します。

   - 2 番目のサブネットの名前
   - Address range - サブネットの予約済 IP アドレスブロックを入力します
   - ネットワークセキュリティグループ-ドロップダウンリストからネットワークセキュリティグループを選択します。

3. [**Create**] をクリックします。

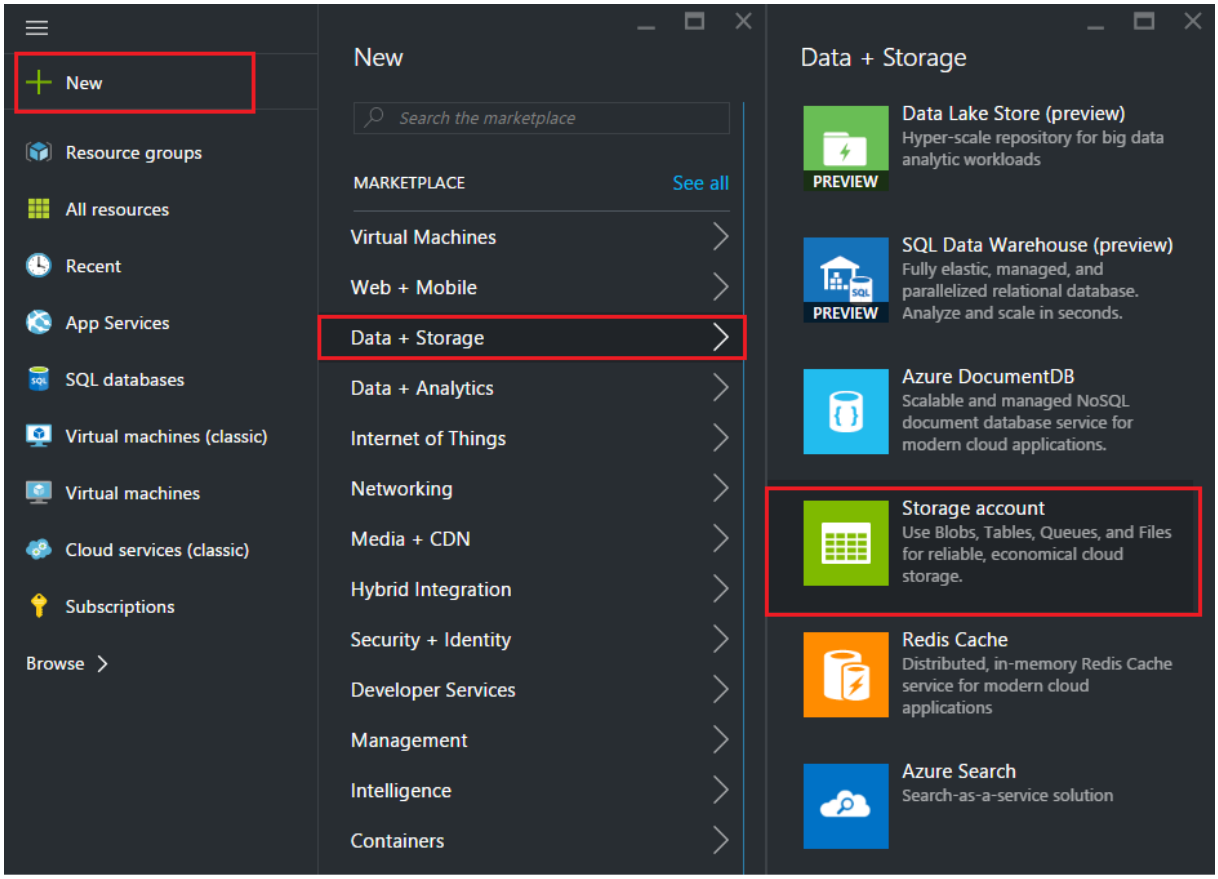


ストレージアカウントの構成

ARM IaaS インフラストラクチャストレージには、BLOB、表、キューおよびファイルの形式でデータを保存できるすべてのサービスが含まれます。ARM では、これらの形式のストレージデータを使用してアプリケーションを作成することもできます。

ストレージアカウントを作成してすべてのデータを保存します。

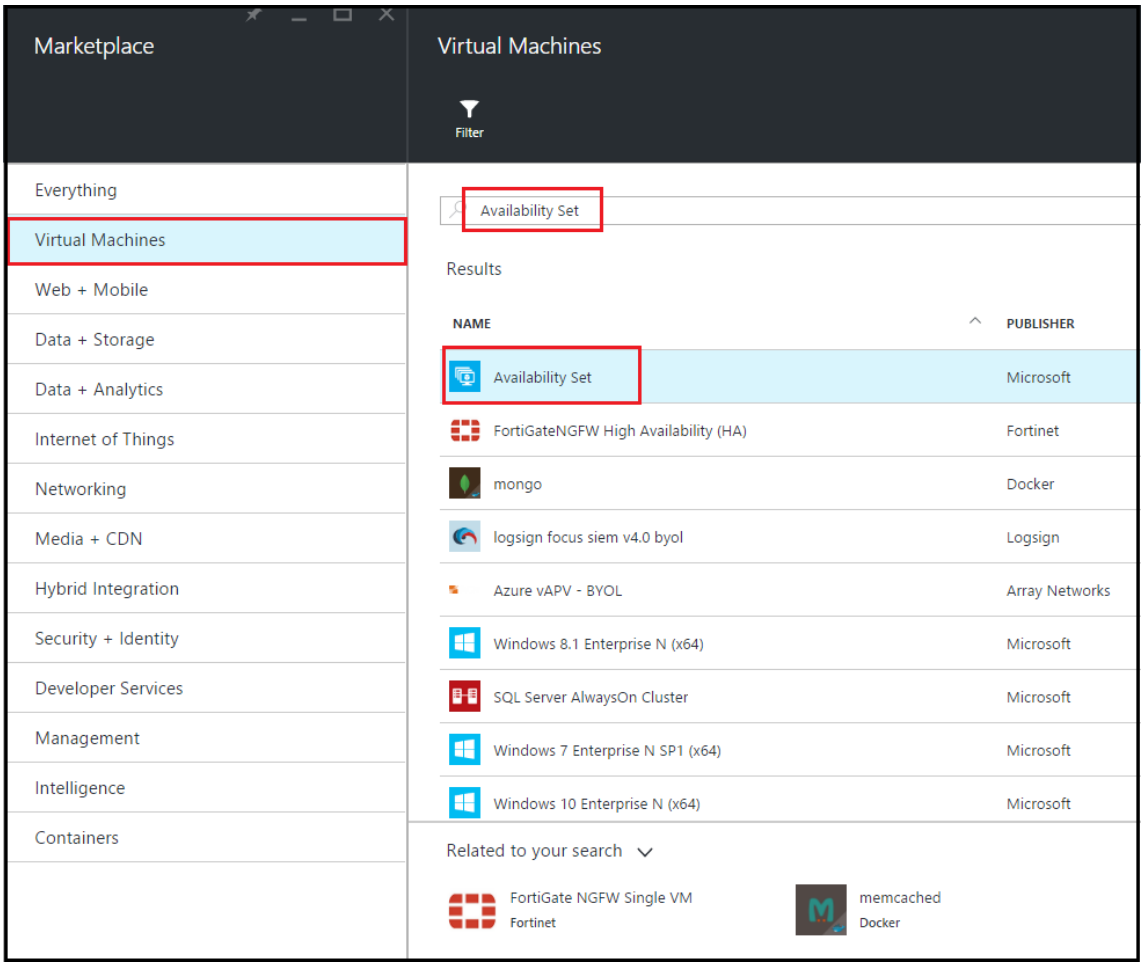
1. [**+** 新規] > [データ**] + [**ストレージ] > [ストレージアカウント] をクリックします。
2. ストレージアカウントの作成ペインで、次の詳細を入力します。
    - アカウントの名前
    - デプロイモード- 必ずリソースマネージャーを選択してください
    - アカウントの種類-ドロップダウンリストから「汎用」を選択します
    - レプリケーション-ドロップダウンリストから「ローカル冗長ストレージ」を選択します
    - Resource group - ボックスの一覧から新しく作成したリソースグループを選択します
3. [**Create**] をクリックします。



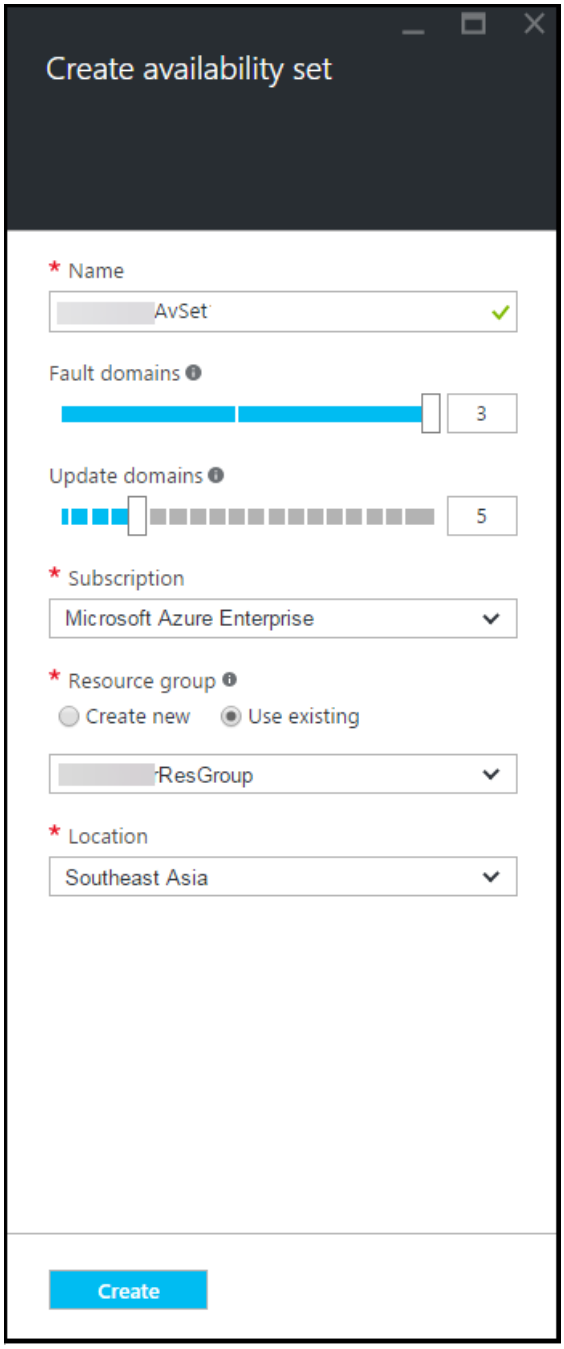
可用性セットの構成

可用性セットにより、計画的または計画外のメンテナンスが発生した場合でも、少なくとも 1 つの VM が稼働し続けることが保証されます。同じ可用性セットに属する 2 台以上の VM は、異なるフォールトドメインに配置されて、サービスの冗長性を確保します。

1. [**+** 新規] をクリックします。
2. MARKETPLACE ペインで「すべて表示」をクリックし、「仮想マシン」をクリックします。
3. 可用性セットを検索し、表示されたリストから [可用性セット エンティティ] を選択します。



4. [作成] をクリックし、 [可用性セットの作成] ウィンドウで、次の詳細を入力します。
   - セットの名前
   - Resource group - ボックスの一覧から新しく作成したリソースグループを選択します
5. [**Create**] をクリックします。



NetScaler VPX インスタンスの構成

仮想ネットワークに NetScaler VPX のインスタンスを作成します。Azure Marketplace から NetScaler VPX イメージを取得し、Azure Resource Manager ポータルを使用して NetScaler VPX インスタンスを作成します。

NetScaler VPX インスタンスの作成を開始する前に、インスタンスが存在する必要なサブネットを持つ仮想ネットワークが作成されていることを確認してください。仮想マシンのプロビジョニング時に仮想ネットワークを作成することもできますが、柔軟性に欠けるため別のサブネットを作成することはできません。仮想ネットワークの作成につ

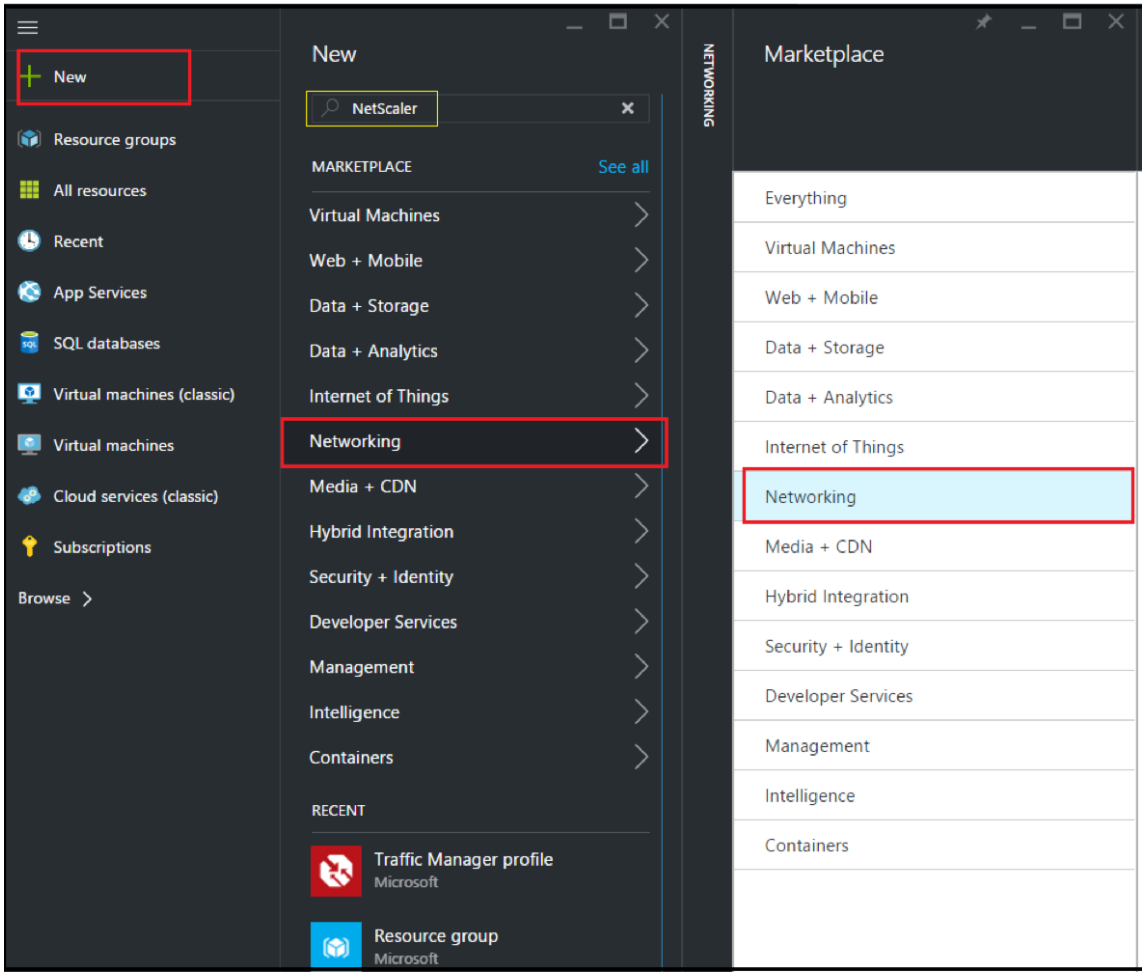
いては、http://azure.microsoft.com/en-us/documentation/articles/create-virtual-network/を参照してください。

オプションで、仮想マシンがインターネットリソースにアクセスできるようにする DNS サーバと VPN 接続を設定します。

> 注
>
> プロビジョニング時にネットワーク情報を利用できるよう、NetScaler VPX VM をプロビジョニングする前に、リソースグループ、ネットワークセキュリティグループ、仮想ネットワークおよび他のエンティティを作成することをお勧めします。

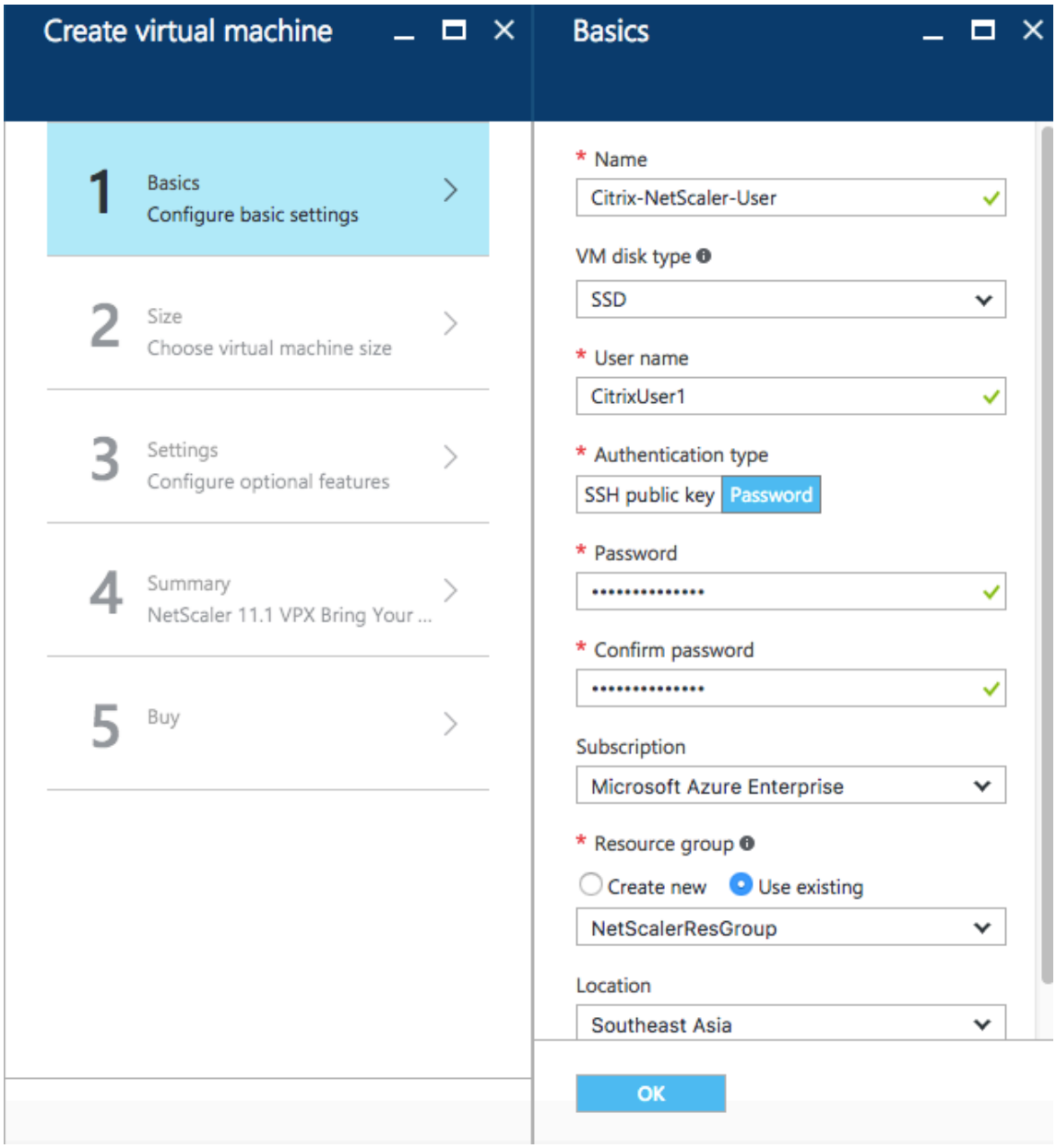
1. [**+** 新規] > [ネットワーク] をクリックします。
2. [すべて表示] をクリックし、[ネットワーク] ペインで [**NetScaler 13.0**] をクリックします。
3. ソフトウェアプランのリストから「**NetScaler 13.0 VPX Bring Your Own License** 」を選択します。

   ARM ポータルでエンティティをすばやく見つける方法として、Azure Marketplace 検索ボックスにエンティティの名前を入力して を押すこともできます <Enter>。検索ボックスに「NetScaler 」と入力して、NetScaler イメージを検索します。



> 注
>
> 最新のイメージを選択するようにしてください。Citrix ADC イメージの名前にリリース番号が含まれている場合があります。

4. [**NetScaler VPX** 自分のライセンスを持参] ページのドロップダウンリストから [リソースマネージャー] を選択し、[作成] をクリックします。



5. [仮想マシンの作成] ウィンドウで、各セクションで必要な値を指定して、仮想マシンを作成します。各セクションで「**OK**」をクリックして設定を保存します。

ベーシック:

- Name - NetScaler VPX インスタンスの名前を指定します
- VM disk type - ボックスの一覧から SSD （デフォルト値）または HDD を選択します。
- User name and Password - 作成したリソースグループのリソースにアクセスするためのユーザー名およびパスワードを指定します
- Authentication Type - ［SSH Public Key］または［Password］を選択します
- Resource group - ボックスの一覧から作成したリソースグループを選択します。

ここではリソースグループを作成できますが、Azure Resource Manager で［Resource groups］からリソースグループを作成して、そのグループをボックスの一覧から選択することをお勧めします

> 注
>
> Azure スタック環境では、基本パラメータに加えて、次のパラメータを指定します。
>
> - Azure スタックドメイン
> - Azure スタックテナント (オプション)
> - Azure クライアント (オプション)
> - Azure クライアントシークレット (オプション)

サイズ:

基本設定で選択した仮想マシンのディスクタイプ、SDD、または HDD に応じて、ディスクサイズが表示されます。

- 必要に応じてディスクサイズを選択し、[選択] をクリックします。

概要:

- デフォルトのディスクタイプ（[Standard]）を選択します
- Storage account - ストレージアカウントを選択します
- Virtual network - 仮想ネットワークを選択します
- Subnet - サブネットアドレスを設定します
- Public IP address - IP アドレス割り当ての種類を選択します
- Network security group - 作成したセキュリティグループを選択します。セキュリティグループで、受信規則および送信規則が構成されていることを確認します。
- アベイラビリティセット-ドロップダウンメニューボックスからアベイラビリティセットを選択します

設定:

構成設定が検証され、［Summary］ページに検証の結果が表示されます。検証が失敗すると、［Summary］ページに障害の理由が表示されます。個別のセクションに戻り、必要に応じて変更します。検証に合格したら、「**OK**」をクリックします。

購入ページでオファーの詳細と法的条件を確認し、「購入」をクリックします。

購入ページでオファーの詳細と法的条件を確認し、「購入」をクリックします。

高可用性導入では、同じ可用性セットおよび同じリソースグループに NetScaler VPX 独立したインスタンスを 2 つ作成して、アクティブ/スタンバイ構成で展開します。

NetScaler VPX スタンドアロンインスタンスに複数の IP アドレスを構成する

October 17, 2024

このセクションでは、Azure Resource Manager（ARM）で複数の IP アドレスを使用してスタンドアロン NetScaler ADC VPX インスタンスを構成する方法について説明します。VPX インスタンスには 1 つ以上の NIC を接続でき、各 NIC には 1 つ以上の静的または動的なパブリックおよびプライベート IP アドレスを割り当てることができます。複数の IP アドレスを NSIP、VIP、SNIP などとして割り当てることができます。

詳細については、Azure のドキュメント「Azure ポータルを使用して仮想マシンに複数の IP アドレスを割り当てる」を参照してください。

PowerShell コマンドを使用する場合は、「PowerShell コマンドを使用してスタンドアロン モードで NetScaler VPX インスタンスに複数の IP アドレスを構成する」を参照してください。

使用例

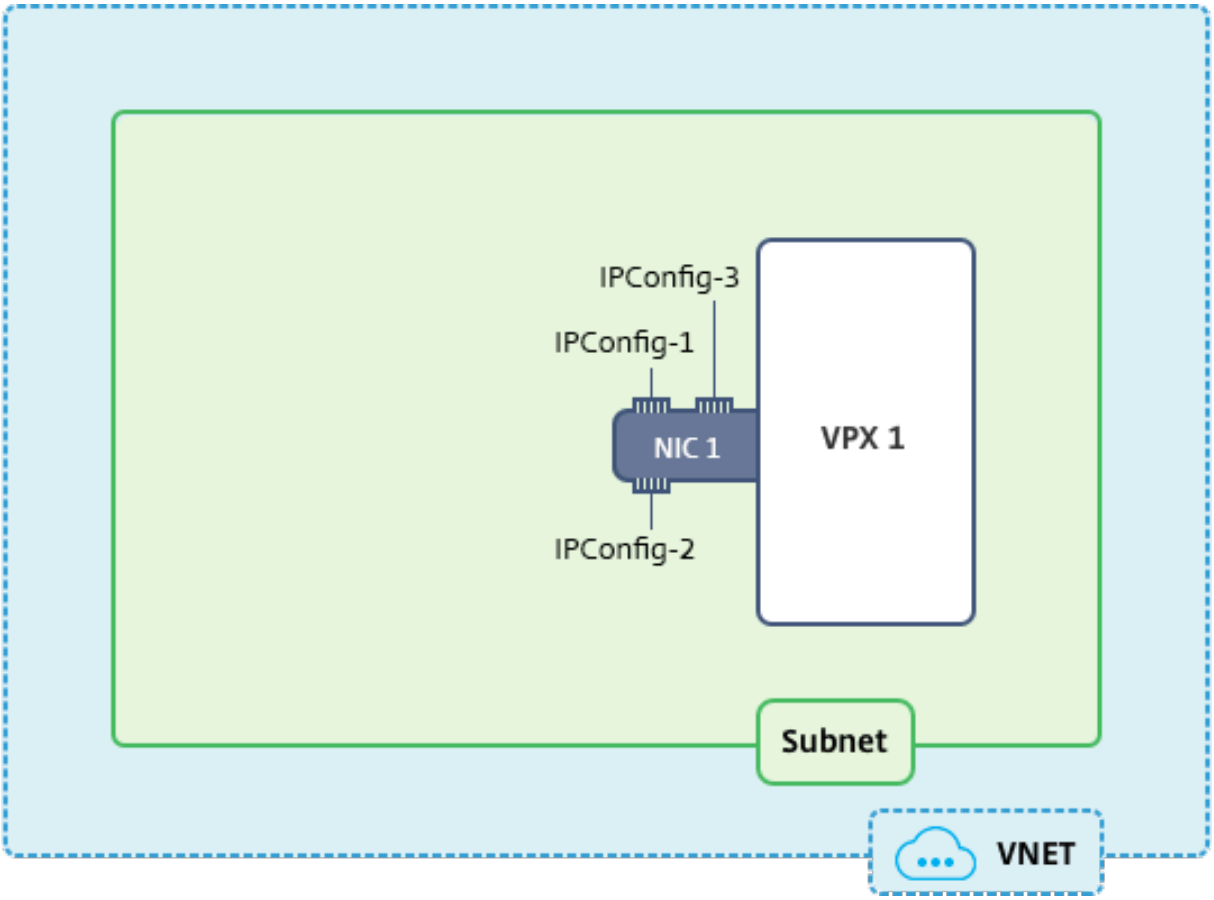
この使用例では、スタンドアロンの NetScaler ADC VPX アプライアンスは、仮想ネットワーク（VNET）に接続された単一の NIC で構成されます。NIC は、表に示すように、3 つの IP 構成（ipconfig）に関連付けられ、各サーバは異なる目的で使用されます。

IP コンフィグ	関連付けられている	目的
ipconfig1	静的パブリック IP アドレス; 静的プライベート IP アドレス	管理トラフィックを提供する
ipconfig2	静的パブリック IP アドレス; 静的プライベートアドレス	クライアント側のトラフィックを提供する
ipconfig3	静的プライベート IP アドレス	バックエンドサーバーと通信する

> 注
>
> `IPConfig-3`はパブリック IP アドレスに関連付けられていません。

図: トポロジ

次の図はこの使用例を視覚的に示しています。



> 注
>
> マルチ NIC、マルチ IP Azure NetScaler VPX 展開では、プライマリ（最初の）NIC のプライマリ（最初）`IPConfig` に関連付けられたプライベート IP が、アプライアンスの管理 NSIP として自動的に追加されます。`IPConfigs`に関連付けられた残りのプライベート IP アドレスは、必要に応じて、`add ns ip` コマンドを使用して VIP または SNIP として VPX インスタンスに追加する必要があります。

はじめに

始める前に、次のリンクに示す手順に従って VPX インスタンスを作成します。

NetScaler VPX スタンドアロンインスタンスを構成する

このユースケースでは、NSDoc0330VM VPX インスタンスが作成されます。

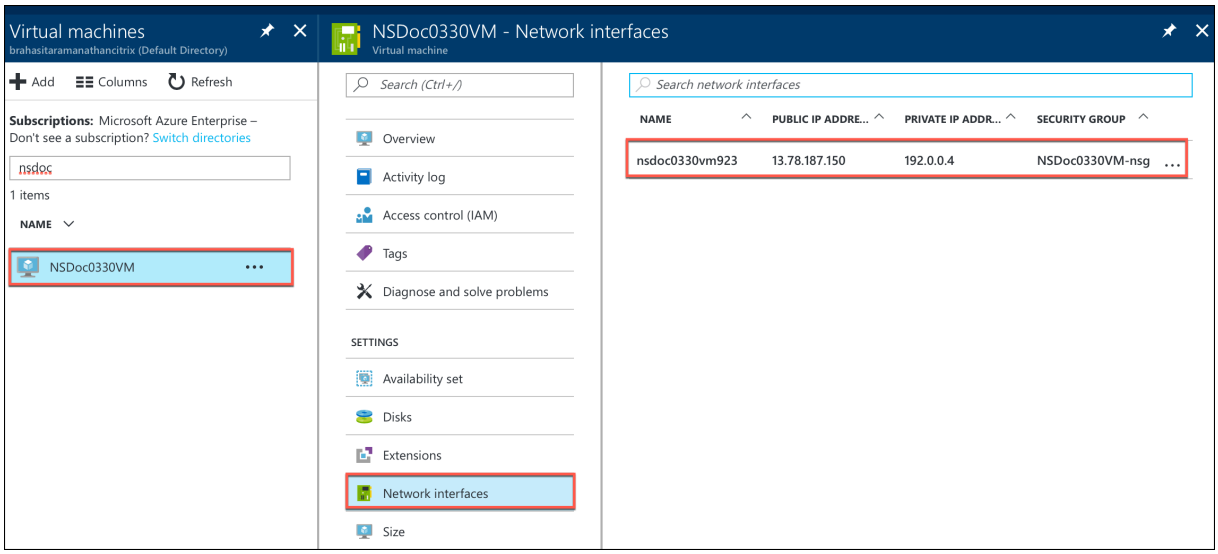
スタンドアロンモードで、**NetScaler VPX** インスタンスに対して複数 **IP** アドレスを構成する手順。

スタンドアロンモードの NetScaler VPX アプライアンスに複数の IP アドレスを構成するには:

1. VM への IP アドレス追加
2. NetScaler が所有する IP アドレスを構成する

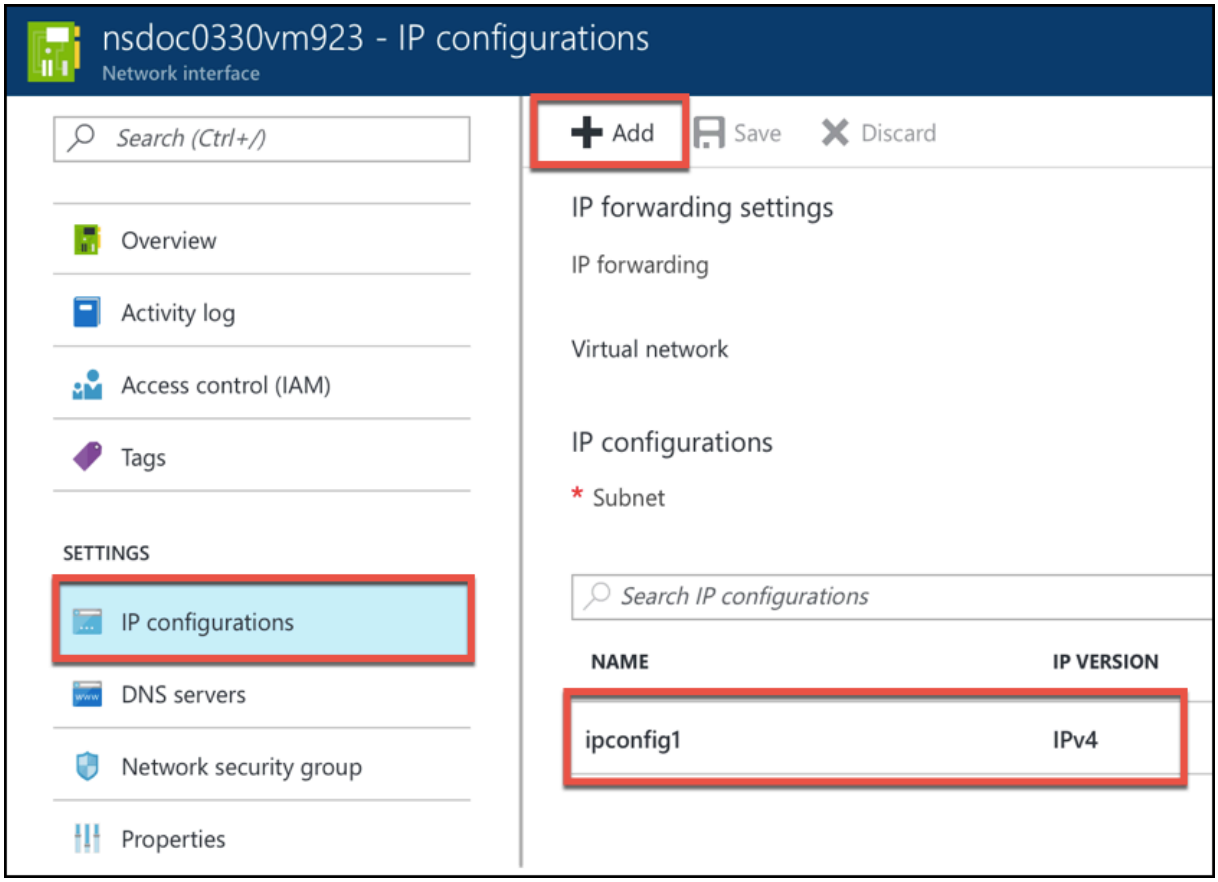
ステップ **1**: VM に IP アドレスを追加する

1. ポータルで [その他のサービス] をクリックし、フィルターボックスに「仮想マシン」と入力し、[仮想マシン] をクリックします。
2. 仮想マシンのブレードで、IP アドレスを追加する仮想マシンをクリックします。表示される仮想マシンブレードの [ネットワーク インターフェイス] をクリックし、ネットワークインターフェイスを選択します。



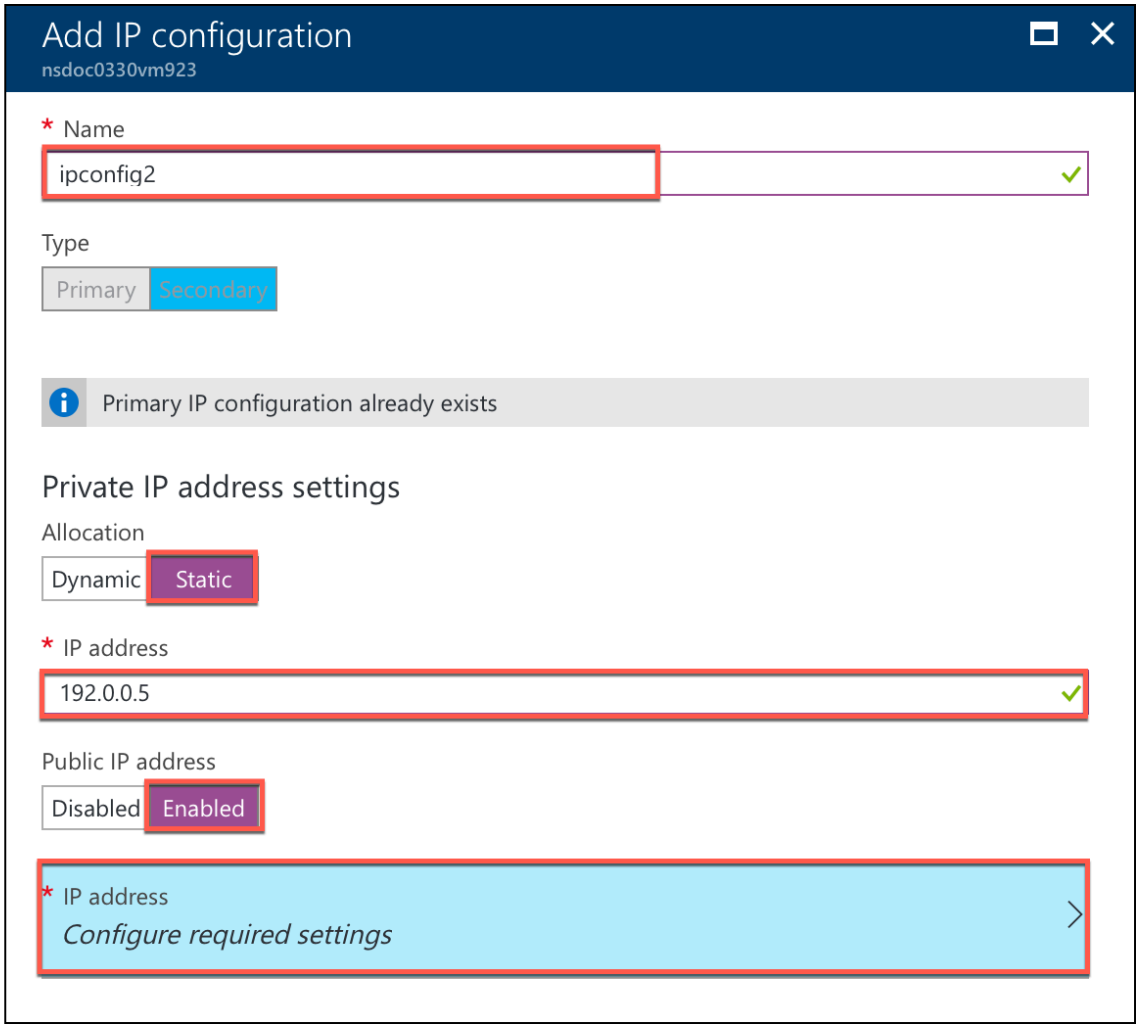
選択した NIC のブレードで、[**IP** 構成] をクリックします。仮想マシンの作成時に割り当てられた既存の IP 構成、**ipconfig1** が表示されます。この使用例では、ipconfig1 に割り当てられている IP アドレスが静的アドレスであることを確認します。次に、さらに 2 つの IP 構成、ipconfig2（VIP）と ipconfig3（SNIP）を作成します。

さらに`ipconfigs`を作成するには、**Add** を作成します。



[**IP** 構成の追加] ウィンドウで、[名前] を入力し、割り当て方法として [静的] を指定し、IP アドレス (このユースケースでは 192.0.0.5) を入力し、[パブリック **IP** アドレス] を有効にします。

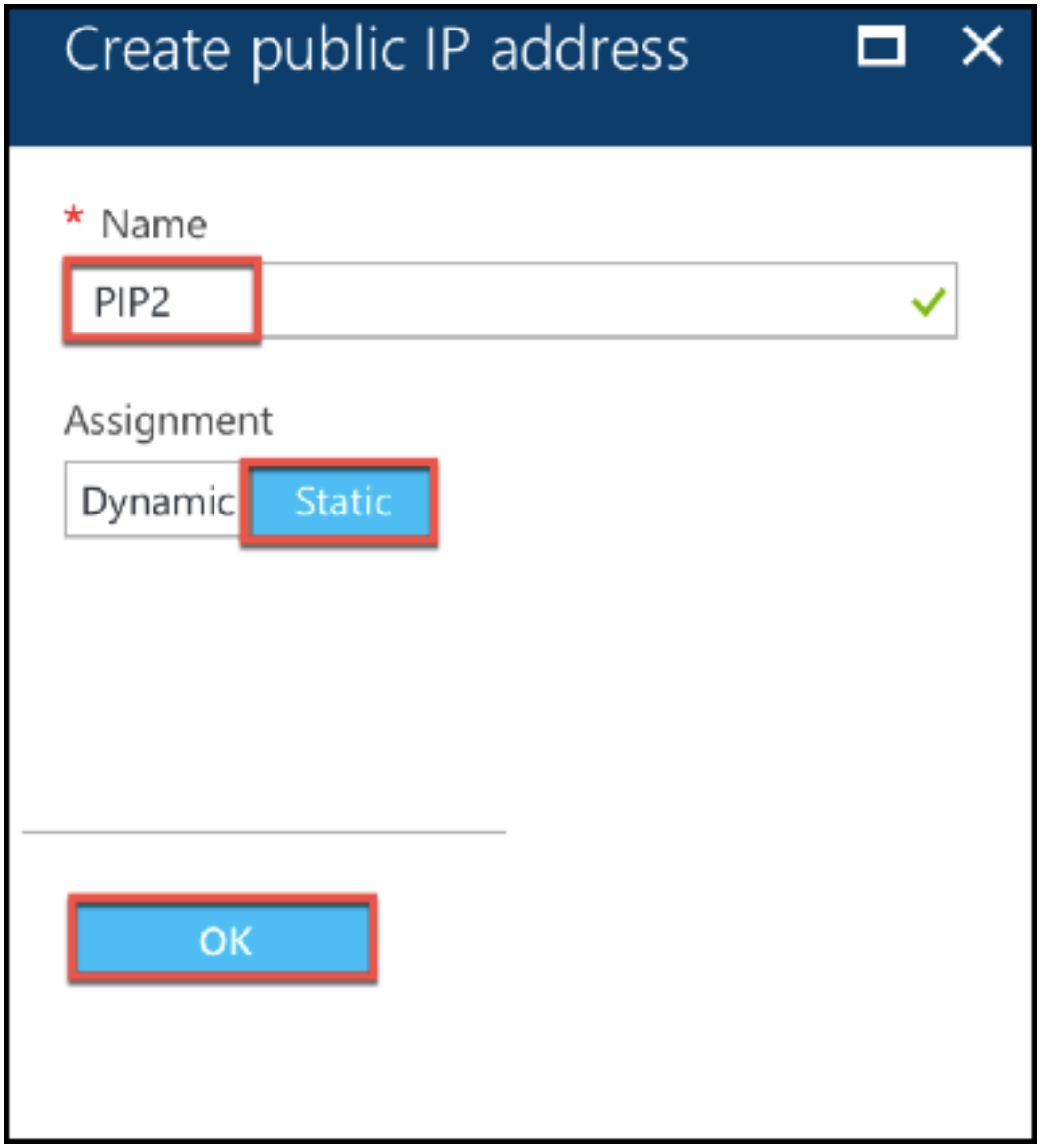
> 注
>
> 静的なプライベート IP アドレスを追加する前に、IP アドレスの可用性をチェックし、その IP アドレスが、NIC の接続先と同じサブネットに属していることを確認します。



次に、[必要な設定の構成] をクリックして ipconfig2 の静的パブリック IP アドレスを作成します。

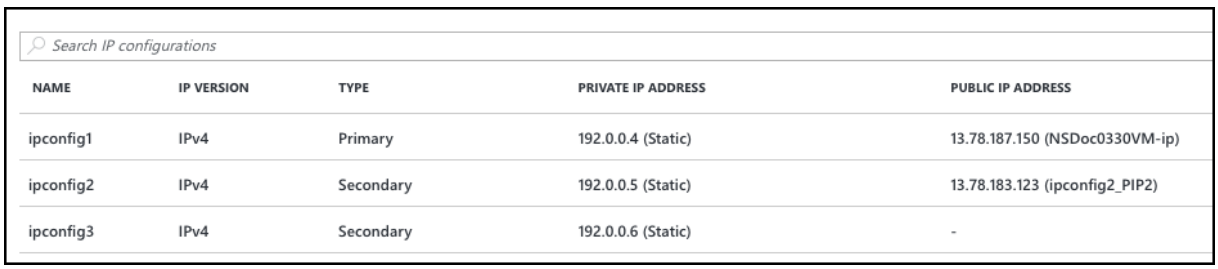
デフォルトでは、パブリック IP アドレスは動的なアドレスです。VM に常に同じパブリック IP アドレスを使用させるために、静的なパブリック IP アドレスを作成します。

「パブリック IP アドレスの作成」ブレードで「名前」を追加し、「割り当て」で「静的」をクリックします。[**OK**] をクリックします。



> 注
>
> 割り当て方式を静的に設定している場合でも、パブリック IP リソースに割り当てられる実際の IP アドレスを指定することはできません。代わりに、リソースが作成された Azure の場所で利用可能な IP アドレスプールから割り当てられます。

手順に従って、もう 1 つの IP 構成 ipconfig3 を追加します。パブリック IP アドレスは必須ではありません。



手順 **2**: **NetScaler** 固有の **IP** アドレスの構成

GUI または`add ns ip`コマンドを使用して、NetScaler が所有する IP アドレスを設定します。詳細については、「Citrix ADC 所有の IP アドレスの構成」を参照してください。

複数の **IP** アドレスと **NIC** を使用して高可用性設定を構成する

October 17, 2024

Microsoft Azure デプロイメントでは、Azure ロードバランサー（ALB）を使用して、2 つの NetScaler VPX インスタンスの高可用性構成を実現します。これは、ALB でヘルスプローブを構成することによって実現されます。ALB は、プライマリインスタンスとセカンダリインスタンスの両方に 5 秒ごとにヘルスプローブを送信することで、各 VPX インスタンスを監視します。

この設定では、プライマリノードだけがヘルスプローブに応答し、セカンダリノードは応答しません。プライマリがヘルスプローブに応答を送信すると、ALB はインスタンスへのデータトラフィックの送信を開始します。プライマリインスタンスが 2 回連続してヘルスプローブに失敗した場合、ALB はトラフィックをそのインスタンスにリダイレクトしません。フェイルオーバー時は、新しいプライマリがヘルスプローブへの応答を開始し、ALB はそのプライマリにトラフィックをリダイレクトします。標準の VPX 高可用性フェイルオーバー時間は 3 秒です。トラフィックスイッチングにかかる合計フェールオーバー時間は、最大 13 秒です。

Azure のアクティブ/パッシブ高可用性（HA）セットアップで、複数の NIC を持つ一対の NetScaler VPX インスタンスを展開できます。各 NIC に複数の IP アドレスを設定できます。

マルチ NIC の高可用性導入では、次のオプションを使用できます。

- Azure 可用性セットを使用した高可用性
- Azure アベイラビリティーゾーンを使用した高可用性

Azure アベイラビリティセットとアベイラビリティーゾーンの詳細については、Azure のドキュメント「Linux 仮想マシンの可用性の管理」を参照してください。

可用性セットを使用した高可用性

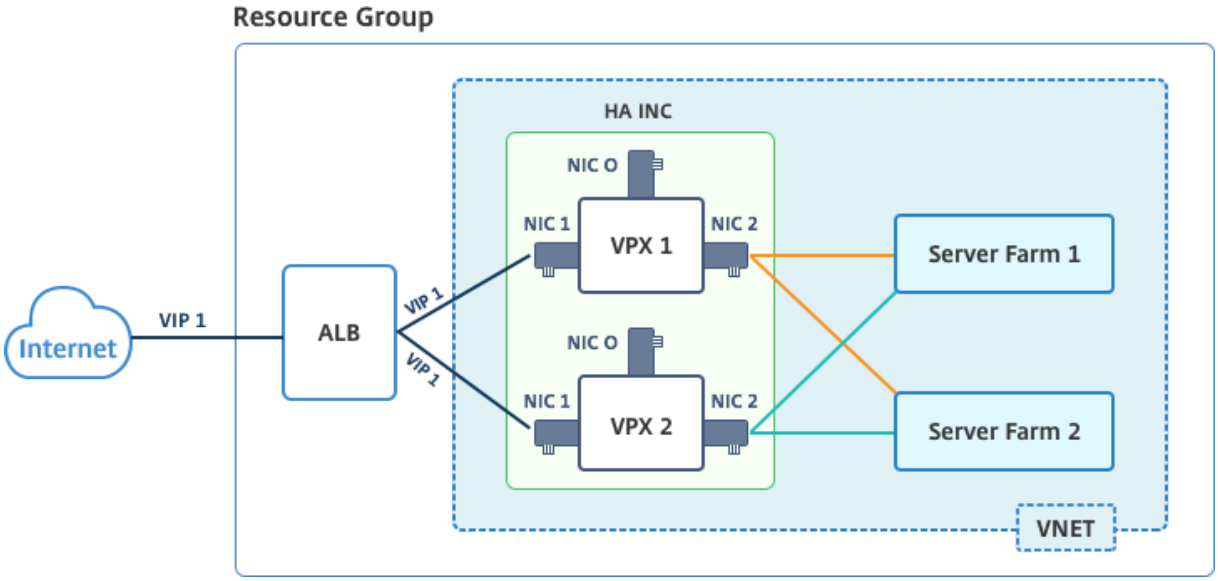
可用性セットを使用した高可用性セットアップは、次の要件を満たす必要があります。

- HA Independent Network Configuration（INC）構成
- Direct Server Return（DSR）モードの Azure Load Balancer（ALB）

すべてのトラフィックはプライマリノードを通過します。セカンダリノードは、プライマリノードが失敗するまでスタンバイモードを維持します。

> 注
>
> Azure クラウド上の NetScaler VPX 高可用性デプロイが機能するには、2 つの VPX ノード間で移動できるフローティングパブリック IP（PIP）が必要です。Azure ロードバランサー (ALB) は、フローティング PIP を提供します。このフローティング PIP は、フェールオーバーが発生した場合に自動的に第 2 ノードに移動されます。

図:Azure 可用性セットを使用した高可用性デプロイアーキテクチャの例



アクティブ/パッシブ展開では、ALB フロントエンドパブリック IP (PIP) アドレスが各 VPX ノードに VIP アドレスとして追加されます。HA-INC 構成では、これらの VIP アドレスはフローティングされ、SNIP アドレスはインスタンス固有のアドレスとなります。

VPX ペアをアクティブ/パッシブ高可用性モードでデプロイするには、次の 2 つの方法があります:

- **NetScaler VPX** 標準高可用性テンプレート: このオプションを使用して、3 つのサブネットと 6 つの NIC のデフォルトオプションで HA ペアを構成します。
- **Windows PowerShell** コマンド: このオプションを使用して、サブネットと NIC の要件に応じて HA ペアを構成します。

このトピックでは、Citrix テンプレートを使用してアクティブ/パッシブ HA セットアップで VPX ペアを展開する方法について説明します。PowerShell コマンドを使用する場合は、「PowerShell コマンドを使用して複数の IP アドレスと NIC で HA セットアップを構成する」を参照してください。

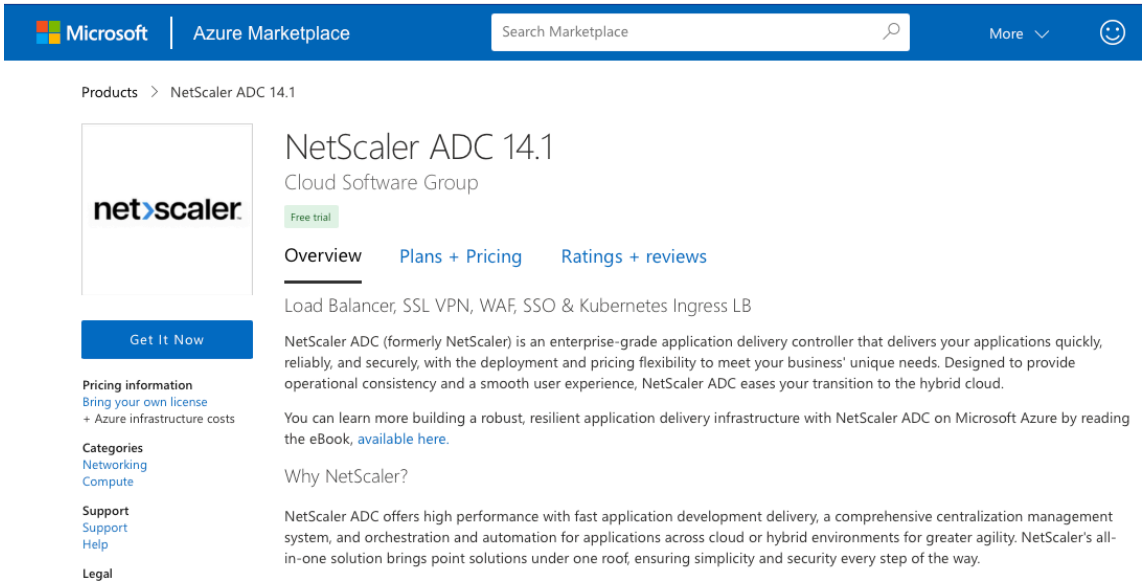
NetScaler の高可用性テンプレートを使用して **HA-INC** ノードを構成する

標準テンプレートを使用すると、一対の VPX インスタンスを HA-INC モードで迅速かつ効率的にデプロイできます。このテンプレートでは、3 つのサブネットと 6 つの NIC を持つ 2 つのノードが作成されます。各サブネットには、両方の VPX インスタンスに対して 2 つの NIC があります。

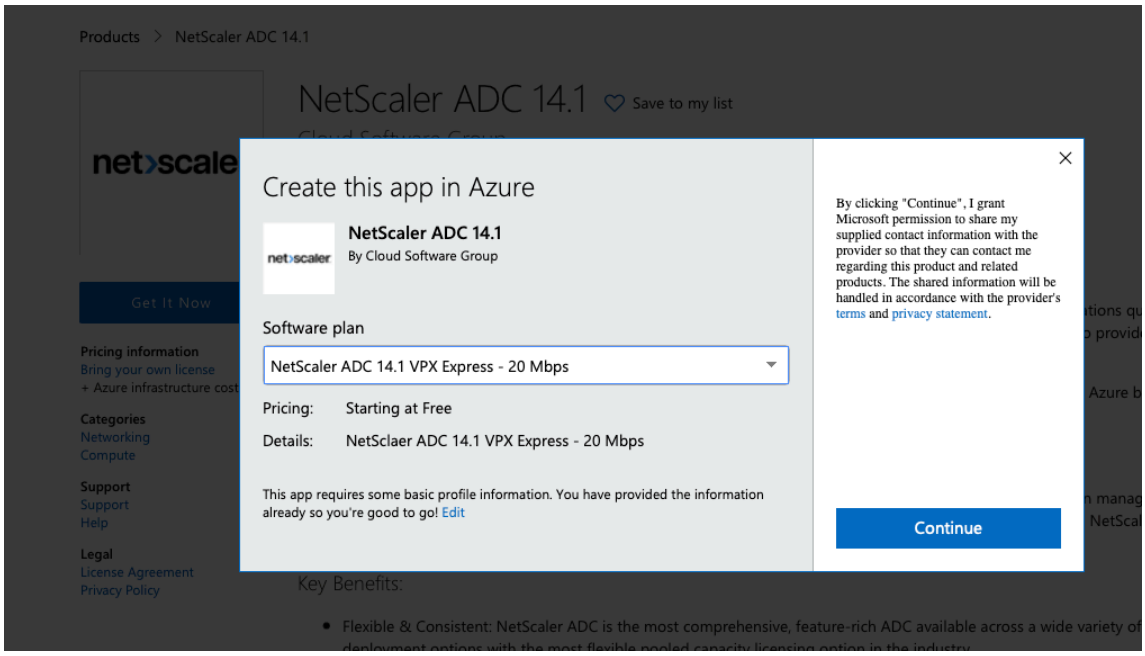
NetScaler HA ペアテンプレートは、Azure マーケットプレイスで入手できます。

次の手順を実行して、Azure 可用性セットを使用して、テンプレートを起動し、高可用性 VPX ペアをデプロイします。

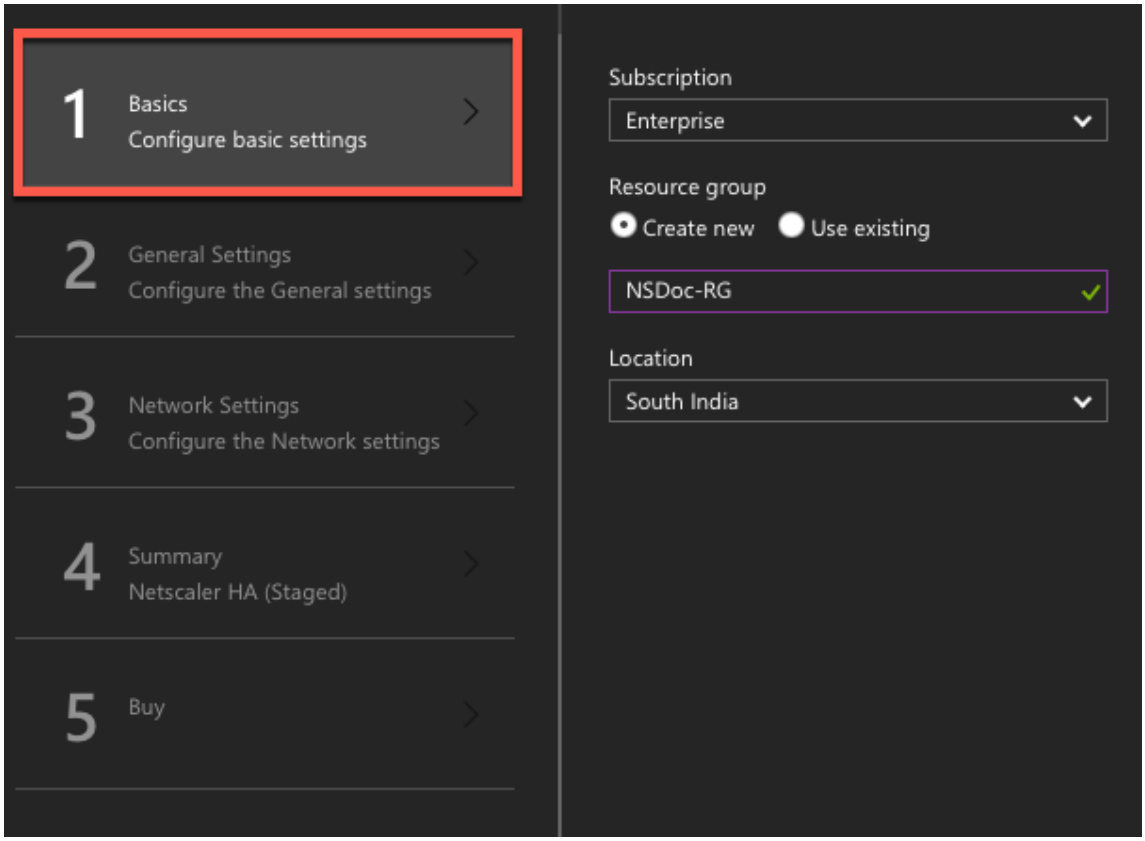
1. Azure Marketplace から **NetScaler** を検索します。



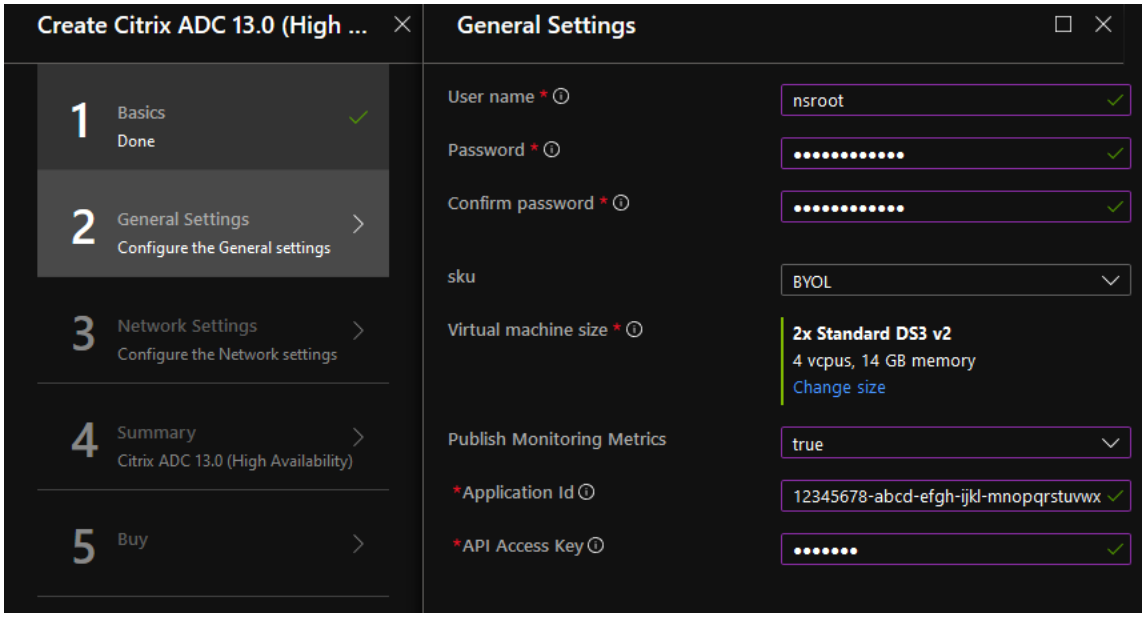
2. [今すぐ入手] をクリックします。
3. 必要な HA 導入とライセンスを選択し、[続行] をクリックします。



4. [基本] ページが表示されます。リソースグループを作成し、**OK** を選択します。



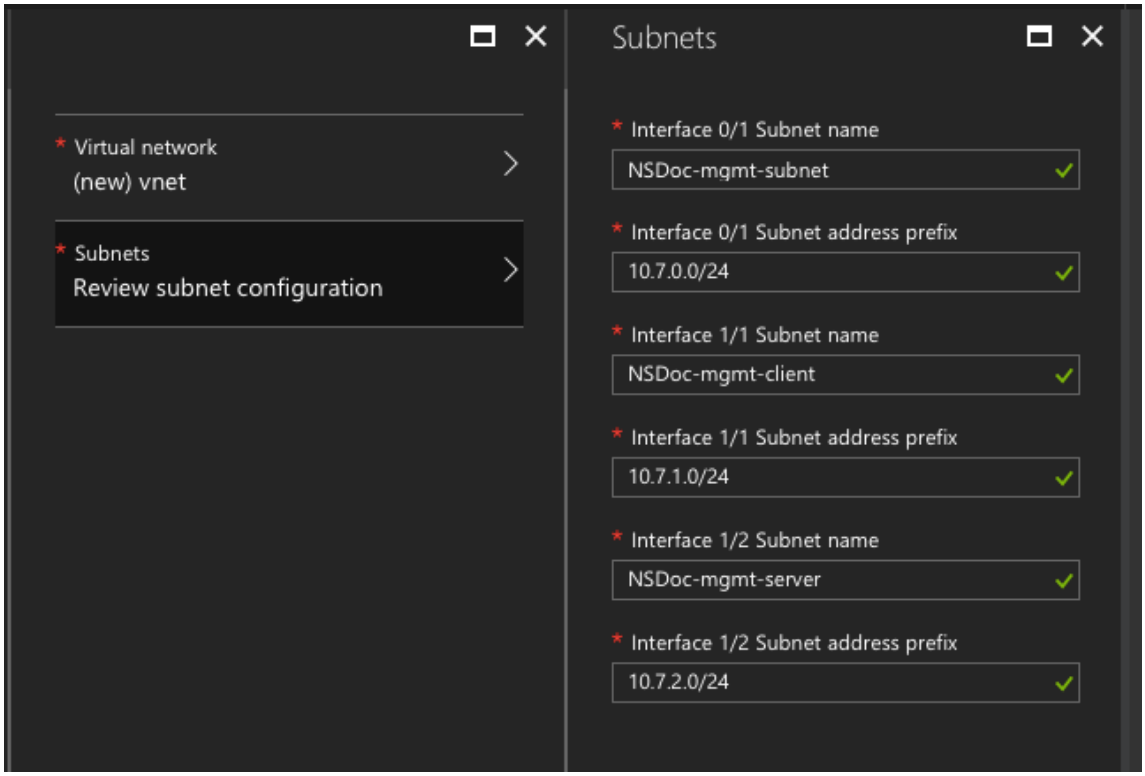
5. [一般設定] ページが表示されます。詳細を入力して「**OK**」を選択します。



> 注
>
> デフォルトでは、監視メトリックの公開 オプションは **false** に設定されています。このオプションを有効にする場合は、**true** を選択します。リソースにアクセスできる AzureActive Directory（ADD）ア

> プリケーションとサービスプリンシパルを作成します。新しく作成された AAD アプリケーションにコントリビュータロールを割り当てます。詳細については、「ポータルを使用してリソースにアクセスできる Azure Active Directory アプリケーションおよびサービスプリンシパルを作成する」を参照してください。

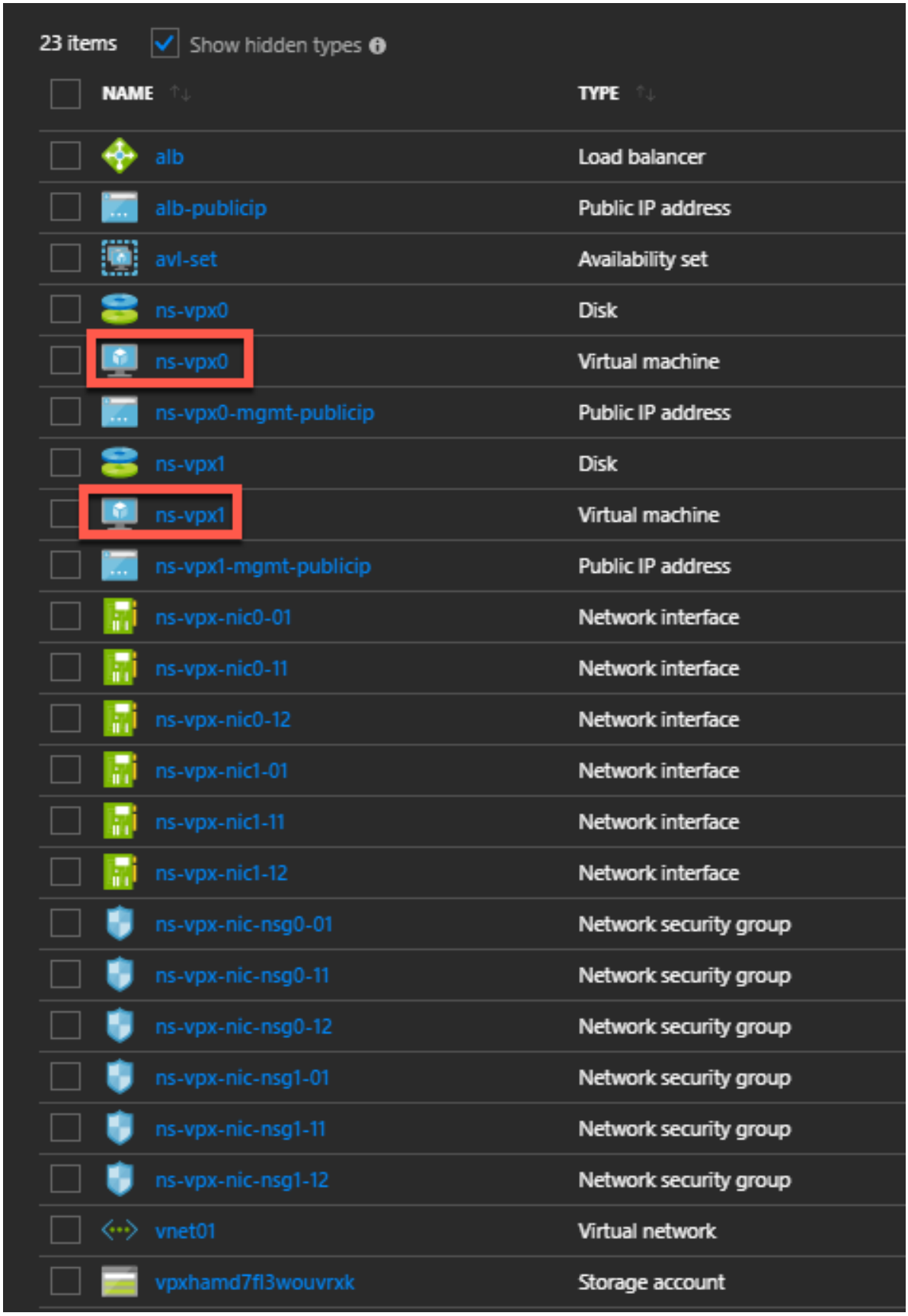
6. [ネットワーク設定] ページが表示されます。VNet とサブネットの構成を確認し、必要な設定を編集して、[**OK]** を選択します。



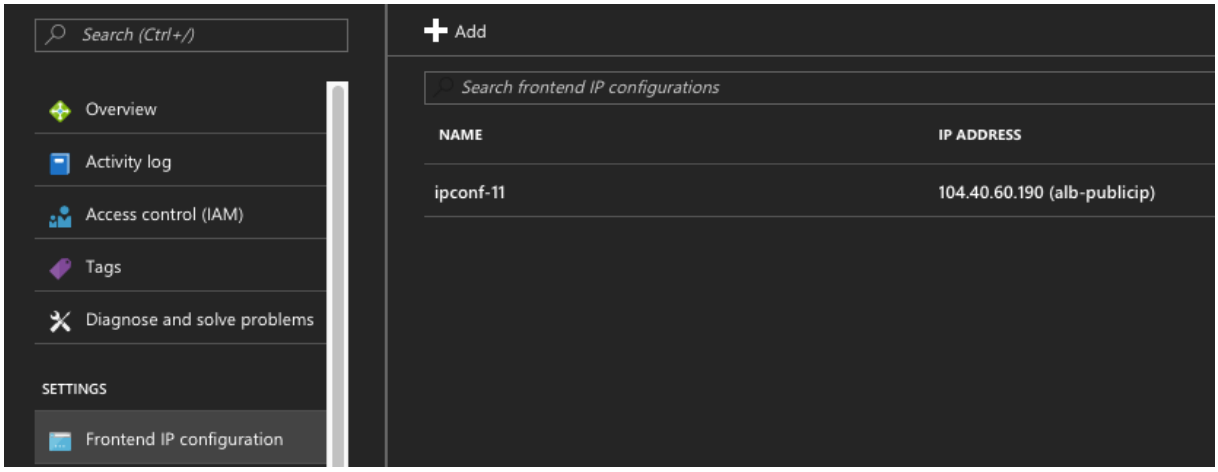
7. [概要] ページが開きます。構成を確認し、適宜編集します。[**OK**] を選択して確定します。
8. 「購入」ページが表示されます。[購入] を選択してデプロイを完了します。

必要な構成で Azure リソースグループが作成されるまで時間がかかることがあります。完了後、Azure ポータルでリソースグループを選択し、LB ルール、バックエンドプール、ヘルスプローブなどの構成の詳細を表示します。高可用性ペアは、ns-vpx0 および ns-vpx1 と表示されます。

追加のセキュリティルールやポートを作成するなど、HA セットアップでさらに変更が必要な場合は、Azure Portal から実行できます。



次に、プライマリノードで **ALB** のフロントエンドパブリック **IP**（**PIP**）アドレスを使用して負荷分散仮想サーバーを構成する必要があります。ALB PIP を検索するには、ALB > フロントエンド IP 設定を選択します。



負荷分散仮想サーバーの構成方法の詳細については、「リソース」セクションを参照してください。

リソース:

次のリンクには、HA の導入と仮想サーバの設定に関する追加情報が表示されます。

- 異なるサブネットでの高可用性ノードの構成
- 基本的な負荷分散を設定する

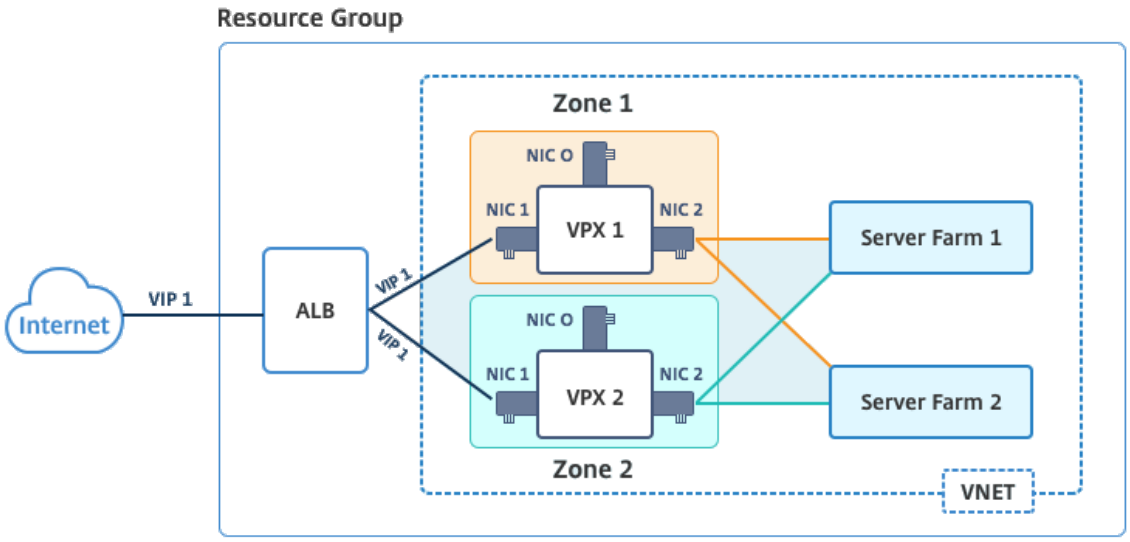
関連リソース:

- PowerShell コマンドを使用して複数の IP アドレスと NIC を使用して高可用性セットアップを構成する
- Azure でのアクティブスタンバイ HA デプロイメントでの GSLB の構成

アベイラビリティーゾーンを使用した高可用性

Azure アベイラビリティーゾーンは、Azure リージョン内の障害分離された場所であり、冗長な電源、冷却、ネットワーキングを提供し、回復力を高めます。特定の Azure リージョンだけがアベイラビリティーゾーンをサポートします。アベイラビリティーゾーンをサポートするリージョンの詳細については、Azure のドキュメントを参照してください。Azure のアベイラビリティーゾーンは何ですか。.

図:Azure アベイラビリティーゾーンを使用した高可用性デプロイアーキテクチャの例



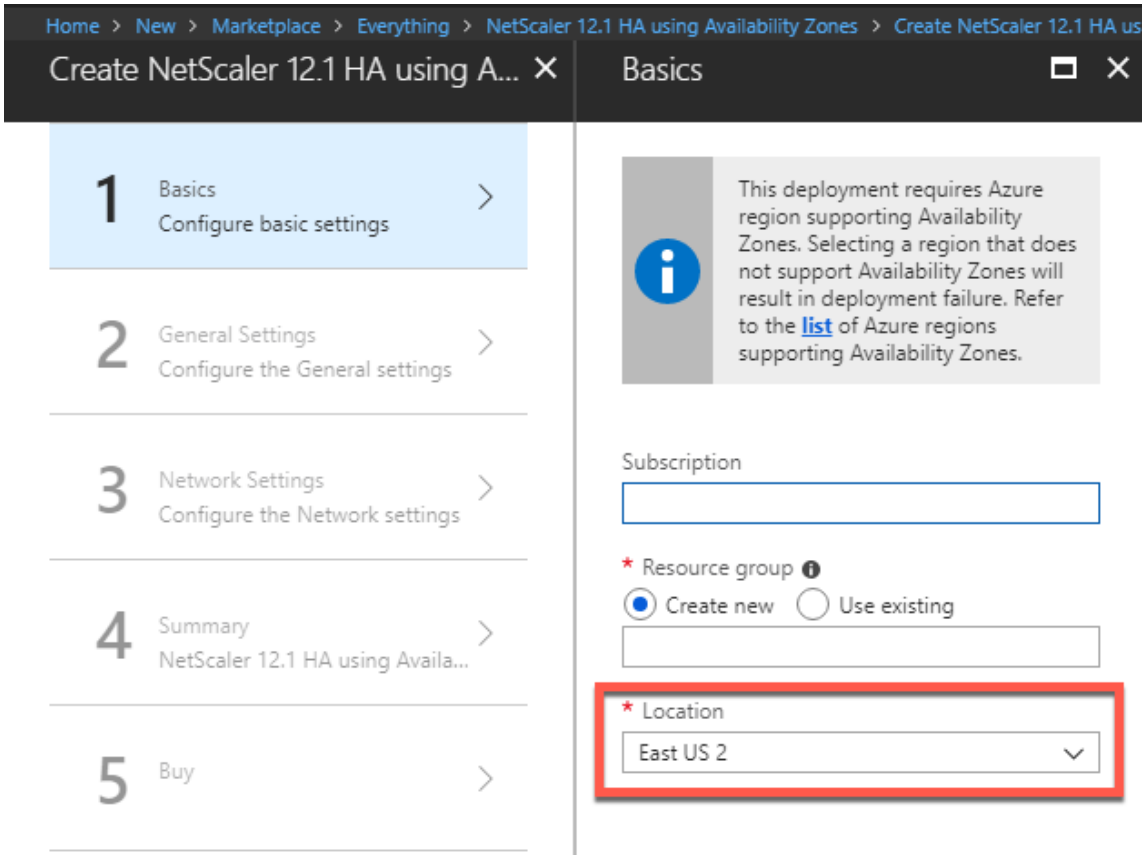
Azure Marketplace e で入手可能な「アベイラビリティーゾーンを使用した NetScaler 13.0 HA」というテンプレートを使用して、VPX ペアを高可用性モードでデプロイできます。

Azure アベイラビリティーゾーンを使用してテンプレートを起動し、高可用性 VPX ペアをデプロイするには、次の手順を実行します。

1. Azure Marketplace から、Citrix ソリューションテンプレートを選択して開始します。

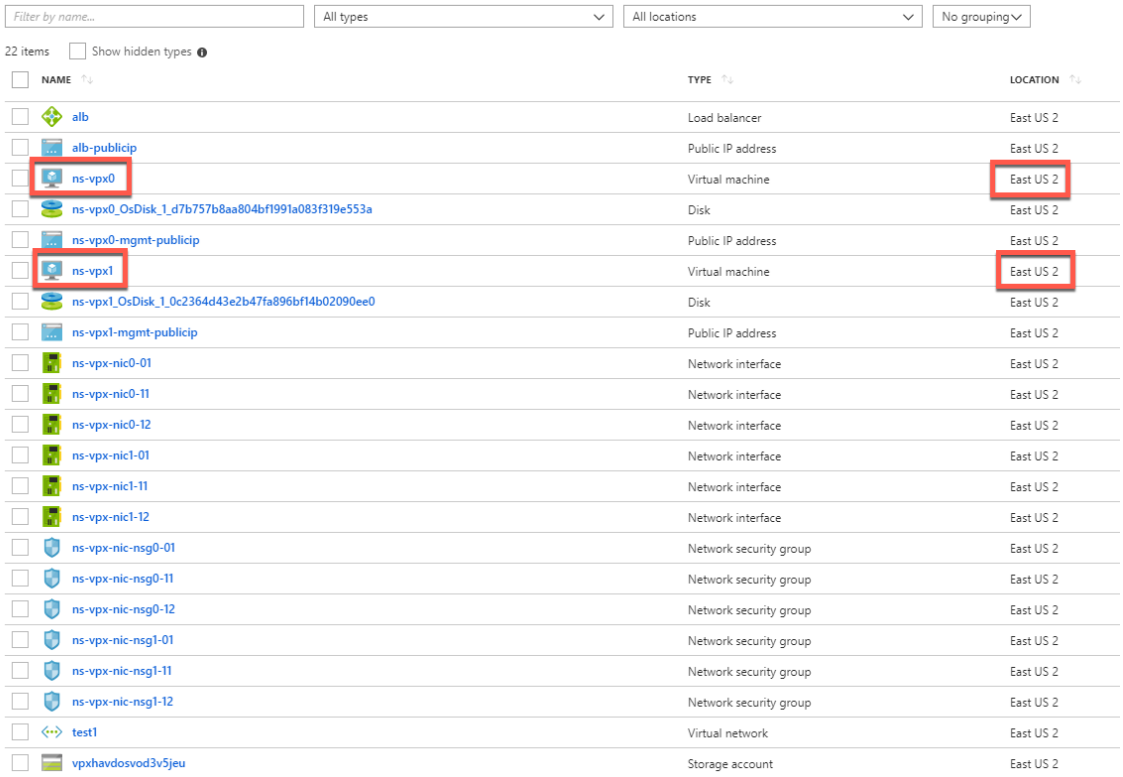
2. デプロイメント・タイプがリソース・マネージャーであることを確認し、「作成」を選択します。
3. [基本] ページが表示されます。詳細を入力し、[**OK**] をクリックします。

> 注
>
> 可用性ゾーンをサポートする Azure リージョンを選択してください。アベイラビリティーゾーンをサポートするリージョンの詳細については、Azure のドキュメントを参照してください。Azure のアベイラビリティーゾーンは何ですか。



4. [一般設定] ページが表示されます。詳細を入力して「**OK**」を選択します。
5. [ネットワーク設定] ページが表示されます。VNet とサブネットの構成を確認し、必要な設定を編集して、[**OK]** を選択します。
6. [概要] ページが開きます。構成を確認し、適宜編集します。[**OK**] を選択して確定します。
7. 「購入」ページが表示されます。[購入] を選択してデプロイを完了します。

   必要な構成で Azure リソースグループが作成されるまで時間がかかることがあります。完了後、リソースグループを選択すると、LB ルール、バックエンドプール、ヘルスプローブなどの構成の詳細が Azure ポータルに表示されます。高可用性ペアは、ns-vpx0 および ns-vpx1 と表示されます。また、「場所」列にも場所が表示されます。



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
vpxhavdosvod3v5jeu	Storage account	East US 2

追加のセキュリティルールやポートを作成するなど、HA セットアップでさらに変更が必要な場合は、Azure Portal から実行できます。

Azure モニターのメトリックを使用してインスタンスを監視する

Azure モニターデータプラットフォームのメトリックを使用して、CPU、メモリ使用率、スループットなどの一連の NetScaler VPX リソースを監視できます。メトリックサービスは、Azure 上で稼働する NetScaler VPX リソースをリアルタイムで監視します。メトリックスエクスプローラーを使用して、収集されたデータにアクセスできます。詳細については、「Azure Monitor メトリックスの概要」を参照してください。

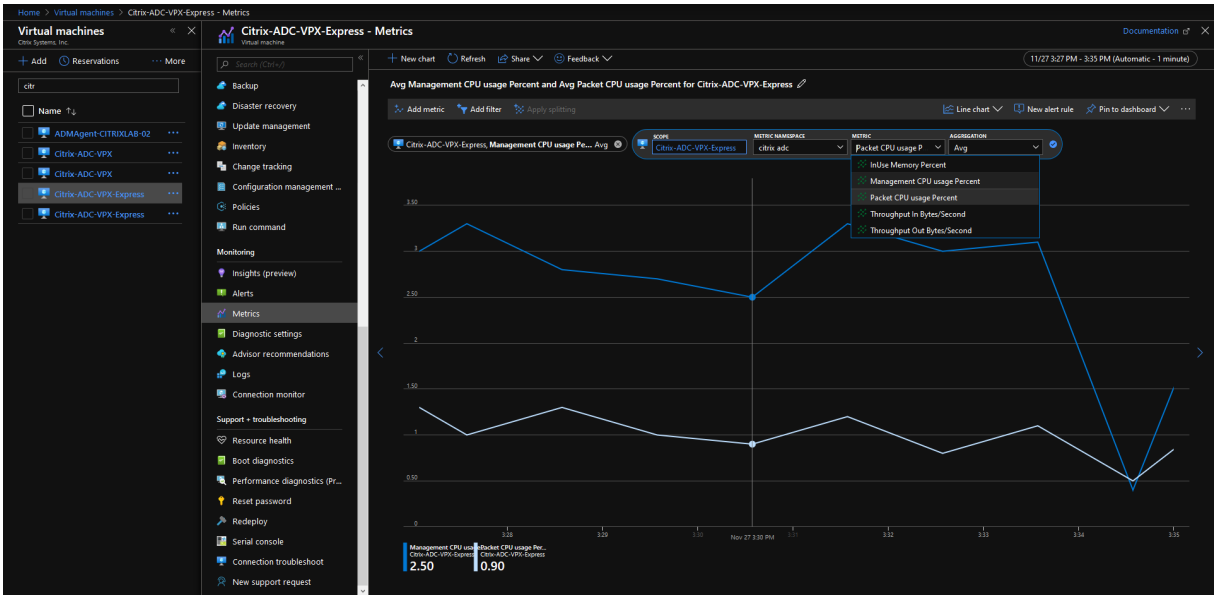
注意事項

- Azure Marketplace オファーを使用して NetScaler VPX インスタンスを Azure にデプロイすると、メトリックスサービスはデフォルトで無効になります。
- メトリックスサービスは Azure CLI ではサポートされていません。
- メトリクスは、CPU (管理およびパケット CPU 使用率)、メモリ、およびスループット (インバウンドとアウトバウンド) で使用できます。

Azure モニターでメトリックを表示する方法

インスタンスの Azure モニターでメトリックスを表示するには、次の手順を実行します。

1. **Azure Portal > Virtual Machines** にログオンします。
2. プライマリノードとなる仮想マシンを選択します。
3. モニタリングセクションで、メトリクスをクリックします。
4. メトリック名前空間のドロップダウンメニューから、**NetScaler** をクリックします。
5.「指標」ドロップダウンメニューの「すべての指標 **」で、表示したい指標をクリックします。
6. [指標を追加] をクリックすると、同じグラフに別の指標が表示されます。チャートオプションを使用してチャートをカスタマイズします。



PowerShell コマンドを使用して複数の **IP** アドレスと **NIC** を使用して高可用性セットアップを構成する

October 17, 2024

Azure のアクティブ/パッシブ高可用性（HA）セットアップで、複数の NIC を持つ一対の NetScaler VPX インスタンスを展開できます。各 NIC に複数の IP アドレスを設定できます。

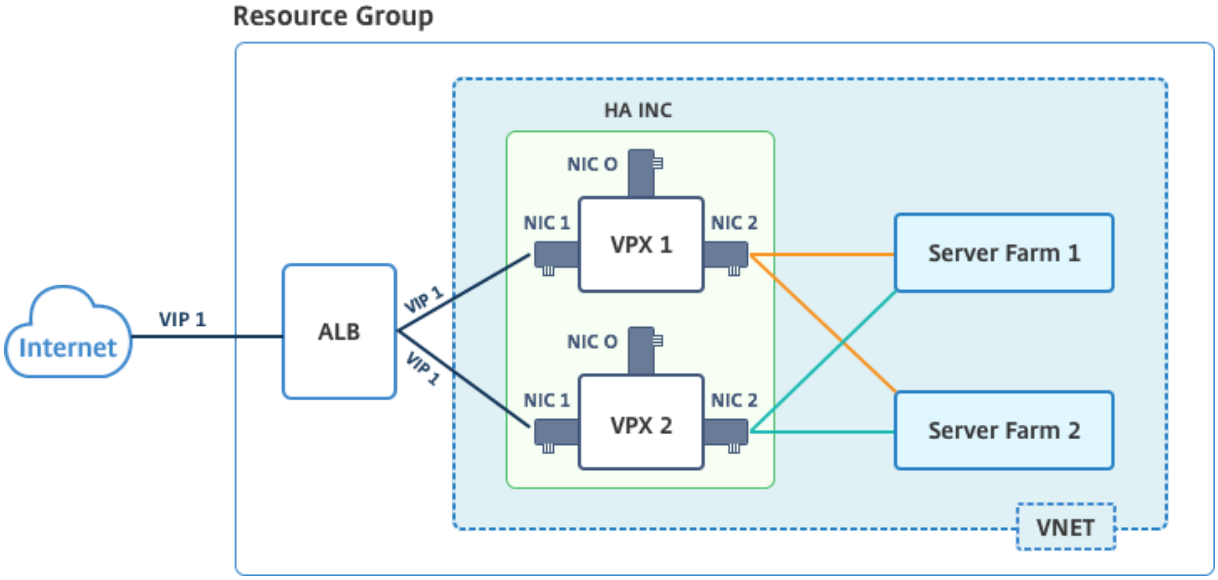
アクティブ/パッシブ展開には以下が必要です。

- HA Independent Network Configuration（INC）構成
- Direct Server Return（DSR）モードの Azure Load Balancer（ALB）

すべてのトラフィックはプライマリノードを通過します。セカンダリノードは、プライマリノードが失敗するまでスタンバイモードを維持します。

> 注
>
> Azure クラウド上の Citrix ADC VPX 高可用性展開を機能させるには、2 つの高可用性ノード間で移動できるフローティングパブリック IP（PIP）が必要です。Azure ロードバランサー (ALB) は、フローティング PIP を提供します。このフローティング PIP は、フェールオーバーが発生した場合に自動的に第 2 ノードに移動されます。

図: アクティブ-パッシブ展開アーキテクチャの例



アクティブ/パッシブ展開では、ALB フローティングパブリック IP（PIP）アドレスが各 VPX ノードに VIP アドレスとして追加されます。HA-INC 構成では、これらの VIP アドレスはフローティングされ、SNIP アドレスはインスタンス固有のアドレスとなります。

ALB は 5 秒ごとにヘルスプローブを送信して各 VPX インスタンスを監視し、定期的にヘルスプローブ応答を送信するトラフィックのみをそのインスタンスにリダイレクトします。そのため、HA セットアップでは、プライマリノードがヘルスプローブに応答し、セカンダリノードは応答しません。プライマリインスタンスが 2 つの連続したヘルスプローブを見逃した場合、ALB はそのインスタンスにトラフィックをリダイレクトしません。フェイルオーバー時は、新しいプライマリがヘルスプローブへの応答を開始し、ALB はそのプライマリにトラフィックをリダイレクトします。標準の VPX 高可用性フェイルオーバー時間は 3 秒です。トラフィック切り替えにかかる合計フェイルオーバー時間は、最大で 13 秒になる可能性があります。

VPX ペアをアクティブ-パッシブ HA セットアップで展開するには、次の 2 つの方法があります。

- **NetScaler VPX** 標準高可用性テンプレート: このオプションを使用して、3 つのサブネットと 6 つの NIC のデフォルトオプションで HA ペアを構成します。
- **Windows PowerShell** コマンド: このオプションを使用して、サブネットと NIC の要件に応じて HA ペアを構成します。

このトピックでは、PowerShell コマンドを使用してアクティブ/パッシブ HA セットアップで VPX ペアを展開する方法について説明します。NetScaler VPX Standard HA テンプレートを使用する場合は、「複数の IP アドレスと NIC を使用した HA セットアップの構成」を参照してください。

PowerShell コマンドを使用して HA-INC ノードを構成する

シナリオ:HA-INC PowerShell の展開

このシナリオでは、表に示されているトポロジを使用して NetScaler VPX ペアをデプロイします。各 VPX インスタンスには 3 つの NIC があり、各 NIC は異なるサブネットに展開されます。各 NIC には IP 構成 が割り当てられます。

ALB	VPX1	VPX2
ALB はパブリック IP 3 (pip3) に関連付けられています。	管理 IP は IPConfig1 を使用して構成され、これには 1 つのパブリック IP (パイプ 1) と 1 つのプライベート IP (12.5.2.24) が含まれます。	管理 IP は IPConfig5 を使用して構成され、これには 1 つのパブリック IP (パイプ 3) と 1 つのプライベート IP (12.5.2.26) が含まれます。
設定された LB ルールおよびポートは、HTTP（80）、SSL（443）、ヘルスプローブ（9000）です。	クライアント側の IP は IPConfig3 を使用して構成され、これには 1 つのプライベート IP (12.5.1.27) が含まれます。	クライアント側の IP は IPConfig7 で構成され、これには 1 つのプライベート IP (12.5.1.28) が含まれます。
-	サーバー側の IP は IPConfig4 を使用して構成され、これには 1 つのプライベート IP (12.5.3.24)、nic3、バックエンドサブネット = 12.5.3.0/24 が含まれます。	サーバー側の IP は IPConfig8 を使用して構成され、これには 1 つのプライベート IP (12.5.3.28)、nic6、バックエンドサブネット = 12.5.3.0/24 が含まれます。
-	NSG のルールとポートは、SSH (22)、HTTP (80)、HTTPS (443)	-

パラメータ設定

このシナリオでは、次のパラメータ設定が使用されます。

$locName = “East Asia”

$rgName = “MulitIP-MultiNIC-RG”

$nicName1= “VM1-NIC1”

$nicName2 = “VM1-NIC2”

$nicName3= “VM1-NIC3”

$nicName4 = “VM2-NIC1”

$nicName5= “VM2-NIC2”

$nicName6 = “VM2-NIC3”

$vNetName = “Azure-MultiIP-ALB-vnet”

$vNetAddressRange= “12.5.0.0/16”

$frontEndSubnetName= “frontEndSubnet”

$frontEndSubnetRange= “12.5.1.0/24”

$mgmtSubnetName= “mgmtSubnet”

$mgmtSubnetRange= “12.5.2.0/24”

$backEndSubnetName = “backEndSubnet”

$backEndSubnetRange = “12.5.3.0/24”

$prmStorageAccountName = “multiipmultinicbstorage”

$avSetName = “multiple-avSet”

$vmSize= “Standard_DS4_V2”

$publisher = “Citrix”

$offer = “netscalervpx-120”

$sku = “netscalerbyol”

$version=” latest”

$pubIPName1=” VPX1MGMT”

$pubIPName2=” VPX2MGMT”

$pubIPName3=” ALBPIP”

$domName1=” vpx1dns”

$domName2=” vpx2dns”

$domName3=” vpxalbdns”

$vmNamePrefix=” VPXMultiIPALB”

$osDiskSuffix1=” osmultiipalbdiskdb1”

$osDiskSuffix2=” osmultiipalbdiskdb2”

$lbName= “MultiIPALB”

$frontEndConfigName1= “FrontEndIP”

$backendPoolName1= “BackendPoolHttp”

$lbRuleName1= “LBRuleHttp”

$healthProbeName= “HealthProbe”

$nsgName=” NSG-MultiIP-ALB”

$rule1Name=” Inbound-HTTP”

$rule2Name=” Inbound-HTTPS”

$rule3Name=” Inbound-SSH”

展開を完了するには、PowerShell コマンドを使用して次の手順を完了します。

1. リソースグループ、ストレージアカウント、高可用性セットの作成
2. ネットワークセキュリティグループの作成と規則の追加
3. 仮想ネットワークと 3 つのサブネットの作成
4. パブリック IP アドレスの作成
5. VPX1 の IP 構成の作成
6. VPX2 の IP 構成の作成
7. VPX1 の NIC の作成
8. VPX2 の NIC の作成
9. VPX1 の作成
10. VPX2 の作成
11. ALB の作成

リソースグループ、ストレージアカウント、および可用性セットを作成します。

```
New-AzureRmResourceGroup -Name $rgName -Location $locName


$prmStorageAccount=New-AzureRMStorageAccount -Name
    $prmStorageAccountName -ResourceGroupName $rgName -Type
    Standard_LRS -Location $locName


$avSet=New-AzureRMAvailabilitySet -Name $avSetName -ResourceGroupName
     $rgName -Location $locName
```

ネットワークセキュリティグループを作成し、ルールを追加します。

```
$rule1 = New-AzureRmNetworkSecurityRuleConfig -Name $rule1Name -
    Description "Allow HTTP" -Access Allow -Protocol Tcp -Direction
    Inbound -Priority 101


-SourceAddressPrefix Internet -SourcePortRange * -
    DestinationAddressPrefix * -DestinationPortRange 80

```



```
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
```

仮想ネットワークと **3** つのサブネットを作成します。

```
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 }
```

パブリック **IP** アドレスを作成します。

```
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
```

VPX1 の **IP** 構成を作成します。

```
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
```

VPX2 の **IP** 構成を作成します。

```
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
```

VPX1 用の **NIC** を作成します。

```
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
```


VPX2 用の **NIC** を作成します。

```
$nic4=New-AzureRmNetworkInterface -Name $nicName4 -ResourceGroupName
    $rgName -Location $locName -IpConfiguration $IpConfig5 -
    NetworkSecurityGroupId $nsg.Id


$nic5=New-AzureRmNetworkInterface -Name $nicName5 -ResourceGroupName
    $rgName -Location $locName -IpConfiguration $IpConfig7 -
    NetworkSecurityGroupId $nsg.Id


$nic6=New-AzureRmNetworkInterface -Name $nicName6 -ResourceGroupName
    $rgName -Location $locName -IpConfiguration $IpConfig8 -
    NetworkSecurityGroupId $nsg.Id
```

VPX1 を作成します。

この手順には、次の下位手順が含まれています。

- VM 設定オブジェクトの作成
- 資格情報、OS、イメージの設定
- NIC の追加
- OS ディスクの指定と VM の作成

```
$suffixNumber = 1

$vmName=$vmNamePrefix + $suffixNumber

$vmConfig=New-AzureRMVMConfig -VMName $vmName -VMSize $vmSize -
    AvailabilitySetId $avSet.Id

$cred=Get-Credential -Message "Type the name and password for
    VPX login."

$vmConfig=Set-AzureRMVMOperatingSystem -VM $vmConfig -Linux -
    ComputerName $vmName -Credential $cred

$vmConfig=Set-AzureRMVMSourceImage -VM $vmConfig -PublisherName
     $publisher -Offer $offer -Skus $sku -Version $version

$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic1
    .Id -Primary

$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic2
    .Id

$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic3
    .Id

$osDiskName=$vmName + "-" + $osDiskSuffix1
```



```
20
21     $osVhdUri=$prmStorageAccount.PrimaryEndpoints.Blob.ToString() +
           "vhds/" + $osDiskName + ".vhd"
22
23     $vmConfig=Set-AzureRMVMOSDisk -VM $vmConfig -Name $osDiskName -
           VhdUri $osVhdUri -CreateOption fromImage
24
25     Set-AzureRmVMPlan -VM $vmConfig -Publisher $publisher -Product
           $offer -Name $sku
26
27     New-AzureRMVM -VM $vmConfig -ResourceGroupName $rgName -
           Location $locName
```

VPX2 を作成します。

```
1  ```
2    $suffixNumber=2
3
4
5    $vmName=$vmNamePrefix + $suffixNumber
6
7
8    $vmConfig=New-AzureRMVMConfig -VMName $vmName -VMSize $vmSize -
         AvailabilitySetId $avSet.Id
9
10
11   $cred=Get-Credential -Message "Type the name and password for VPX
         login."
12
13
14   $vmConfig=Set-AzureRMVMOperatingSystem -VM $vmConfig -Linux -
         ComputerName $vmName -Credential $cred
15
16
17   $vmConfig=Set-AzureRMVMSourceImage -VM $vmConfig -PublisherName
         $publisher -Offer $offer -Skus $sku -Version $version
18
19
20   $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic4.Id -
         Primary
21
22
23   $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic5.Id
24
25
26   $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic6.Id
27
28
29   $osDiskName=$vmName + "-" + $osDiskSuffix2
30
31
32   $osVhdUri=$prmStorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds
         /" + $osDiskName + ".vhd"
```



```
33
34
35    $vmConfig=Set-AzureRMVMOSDisk -VM $vmConfig -Name $osDiskName -VhdUri
          $osVhdUri -CreateOption fromImage
36
37
38    Set-AzureRmVMPlan -VM $vmConfig -Publisher $publisher -Product $offer
          -Name $sku
39
40
41    New-AzureRMVM -VM $vmConfig -ResourceGroupName $rgName -Location
          $locName
42 ```
```

NIC に割り当てられたプライベート IP アドレスとパブリック IP アドレスを表示するには、次のコマンドを入力します。

```
1  ```
2    $nic1.IPConfig
3
4
5    $nic2.IPConfig
6
7
8    $nic3.IPConfig
9
10
11   $nic4.IPConfig
12
13
14   $nic5.IPConfig
15
16
17   $nic6.IPConfig
18 ```
```

Azure の負荷分散 **(ALB)** を作成します。

この手順には、次の下位手順が含まれています。

- フロントエンド IP 構成を作成する
- ヘルスプローブの作成
- バックエンドアドレスプールを作成する
- 負荷分散規則（HTTP および SSL）の作成
- フロントエンド IP 設定、バックエンドアドレスプール、および LB ルールを使用して ALB を作成します。
- IP 構成をバックエンドプールに関連付ける

  ```
  $frontEndIP1=New-AzureRmLoadBalancerFrontendIpConfig -Name
  $frontEndConfigName1 -PublicIpAddress $pip3
  ```



```
$healthProbe=New-AzureRmLoadBalancerProbeConfig -Name $healthProbeName
 -Protocol Tcp -Port 9000 -IntervalInSeconds 5 -ProbeCount 2

$beAddressPool1=New-AzureRmLoadBalancerBackendAddressPoolConfig -
Name $backendPoolName1

$lbRule1=New-AzureRmLoadBalancerRuleConfig -Name $lbRuleName1
 -FrontendIpConfiguration $frontEndIP1 -BackendAddressPool
$beAddressPool1 -Probe $healthProbe -Protocol Tcp -FrontendPort
80 -BackendPort 80 -EnableFloatingIP

$lb=New-AzureRmLoadBalancer -ResourceGroupName $rgName -Name
$lbName -Location $locName -FrontendIpConfiguration $frontEndIP1
-LoadBalancingRule $lbRule1 -BackendAddressPool $beAddressPool1 -
Probe $healthProbe

$nic2.IpConfigurations[0].LoadBalancerBackendAddressPools.Add($lb
.BackendAddressPools[0])

$nic5.IpConfigurations[0].LoadBalancerBackendAddressPools.Add($lb
.BackendAddressPools[0])

$lb=$lb |Set-AzureRmLoadBalancer

$nic2=$nic2 | Set-AzureRmNetworkInterface

$nic5=$nic5 | Set-AzureRmNetworkInterface
```

NetScaler VPX ペアを正常に展開したら、各 VPX インスタンスにログオンして HA-INC、SNIP アドレス、および VIP アドレスを構成します。

1. 次のコマンドを入力して HA ノードを追加します。

   ```
   add ha node 1 PeerNodeNSIP -inc Enabled
   ```

2. クライアント側 NIC のプライベート IP アドレスを VPX1（NIC2）および VPX2（NIC5）の SNIP として追加する

   ```
   nsip privateIPofNIC2 255.255.255.0 -type SNIP を追加します       nsip
   privateIPofNIC5 255.255.255.0 -type SNIP を追加します
   ```

3. ALB のフロントエンド IP アドレス (パブリック IP) を持つプライマリノードに負荷分散仮想サーバーを追加します。

   ```
   add lb virtual server v1 HTTP FrontEndIPofALB 80
   ```

関連リソース:

Azure でのアクティブスタンバイ HA デプロイメントでの GSLB の構成

フローティング **IP** 無効モードの **ALB** を使用して **Azure** に **NetScaler** 高可用性ペアをデプロイする

October 17, 2024

Azure のアクティブ/パッシブ高可用性（HA）セットアップで、複数の NIC を持つ一対の NetScaler VPX インスタンスを展開できます。各 NIC には多数の IP アドレスを含めることができます。

アクティブ/パッシブ展開には以下が必要です。

- HA Independent Network Configuration（INC）構成
- Azure Load Balancer (ALB) には次の機能があります。
  - フローティング IP 対応モードまたはダイレクトサーバーリターン (DSR) モード
  - フローティング IP 無効モード

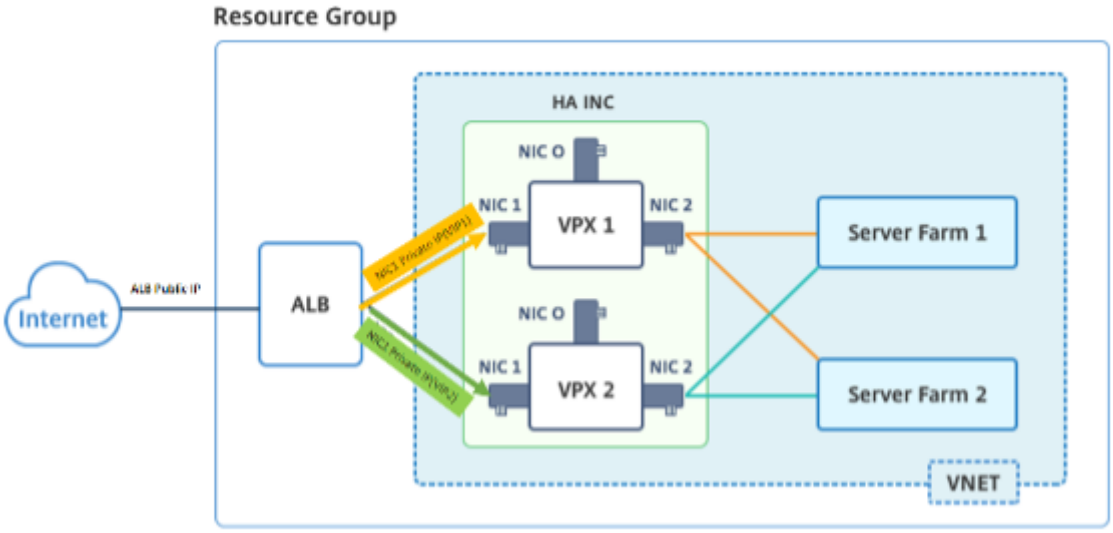
ALB Floating IP オプションの詳細については、Azure のドキュメントを参照してください。

ALB フローティング IP を有効にして、Azure でアクティブ/パッシブ HA セットアップで VPX ペアをデプロイする場合は、「PowerShell コマンドを使用して複数の IP アドレスと NIC で高可用性セットアップを構成する」を参照してください。

フローティング **IP** 無効モードの **ALB** を使用した **HA** 導入アーキテクチャ

アクティブ/パッシブ展開では、各インスタンスのクライアントインターフェイスのプライベート IP アドレスが各 VPX インスタンスの VIP アドレスとして追加されます。HA-INC モードで、IP セットを使用して VIP アドレスを共有し、SNIP アドレスをインスタンス固有に設定します。すべてのトラフィックはプライマリインスタンスを通過します。セカンダリインスタンスは、プライマリインスタンスに障害が発生するまでスタンバイモードです。

図: アクティブ-パッシブ展開アーキテクチャの例



前提条件

NetScaler VPX インスタンスを Azure に展開する前に、次の情報を理解している必要があります。

- Azure の用語とネットワークの詳細。詳細については、「Azure 用語」を参照してください。
- NetScaler アプライアンスの動作。詳しくは、NetScaler のドキュメントを参照してください。
- NetScaler ネットワーキング。詳細については、ADC ネットワークを参照してください。
- Azure ロードバランサーと負荷分散ルール設定。詳細については、Azure ALB のドキュメントを参照してください。

ALB フローティング **IP** を無効にして **VPX HA** ペアを **Azure** にデプロイする方法

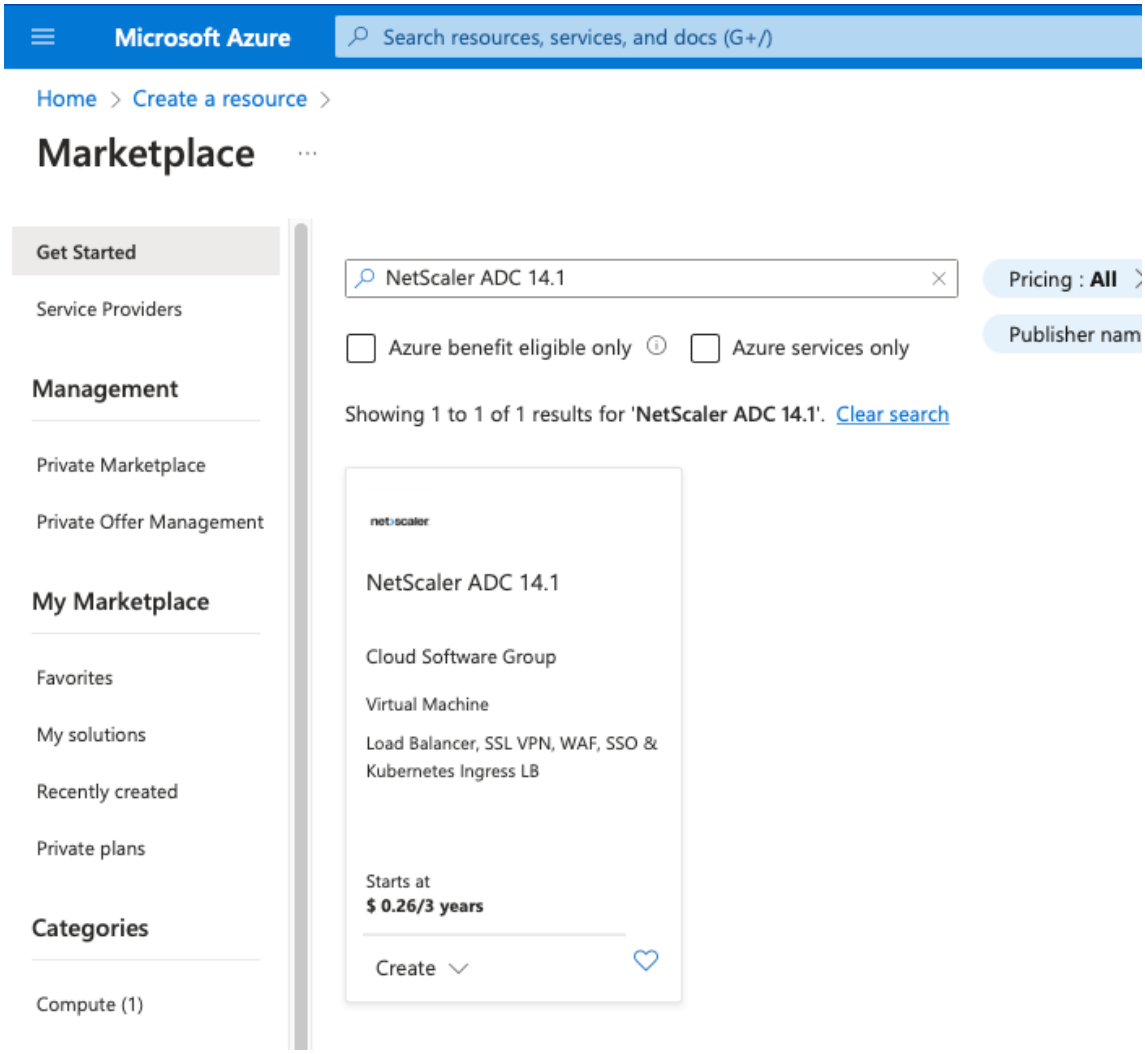
HA と ALB の導入手順の概要は次のとおりです。

1. Azure に 2 つの VPX インスタンス (プライマリインスタンスとセカンダリインスタンス) をデプロイします。
2. 両方のインスタンスにクライアントとサーバーの NIC を追加します。
3. フローティング IP モードが無効になっている負荷分散ルールを持つ ALB をデプロイします。
4. NetScaler GUI を使用して、両方のインスタンスで高可用性設定を構成します。

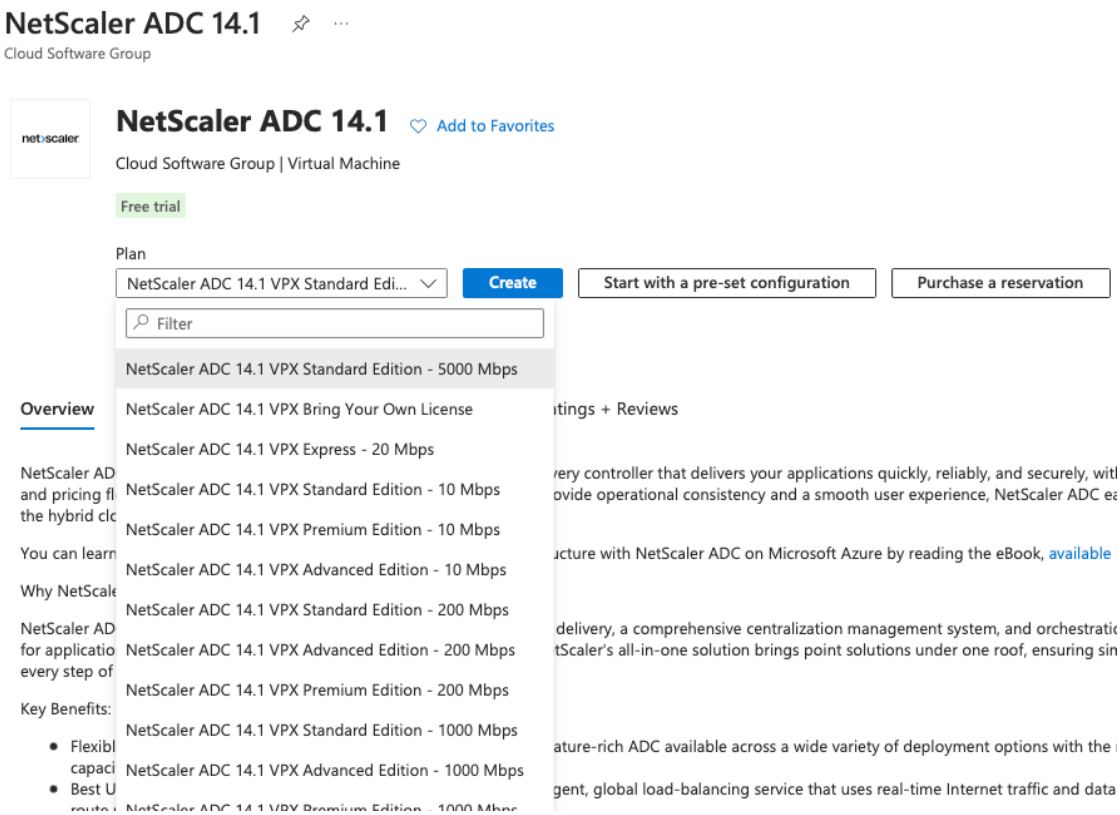
手順 **1. Azure** に **2** つの **VPX** インスタンスをデプロイします。

次の手順に従って、2 つの VPX インスタンスを作成します。

1. Azure Marketplace から NetScaler バージョンを選択します（この例では、NetScaler リリース 13.1 が使用されています)。

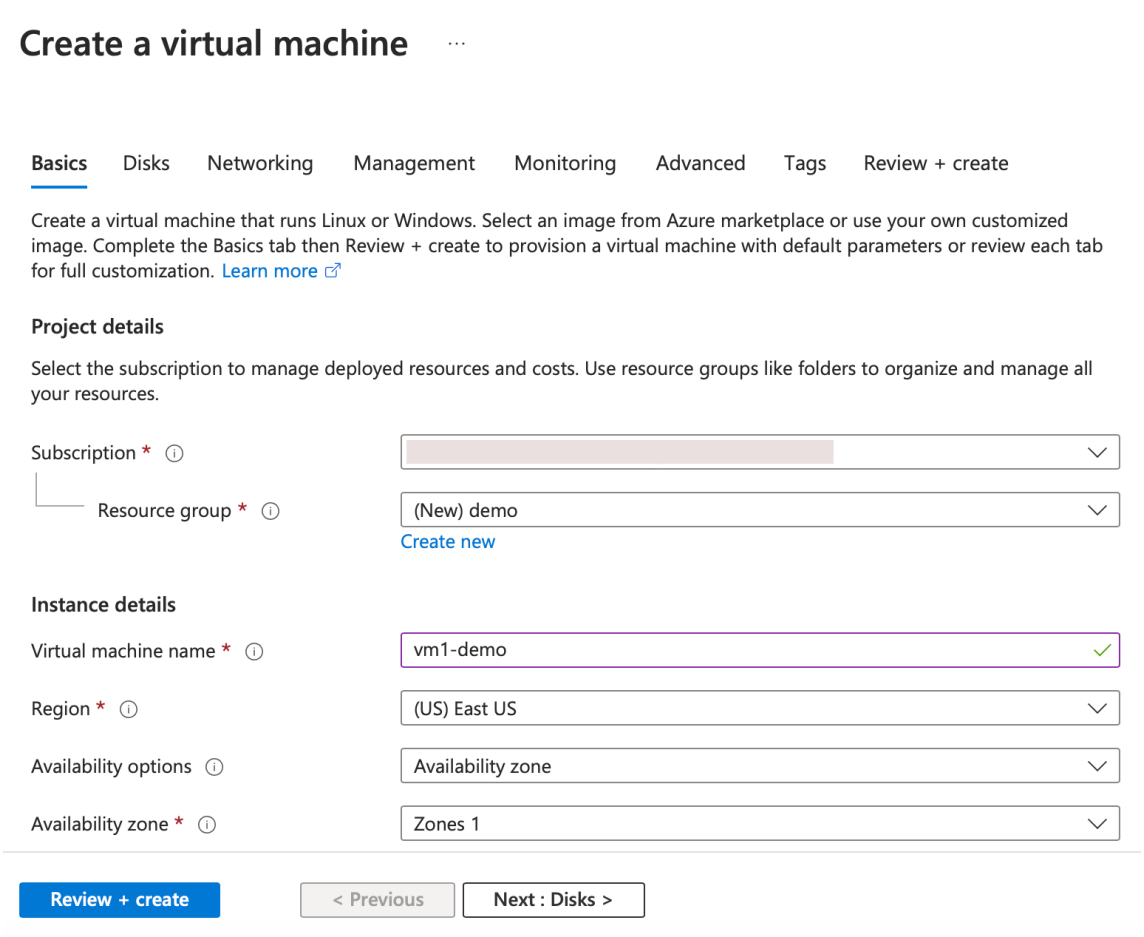


2. 必要な ADC ライセンスモードを選択し、[作成] をクリックします。



[仮想マシンの作成] ページが開きます。

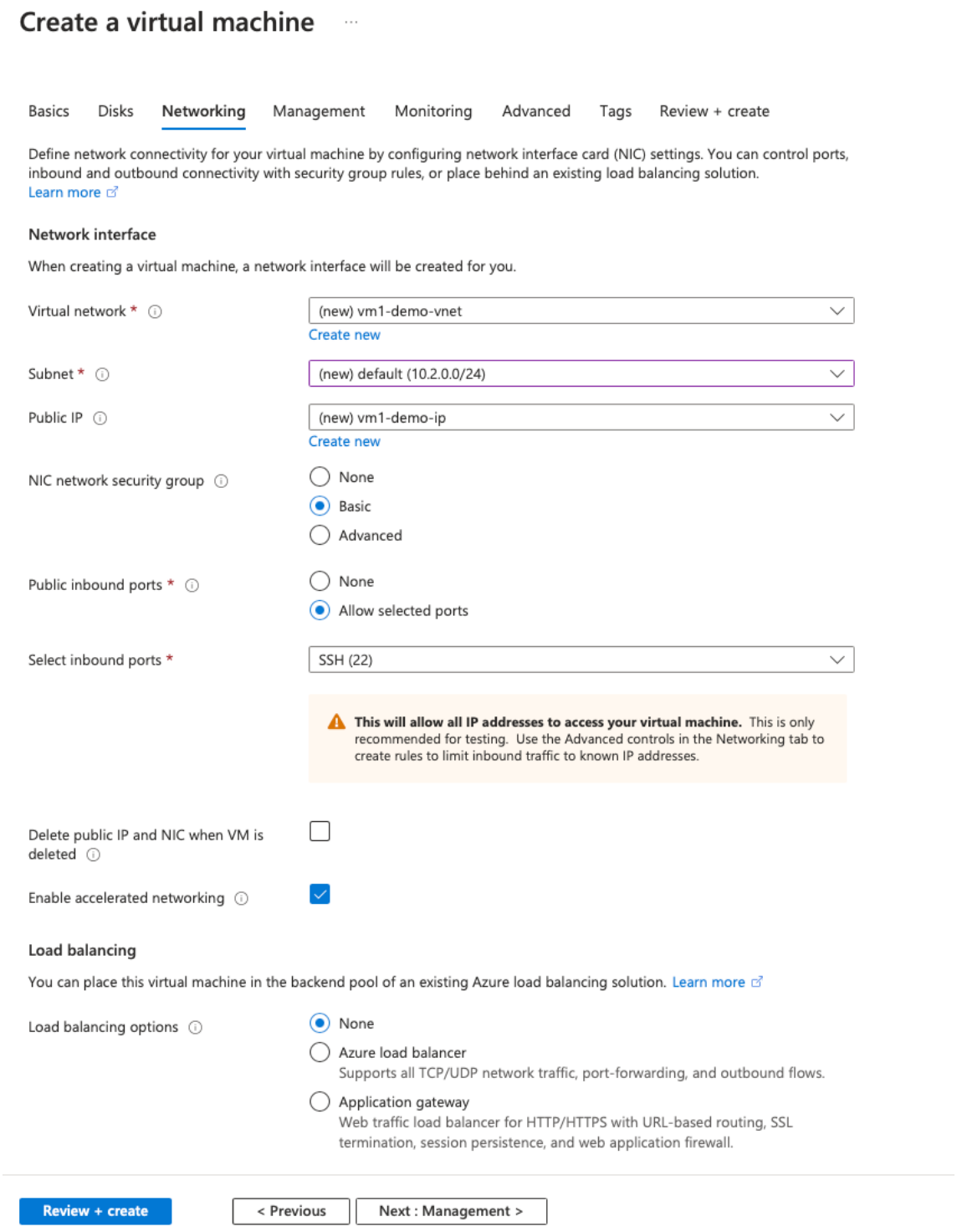
3. 展開を成功させるには、各タブ (基本、ディスク、ネットワーク、管理、監視、詳細、タグ) で必要な詳細を入力します。



[ネットワーク] タブで、管理、クライアント、サーバーの NIC の 3 つのサブネットを持つ新しい仮想ネットワークを作成します。それ以外の場合は、既存の仮想ネットワークを使用することもできます。管理 NIC は、VM の展開中に作成されます。クライアントとサーバーの NIC は、仮想マシンの作成後に作成および接続されます。NIC ネットワークセキュリティグループでは、次のいずれかを実行できます。

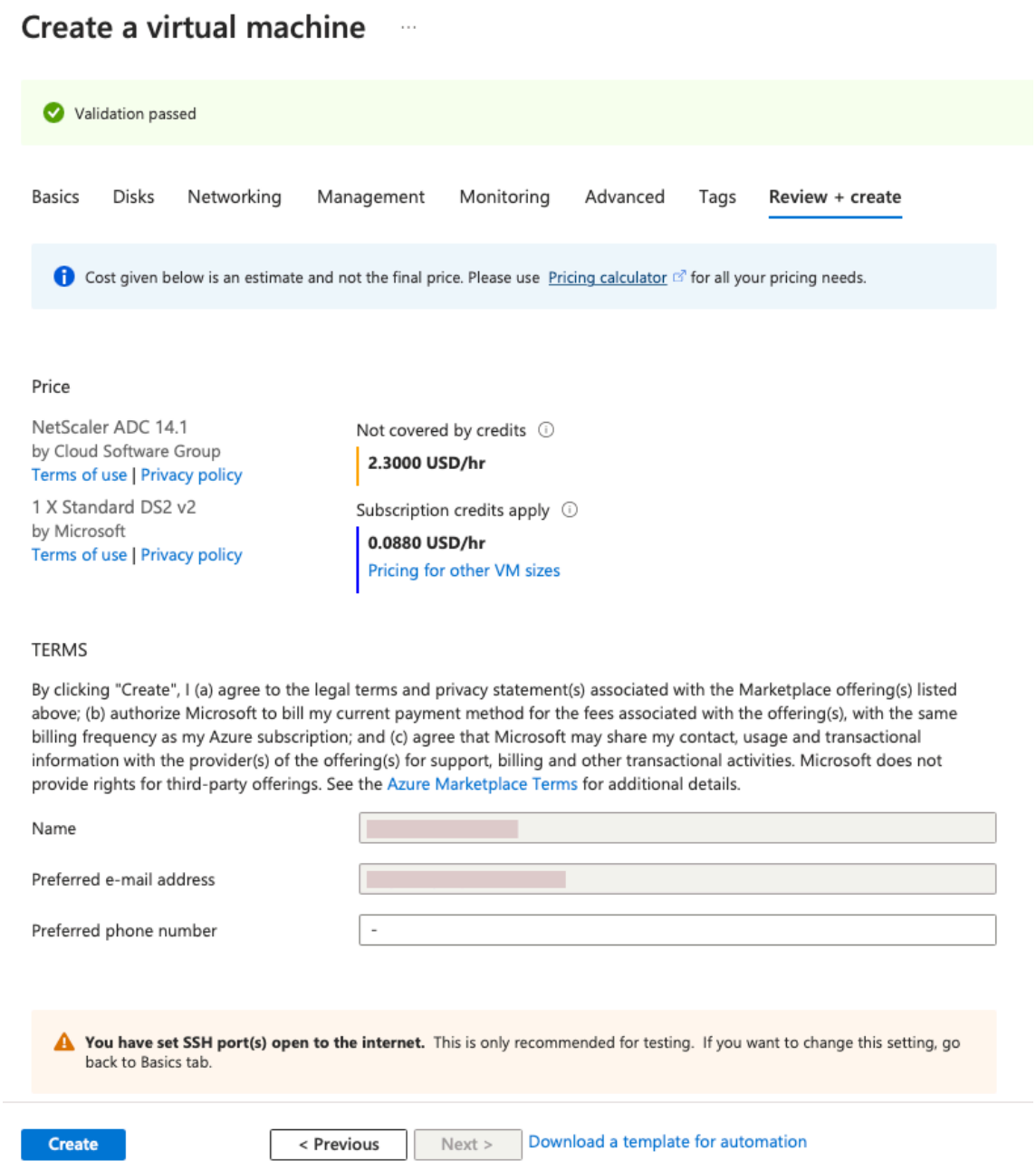
- [詳細] を選択し、要件に合った既存のネットワークセキュリティグループを使用します。
- [基本] を選択し、必要なポートを選択します。

> 注
>
> 仮想マシンのデプロイが完了した後に、ネットワークセキュリティグループの設定を変更することもできます。

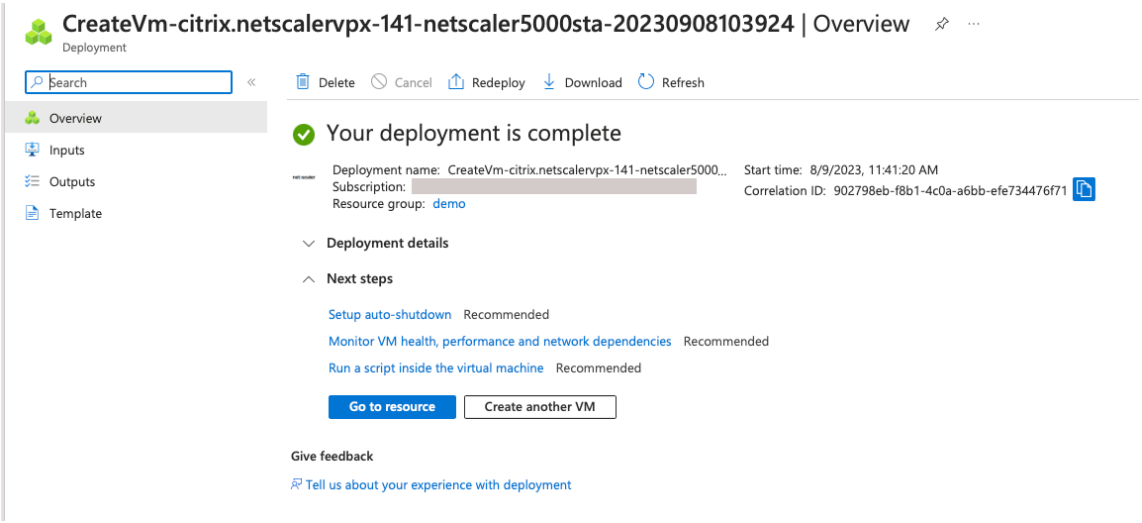


4. 次へをクリックします: 確認 **+** 作成 **>**。

   検証が成功したら、基本設定、仮想マシンの構成、ネットワーク、および追加設定を確認し、[作成] をクリックします。



5. デプロイが完了したら、「リソースに移動」をクリックして設定の詳細を確認します。



同様に、2 つ目の NetScaler VPX インスタンスを展開します。

手順 **3.** クライアントとサーバーの **NIC** を両方のインスタンスに追加します。

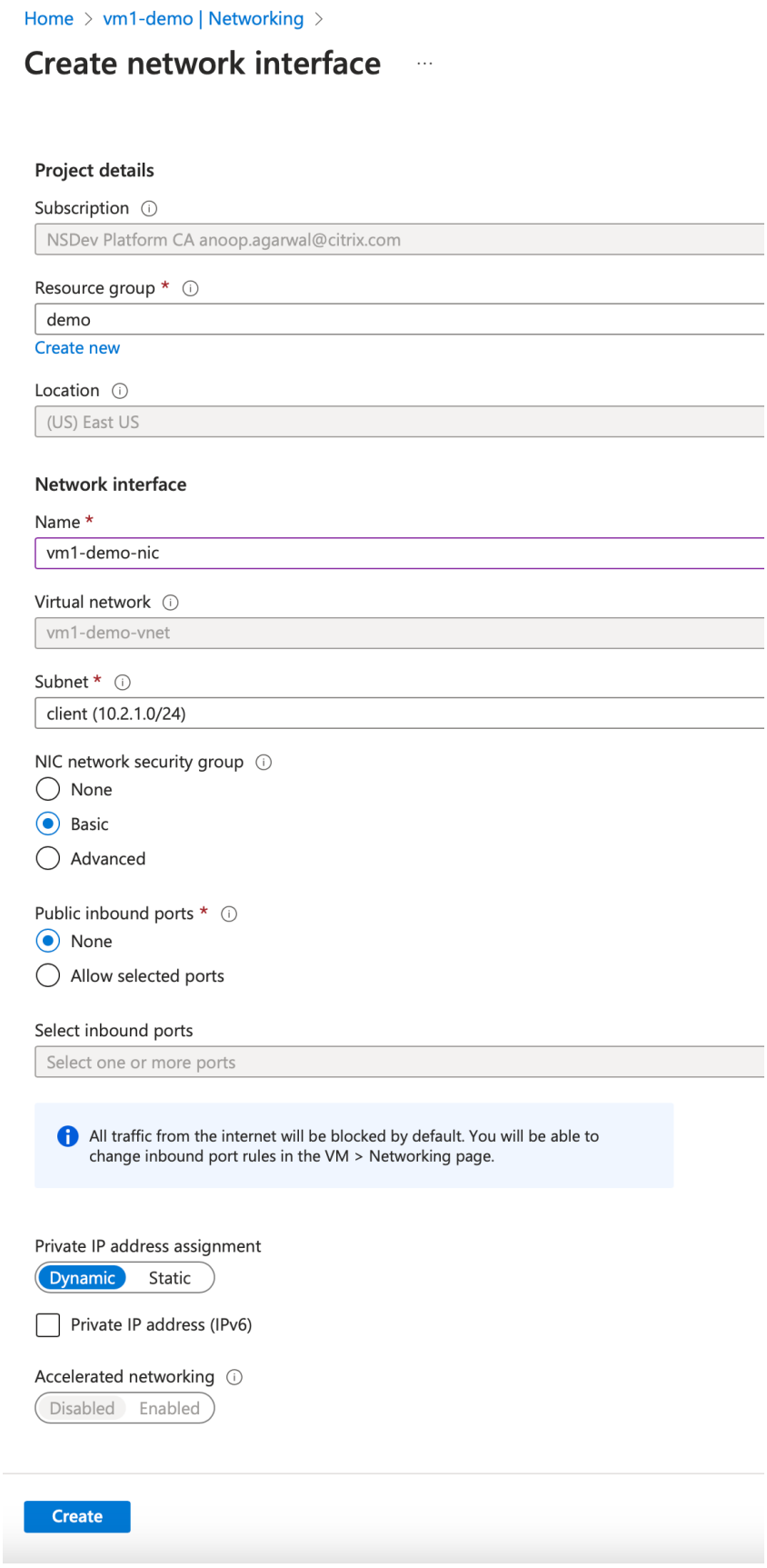
> 注
>
> さらに NIC を接続するには、まず仮想マシンを停止する必要があります。Azure ポータルで、停止する VM を選択します。[概要] タブで、[停止] をクリックします。ステータスが [停止] と表示されるまで待ちます。

プライマリインスタンスにクライアント NIC を追加するには、次の手順に従います。

1. [ネットワーク**] > [**ネットワークインターフェイスの接続] に移動します。

   既存の NIC を選択するか、新しいインターフェイスを作成して接続できます。

2. NIC ネットワークセキュリティグループについては、[詳細] を選択して既存のネットワークセキュリティグループを使用するか、[基本] を選択して作成できます。



サーバ NIC を追加するには、クライアント NIC を追加する場合と同じ手順に従います。

NetScaler VPX インスタンスには、3 つの NIC（管理 NIC、クライアント NIC、およびサーバー NIC）がすべて接続されています。

前の手順を繰り返して、セカンダリインスタンスに NIC を追加します。

両方のインスタンスで NIC を作成してアタッチしたら、[**Overview] > [Start**] に移動して両方のインスタンスを再起動します。

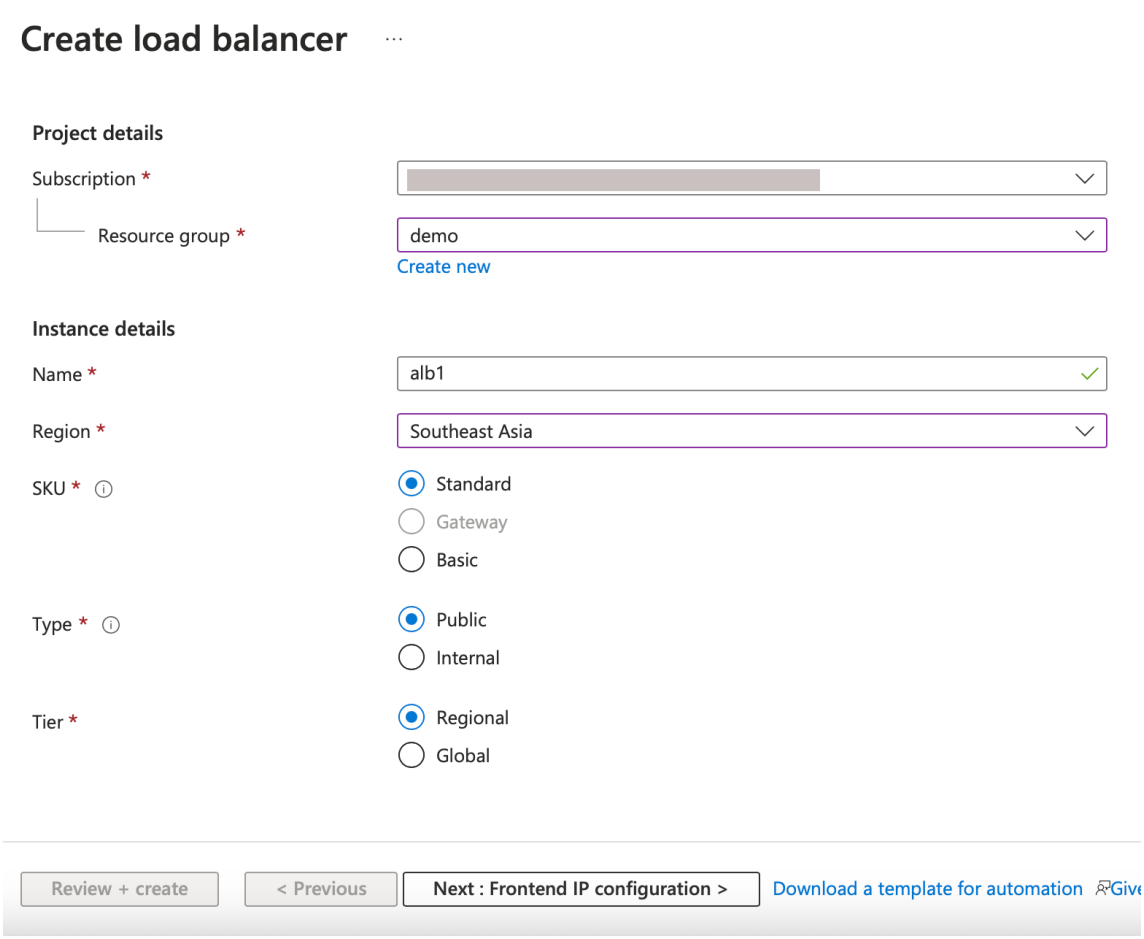
> 注
>
> クライアント NIC インバウンドルールでは、ポートを通過するトラフィックを許可する必要があります。このルールは、後で NetScaler VPX インスタンスの構成時に負荷分散仮想サーバーを作成するために使用されます。

手順 **3.** フローティング **IP** モードが無効になっている負荷分散ルールを持つ **ALB** をデプロイします。

ALB の設定を開始するには、次の手順に従います。

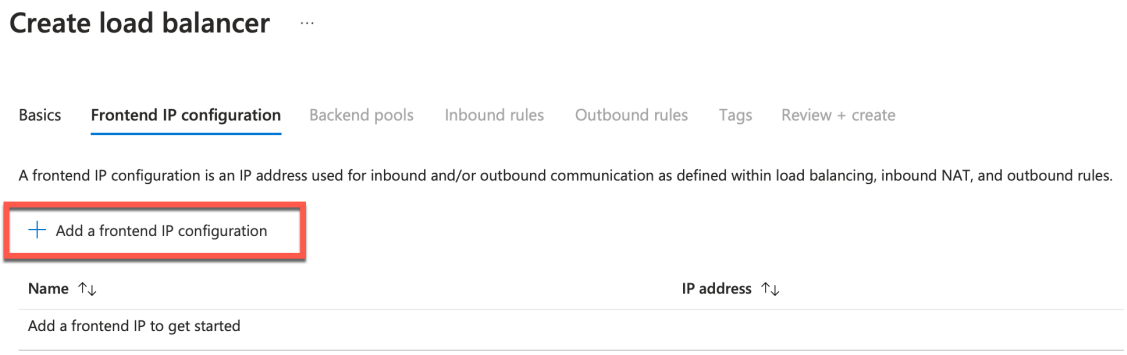
1. [ロードバランサー] ページに移動し、[作成] をクリックします。
2. **[**ロードバランサーの作成] ページで、必要に応じて詳細を入力します。

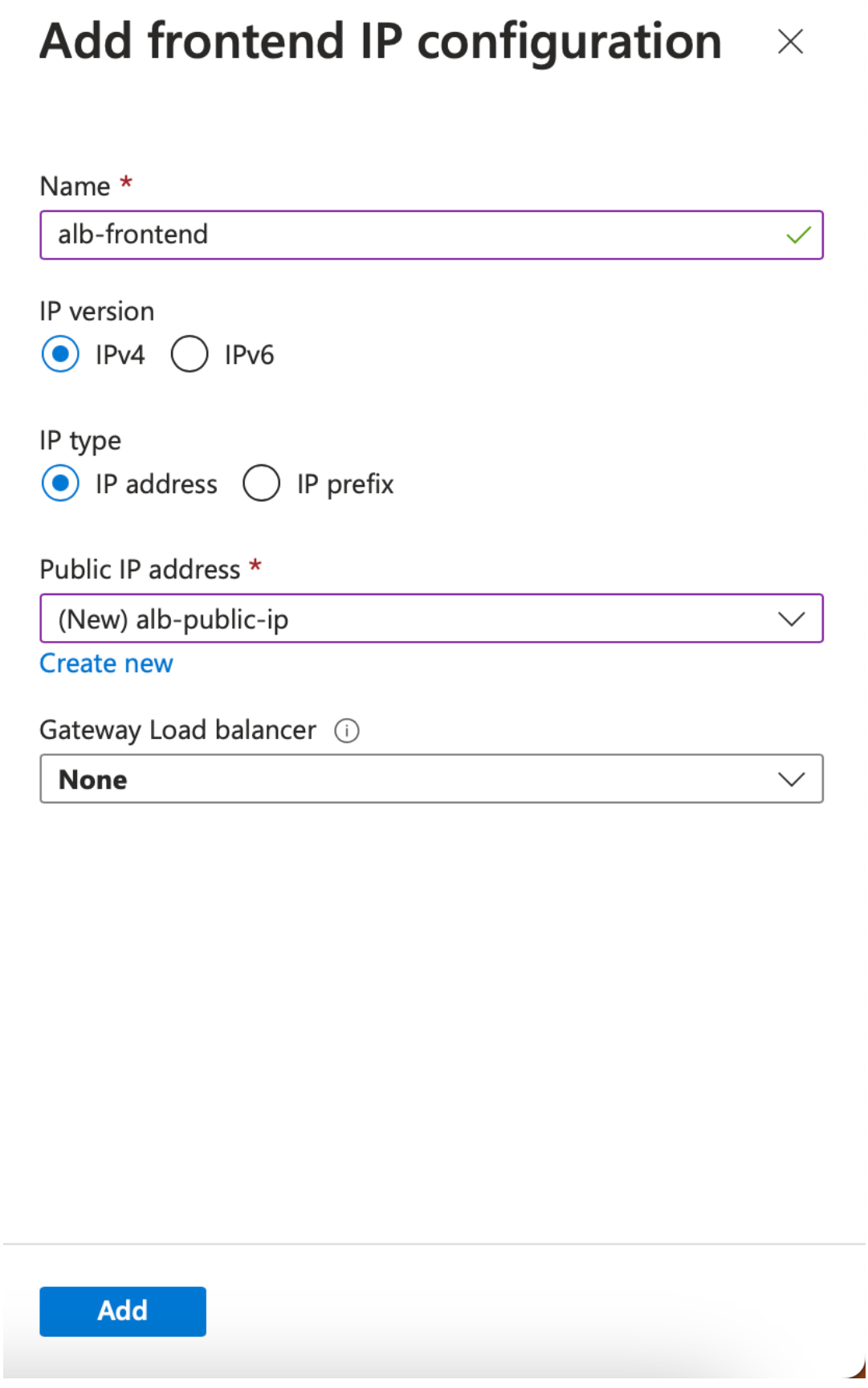
   次の例では、Standard SKU のリージョンパブリックロードバランサーをデプロイします。



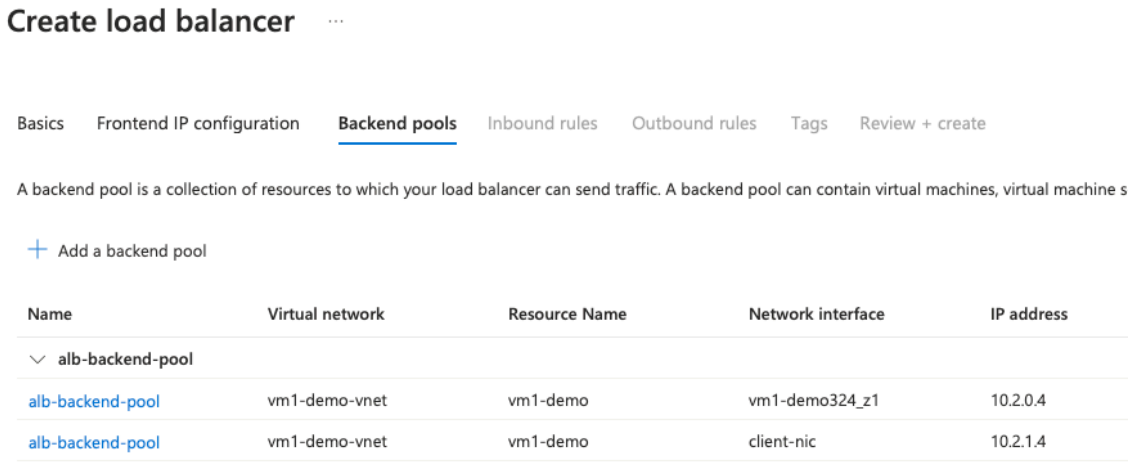
> 注
>
> NetScaler 仮想マシンに接続されているすべてのパブリック IP は、ALB の SKU と同じ SKU を持つ必要があります。ALB SKU の詳細については、Azure Load Balancer SKU のドキュメントを参照してください。

3. [フロントエンド **IP** 設定] タブで、IP アドレスを作成するか、既存の IP アドレスを使用します。

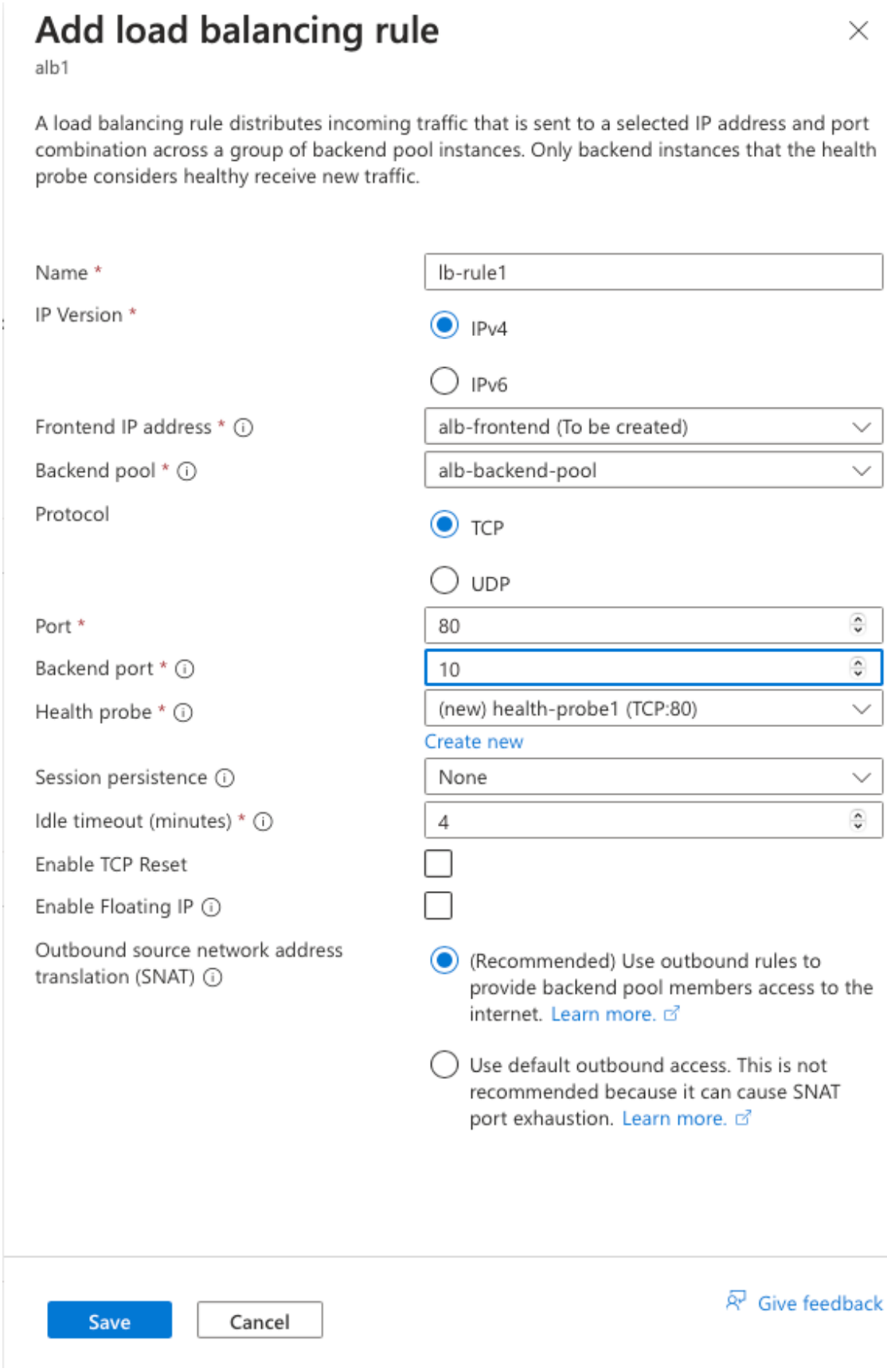




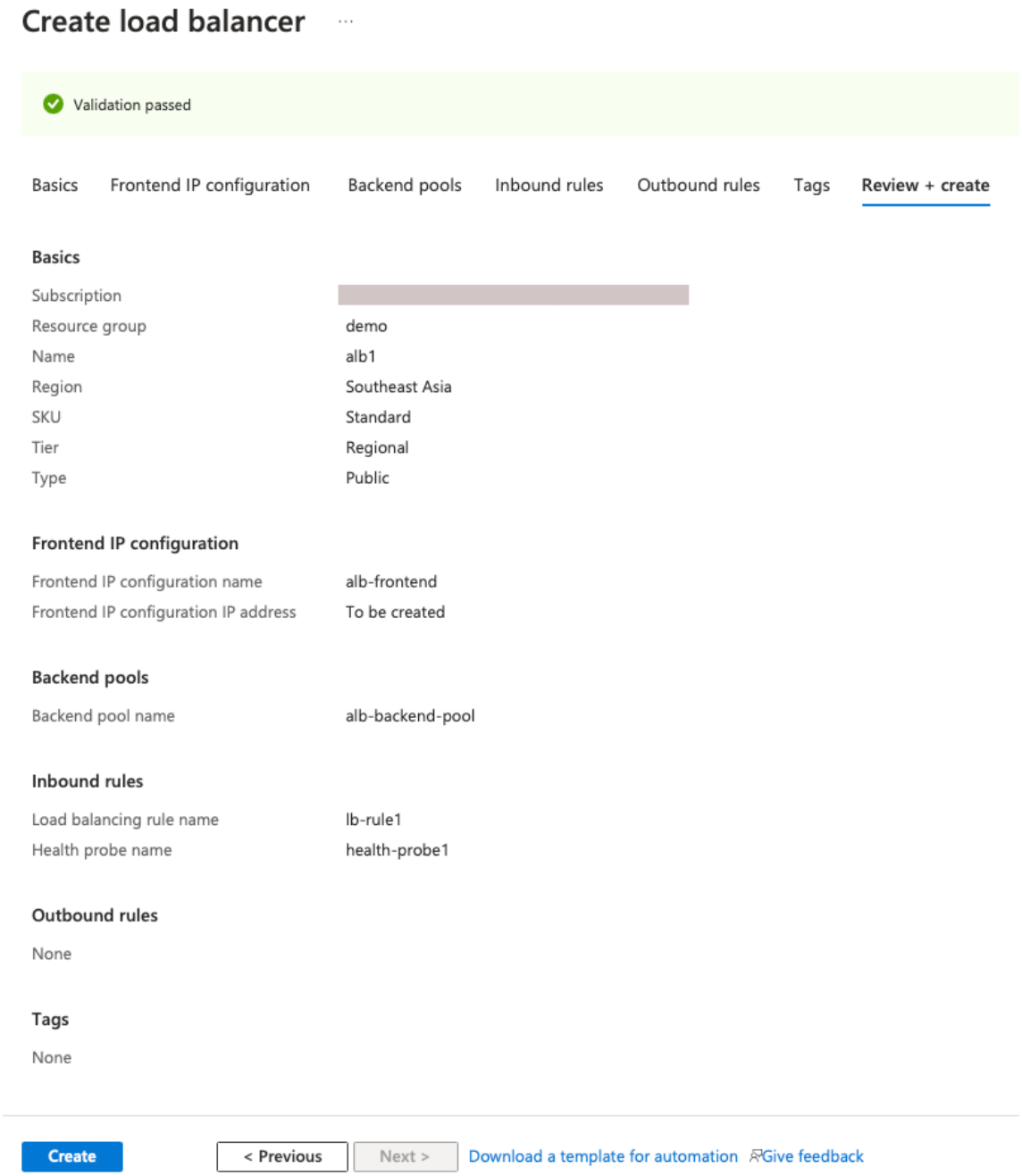
4. ［バックエンドプール］タブで、［NIC ベースのバックエンドプール構成］を選択し、両方の NetScaler 仮想マシンのクライアント NIC を追加します。



5. [受信ルール] タブで、[負荷分散ルールの追加] をクリックし、前の手順で作成したフロントエンド IP アドレスとバックエンドプールを指定します。要件に基づいてプロトコルとポートを選択します。ヘルスプローブを作成するか、既存のヘルスプローブを使用します。「フローティング **IP** を有効にする」チェックボックスをオフにします。



6. [レビュー**] + [**作成] をクリックします。検証に合格したら、[作成] をクリックします。



手順 **4**: **IP** セットを追加し、両方のインスタンスで **IP** セットをセカンダリ **VIP** にバインドします。

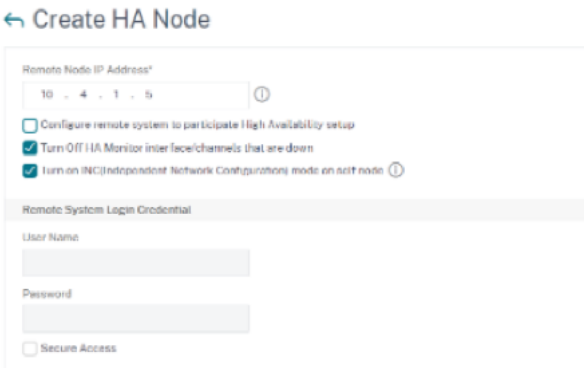
Azure で NetScaler VPX インスタンスを作成したら、NetScaler GUI を使用して HA を構成できます。

手順 **1.** 両方のインスタンスで **INC** モードで高可用性をセットアップします。

プライマリ・インスタンスで、次の手順を実行します。

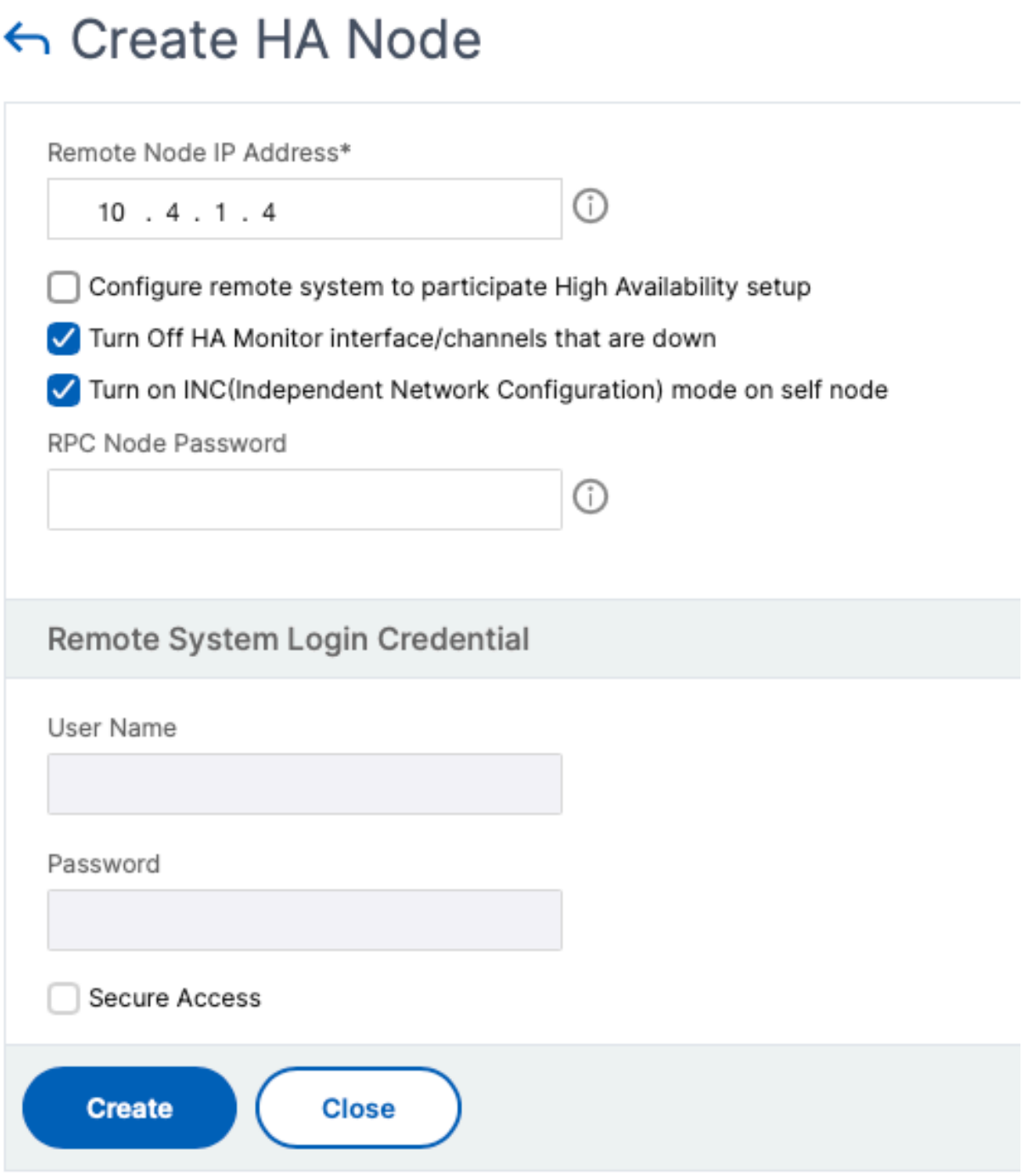
1. インスタンスのデプロイ時に指定したユーザー名`nsroot`とパスワードを使用して、インスタンスにログオンします。

2. 構成 **>** システム **>** 高可用性 **>** ノードに移動し、追加をクリックします。
3. [リモートノードの **IP** アドレス] フィールドに、セカンダリインスタンスの管理 NIC のプライベート IP アドレス（例：10.4.1.5）を入力します。
4. 「セルフノードで **INC (**独立ネットワーク構成**)** モードを有効にする」チェックボックスを選択します。
5. [**Create**] をクリックします。



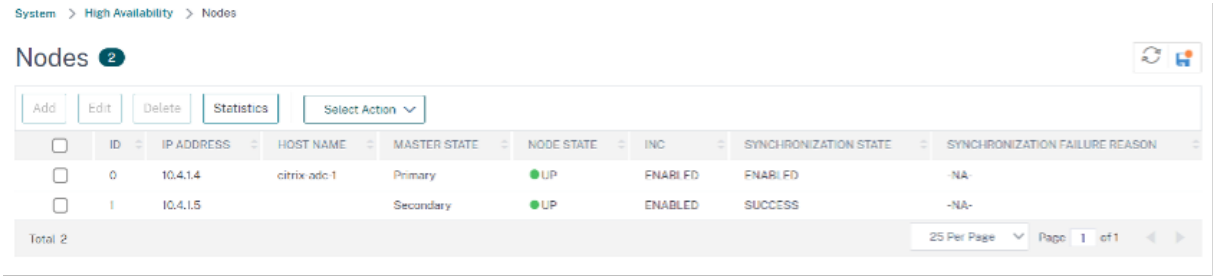
セカンダリインスタンスで、次の手順を実行します。

1. インスタンスのデプロイ時に指定したユーザー名`nsroot`とパスワードを使用して、インスタンスにログオンします。
2. 構成 **>** システム **>** 高可用性 **>** ノードに移動し、追加をクリックします。
3. [リモートノードの **IP** アドレス] フィールドに、プライマリインスタンスの管理 NIC のプライベート IP アドレス（例：10.4.1.4）を入力します。
4. 「セルフノードで **INC (**独立ネットワーク構成**)** モードを有効にする」チェックボックスを選択します。
5. [**Create**] をクリックします。



先に進む前に、セカンダリインスタンスの同期状態が [ノード] ページで [**SUCCESS**] と表示されていることを確認します。

> 注
>
> これで、セカンダリインスタンスはプライマリインスタンスと同じログオン認証情報を持つようになりました。



手順 **3.** 両方のインスタンスに仮想 **IP** アドレスとサブネット **IP** アドレスを追加します。

プライマリ・インスタンスで、次の手順を実行します。

1. **System > Network > IP Sets > Add** に移動します。
2. 次の手順に従って、プライマリ VIP アドレスを追加します。

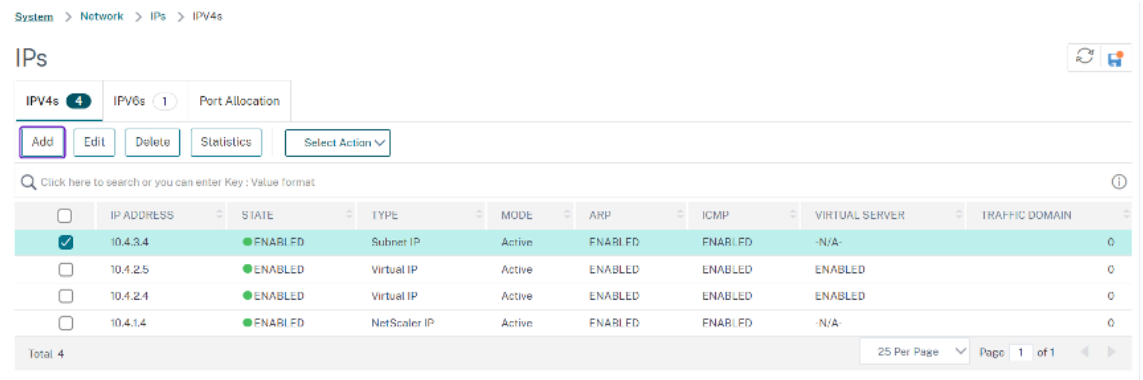
   a) プライマリインスタンスのクライアント NIC のプライベート IP アドレスと、VM インスタンスのクライアントサブネットに設定されているネットマスクを入力します。
   b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Virtual IP**] を選択します。
   c) [**Create**] をクリックします。

3. 次の手順に従って、プライマリ SNIP アドレスを追加します。

   a) プライマリ・インスタンスのサーバ NIC の内部 IP アドレスと、プライマリ・インスタンスのサーバ・サブネットに設定されているネットマスクを入力します。
   b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Subnet IP**] を選択します。
   c) [**Create**] をクリックします。

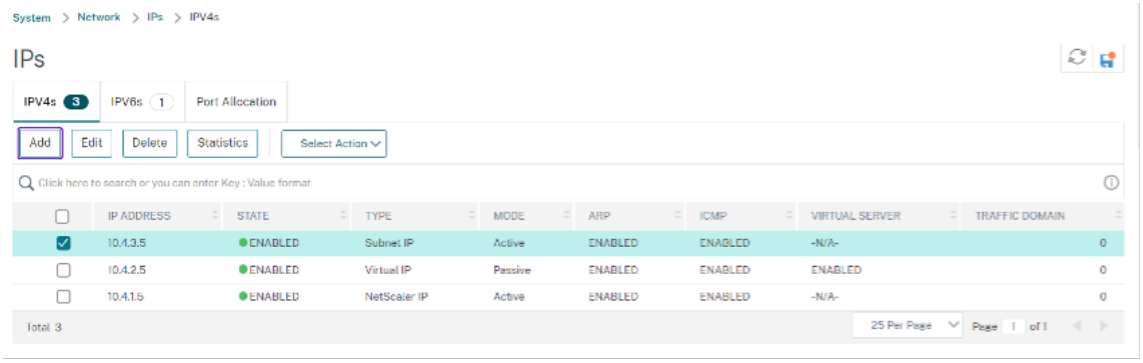
4. 次の手順に従って、セカンダリ VIP アドレスを追加します。

   a) セカンダリインスタンスのクライアント NIC の内部 IP アドレスと、VM インスタンスのクライアントサブネットに設定されているネットマスクを入力します。
   b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Virtual IP**] を選択します。
   c) [**Create**] をクリックします。



セカンダリインスタンスで、次の手順を実行します。

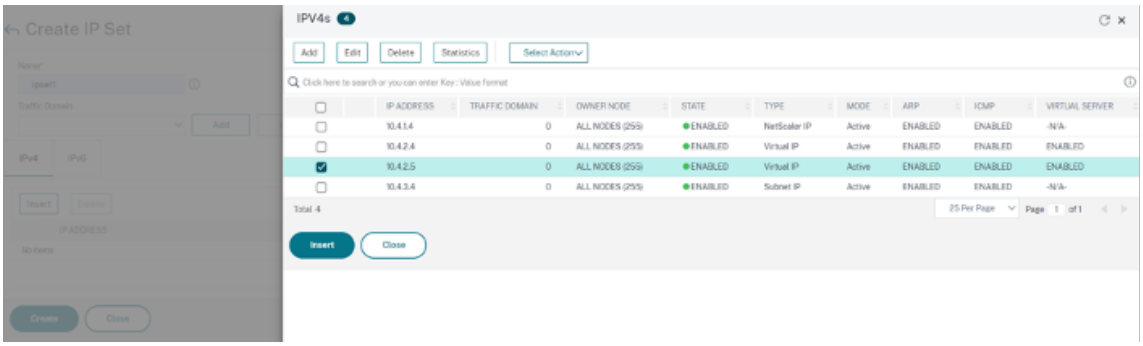
1. **System > Network > IP Sets > Add** に移動します。
2. 次の手順に従って、セカンダリ VIP アドレスを追加します。
   a) セカンダリインスタンスのクライアント NIC の内部 IP アドレスと、VM インスタンスのクライアントサブネットに設定されているネットマスクを入力します。
   b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Virtual IP**] を選択します。
3. 次の手順に従って、セカンダリ SNIP アドレスを追加します。
   a) セカンダリインスタンスのサーバ NIC の内部 IP アドレスと、セカンダリインスタンスのサーバサブネットに設定されているネットマスクを入力します。
   b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Subnet IP**] を選択します。
   c) [**Create**] をクリックします。



手順 **3. IP** セットを追加し、両方のインスタンスで **IP** セットをセカンダリ **VIP** にバインドします。

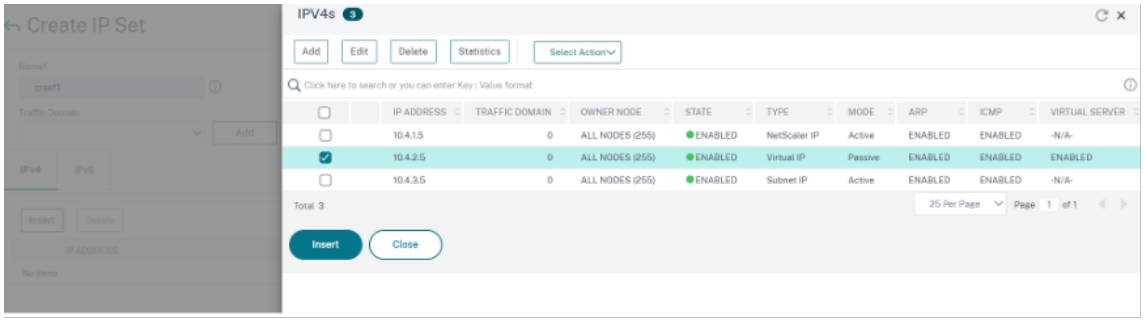
プライマリ・インスタンスで、次の手順を実行します。

1. ステップ **2:** 両方のインスタンスに IP セットを追加します。
2. IP セット名を追加し、[**Insert**] をクリックします。
3. [**IPv4**] ページで、仮想 IP（セカンダリ VIP）を選択し、[挿入] をクリックします。
4. [**Create**] をクリックして IP セットを作成します。



セカンダリインスタンスで、次の手順を実行します。

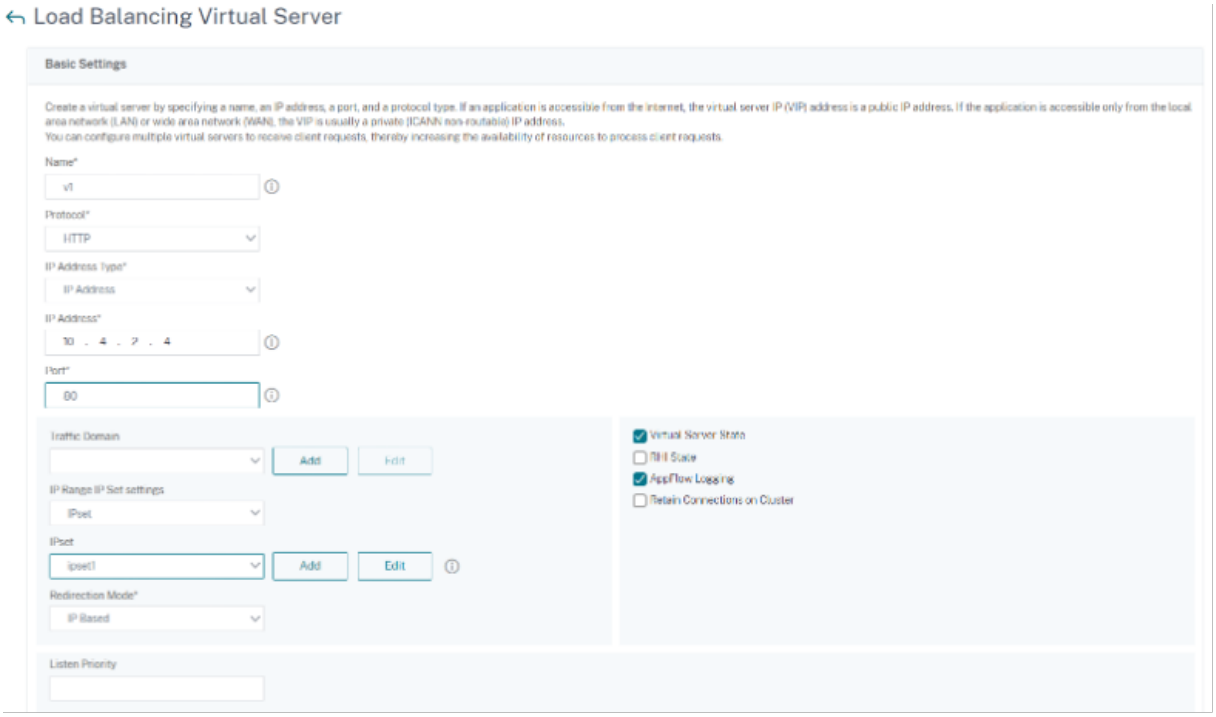
1. ステップ **2:** 両方のインスタンスに IP セットを追加します。
2. IP セット名を追加し、[**Insert**] をクリックします。
3. [**IPv4s**] ページで、仮想 IP (セカンダリ VIP) を選択し、[挿入] をクリックします。
4. [**Create**] をクリックして IP セットを作成します。



> 注
>
> IP セット名は、プライマリインスタンスとセカンダリインスタンスの両方で同じである必要があります。

手順 **4**: プライマリインスタンスにサービスまたはサービスグループを追加します。

1. [設定] **>** [トラフィック管理] **>** [負荷分散] **>** [仮想サーバー] **>** [追加] に移動します。
2. [名前]、[プロトコル]、[IP アドレスタイプ (IP アドレス)]、[IP アドレス] (プライマリ VIP)、および [ポート] に必要な値を追加します。
3. [詳細] クリックします。[**IP** 範囲 **IP** セット設定] に移動し、ドロップダウンメニューから [**IPSet**] を選択し、ステップ **3** で作成した IPSet を指定します。
4. **OK** をクリックして、負荷分散仮想サーバーを作成します。

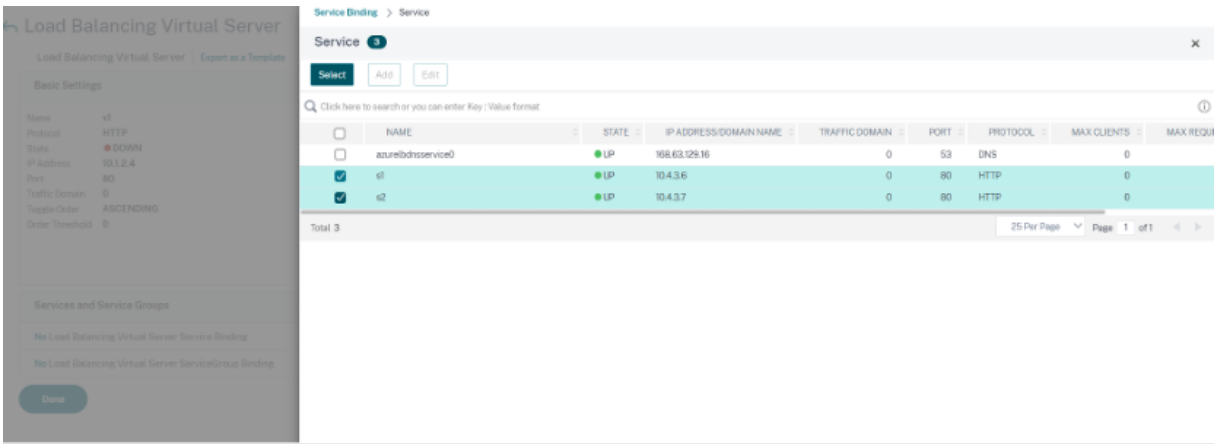


手順 **5**：プライマリインスタンスにサービスまたはサービスグループを追加します。

1. **[設定] > [**トラフィック管理**] > [**負荷分散**] > [**サービス**] >[**追加**]** に移動します。
2. サービス名、IP アドレス、プロトコル、およびポートに必要な値を追加し、[**OK**] をクリックします。

手順 **6**：手順 **6**：サービスまたはサービスグループを、プライマリインスタンスの負荷分散仮想サーバにバインドします。

1. **[設定] > [**トラフィック管理**] > [**負荷分散**] > [**仮想サーバー**]** に移動します。
2. 手順 **4** で構成した負荷分散仮想サーバーを選択し、[編集] をクリックします。
3. [サービスとサービスグループ] タブで、[負荷分散仮想サーバーサービスバインドなし] をクリックします。
4. 手順 **5** で構成したサービスを選択し、[バインド] をクリックします。



手順 **7**：構成を保存します。

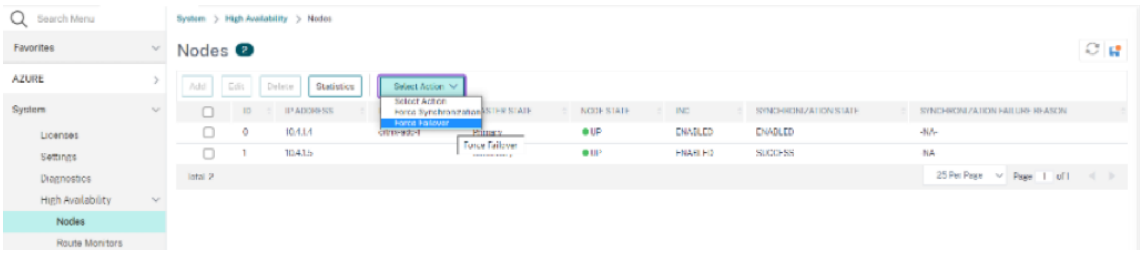
そうしないと、再起動後、または即時再起動が行われた場合、すべての設定が失われます。

手順 **8**：設定を確認します。

フェイルオーバー後に ALB フロントエンド IP アドレスに到達できることを確認します。

1. ALB フロントエンド IP アドレスをコピーします。
2. ブラウザに IP アドレスを貼り付け、バックエンドサーバーにアクセスできることを確認します。
3. プライマリインスタンスで、フェイルオーバーを実行します：

   NetScaler GUI から、［構成］**>**［システム］**>**［高可用性］**>**［アクション］**>**［強制フェールオーバー］に移動します。



4. フェイルオーバー後、以前に使用した ALB フロントエンド IP を介してバックエンドサーバーにアクセスできることを確認してください。

Azure DNS プライベートゾーン用 NetScaler デプロイ

October 17, 2024

Azure DNS は、DNS ドメインをホストし、名前解決を行うための Microsoft Azure インフラストラクチャ上のサービスです。

Azure DNS プライベートゾーンは、プライベートネットワーク内のドメイン名の解決に重点を置いたサービスです。プライベートゾーンでは、顧客は Azure が提供している現在提供されている名前ではなく、独自のカスタムドメイン名を使用できます。

業界をリードするアプリケーション配信ソリューションである NetScaler は、Azure DNS プライベートゾーンの負荷分散と GSLB 機能を提供するのに最適です。Azure DNS プライベートゾーンに登録することで、企業は NetScaler Global Server Load Balancing（GSLB）の機能とインテリジェンスを利用して、安全な VPN トンネルを介して接続された複数の地域のワークロードやデータセンターにイントラネットトラフィックを分散できます。このコラボレーションにより、企業は Azure パブリッククラウドに移行したいワークロードの一部にシームレスにアクセスできるようになります。

Azure DNS の概要

ドメインネームシステム (DNS) は、サービス名をその IP アドレスに変換または解決します。DNS ドメイン用のホスティングサービスである Azure DNS は、Microsoft Azure インフラストラクチャを使用して名前解決を行います。Azure DNS は、インターネットに直接接続する DNS ドメインをサポートするだけでなく、プライベート DNS ドメインもサポートするようになりました。

Azure DNS は、カスタム DNS ソリューションを必要とせずに仮想ネットワーク内のドメイン名を管理および解決するための、信頼性が高く安全な DNS サービスを提供します。プライベート DNS ゾーンを使用すると、Azure が提供する名前ではなく、独自のカスタムドメイン名を使用できます。カスタムドメイン名を使用すると、組織のニーズに合わせて仮想ネットワークアーキテクチャを調整できます。仮想ネットワーク内および仮想ネットワーク間の仮想マシン (VM) の名前解決を行います。また、お客様はスプリットホライズンビューでゾーン名を設定できます。これにより、プライベート DNS ゾーンとパブリック DNS ゾーンで名前を共有できます。

Azure DNS プライベートゾーン用の **NetScaler GSLB** を選ぶ理由

今日の世界では、企業はワークロードをオンプレミスから Azure クラウドに移行したいと考えています。クラウドへの移行により、市場投入までの時間、設備投資/価格、導入のしやすさ、セキュリティを確保できます。Azure DNS プライベートゾーンサービスは、ワークロードの一部を Azure クラウドに移行する企業に独自の提案を提供します。これらの企業は、プライベートゾーンサービスを使用するときに、オンプレミス展開で長年使用していたプライベート DNS 名を作成できます。このハイブリッドモデルのイントラネットアプリケーションサーバーはオンプレミスにあり、Azure クラウドは安全な VPN トンネルを介して接続されているため、課題の 1 つは、これらのイントラネットアプリケーションにシームレスにアクセスできるようにすることです。NetScaler は、アプリケーショントラフィックをオンプレミスまたは Azure クラウド上の最適な分散ワークロード/サーバーにルーティングし、アプリケーションサーバーのヘルスステータスを提供するグローバル負荷分散機能によって、このユニークなユースケースを解決します。

使用例

オンプレミスネットワークと異なる Azure VNet のユーザーは、内部ネットワーク内の最適なサーバーに接続して、必要なコンテンツにアクセスできます。これにより、アプリケーションが常に利用可能になり、コストが最適化され、ユーザーエクスペリエンスが向上します。ここでは、Azure プライベートトラフィック管理 (PTM) が主な要件です。Azure PTM は、ユーザーの DNS クエリがアプリケーションサーバーの適切なプライベート IP アドレスに解決されるようにします。

ユースケースソリューション

NetScaler には、Azure PTM の要件を満たすグローバルサーバー負荷分散（GSLB）機能が含まれています。GSLB は DNS サーバーのように機能し、DNS リクエストを受け取り、DNS リクエストを適切な IP アドレスに解決して以下を提供します。

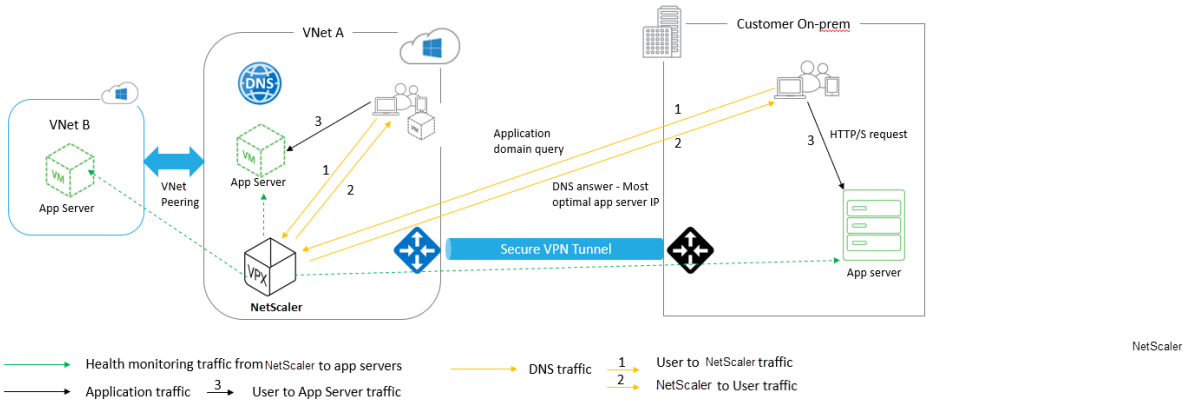
- DNS ベースのシームレスなフェイルオーバー。
- オンプレミスからクラウドへの段階的移行。
- 新機能の A/B テスト。

サポートされている多くの負荷分散方法の中で、このソリューションでは次の方法が役立ちます。

1. ラウンドロビン
2. 静的近接性 (ロケーションベースのサーバー選択)。次の 2 つの方法で導入できます。
    a) NetScaler 上の EDNS クライアントサブネット (ECS) ベースの GSLB。
    b) すべての仮想ネットワークに DNS フォワーダーをデプロイします。

トポロジ

次の図は、Azure プライベート DNS ゾーンの NetScaler GSLB デプロイメントを示しています。



NetScaler

ユーザーは、Azure プライベート DNS ゾーンの NetScaler GSLB メソッドに基づいて、Azure またはオンプレミスの任意のアプリケーションサーバーにアクセスできます。オンプレミスと Azure 仮想ネットワーク間のすべてのトラフィックは、安全な VPN トンネルのみを経由します。アプリケーショントラフィック、DNS トラフィック、およびモニタリングトラフィックは、前述のトポロジに示されています。必要な冗長性に応じて、NetScaler と DNS フォワーダーを仮想ネットワークとデータセンターに導入できます。わかりやすくするために、ここでは NetScaler を 1 つだけ表示していますが、Azure リージョンには少なくとも 1 セットの NetScaler と DNS フォワーダーを使用することをお勧めします。すべてのユーザー DNS クエリは、まずクエリを適切な DNS サーバーに転送するためのルールが定義されている DNS フォワーダーに送られます。

Azure DNS プライベートゾーン用 **NetScaler** 構成

テストした製品とバージョン:

Product	バージョン
Azure	クラウドサブスクリプション
NetScaler VPX	BYOL (独自のライセンスを持参)

> 注
>
> 導入はテスト済みで、NetScaler バージョン 12.0 以降と変わりません。

前提条件

一般的な前提条件は次のとおりです。

- サブスクリプションが有効な Microsoft Azure ポータルアカウント。
- オンプレミスと Azure クラウド間の接続 (セキュア VPN トンネル) を確認します。Azure で安全な VPN トンネルを設定するには、「ステップバイステップ:Azure とオンプレミス間のサイト間 VPN ゲートウェイの設定」を参照してください。

ソリューションの説明

HTTP 上で動作し、ラウンドロビン GSLB 負荷分散方式に基づくイントラネットアクセスで Azure とオンプレミス全体にデプロイされる Azure DNS プライベートゾーン (rr.ptm.mysite.net) を 1 つのアプリケーションをホストする場合。この展開を実現するには、NetScaler を使用して Azure プライベート DNS ゾーンの GSLB を有効にします。このゾーンには次の構成が含まれます。

1. Azure とオンプレミスのセットアップを設定します。
2. Azure 仮想ネットワーク上の NetScaler アプライアンス。

Azure とオンプレミスセットアップの設定

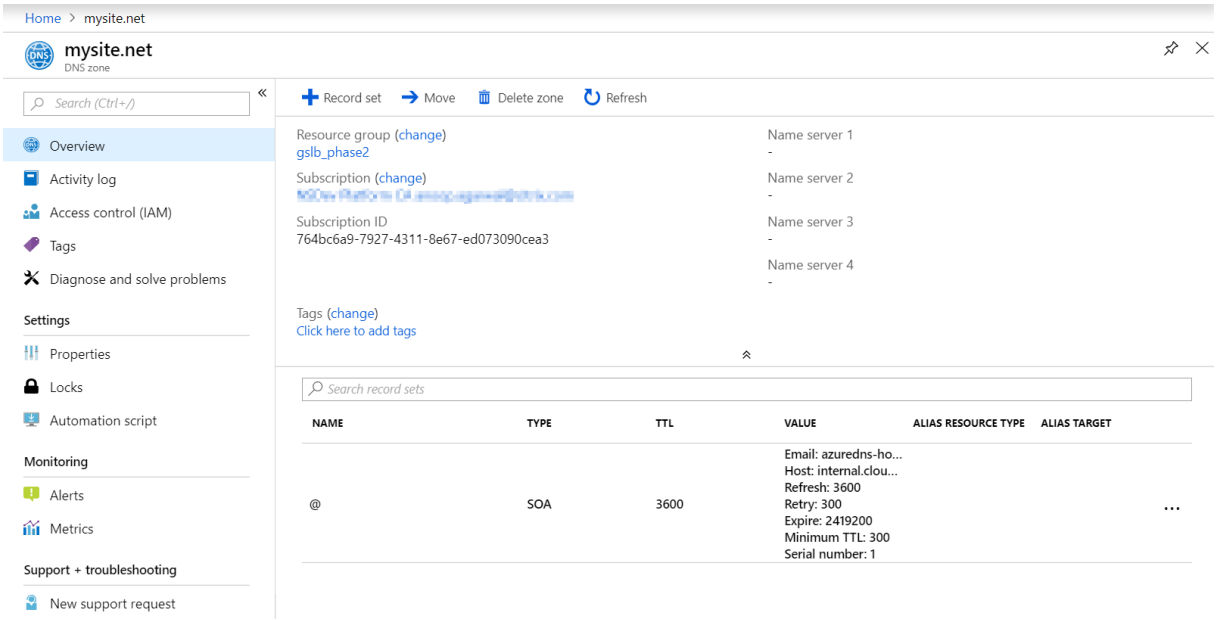
トポロジに示されているように、Azure 仮想ネットワーク (この場合は VNet A、VNet B) とオンプレミスセットアップを設定します。

1. ドメイン名 (mysite.net) を使用して Azure プライベート DNS ゾーンを作成します。
2. Azure リージョンのハブアンドスポークモデルに 2 つの仮想ネットワーク (VNet A、VNet B) を作成します。
3. VNet A にアプリケーションサーバー、DNS フォワーダー、Windows 10 Pro クライアント、NetScaler をデプロイします。
4. アプリケーションサーバーをデプロイし、VNet B にクライアントがある場合は DNS フォワーダーをデプロイします。
5. アプリケーションサーバー、DNS フォワーダー、および Windows 10 pro クライアントをオンプレミスにデプロイします。

Azure プライベート **DNS** ゾーン

ドメイン名を使用して Azure プライベート DNS ゾーンを作成します。

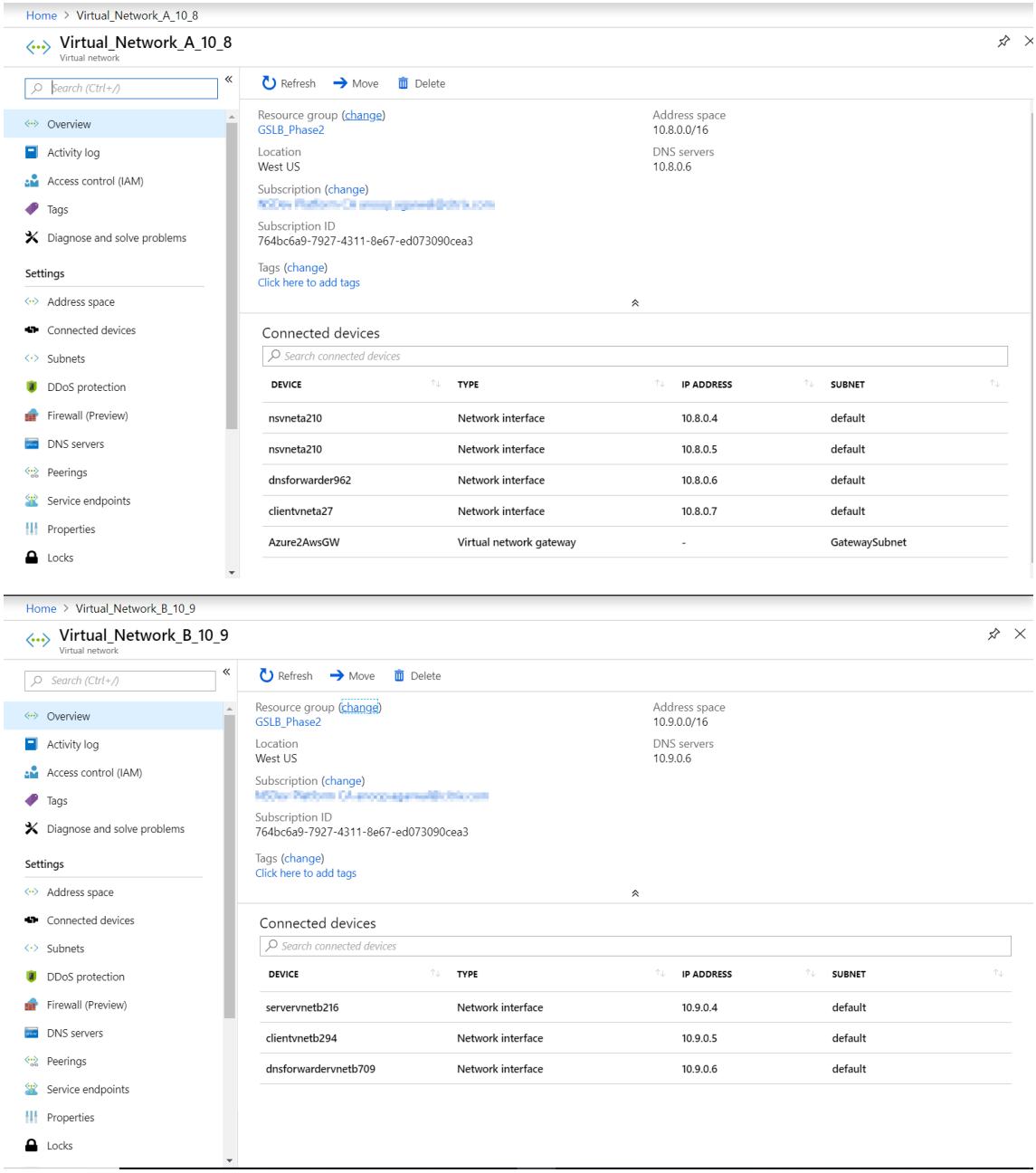
1. Azure Portal にログインし、ダッシュボードを選択または作成します。
2. [リソースの作成] をクリックし、DNS ゾーンを検索して、ドメイン名 (mysite.net) の Azure プライベート DNS ゾーン (この場合は mysite.net) を作成します。



ハブアンドスポークモデルの **Azure** 仮想ネットワーク **(VNet A**、**VNet B)**

Azure リージョンのハブアンドスポークモデルに 2 つの仮想ネットワーク (VNet A、VNet B) を作成します。

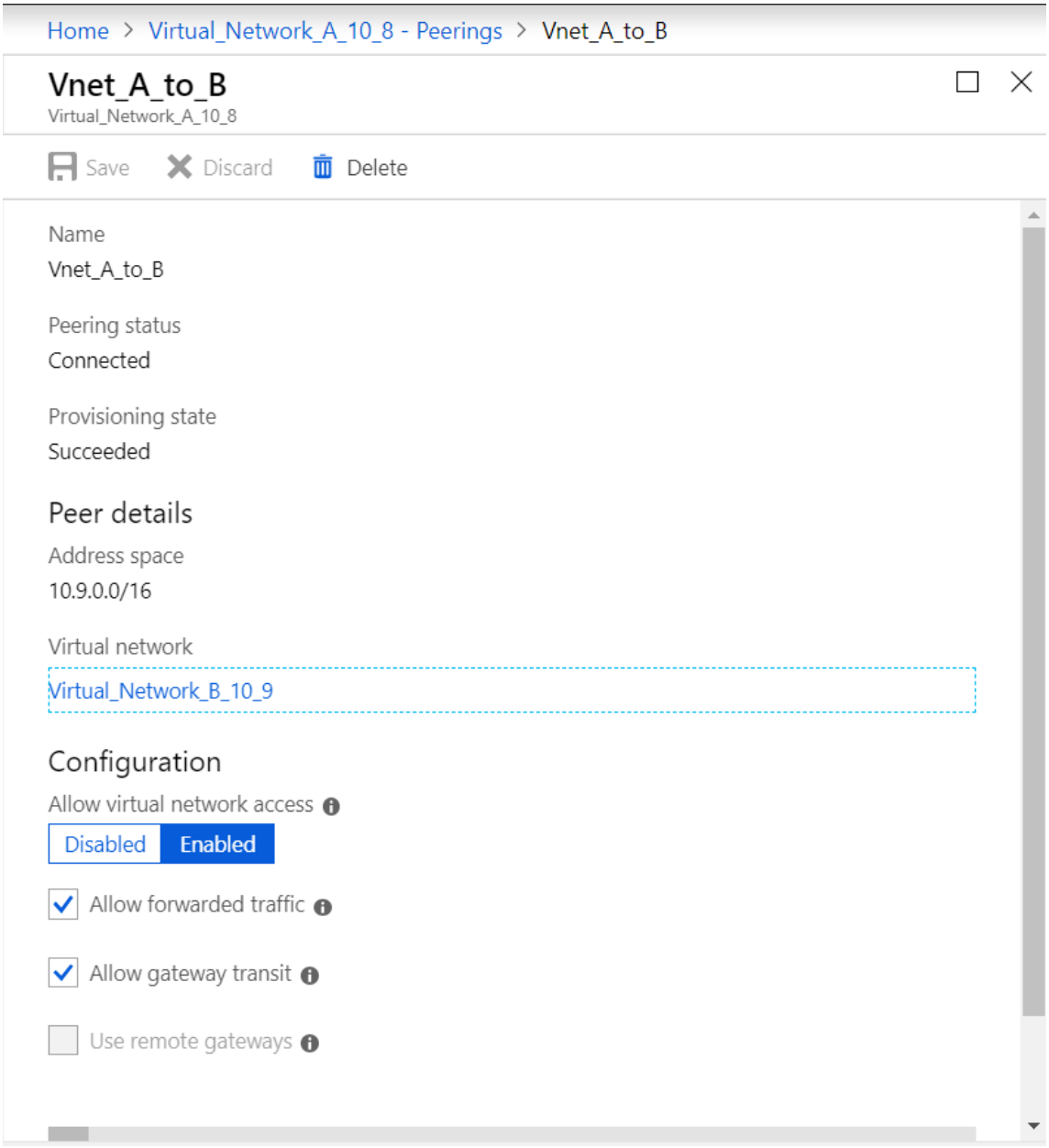
1. 2 つの仮想ネットワークを作成します。
2. 同じダッシュボードを選択し、リソースの作成 をクリックして仮想ネットワークを検索し、同じリージョンに VNet A と VNet B という 2 つの仮想ネットワークを作成し、それらをピアリングして、次の図に示すようにハブ アンド スポーク モデルを形成します。ハブ アンド スポーク トポロジを設定する方法の詳細については、「Azure でハブ スポーク ネットワーク トポロジを実装する」を参照してください。



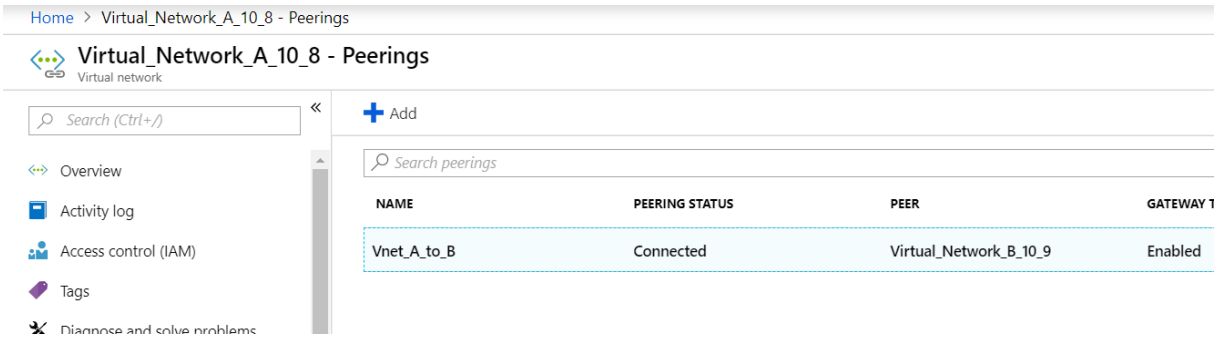
VNet A から **VNet B** へのピアリング

VNet A と VNet B をピアリングするには:

1. VNet A とピア **VNet B** の **[**** 設定] メニューから [ピアリング **] をクリックします。
2. 次の図に示すように、[転送トラフィックを許可する**]** と **[**ゲートウェイトランジットを許可する] を有効にします。



次の図は、VNet A と VNet B の正常なピアリングを示しています。



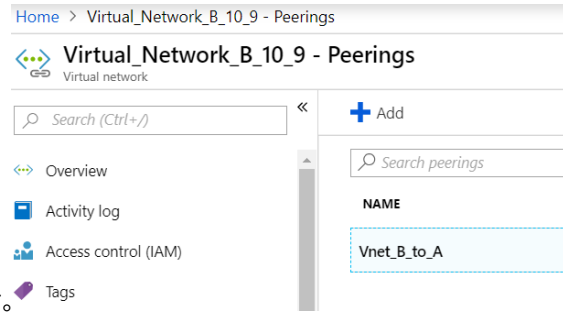
VNet B から **VNet A** へのピアリング

VNet B と VNet A をピアリングするには:

1. VNet B とピア **VNet A** の **[**** 設定] メニューから [ピアリング **] をクリックします。
2. 次の図に示すように、[転送トラフィックを許可し、リモートゲートウェイを使用する] を有効にします。

```
1 ![VNet B to A](/en-us/vpx/media/image-07.png)
```

次の図は、VNet B から VNet A へのピアリングが成功したことを示しています。



VNet A にアプリケーションサーバー、**DNS** フォワーダー、**Windows 10 Pro** クライアント、**NetScaler** をデプロイします

アプリケーションサーバー、DNS フォワーダー、Windows 10 プロクライアント、および VNet A 上の NetScaler について簡単に説明します。

1. 同じダッシュボードを選択し、[リソースを作成] をクリックします。
2. それぞれのインスタンスを検索し、VNet A サブネットから IP を割り当てます。

アプリケーションサーバー　アプリケーションサーバーは、Ubuntu サーバー 16.04 が Azure またはオンプレミス VM にインスタンスとしてデプロイされている Web サーバー（HTTP サーバー）に他なりません。Web サーバーとして設定するには、コマンドプロンプトで次のように入力します。

```
sudo apt install apache2
```

Windows 10 Pro Client　Windows 10 Pro インスタンスを VNet A およびオンプレミスのクライアントマシンとして起動します。

NetScaler　NetScaler は、NetScaler MAS のヘルスチェックと分析によって Azure DNA プライベートゾーンを補完しています。要件に基づいて Azure Marketplace から NetScaler を起動します。ここでは、NetScaler (BYOL) を使用してデプロイしました。

Microsoft Azure に NetScaler をデプロイする方法の詳細な手順については、こちらをご覧ください。Microsoft Azure に NetScaler VPX インスタンスをデプロイするを参照してください。

展開後、NetScaler IP を使用して NetScaler ADC GSLB を構成します。

DNS フォワーダー　NetScaler GSLB (ADNS IP) にバインドされたホストドメインのクライアント要求を転送するために使用されます。Ubuntu サーバー 16.04 を Linux インスタンス（Ubuntu サーバー 16.04）として起動し、DNS フォワーダーとして設定する方法については、以下の URL を参照してください。

> 注
>
> ラウンドロビン GSLB 負荷分散方法では、Azure リージョン用の DNS フォワーダーは 1 つで十分ですが、静的近接の場合は、仮想ネットワークごとに 1 つの DNS フォワーダーが必要です。

1. フォワーダーをデプロイしたら、次の図に示すように、仮想ネットワーク A の DNS サーバー設定をデフォルトから VNet A の DNS フォワーダー IP を使用してカスタムに変更します。
2. VNet A DNS `named.conf.options` フォワーダー内のファイルを変更して、ドメイン (mysite.net) とサブドメイン (ptm.mysite.net) の転送ルールを NetScaler GSLB の ADNS IP に追加します。
3. DNS フォワーダーを再起動して、`named.conf.options`ファイルに加えられた変更を反映します。

VNet A の **DNS** フォワーダー設定

```
      zone "mysite.net" {

                  type forward;
      forwarders {
  168.63.129.16;  }
 ;
       }
 ;
      zone "ptm.mysite.net" {

          type forward;
          forwarders {
  10.8.0.5;  }
 ;
       }
 ;
```

> 注
>
> ドメイン (「mysite.net」) ゾーンの IP アドレスには、Azure リージョンの DNS IP アドレスを使用してください。サブドメイン (「ptm.mysite.net」) ゾーン IP アドレスには、GSLB インスタンスのすべての ADNS IP アドレスを使用してください。

VNet B にクライアントがある場合は、アプリケーションサーバーと **DNS** フォワーダーをデプロイします

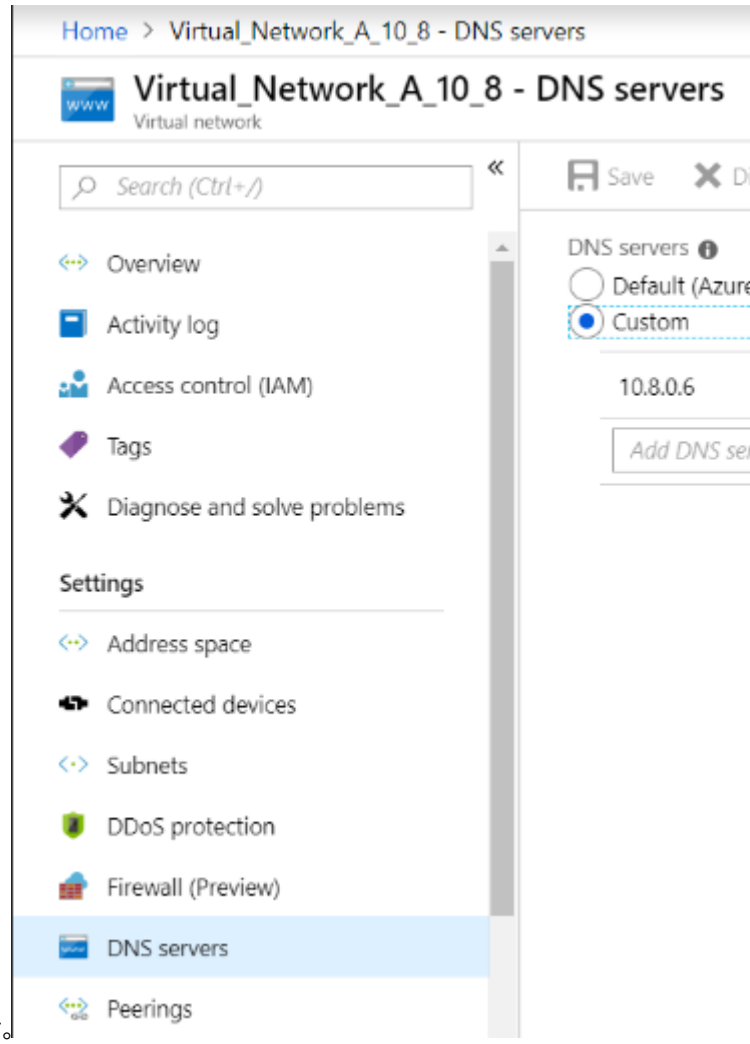
1. 仮想ネットワーク B の場合は、同じダッシュボードを選択し、「リソースを作成」をクリックします。
2. それぞれのインスタンスを検索し、VNet B サブネットから IP を割り当てます。

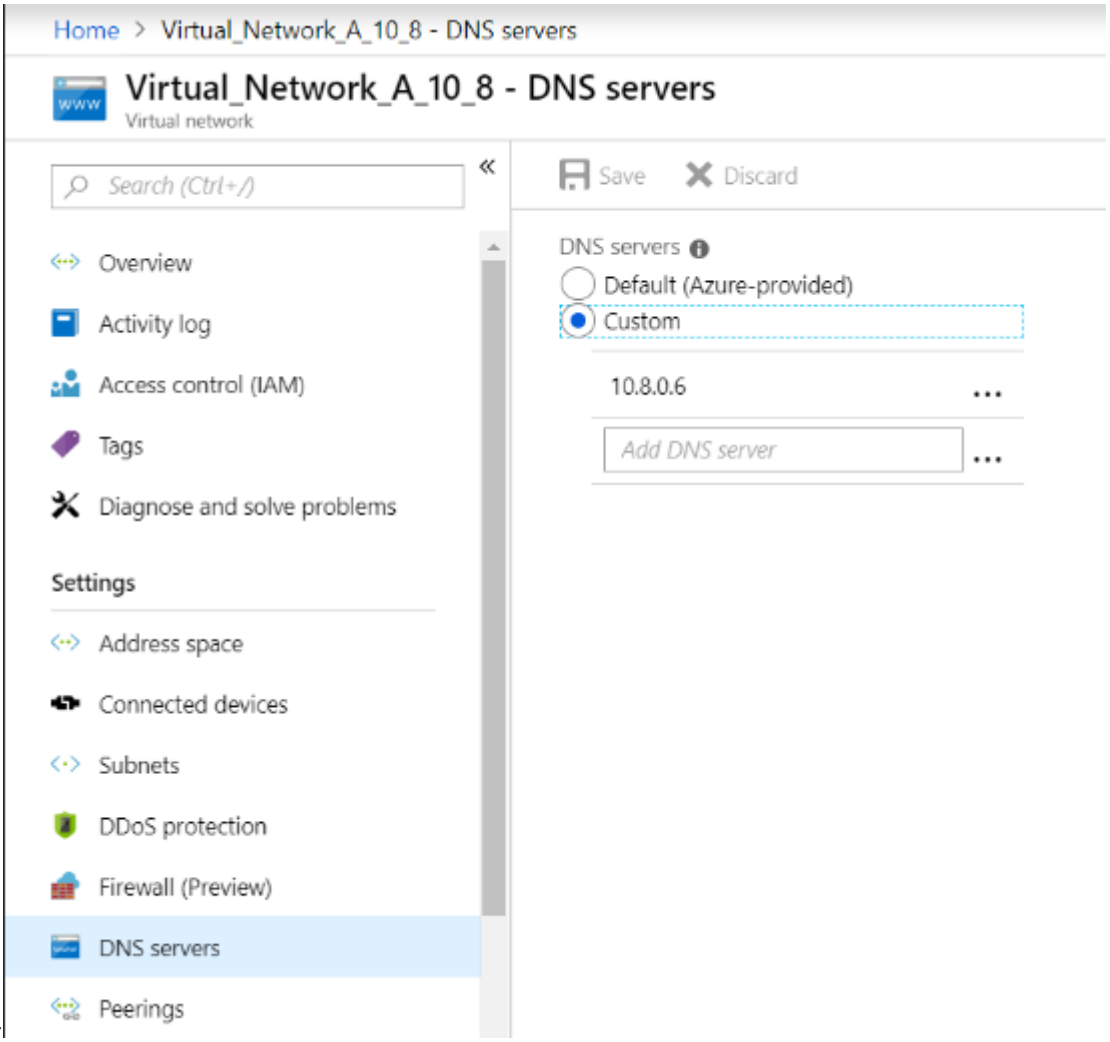
3. VNet A と同様の静的近接 GSLB 負荷分散がある場合は、アプリケーションサーバーと DNS フォワーダーを起動します。
4. 次の設定に示すように、`named.conf.options`の VNet B の DNS フォワーダ設定を編集します。

   VNet B の DNS フォワーダー設定:

```
1        zone "ptm.mysite.net" {
2
3            type forward;
4            forwarders {
5     10.8.0.5;  }
6    ;
7          }
8    ;
```

次の図は、VNet B の DNS フォワーダー設定を示しています。





DNS サーバー

アプリケーションサーバー、**DNS** フォワーダー、および **Windows 10 pro** クライアントをオンプレミスにデプロイ

1. オンプレミスの場合は、ベアメタルで仮想マシンを起動し、アプリケーションサーバー、DNS フォワーダー、および VNet A と同様の Windows 10 プロクライアントを用意します。
2. 次の例に示すように、`named.conf.options`のオンプレミス DNS フォワーダー設定を編集します。

オンプレミス **DNS** フォワーダー設定

```
      zone "mysite.net" {

                  type forward;
                  forwarders {
 10.8.0.6;  }
;
       }
;
      zone "ptm.mysite.net" {

```



```
11            type forward;
12            forwarders {
13    10.8.0.5;  }
14   ;
15          }
16   ;
```

mysite.netについては、Azure プライベート DNS ゾーンサーバー IP の代わりに VNet A の DNS フォワーダー IP を指定しました。そのため、オンプレミスの DNS フォワーダー設定ではこの変更が必要です。

Azure 仮想ネットワーク上で NetScaler を構成します

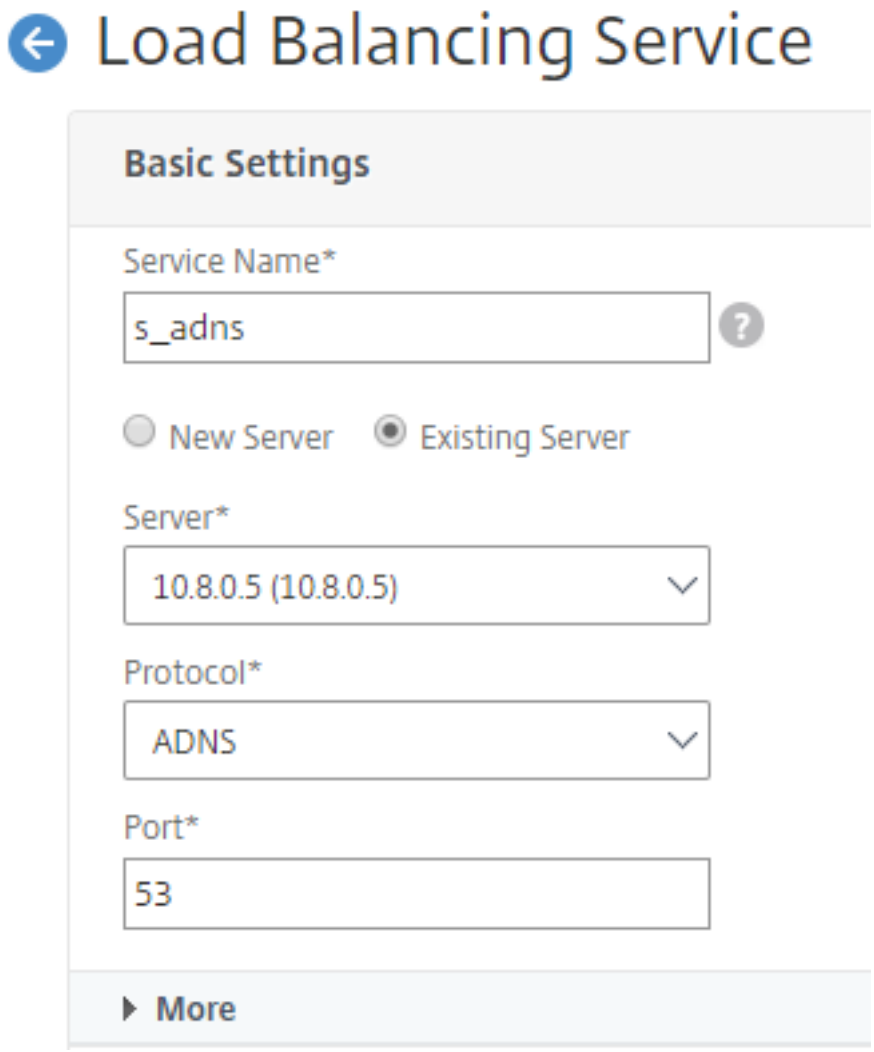
トポロジーに示されているように、NetScaler を Azure 仮想ネットワーク（この場合は VNet A）にデプロイし、NetScaler GUI を介してアクセスします。

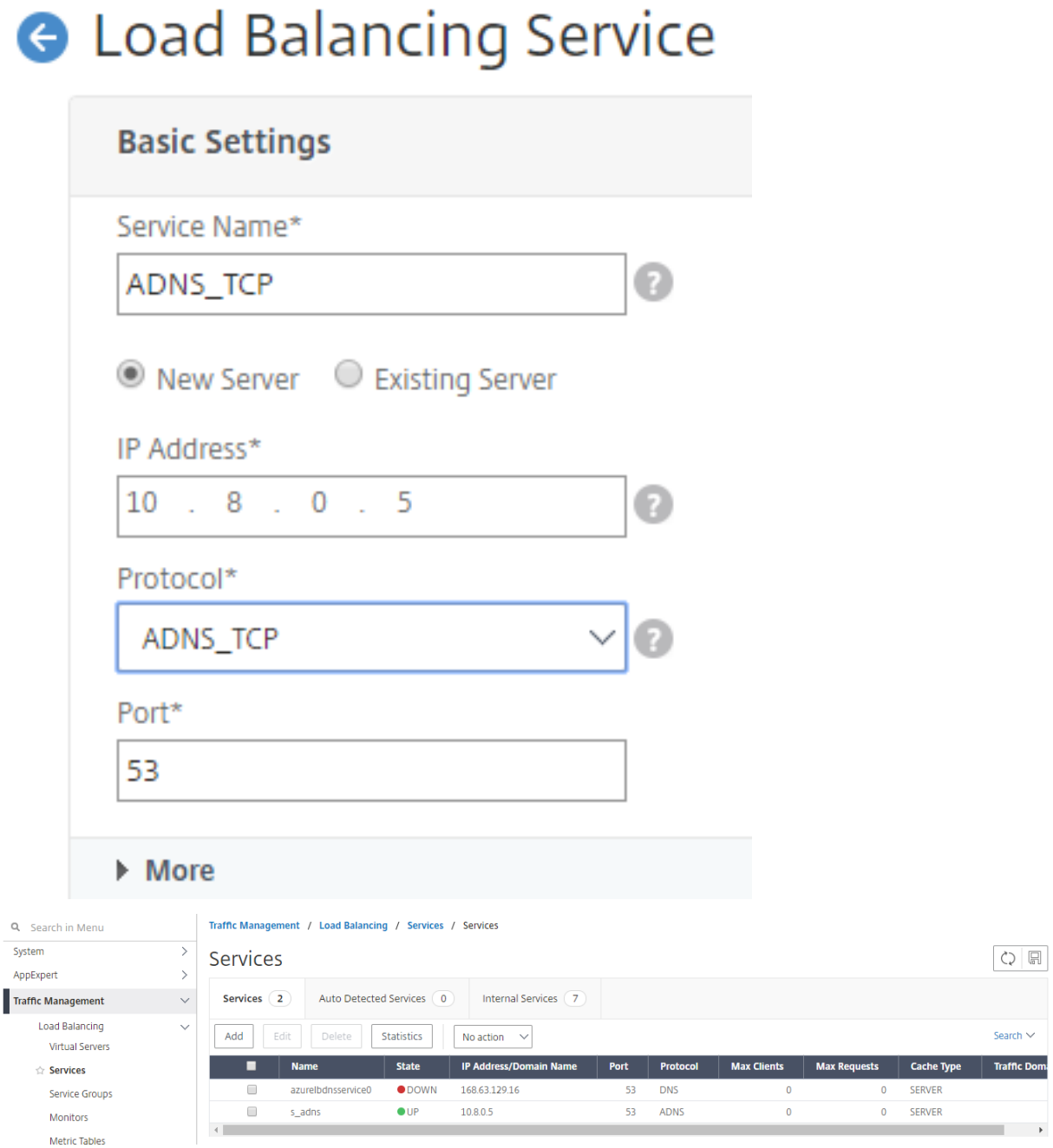
NetScaler GSLB の設定

1. ADNS サービスを作成します。
2. ローカルサイトとリモートサイトを作成します。
3. ローカル仮想サーバー用のサービスを作成します。
4. GSLB サービス用の仮想サーバーを作成します。

ADNS サービスを追加

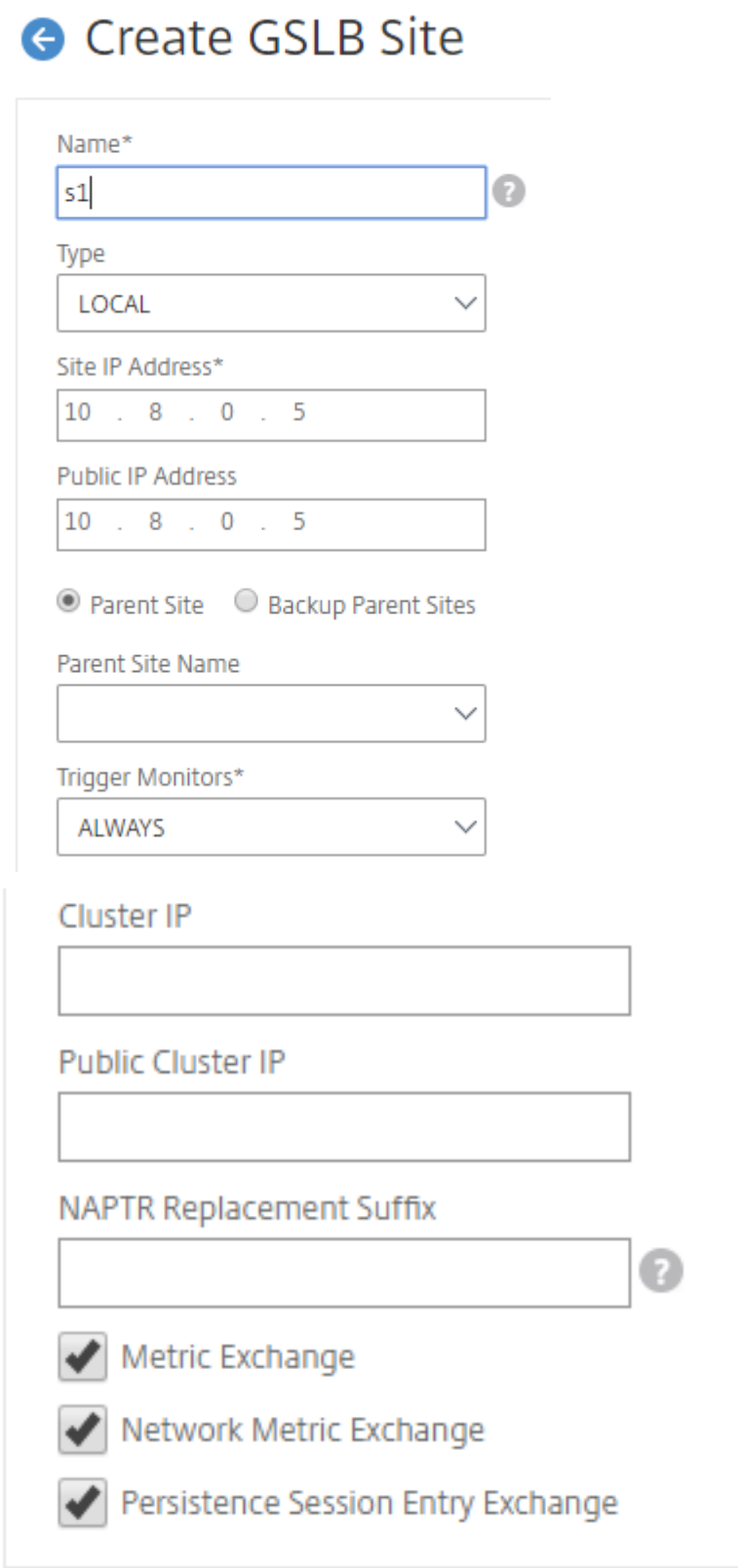
1. NetScaler ユーザーインターフェイスにログインします。
2. [設定] タブで、[トラフィック管理] **>** [負荷分散] **>** [サービス] に移動します。
3. サービスを追加します。サービスを追加します。次の図に示すように、ADNS サービスを TCP と UDP の両方で設定することをお勧めします。

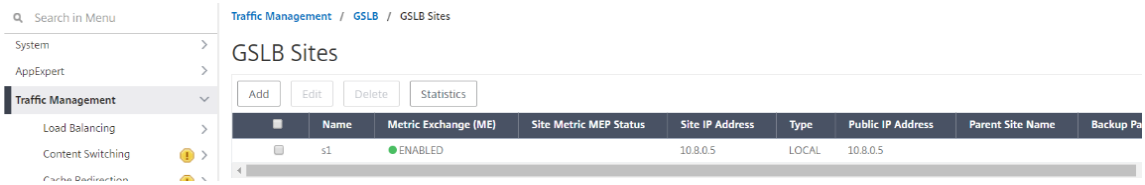




GSLB サイトを追加する

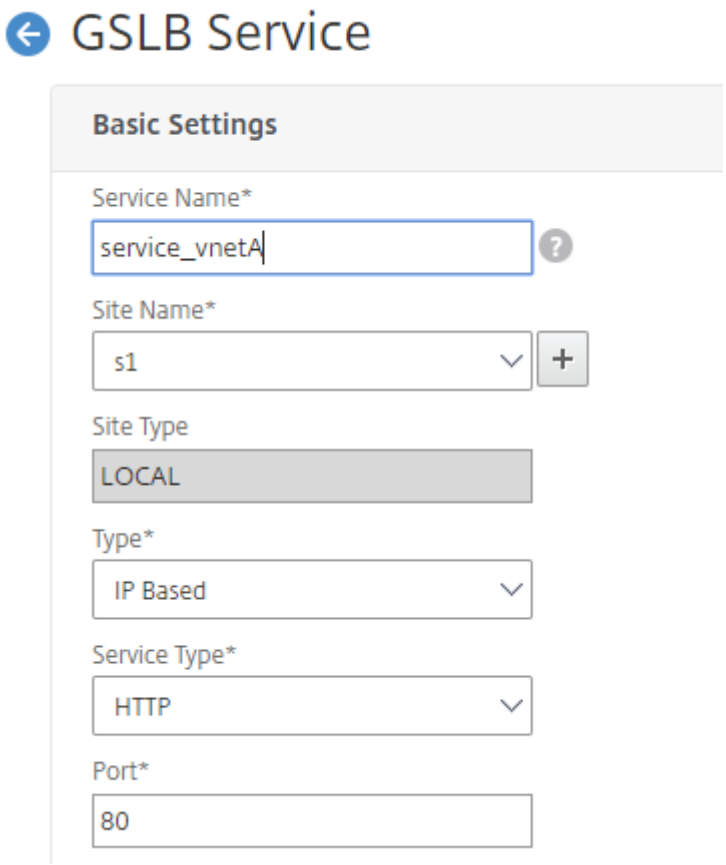
1. GSLB を設定するローカルサイトとリモートサイトを追加します。
2. [設定] タブで、[トラフィック管理**] > [GSLB] > [GSLB** サイト] に移動します。次の例のようにサイトを追加し、他のサイトについても同じ手順を繰り返します。次の例に示すようにサイトを追加し、他のサイトに対しても同じ手順を繰り返します。





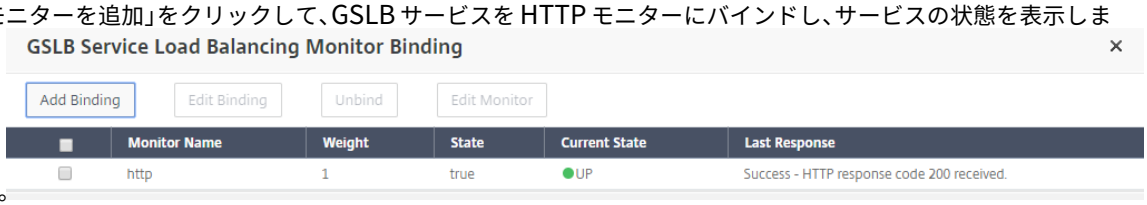
GSLB サービスの追加

1. アプリケーションサーバーの負荷分散を行うローカルおよびリモートの仮想サーバー用の GSLB サービスを追加します。
2. [設定] タブで、[トラフィック管理] **> [GSLB] > [GSLB** サービス] に移動します。
3. 次の例に示すようにサービスを追加します。
4. HTTP モニターをバインドしてサーバーのステータスを確認します。

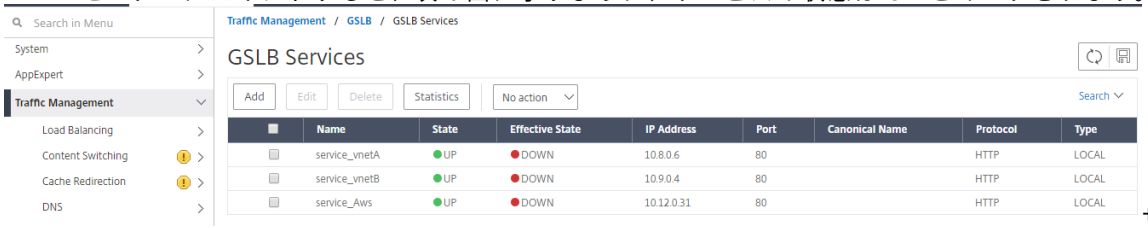


5. サービスを作成したら、GSLB サービス内の [詳細設定] タブに移動します。

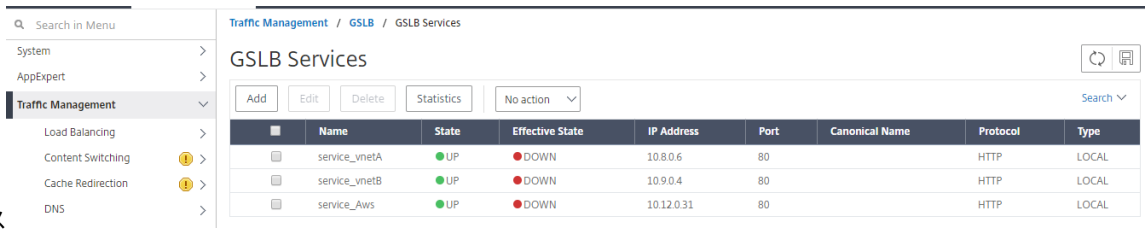
6. 「モニターを追加」をクリックして、GSLB サービスを HTTP モニターにバインドし、サービスの状態を表示します。



7. HTTP モニターにバインドすると、次の図に示すように、サービスの状態は UP とマークされます。



サ

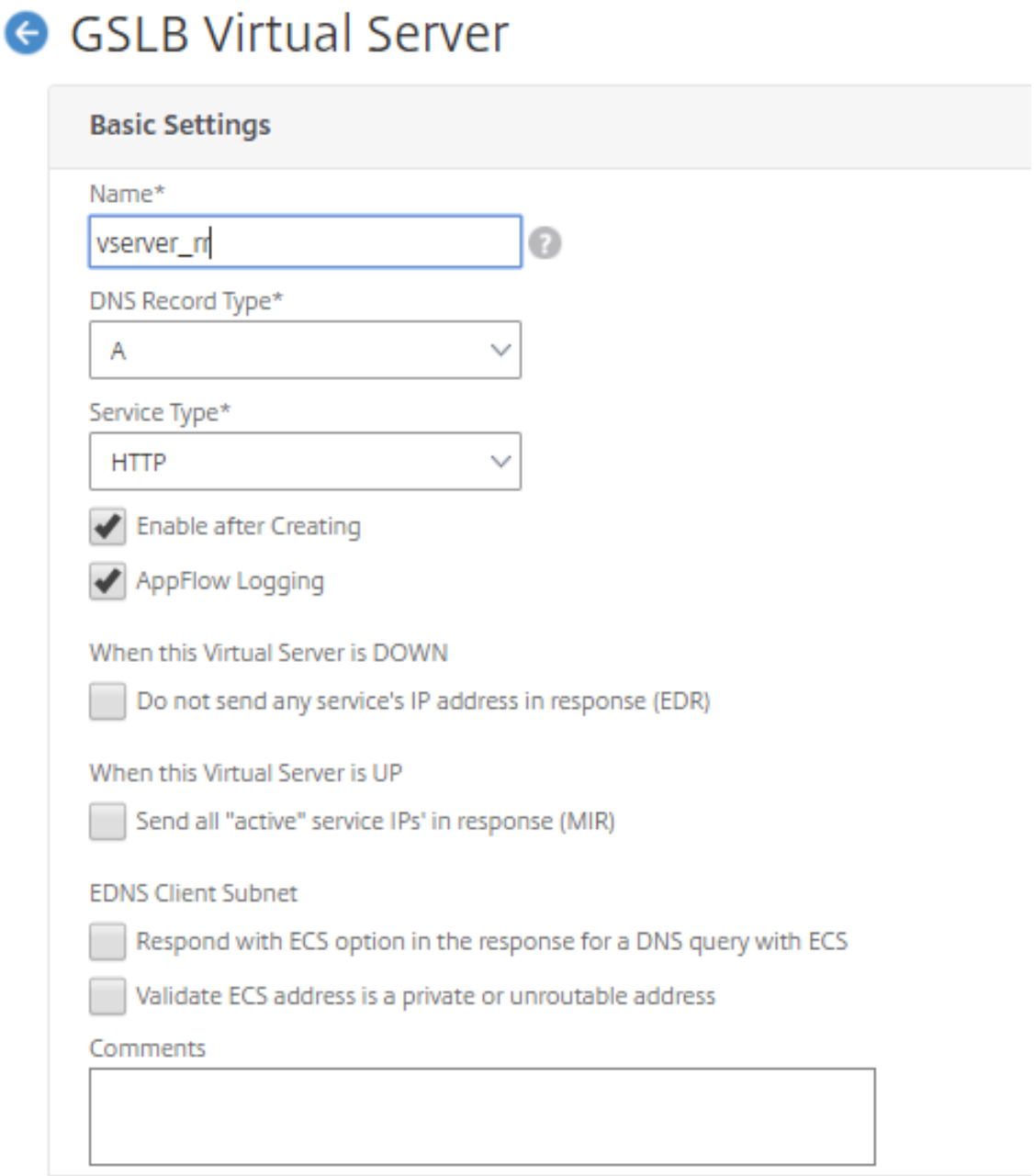


ービス

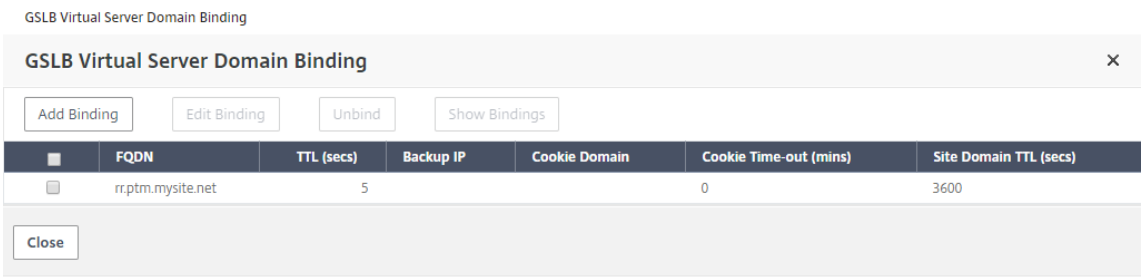
GSLB 仮想サーバーの追加

アプリケーションサーバーのエイリアス GSLB サービスにアクセスできる GSLB 仮想サーバーを追加します。

1. [設定] タブで、[トラフィック管理] **> [GSLB] > [GSLB** 仮想サーバー] に移動します。
2. 次の例のように仮想サーバーを追加します。
3. GSLB サービスとドメイン名をそれにバインドします。

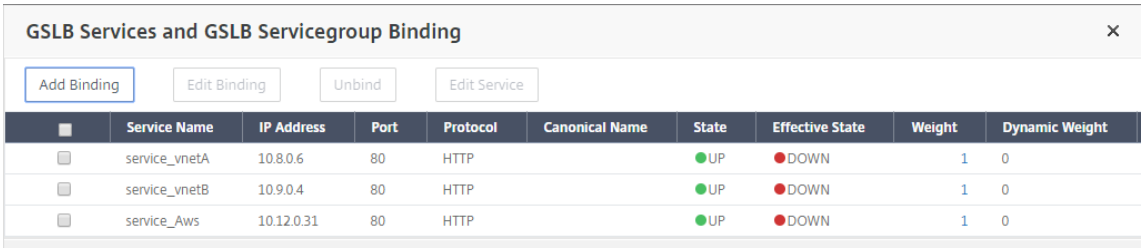


4. GSLB 仮想サーバーを作成し、適切な負荷分散方法 (この場合はラウンドロビン) を選択したら、GSLB サービスとドメインをバインドして手順を完了します。

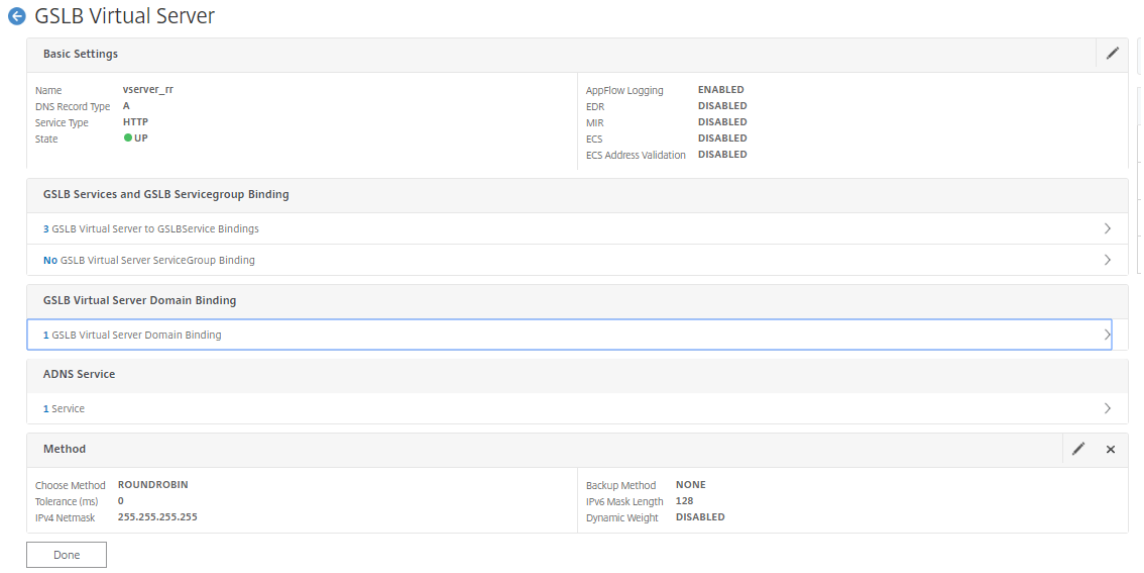


5. 仮想サーバー内の [詳細設定] タブに移動し、[ドメインの追加] タブをクリックしてドメインをバインドします。

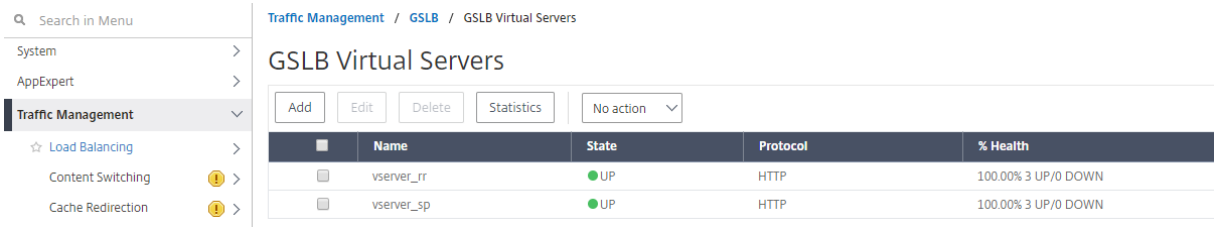
6. [詳細設定**] > [**サービス] に移動し、矢印をクリックして GSLB サービスをバインドし、3 つのサービス (VNet A、VNet B、オンプレミス) すべてを仮想サーバーにバインドします。



GSLB サービスとドメインを仮想サーバーにバインドすると、次の図のように表示されます。



GSLB 仮想サーバーが稼働していて、100% 正常かどうかを確認します。モニターにサーバーが稼働していて正常であることが示されたら、サイトが同期されており、バックエンドサービスが利用可能であることを意味します。



デプロイをテストするには、クラウドクライアントマシンまたはオンプレミスクライアントマシンからドメイン URL `rr.ptm.mysite.net`にアクセスします。クラウド Windows クライアントマシンからアクセスする場合は、サードパーティの DNS ソリューションやカスタム DNS ソリューションを必要とせずに、プライベート DNS ゾーンでオンプレミスのアプリケーションサーバーにアクセスするようにしてください。

Azure アクセラレーションネットワークを使用するように NetScaler VPX インスタンスを構成する

October 17, 2024

高速ネットワーキングにより、仮想マシンへのシングルルート I/O 仮想化 (SR-IOV) 仮想機能 (VF) NIC が有効になり、ネットワークのパフォーマンスが向上します。この機能は、信頼性の高いストリーミングと低い CPU 使用率で、より高いスループットでデータを送受信する必要がある負荷の高いワークロードで使用できます。NIC で高速ネットワークが有効になっている場合、Azure は NIC の既存の準仮想化 (PV) インターフェイスを SR-IOV VF インターフェイスとバンドルします。SR-IOV VF インターフェイスのサポートにより、NetScaler VPX インスタンスのスループットが有効になり、向上します。

高速ネットワーキングには、次の利点があります。

- 低レイテンシ
- 1 秒あたりのパケット数 (pps) のパフォーマンスが向上
- スループットの強化
- ジッタの低減
- CPU 使用率の低下

> 注
>
> Azure アクセラレーションネットワーキングは、リリース 13.0 ビルド 76.29 以降の NetScaler VPX インスタンスでサポートされています。

前提条件

- VM のサイズが Azure アクセラレーションネットワーキングの要件と一致していることを確認します。
- 任意の NIC で高速ネットワーキングを有効にする前に、VM（個別または可用性セット内）を停止します。

制限事項

高速ネットワーキングは、一部のインスタンスタイプでのみ有効にできます。詳細については、「サポートされるインスタンスタイプ」を参照してください。

高速ネットワーキングでサポートされる NIC

Azure では、ネットワークを高速化するために SR-IOV モードの Mellanox ConnectX3、ConnectX4、および ConnectX5 NIC が提供されています。

NetScaler VPX インターフェイスでアクセラレーテッドネットワーキングが有効になっている場合、Azure は ConnectX3、ConnectX4、または ConnectX5 インターフェイスのいずれかを NetScaler VPX アプライアンスの既存の PV インターフェイスにバンドルします。

仮想マシンにインターフェイスをアタッチする前に高速ネットワークを有効にする方法の詳細については、「高速ネットワークを使用したネットワークインターフェイスの作成」を参照してください。

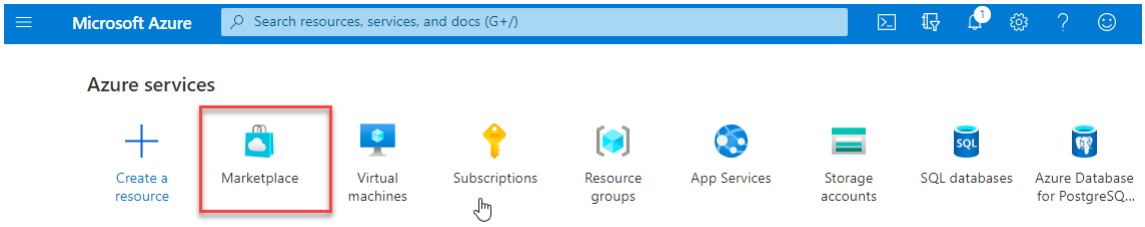
仮想マシンの既存のインターフェイスで高速ネットワーキングを有効にする方法の詳細については、「仮想マシンで既存のインターフェイスを有効にする」を参照してください。

Azure コンソールを使用して NetScaler VPX インスタンスで高速ネットワークを有効にする方法

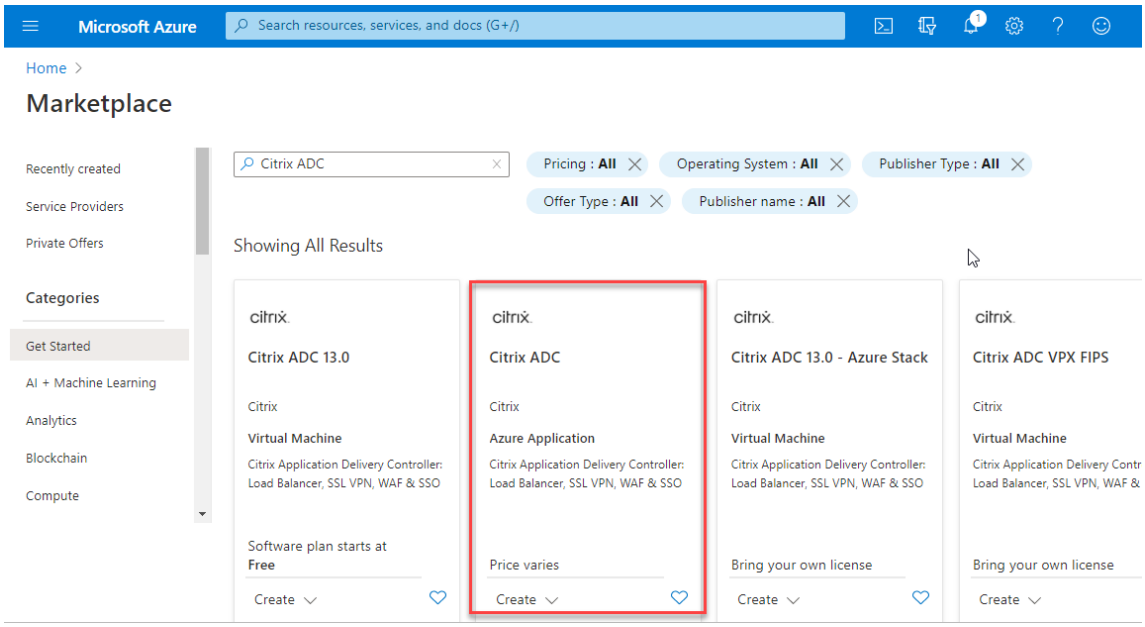
Azure コンソールまたは Azure PowerShell を使用して、特定のインターフェイスで高速ネットワークを有効にできます。

Azure のアベイラビリティセットまたはアベイラビリティーゾーンを使用して高速ネットワークを有効にするには、次の手順を実行します。

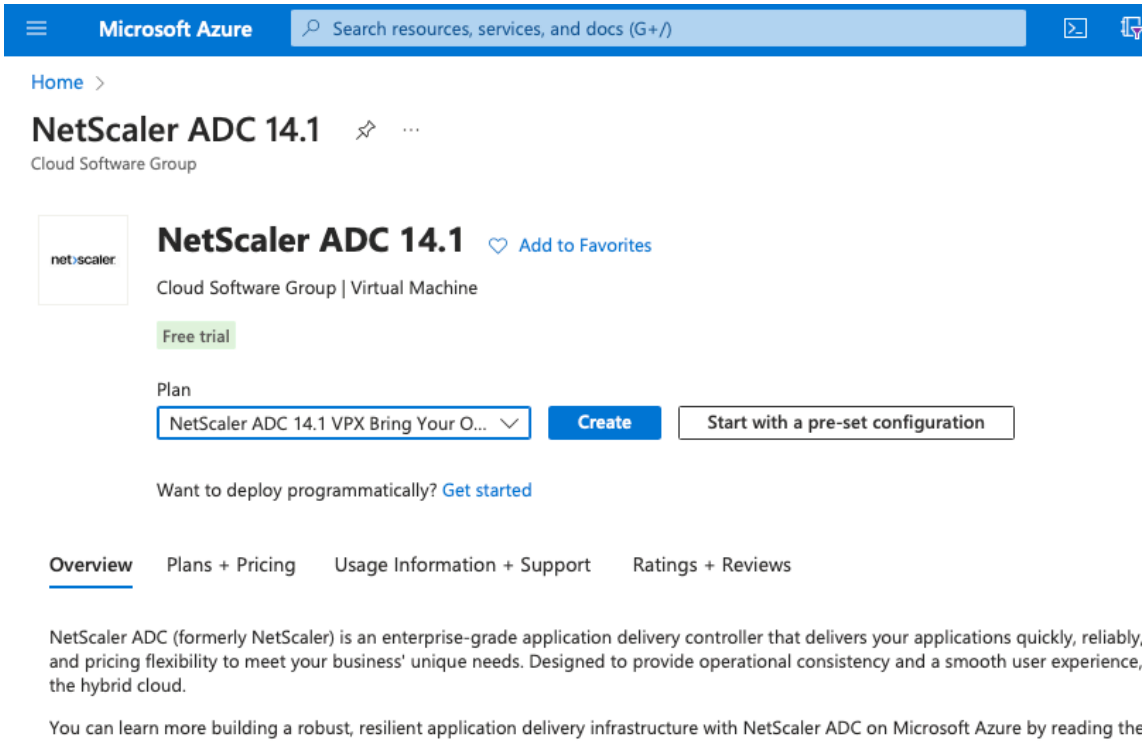
1. Azure ポータルにログインし、Azureマーケットプレイスにナビゲートします。



2. **Azure Marketplace** から **NetScaler** を検索してください。

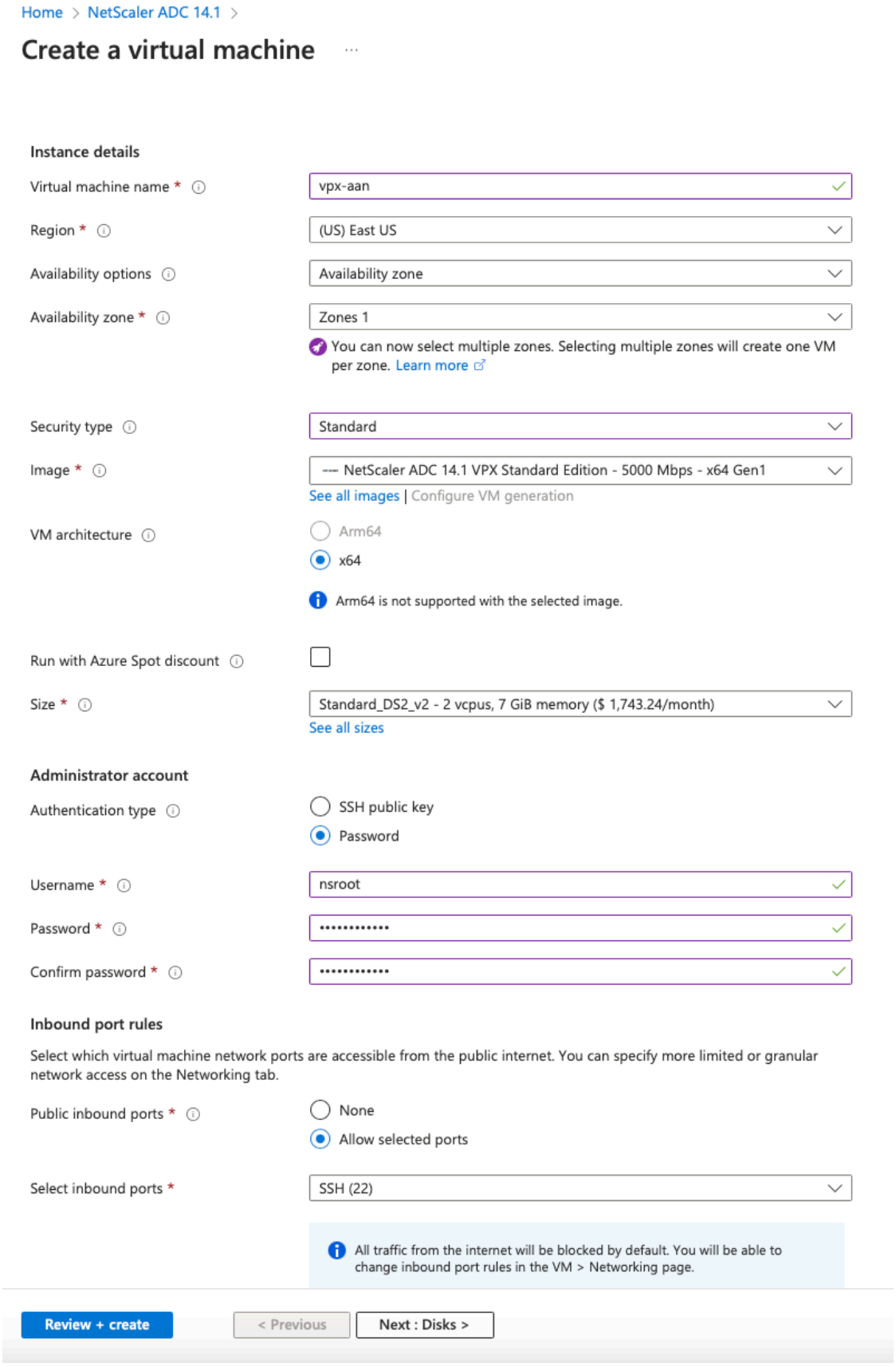


3. ライセンスとともに FIPS 以外の NetScaler プランを選択し、［作成］をクリックします。



［**NetScaler** の作成］ページが表示されます。

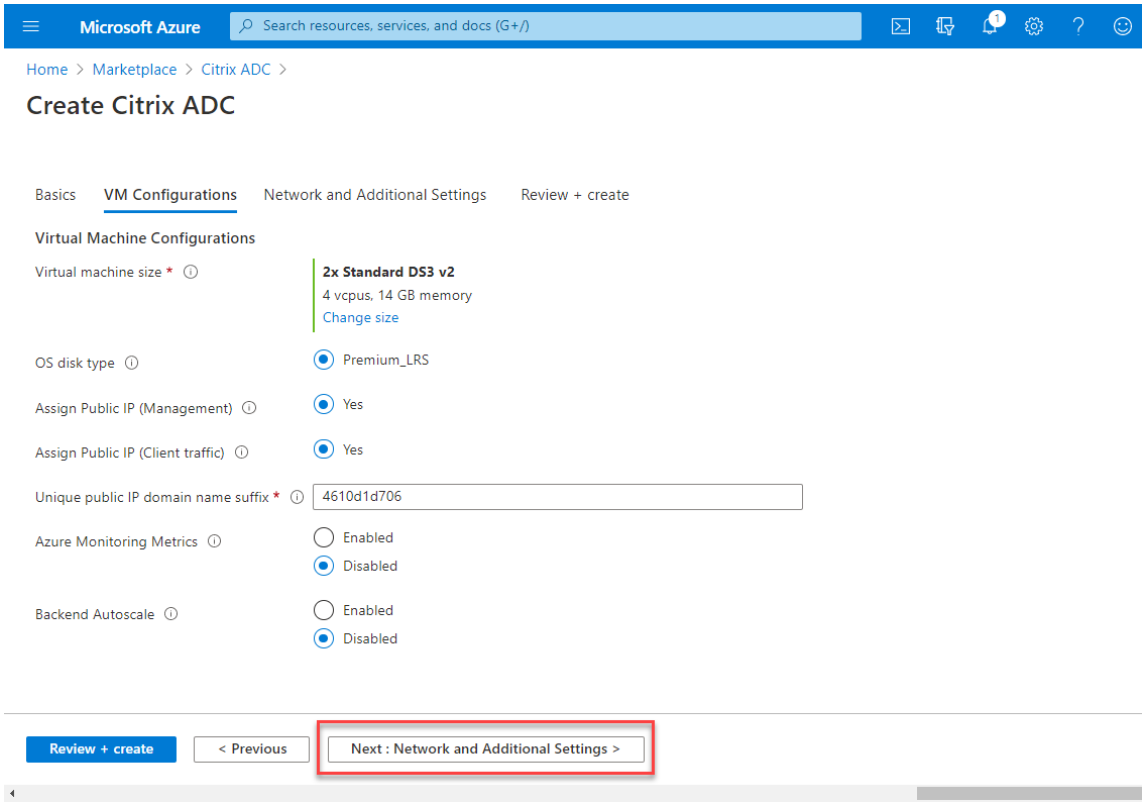
4. [基本] タブで、リソースグループを作成します。[パラメータ] タブで、地域、管理者ユーザー名、管理者パスワード、ライセンスタイプ（VM SKU）、およびその他のフィールドの詳細を入力します。



5. [次へ] をクリックします: VM 構成 > 。

   [**VM** 構成] ページで、次の手順を実行します。

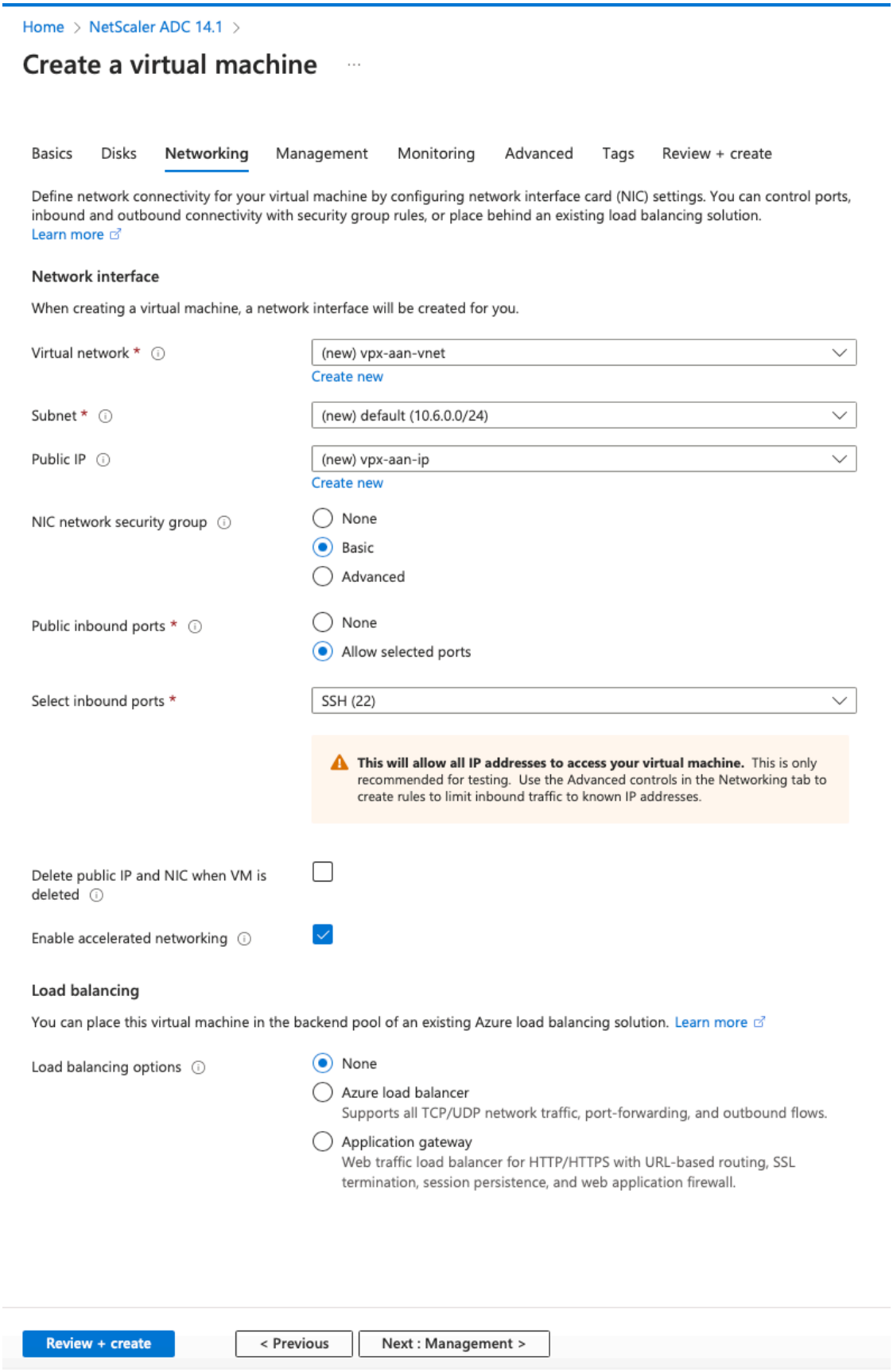
   a) パブリック IP ドメイン名サフィックスを設定します。
   b) **Azure** モニタリングメトリクスを有効または無効にします。
   c) バックエンドオートスケールを有効または無効にします。



6. 次へ**:** ネットワークと追加設定をクリックします。

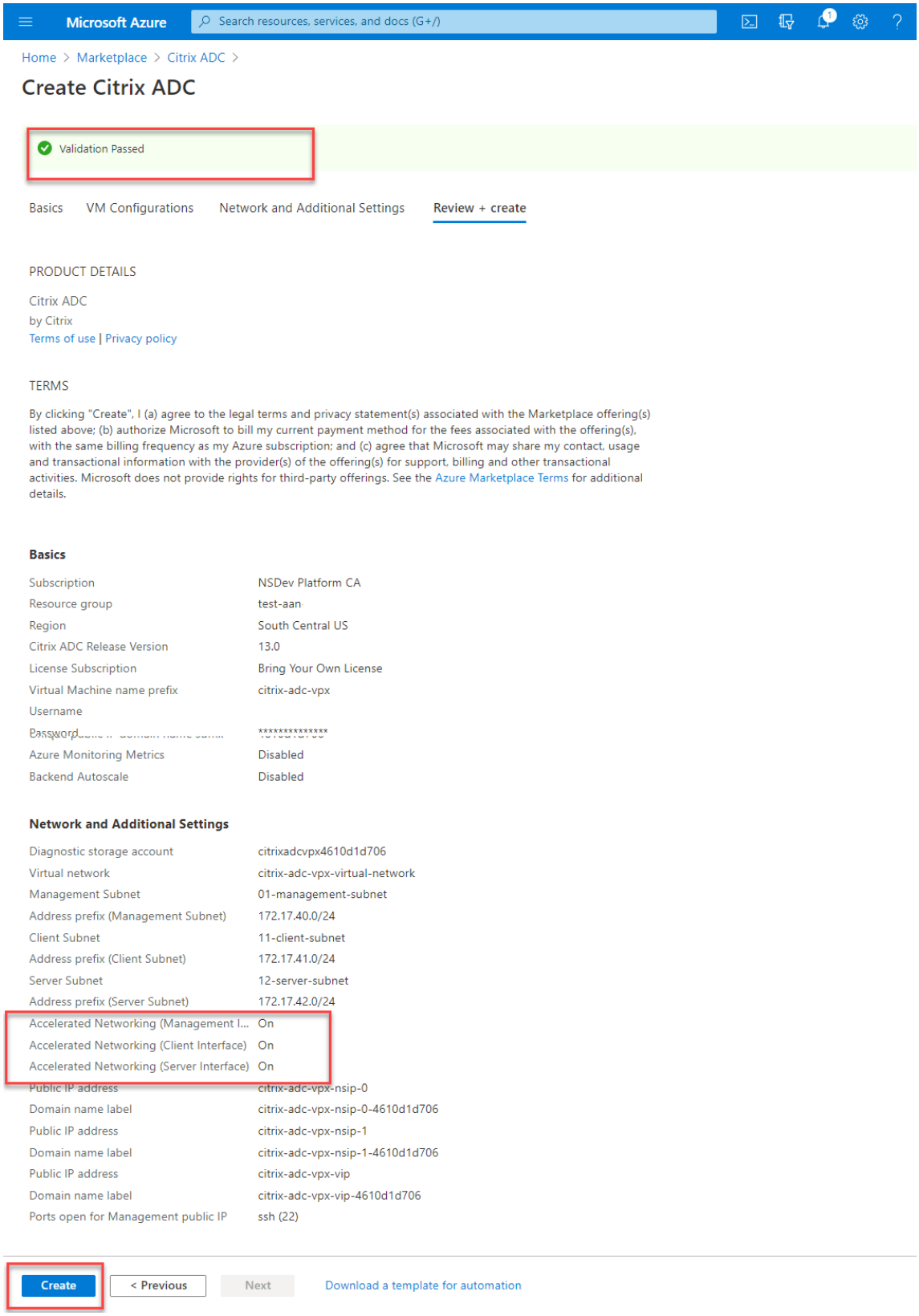
   [ネットワークとその他の設定] ページで、ブート診断アカウントを作成し、ネットワーク設定を構成します。

   [高速ネットワーキング] セクションには、管理インターフェイス、クライアントインターフェイス、およびサーバインターフェイスについて、アクセラレーションネットワーキングを個別に有効または無効にするオプションがあります。

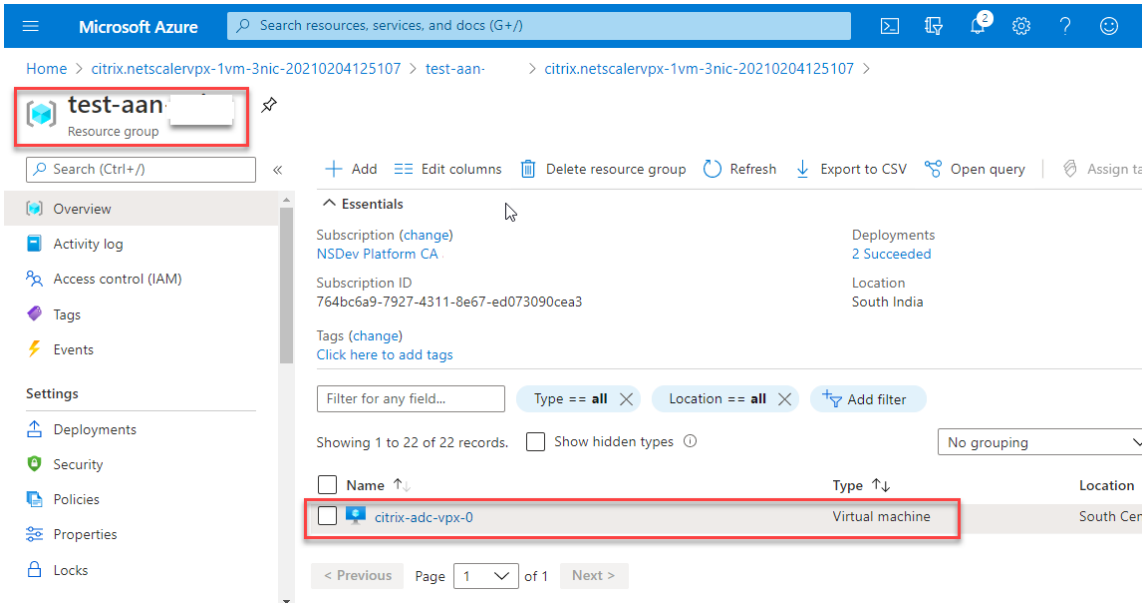


7. [次へ] をクリックします: レビュー + 作成する > 。

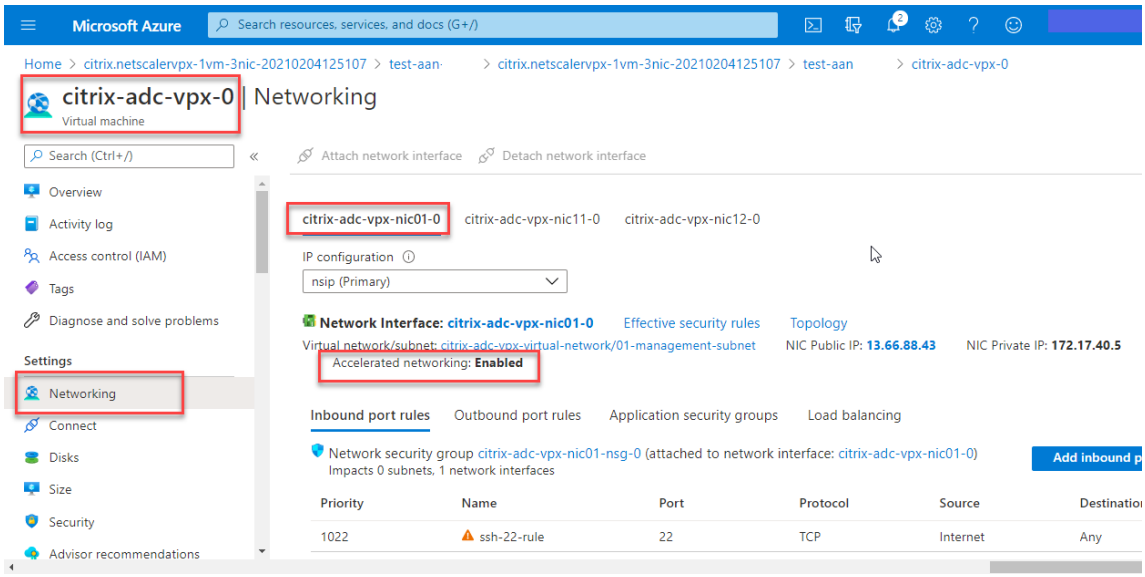
   検証が成功したら、基本設定、仮想マシンの構成、ネットワーク、および追加設定を確認し、[作成] をクリックします。Azure リソースグループが必要な構成で作成されるまでに時間がかかる場合があります。



8. デプロイが完了したら、リソースグループを選択して設定の詳細を確認します。



9. 高速ネットワーク構成を確認するには、[仮想マシン] **>** [ネットワーク] を選択します。アクセラレートネットワーキングのステータスは、NIC ごとに [有効] または [無効] と表示されます。



Azure PowerShell を使用して高速ネットワーキングを有効にする

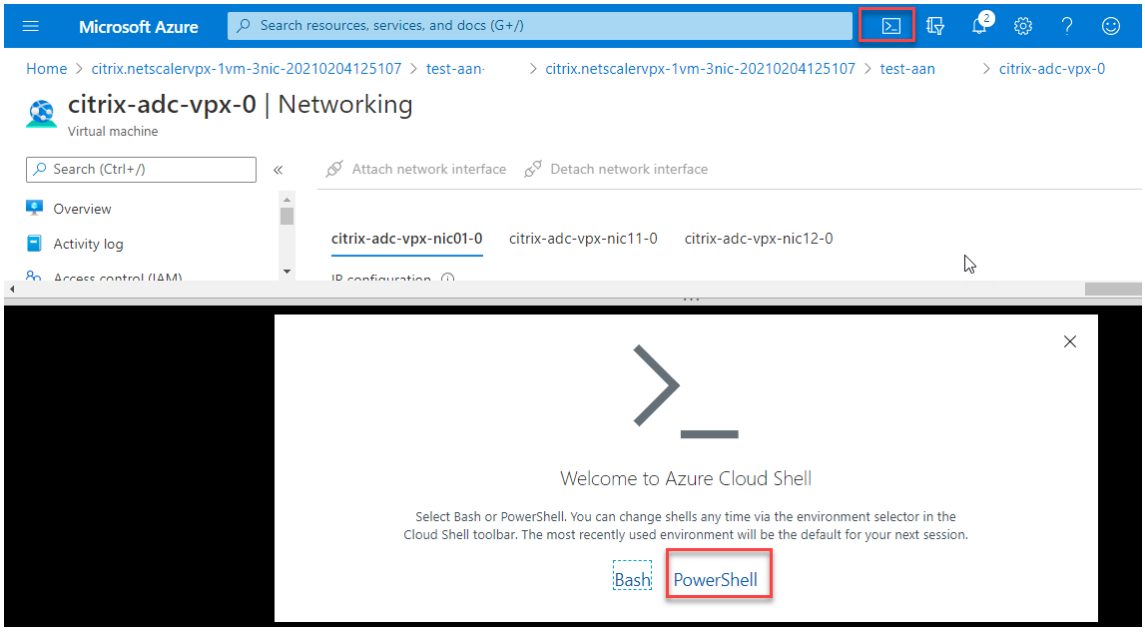
仮想マシンの作成後に高速ネットワークを有効にする必要がある場合は、Azure PowerShell を使用して有効化できます。

> 注
>
> Azure PowerShell を使用した高速ネットワーキングを有効にする前に、仮想マシンを停止してください。

Azure PowerShell を使用して高速ネットワークを有効にするには、次の手順を実行します。

1. **Azure** ポータルに移動し、右上隅にある **PowerShell** アイコンをクリックします。

   > 注
   >
   > Bash モードの場合は、PowerShell モードに切り替えます。



2. コマンドプロンプトで、次のコマンドを実行します:

```
az network nic update --name <nic-name> --accelerated-networking
    [true | false] --resource-group <resourcegroup-name>
```

アクセラレートネットワーキングパラメータは、次のいずれかの値を受け入れます。

- **True**: 指定した NIC で高速ネットワーキングを有効にします。
- **False**: 指定された NIC のアクセラレーションネットワーキングを無効にします。

特定の **NIC** で高速ネットワーキングを有効にするには、次の手順を実行します。

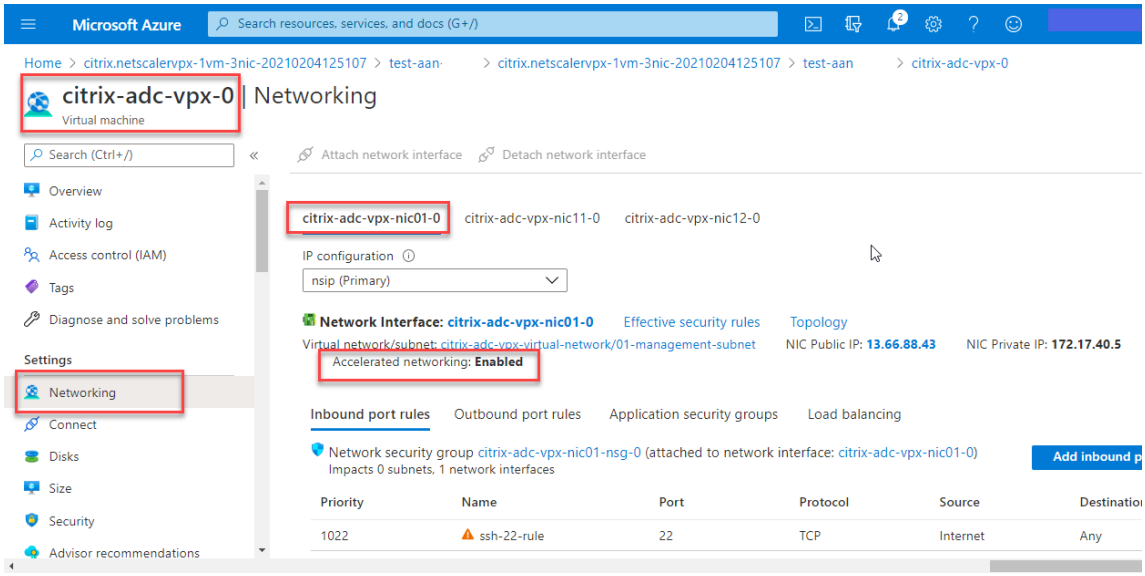
```
az network nic update --name citrix-adc-vpx-nic01-0 --
    accelerated-networking true --resource-group rsgp1-aan
```

特定の **NIC** で高速ネットワーキングを無効にするには、次の手順を実行します。

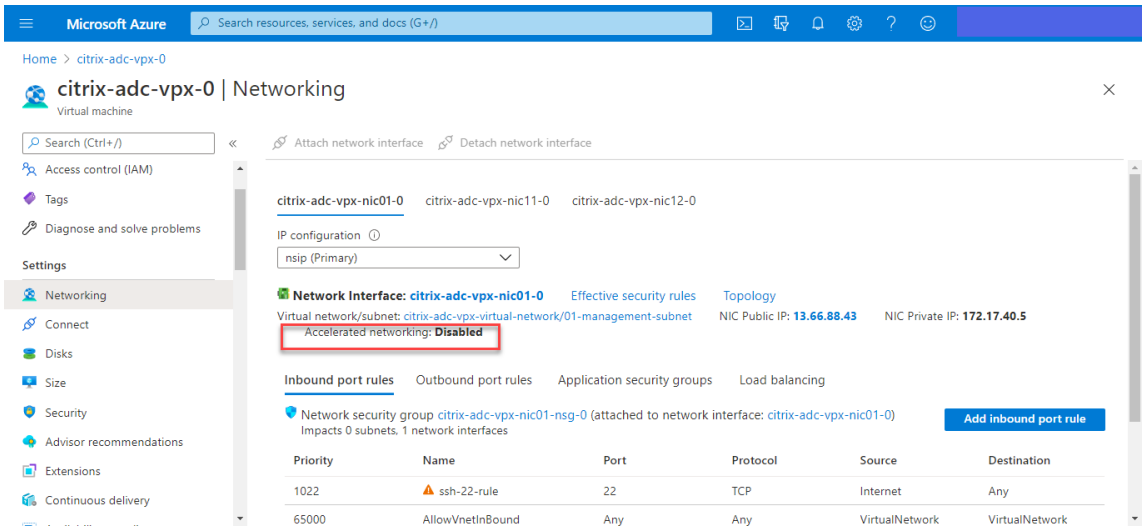
```
az network nic update --name citrix-adc-vpx-nic01-0 --
    accelerated-networking false --resource-group rsgp1-aan
```

3. デプロイが完了した後にアクセラレーテッドネットワーキングのステータスを確認するには、[**VM**] > [ネットワーク] に移動します。

   次の例では、アクセラレートネットワーキングが有効になっていることを確認します。



次の例では、アクセラレートネットワーキングが無効になっていることがわかります。



NetScaler FreeBSD Shell を使用してインターフェイス上で高速ネットワークを検証するには

NetScaler の FreeBSD シェルにログインし、次のコマンドを実行してアクセラレーションネットワークのステータスを確認できます。

ConnectX3 NIC の例**:**

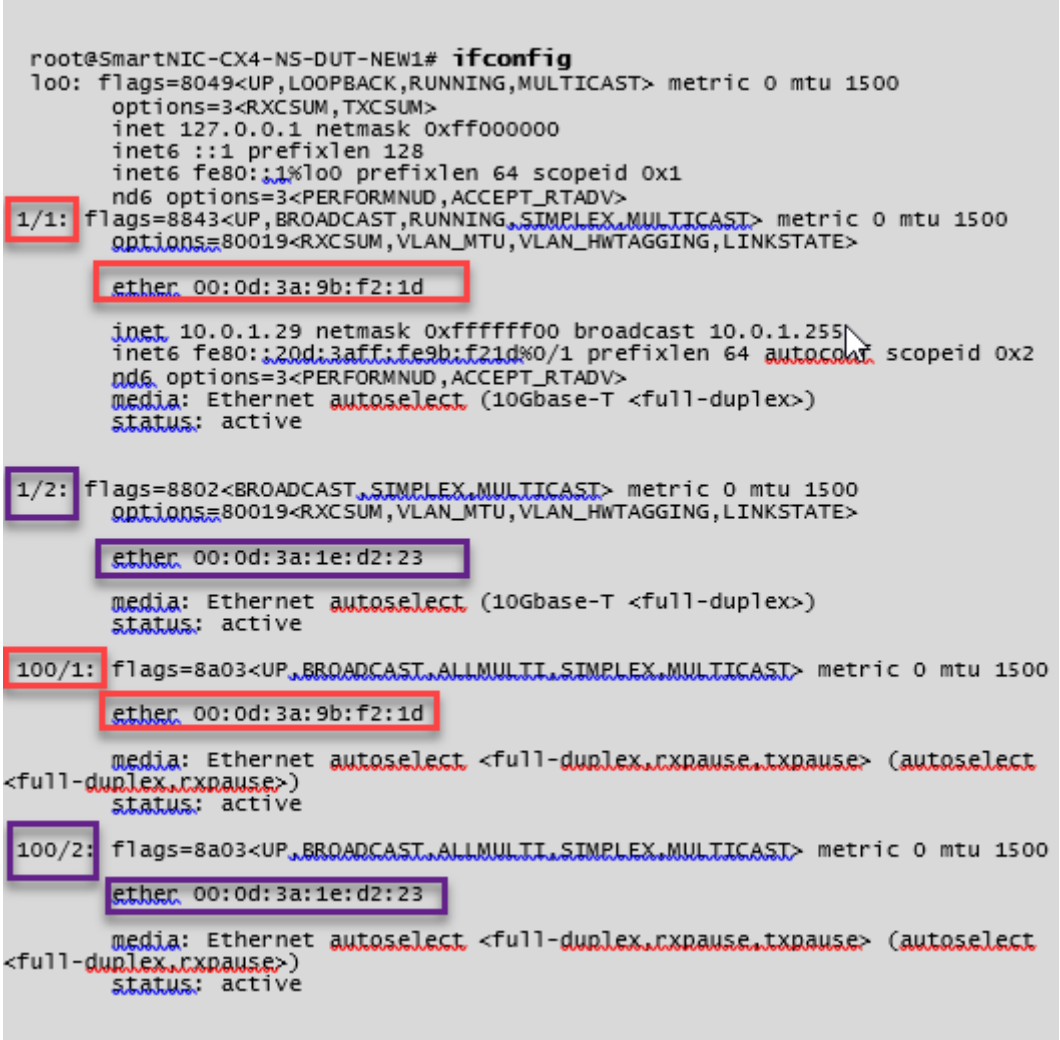
次の例は、Mellanox ConnectX3 NIC の「ifconfig」コマンド出力を示しています。「50/n」は、Mellanox ConnectX3 NIC の VF インターフェイスを示します。0/1 と 1/1 は、NetScaler VPX インスタンスの PV インターフェイスを示します。PV インターフェイス（1/1）と CX3 VF インターフェイス（50/1）の両方が同じ MAC アドレス（00:22:48:1c: 99:3 e）を持つことがわかります。これは、2 つのインターフェイスが一緒にバンドルされていることを示します。


```
root@nvr-us-cx3# ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 1500
        options=3<RXCSUM,TXCSUM>
        inet 127.0.0.1 netmask 0xff000000
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
        nd6 options=3<PERFORMNUD,ACCEPT_RTADV>
0/1: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=80019<RXCSUM,VLAN_MTU,VLAN_HWTAGGING,LINKSTATE>
        ether 00:0d:3a:98:71:be
        inet 172.16.27.11 netmask 0xffffff00 broadcast 172.16.27.255
        inet6 fe80::20d:3aff:fe98:71be%0/1 prefixlen 64 autoconf scopeid 0x2
        nd6 options=3<PERFORMNUD,ACCEPT_RTADV>
        media: Ethernet autoselect (10Gbase-T <full-duplex>)
        status: active
1/1: flags=8802<BROADCAST,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=80019<RXCSUM,VLAN_MTU,VLAN_HWTAGGING,LINKSTATE>
        ether 00:22:48:1c:99:3e
        media: Ethernet autoselect (10Gbase-T <full-duplex>)
        status: active
50/1: flags=8842<BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500

options=900b8<VLAN_MTU,VLAN_HWTAGGING,JUMBO_MTU,VLAN_HWCSUM,VLAN_HWFILTER,LINKSTATE>
        ether 00:22:48:1c:99:3e
        media: Ethernet autoselect (<unknown subtype>)
        status: active
```

ConnectX4 NIC の例**:**

次の例は、Mellanox ConnectX4 NIC の「ifconfig」コマンド出力を示しています。「100/n」は、Mellanox ConnectX4 NIC の VF インターフェイスを示します。0/1、1/1、および 1/2 は、NetScaler VPX インスタンスの PV インターフェイスを示します。PV インターフェイス（1/1）と CX4 VF インターフェイス（100/1）の両方が同じ MAC アドレス（00:0 d: 3a: 9b: f 2:1 d）を持つことがわかります。PV インターフェイス (1/1) と CX4 VF インターフェイス (100/1) の両方に同じ MAC アドレス (00:0d:3a:9b:f2:1d) があることがわかります。これは、2 つのインターフェイスが一緒にバンドルされていることを示します。同様に、PV インターフェイス（1/2）と CX4 VF イン

ターフェイス（100/2）は同じ MAC アドレス（00:0 d: 3a: 1:D 2:23）を持ちます。



ADC CLI を使用してインターフェイスで高速ネットワーキングを検証するには

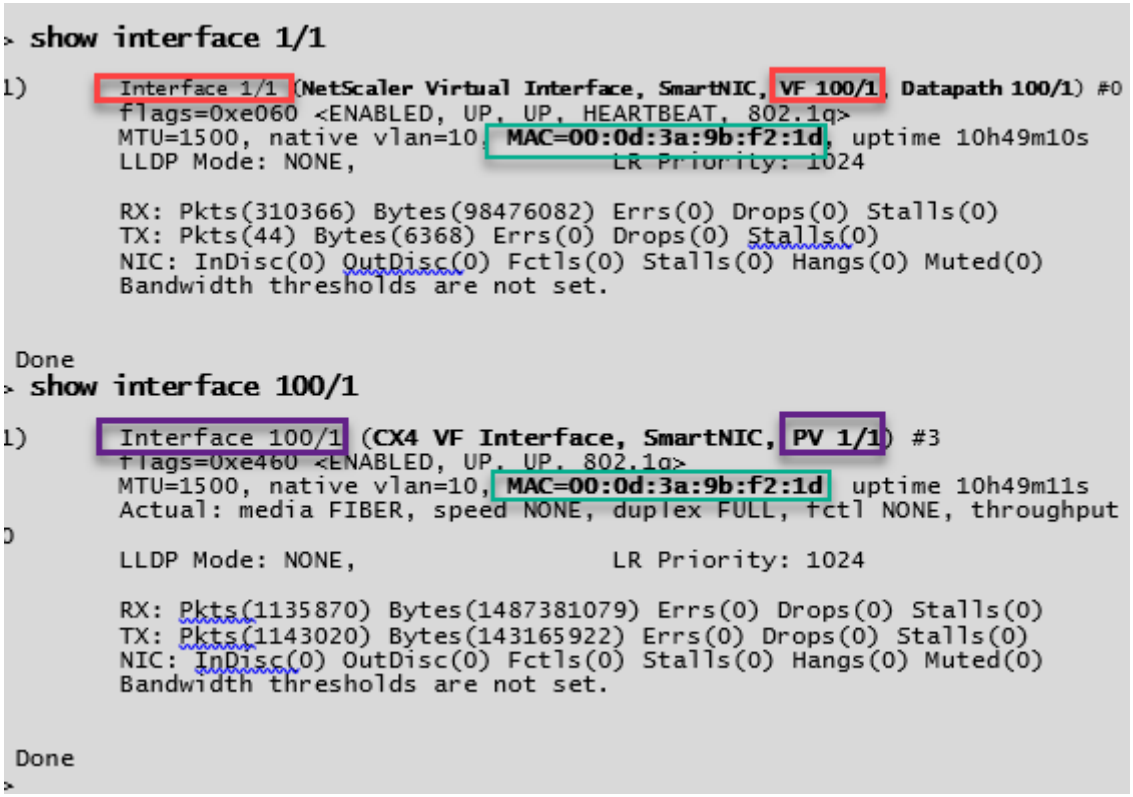
ConnectX3 NIC の例**:**

次の show interface コマンドの出力は、PV インターフェイス 1/1 が SR-IOV VF NIC である仮想機能 50/1 にバンドルされていることを示しています。1/1 と 50/1 の NIC の両方の MAC アドレスは同じです。高速ネットワーキングが有効になると、1/1 インターフェイスのデータは、ConnectX3 インターフェイスである 50/1 インターフェイスのデータパスを介して送信されます。PV インターフェイス (1/1) の「show interface」出力が VF (50/1) を指していることがわかります。同様に、VF インターフェイス (50/1) の「show interface」出力は PV インターフェイス (1/1) を指します。


```
> show interface 1/1

        Interface 1/1 (NetScaler Virtual Interface, SmartNIC, VF 50/1 Datapath 50/1) #1
        flags=0xe060 <ENABLED, UP, UP, HEARTBEAT, 802.1q>
        MTU=1500, native vlan=1, MAC=00:22:48:1c:99:3e, uptime 0h00m07s
        LLDP Mode: NONE,                LR Priority: 1024

        RX: Pkts(0) Bytes(0) Errs(0) Drops(0) Stalls(0)
        TX: Pkts(0) Bytes(0) Errs(0) Drops(0) Stalls(0)
        NIC: InDisc(0) OutDisc(0) Fctls(0) Stalls(0) Hangs(0) Muted(0)
        Bandwidth thresholds are not set.

Done

> show interface 50/1

      Interface 50/1 (CX3 VF Interface, SmartNIC, PV 1/1) #2
        flags=0xe460 <ENABLED, UP, UP, 802.1q>
        MTU=1500, native vlan=1, MAC=00:22:48:1c:99:3e, uptime 0h00m08s
        Actual: media NONE, speed 50000, duplex FULL, fctl NONE, throughput 50000
        LLDP Mode: NONE,                LR Priority: 1024

        RX: Pkts(0) Bytes(0) Errs(0) Drops(0) Stalls(0)
        TX: Pkts(0) Bytes(0) Errs(0) Drops(0) Stalls(0)
        NIC: InDisc(0) OutDisc(0) Fctls(0) Stalls(0) Hangs(0) Muted(0)
        Bandwidth thresholds are not set.
```

ConnectX4 NIC の例**:**

次の show interface コマンドの出力は、PV インターフェイス 1/1 が SR-IOV VF NIC である仮想機能 100/1 にバンドルされていることを示しています。1/1 と 100/1 の NIC の両方の MAC アドレスは同じです。高速ネットワーキングが有効になると、1/1 インターフェイスのデータは、ConnectX4 インターフェイスである 100/1 インターフェイスのデータパスを介して送信されます。PV インターフェイス (1/1) の「show interface」出力が VF (100/1) を指すことがわかります。同様に、VF インターフェイス (100/1) の「show interface」出力は PV インターフェイス (1/1) を指します。



NetScaler での注意点

- PV インターフェイスは、必要なすべての操作のプライマリインターフェイスまたはメインインターフェイスと見なされます。設定は PV インターフェイスでのみ実行する必要があります。
- VF インターフェイスでのすべての「set」操作は、以下を除いてブロックされます。
  - インターフェイスを有効にする
  - インターフェイスを無効にする
  - インターフェイスをリセット
  - 統計をクリアする

  > 注
  >
  > VF インターフェイスでは操作を実行しないことをお勧めします。

- `show interface`コマンドを使用して、PV インターフェイスと VF インターフェイスとのバインディングを確認できます。
- NetScaler リリース 13.1～33.x 以降、NetScaler VPX インスタンスは、Azure アクセラレーテッドネットワークでの動的な NIC の取り外しと、取り外された NIC の再接続をシームレスに処理できます。Azure は、ホストのメンテナンス作業のために、高速ネットワークの SR-IOV VF NIC を削除できます。NIC が Azure VM から削除されると、NetScaler VPX インスタンスのインターフェイスステータスが「リンクダウン」と表示さ

れ、トラフィックは仮想インターフェイスのみを経由します。取り外した NIC が再接続されると、VPX インスタンスは再接続された SR-IOV VF NIC を使用します。このプロセスはシームレスに行われ、設定は不要です。

PV インターフェースへの VLAN の構成

PV インターフェイスが VLAN にバインドされている場合、関連するアクセラレーション VF インターフェイスも PV インターフェイスと同じ VLAN にバインドされます。この例では、PV インターフェイス（1/1）は VLAN（20）にバインドされています。PV インターフェイス（1/1）にバンドルされている VF インターフェイス（100/1）も VLAN 20 にバインドされます。

例

1. VLAN を作成します。

```
add vlan 20
```

2. VLAN を PV インターフェイスにバインドします。

```
bind vlan 20  - ifnum 1/1

show vlan

1)  VLAN ID: 1
    Link-local IPv6 addr: fe80::20d:3aff:fe9b:f21d/64
    Interfaces : LO/1

2)  VLAN ID: 10     VLAN Alias Name:
    Interfaces : 0/1 100/1
    IPs : 10.0.1.29  Mask: 255.255.255.0

3)  VLAN ID: 20     VLAN Alias Name:
    Interfaces :  1/1 100/2
```

> 注
>
> VLAN バインディング操作は、アクセラレーション VF インターフェイスでは許可されません。

```
bind vlan 1 -ifnum 100/1
ERROR: Operation not permitted
```

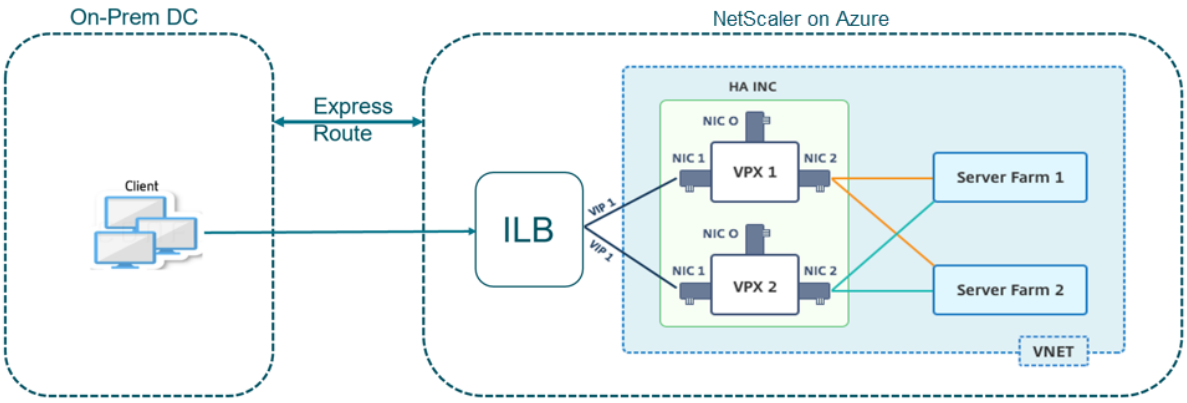
Azure ILB で NetScaler 高可用性テンプレートを使用して HA-INC ノードを構成する

October 17, 2024

イントラネットアプリケーション用の標準テンプレートを使用すると、HA-INC モードで一対の VPX インスタンスを迅速かつ効率的にデプロイできます。Azure 内部ロードバランサー (ILB) は、図 1 に示すように、フロントエンドに

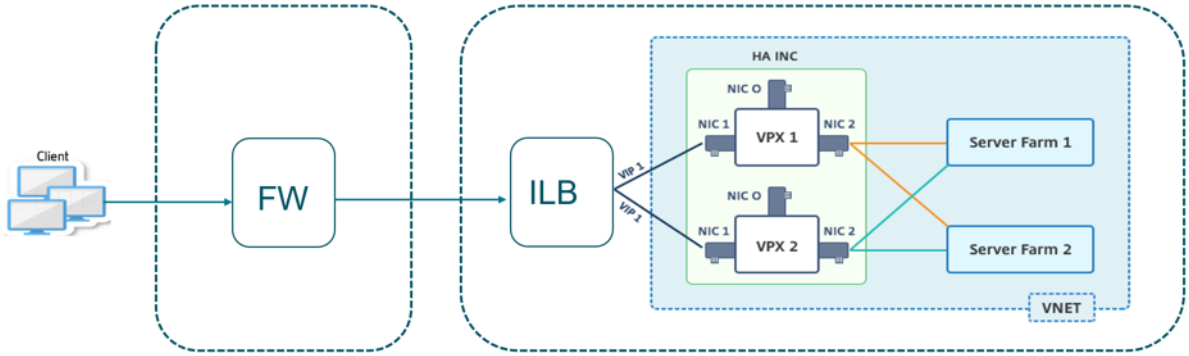
内部 IP アドレスまたはプライベート IP アドレスを使用します。このテンプレートでは、3 つのサブネットと 6 つの NIC を持つ 2 つのノードが作成されます。サブネットは、管理、クライアント、およびサーバー側のトラフィック用で、各サブネットはデバイスごとに異なる NIC に属します。

図 1: 内部ネットワーク内のクライアント用の NetScaler ADC HA ペア



この展開は、図 2 に示すように、NetScaler HA ペアがファイアウォールの内側にある場合にも使用できます。パブリック IP アドレスはファイアウォールに属し、ILB のフロントエンド IP アドレスに NAT されます。

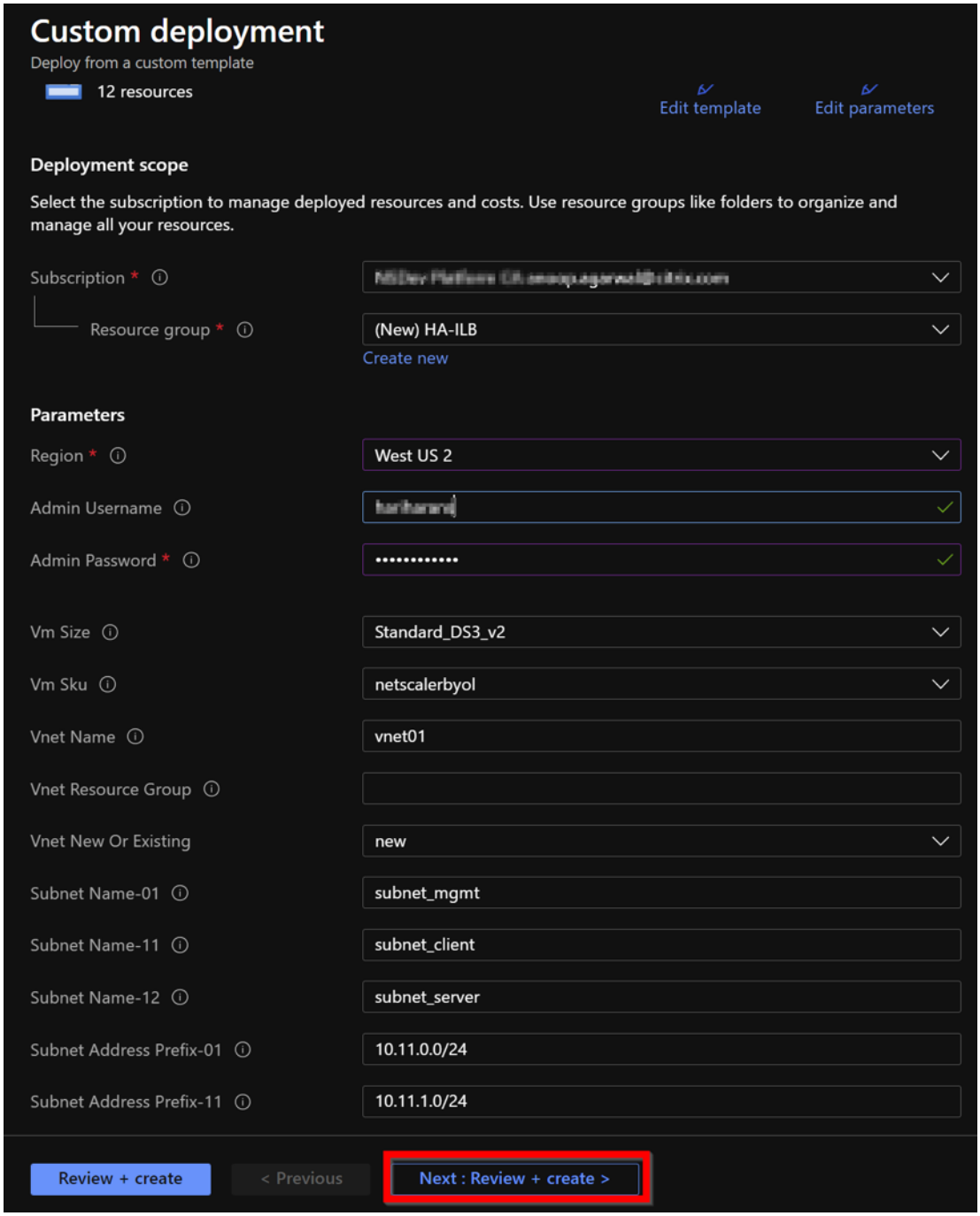
図 2: パブリック IP アドレスを持つファイアウォールと NetScaler ADC HA のペア



イントラネットアプリケーション用の NetScaler ADC HA ペアテンプレートは、Azure ポータルで入手できます。

次の手順を実行してテンプレートを起動し、Azure 可用性セットを使用して高可用性 VPX ペアをデプロイします。

1. Azure Portal から、[カスタム展開] ページに移動します。
2. [基本] ページが表示されます。リソースグループを作成します。[パラメータ] タブで、[リージョン]、[管理者ユーザー名]、[管理者パスワード]、[ライセンスタイプ] (VM sku)、およびその他のフィールドの詳細を入力します。

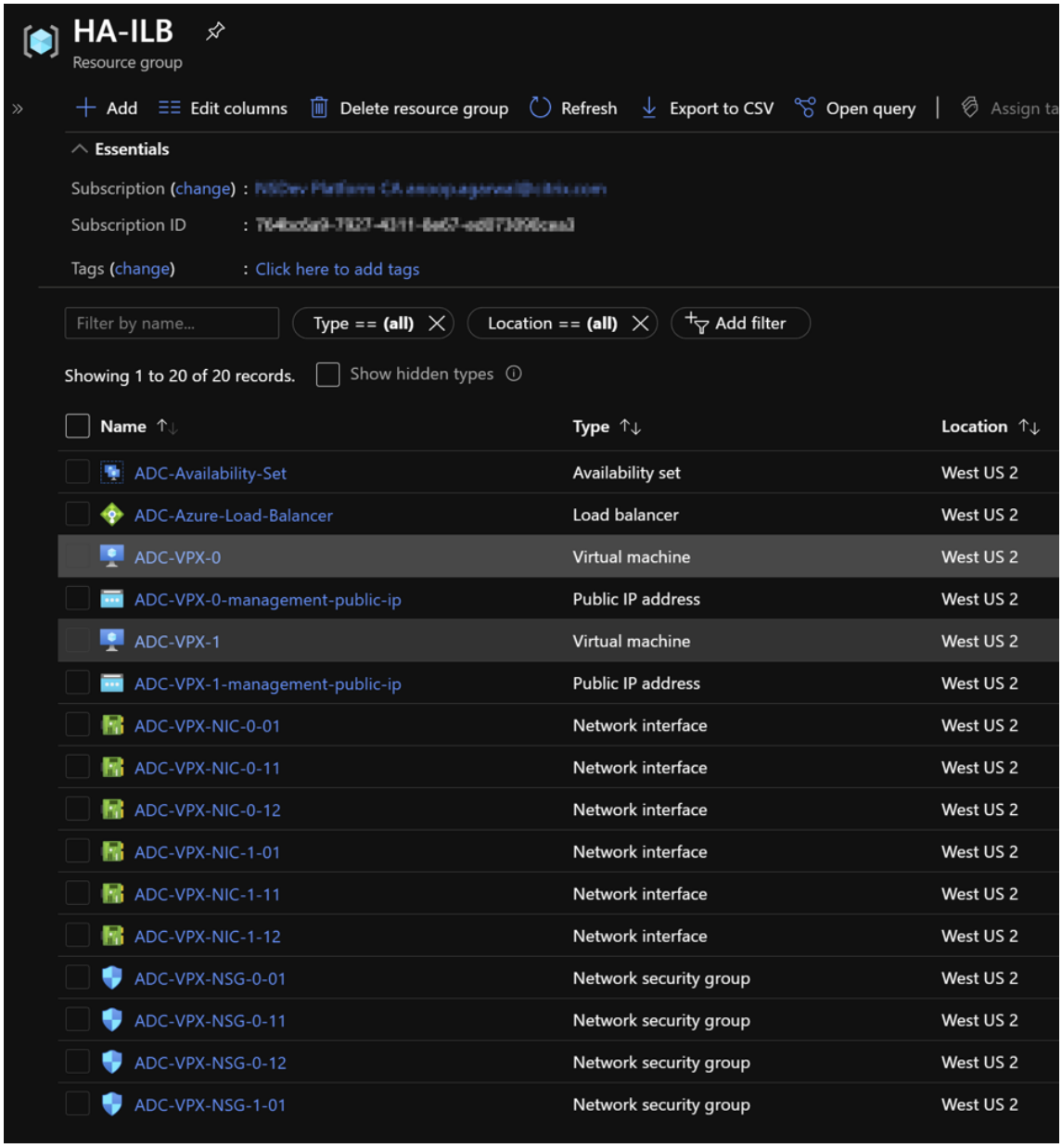


3. 次へ**:** 確認 **+** 作成 **>** をクリックします。

   必要な構成で Azure リソースグループが作成されるまで時間がかかることがあります。完了後、Azure ポータルでリソースグループを選択して、LB ルール、バックエンドプール、ヘルスプローブなどの構成の詳細を確認します。高可用性ペアは ADC-VPX-0 と ADC-VPX-1 として表示されます。

   追加のセキュリティルールやポートを作成するなど、HA セットアップでさらに変更が必要な場合は、Azure Portal から実行できます。

必要な構成が完了すると、次のリソースが作成されます。



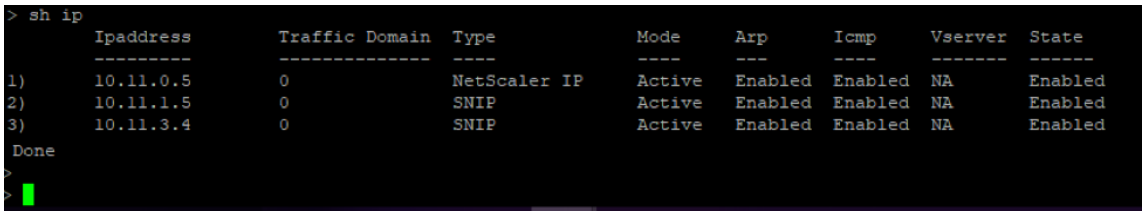
4. **ADC-VPX-0** および **ADC-VPX-1** ノードにログオンして、次の設定を検証します。

   - 両方のノードの NSIP アドレスは管理サブネットに存在する必要があります。
   - プライマリ（ADC-VPX-0）ノードとセカンダリ（ADC-VPX-1）ノードには、2 つの SNIP アドレスが表示される必要があります。一方の SNIP（クライアントサブネット）は ILB プローブへの応答に使用され、もう 1 つの SNIP（サーバーサブネット）はバックエンドサーバー通信に使用されます。

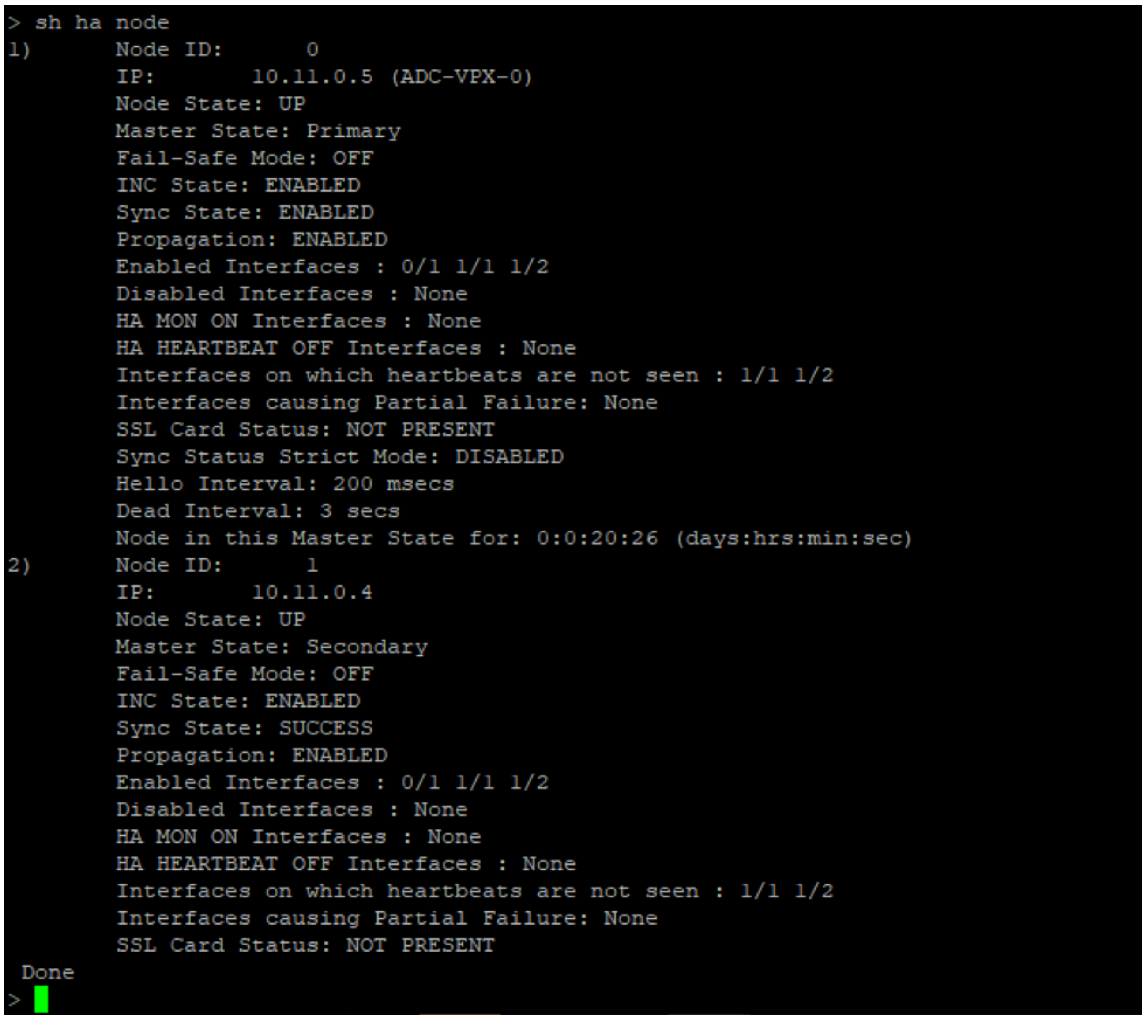
   > 注
   >
   > HA-INC モードでは、ADC-VPX-0VM と ADC-VPX-1VM の SNIP アドレスは、両方が同じである従来のオンプレミス ADC HA 展開とは異なり、同じサブネット内では異なります。VPX ペア SNIP が異なるサ

> ブネットにある場合、または VIP が SNIP と同じサブネット内にない場合に展開をサポートするには、Mac ベース転送（MBF）を有効にするか、各 VIP の静的ホストルートを各 VPX ノードに追加する必要があります。VPX ペアの SNIP が異なるサブネットにある場合、または VIP が SNIP と同じサブネットにない場合に展開をサポートするには、Mac ベース転送 (MBF) を有効にするか、各 VIP の静的ホストルートを各 VPX ノードに追加する必要があります。

プライマリノード（ADC-VPX-0）



```
> sh ip
        Ipaddress        Traffic Domain  Type             Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----             ----     ---      ----     -------  ------
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
>
>
```



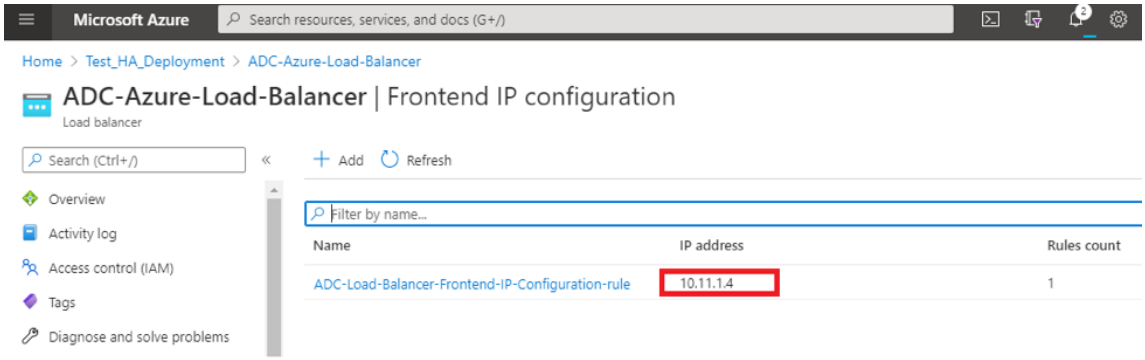
```
> sh ha node
1)      Node ID:      0
        IP:       10.11.0.5 (ADC-VPX-0)
        Node State: UP
        Master State: Primary
        Fail-Safe Mode: OFF
        INC State: ENABLED
        Sync State: ENABLED
        Propagation: ENABLED
        Enabled Interfaces : 0/1 1/1 1/2
        Disabled 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 msecs
        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
>
```

セカンダリノード（ADC-VPX-1）

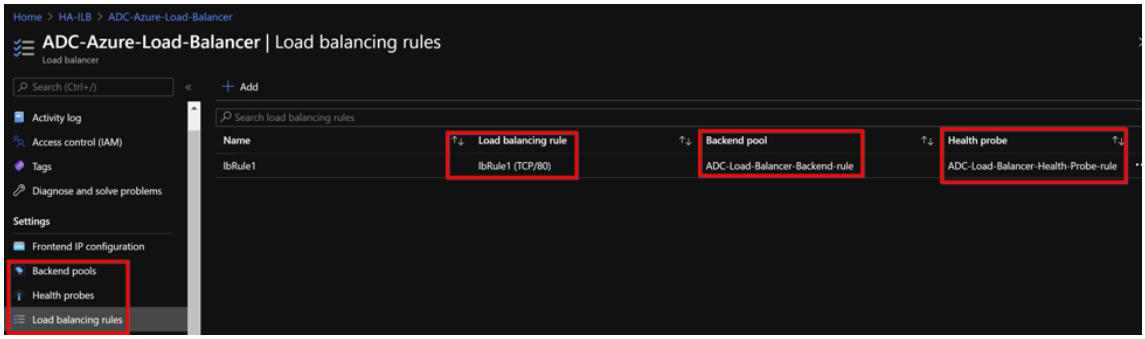

```
> sh ip
        Ipaddress        Traffic Domain  Type             Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----             ----     ---      ----     -------  ------
1)      10.11.0.4        0               NetScaler IP     Active   Enabled  Enabled  NA       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 msecs
        Dead Interval: 3 secs
        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
>
```

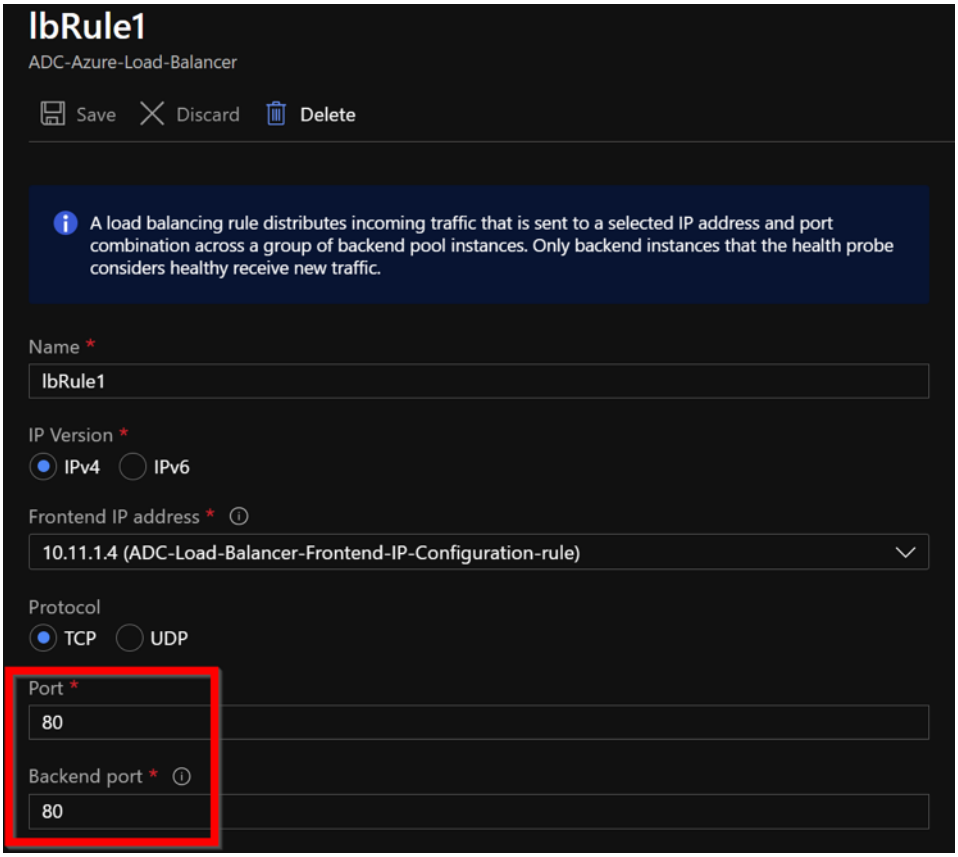
5. プライマリノードとセカンダリノードが UP で、同期ステータスが **SUCCESS** になったら、プライマリノード (ADC-VPX-0) の負荷分散仮想サーバーまたはゲートウェイ仮想サーバーを、ADC Azure ロードバランサーのプライベートフローティング IP (FIP) アドレスで構成する必要があります。詳細については、「サンプル設定」セクションを参照してください。
6. ADC Azure 負荷分散サーバーのプライベート IP アドレスを見つけるには、**Azure portal > ADC Azure Load Balancer > Frontend IP configuration** に移動します。



7. **Azure Load Balancer** の 構成ページで、ARM テンプレートの展開は、LB ルール、バックエンドプール、およびヘルスプローブの作成に役立ちます。

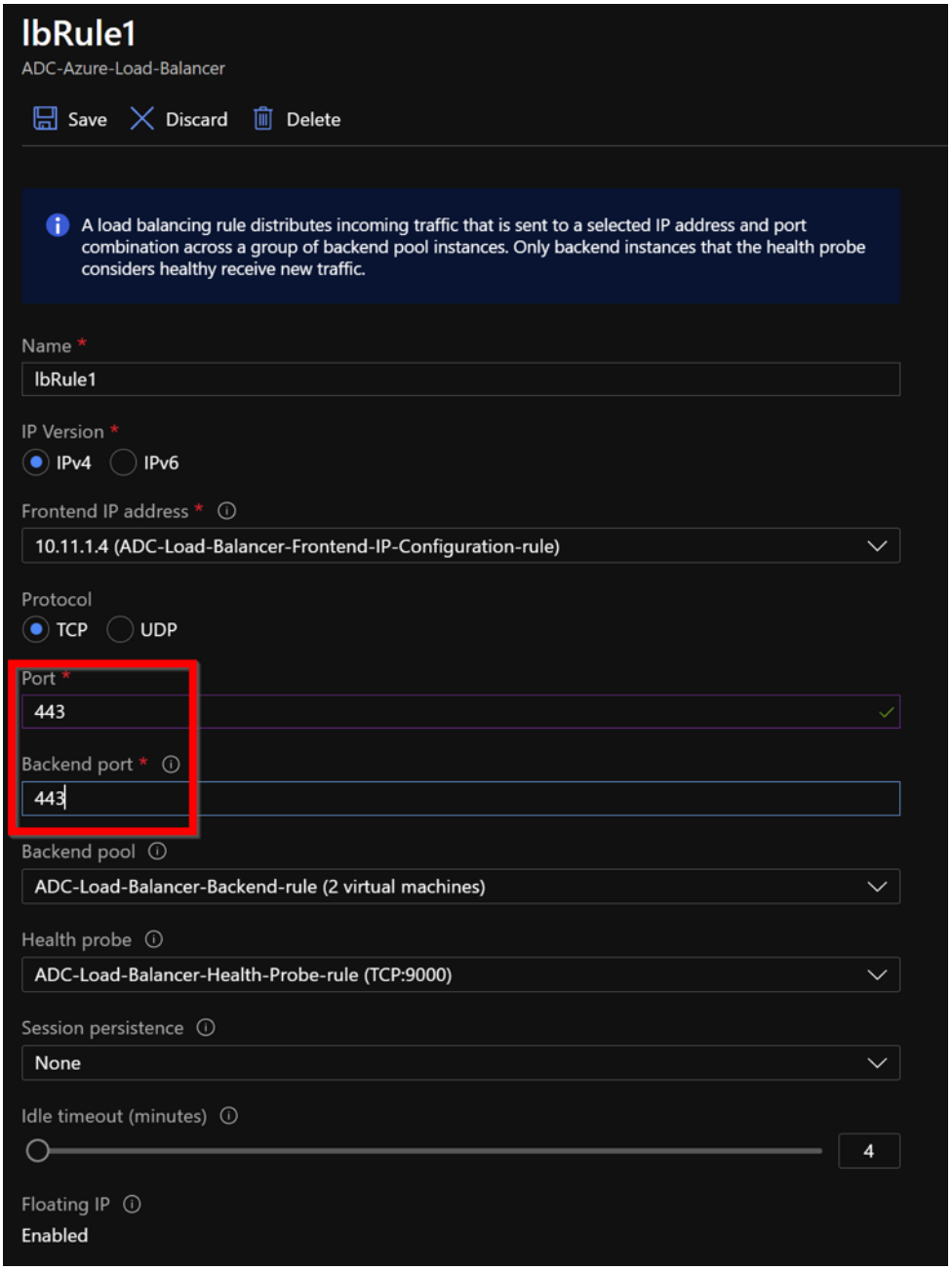


- LB ルール (lbrule1) はデフォルトでポート 80 を使用します。



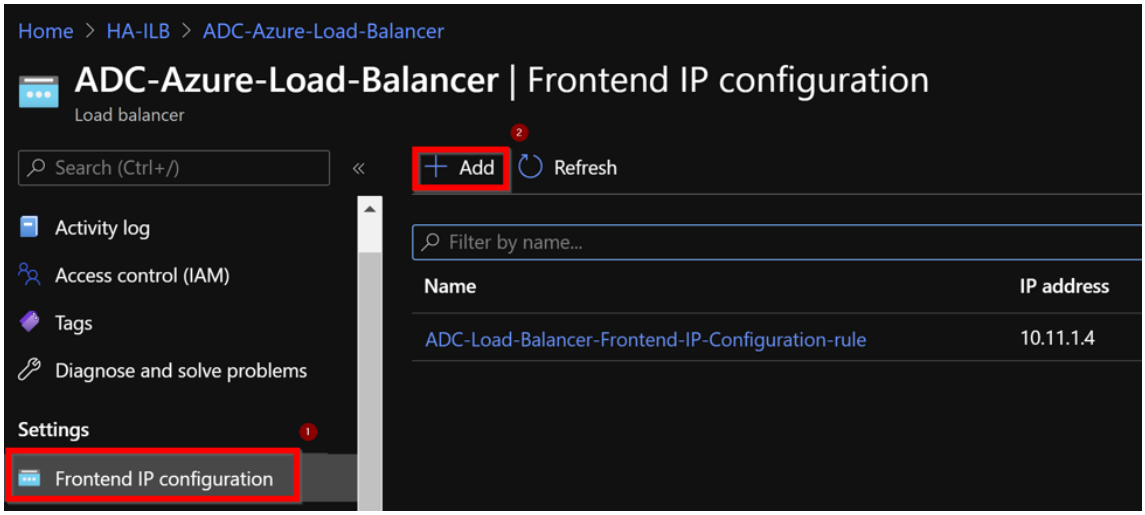
- ポート 443 を使用するようにルールを編集し、変更を保存します。

  > 注
  >
  > セキュリティを強化するため、LB 仮想サーバーまたはゲートウェイ仮想サーバーには SSL ポート 443 を使用することをお勧めします。

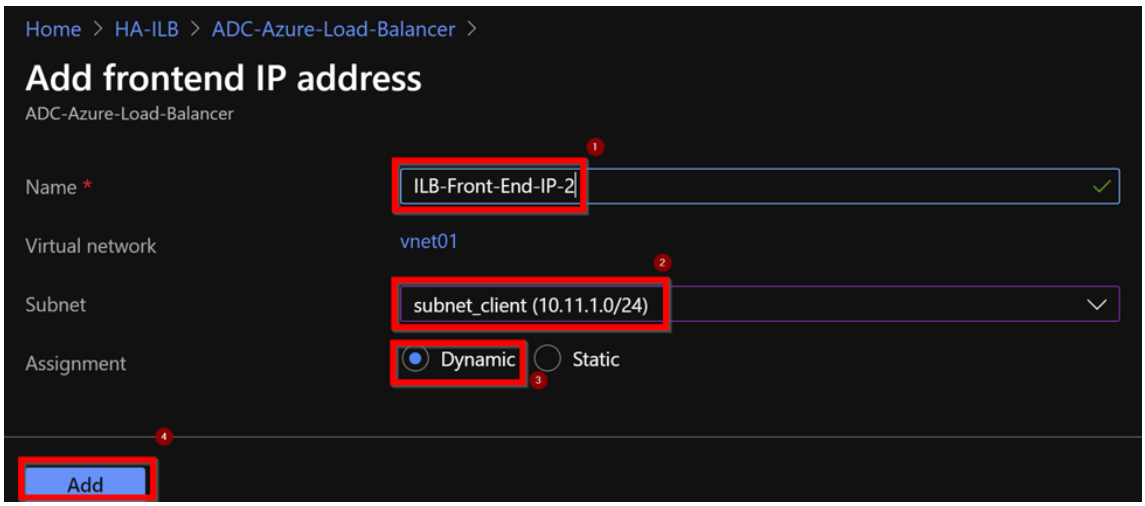


ADC に VIP アドレスを追加するには、次の手順に従います。

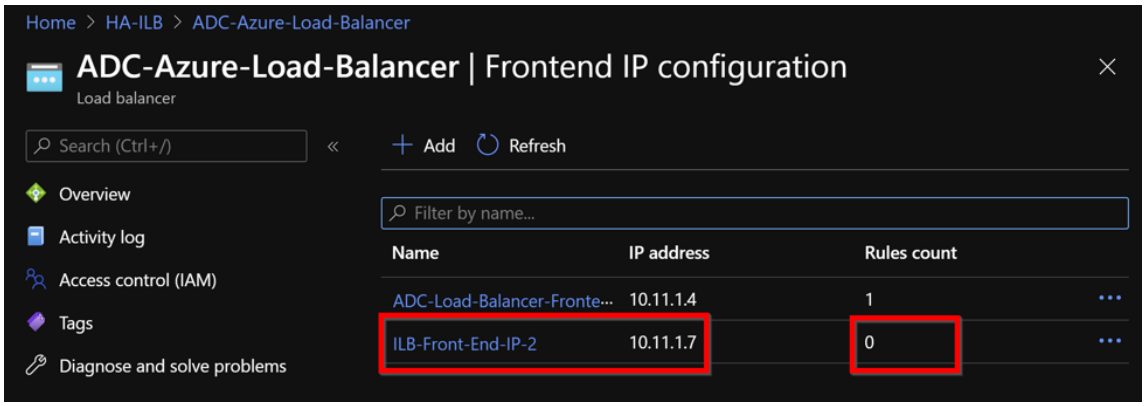
1. **Azure Load Balancer > Frontend IP** 構成に移動し、[追加] をクリックして新しい内部ロードバランサー IP アドレスを作成します。



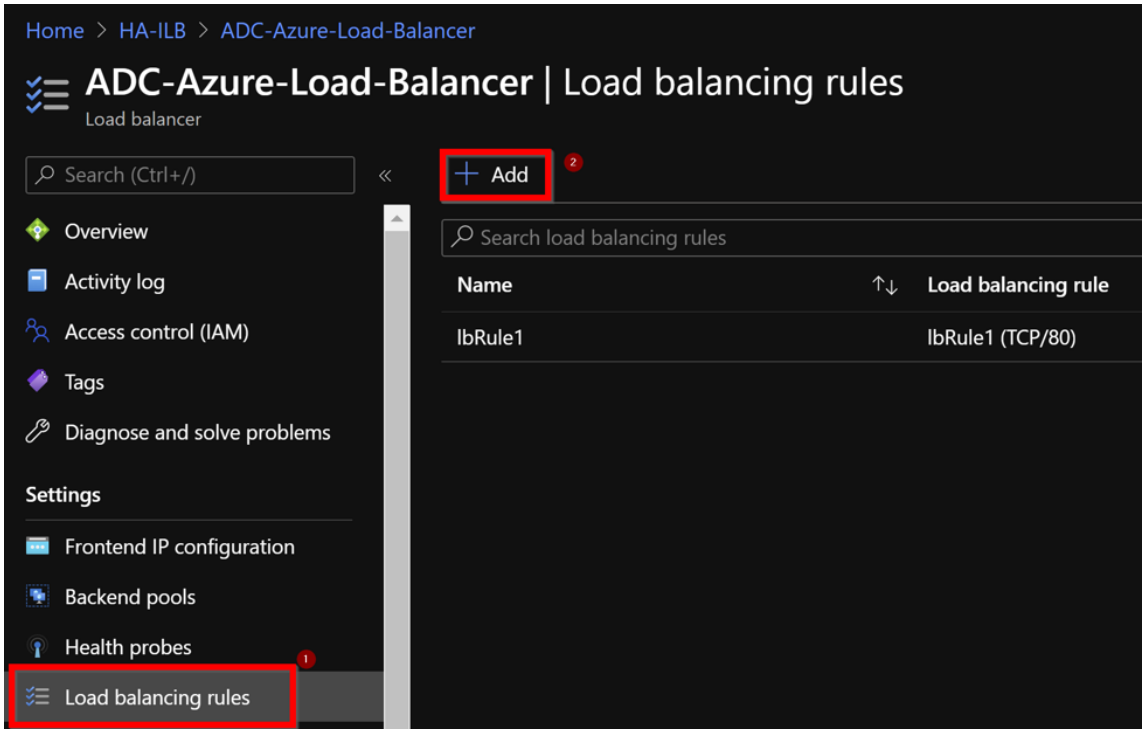
2. **[Add frontend IP address**] ページで、名前を入力し、クライアントサブネットを選択し、動的 IP アドレスまたは静的 IP アドレスを割り当てて、[**Add**] をクリックします。



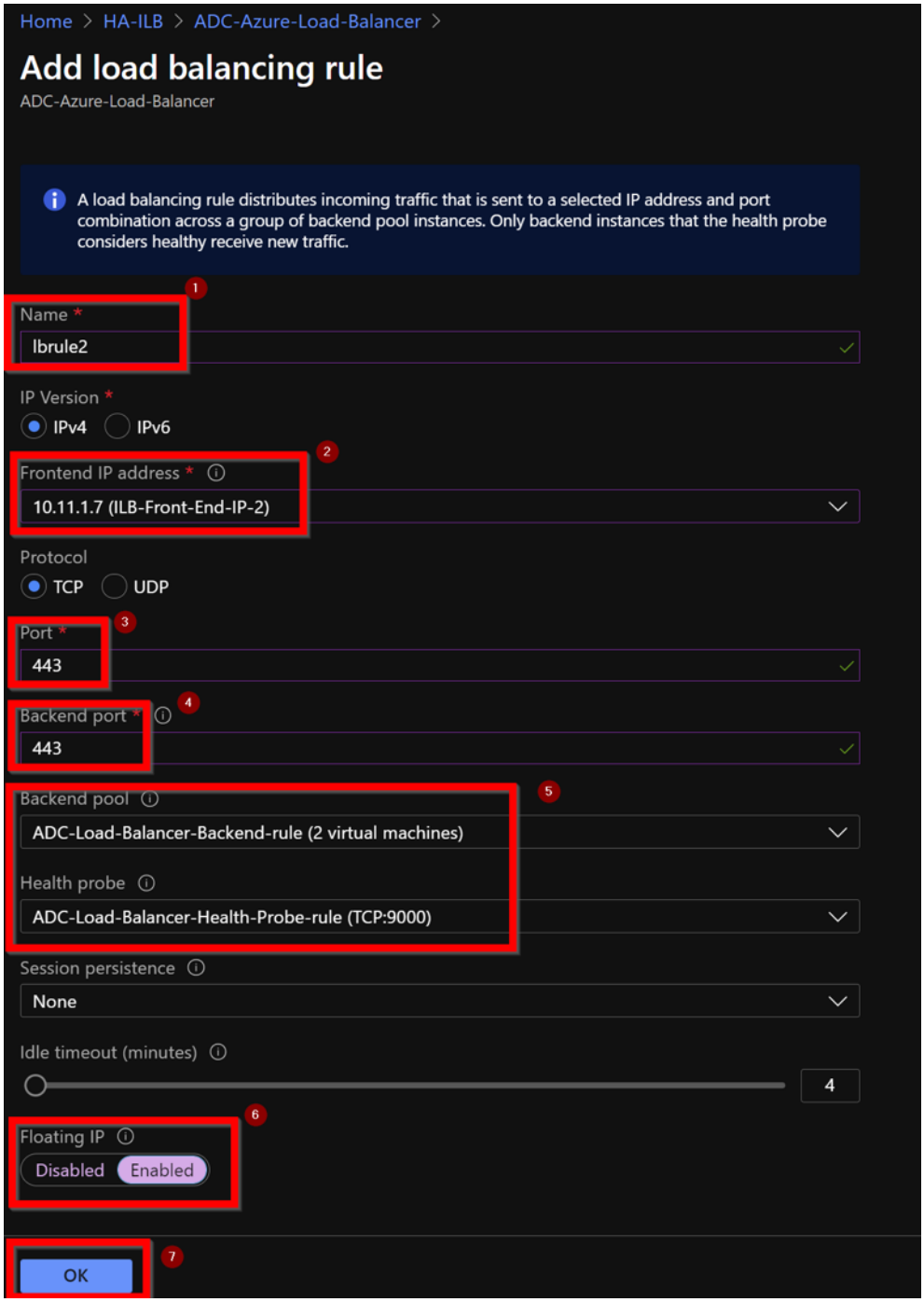
3. フロントエンド IP アドレスは作成されますが、LB ルールは関連付けられていません。新しい負荷分散ルールを作成し、フロントエンド IP アドレスに関連付けます。



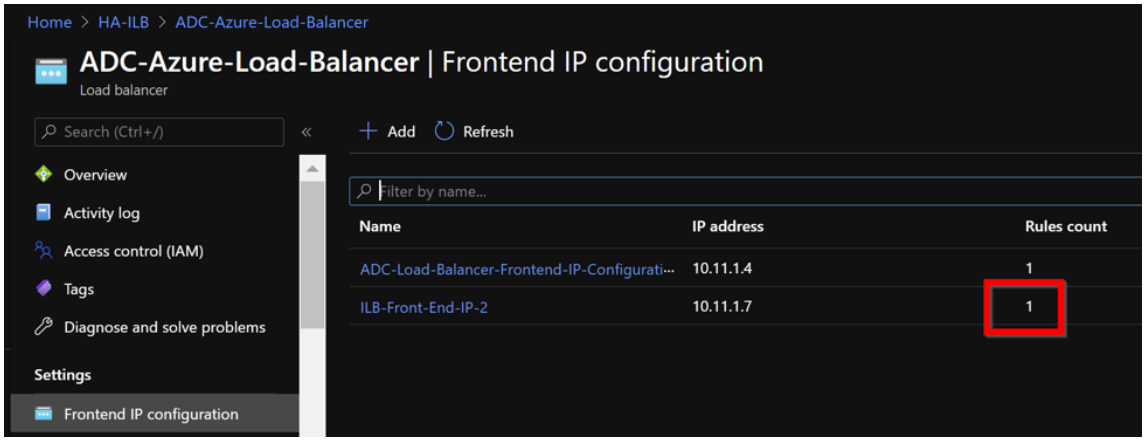
4. [**Azure** ロードバランサー] ページで、[負荷分散ルール] を選択し、[追加] をクリックします。



5. 新しいフロントエンド IP アドレスとポートを選択して、新しい LB ルールを作成します。[フローティング **IP**] フィールドは [有効] に設定する必要があります。



6. これで、フロントエンド **IP** 設定に、適用されている LB ルールが表示されます。



設定例

ゲートウェイ VPN 仮想サーバーと負荷分散仮想サーバーを設定するには、プライマリノード (ADC-VPX-0) で次のコマンドを実行します。設定はセカンダリノード（ADC-VPX-1）に自動的に同期されます。

ゲートウェイのサンプル構成

```
enable feature aaa LB SSL SSLVPN
enable ns mode MBF
add vpn vserver vpn_ssl SSL 10.11.1.4 443
add ssl certKey ckp -cert wild-cgwsanity.cer -key wild-cgwsanity.key
bind ssl vserver vpn_ssl -certkeyName ckp
```

負荷分散のサンプル構成

```
enable feature LB SSL
enable ns mode MBF
add lb vserver lb_vs1 SSL 10.11.1.7 443
bind ssl vserver lb_vs1 -certkeyName ckp
```

ILB の内部 IP アドレスに関連付けられている完全修飾ドメイン名 (FQDN) を使用して、負荷分散または VPN 仮想サーバーにアクセスできるようになりました。

負荷分散仮想サーバーの構成方法の詳細については、「リソース」セクションを参照してください。

リソース:

次のリンクには、HA の導入と仮想サーバの設定に関する追加情報が表示されます。

- 異なるサブネットでの高可用性ノードの構成
- 基本的な負荷分散を設定する

関連リソース:

- PowerShell コマンドを使用して複数の IP アドレスと NIC を使用して高可用性セットアップを構成する
- Azure でのアクティブスタンバイ HA デプロイメントでの GSLB の構成

インターネット向けアプリケーション用の **NetScaler** 高可用性テンプレートを使用して **HA-INC** ノードを構成する

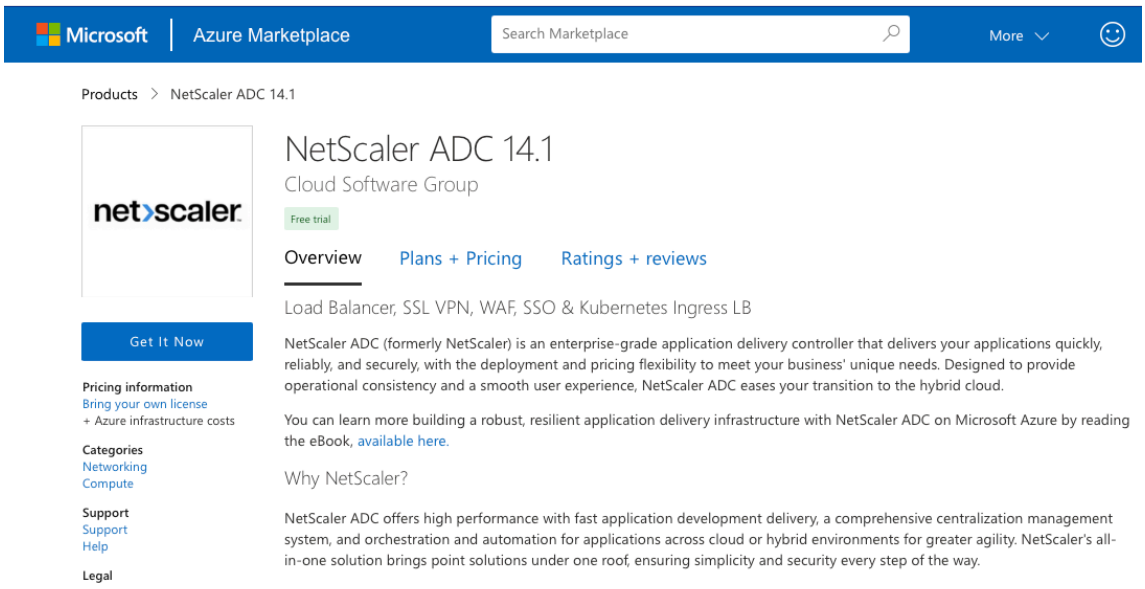
October 17, 2024

インターネット向けアプリケーションの標準テンプレートを使用すると、一対の VPX インスタンスを HA-INC モードで迅速かつ効率的にデプロイできます。Azure ロードバランサー (ALB) は、フロントエンドにパブリック IP アドレスを使用します。このテンプレートでは、3 つのサブネットと 6 つの NIC を持つ 2 つのノードが作成されます。サブネットは、管理、クライアント、およびサーバー側のトラフィック用です。各サブネットには、両方の VPX インスタンス用に 2 つの NIC があります。

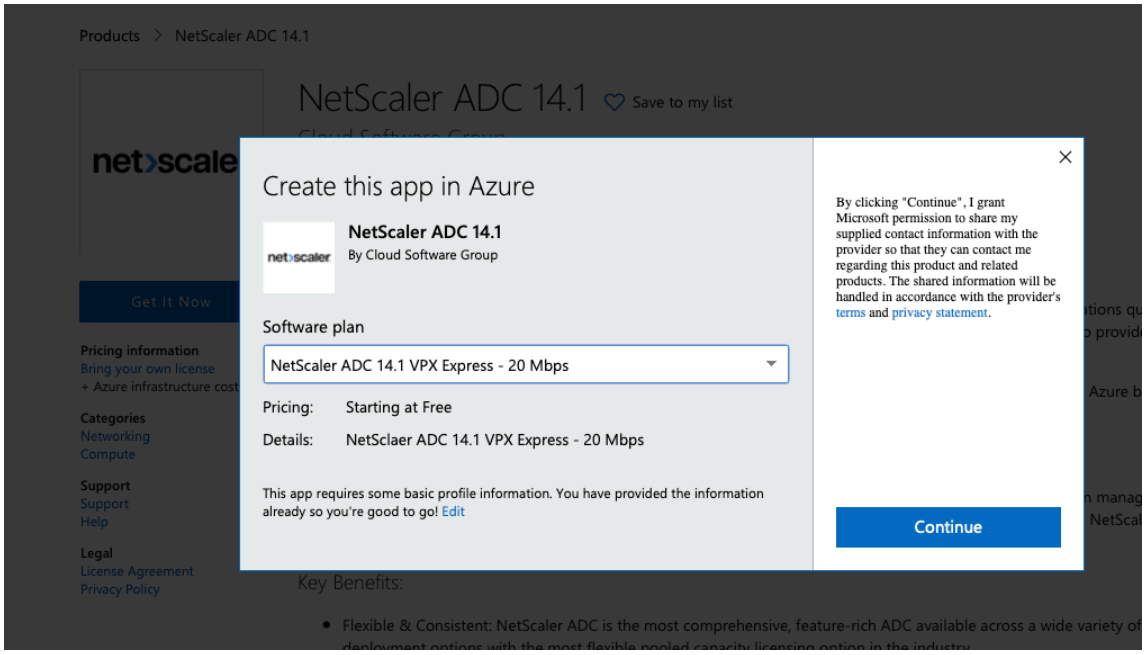
インターネット向けアプリケーションの Citrix ADC HA ペアテンプレートは、Azure Marketplace で入手できます。

次の手順を実行してテンプレートを起動し、Azure 可用性セットまたは可用性ゾーンを使用して高可用性 VPX ペアをデプロイします。

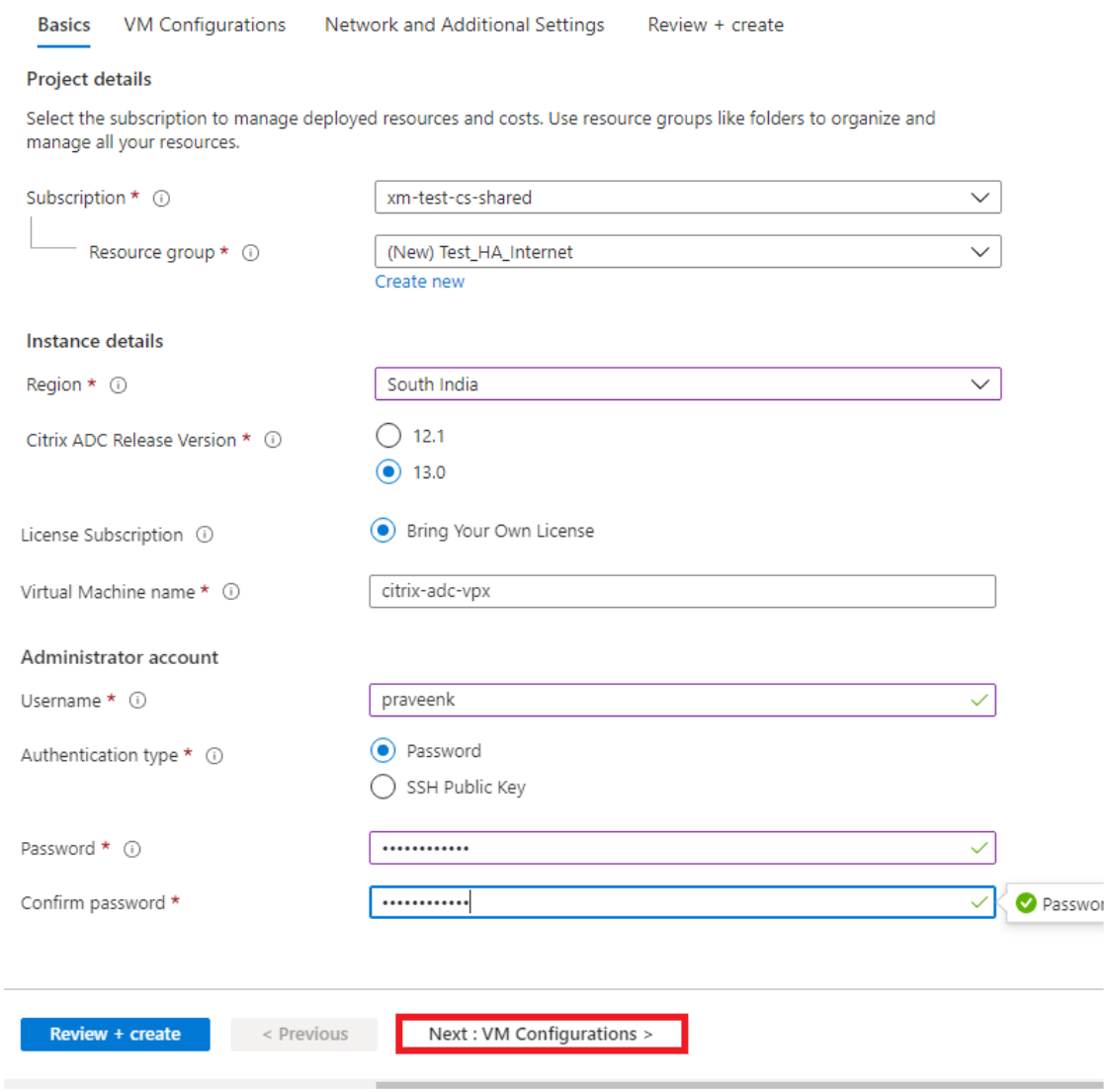
1. Azure Marketplace から **NetScaler** を検索してください。
2. [今すぐ入手] をクリックします。



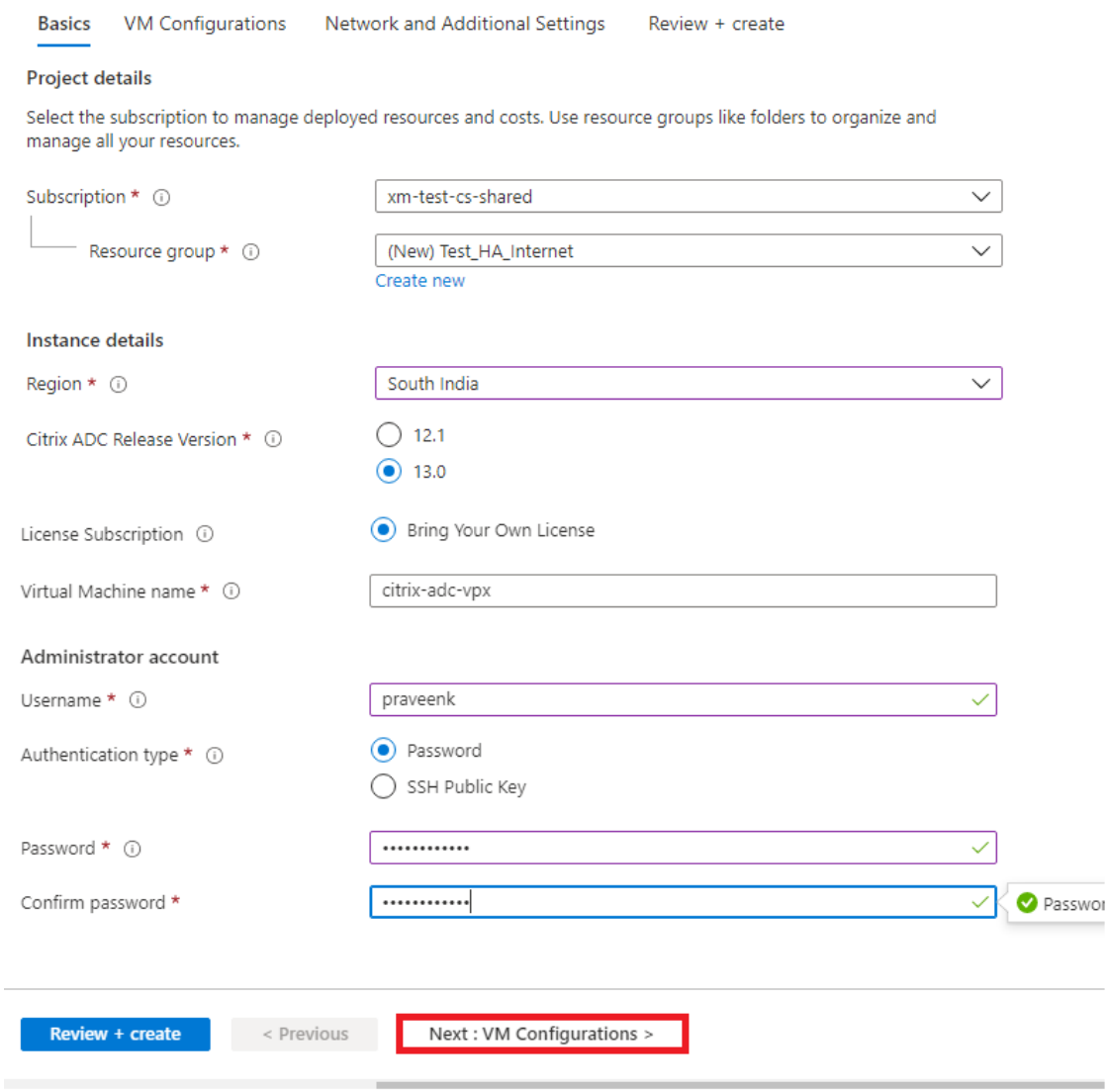
3. 必要な HA 導入とライセンスを選択し、[続行] をクリックします。



4. [基本] ページが表示されます。リソースグループを作成します。[パラメータ] タブで、地域、管理者ユーザー名、管理者パスワード、ライセンスタイプ（VM SKU）、およびその他のフィールドの詳細を入力します。



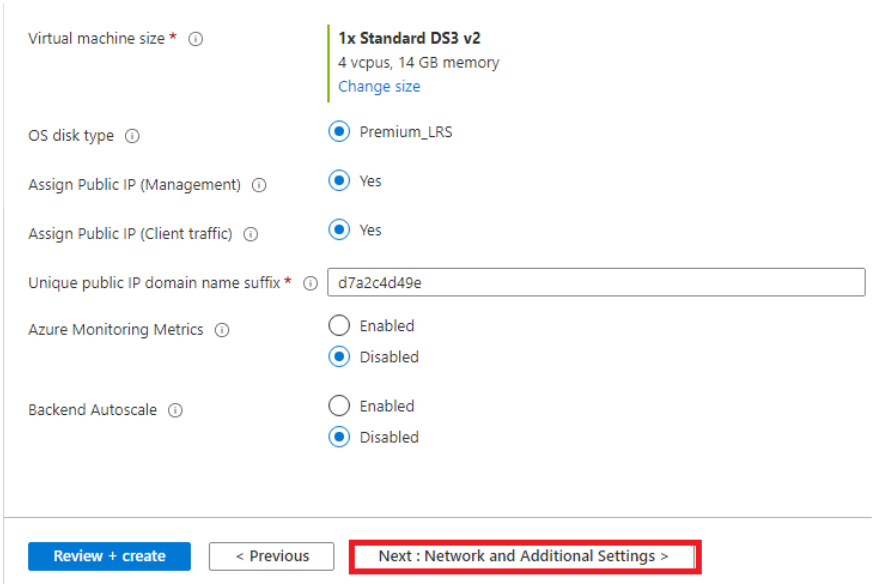
5. [次へ] をクリックします: VM 構成 > 。



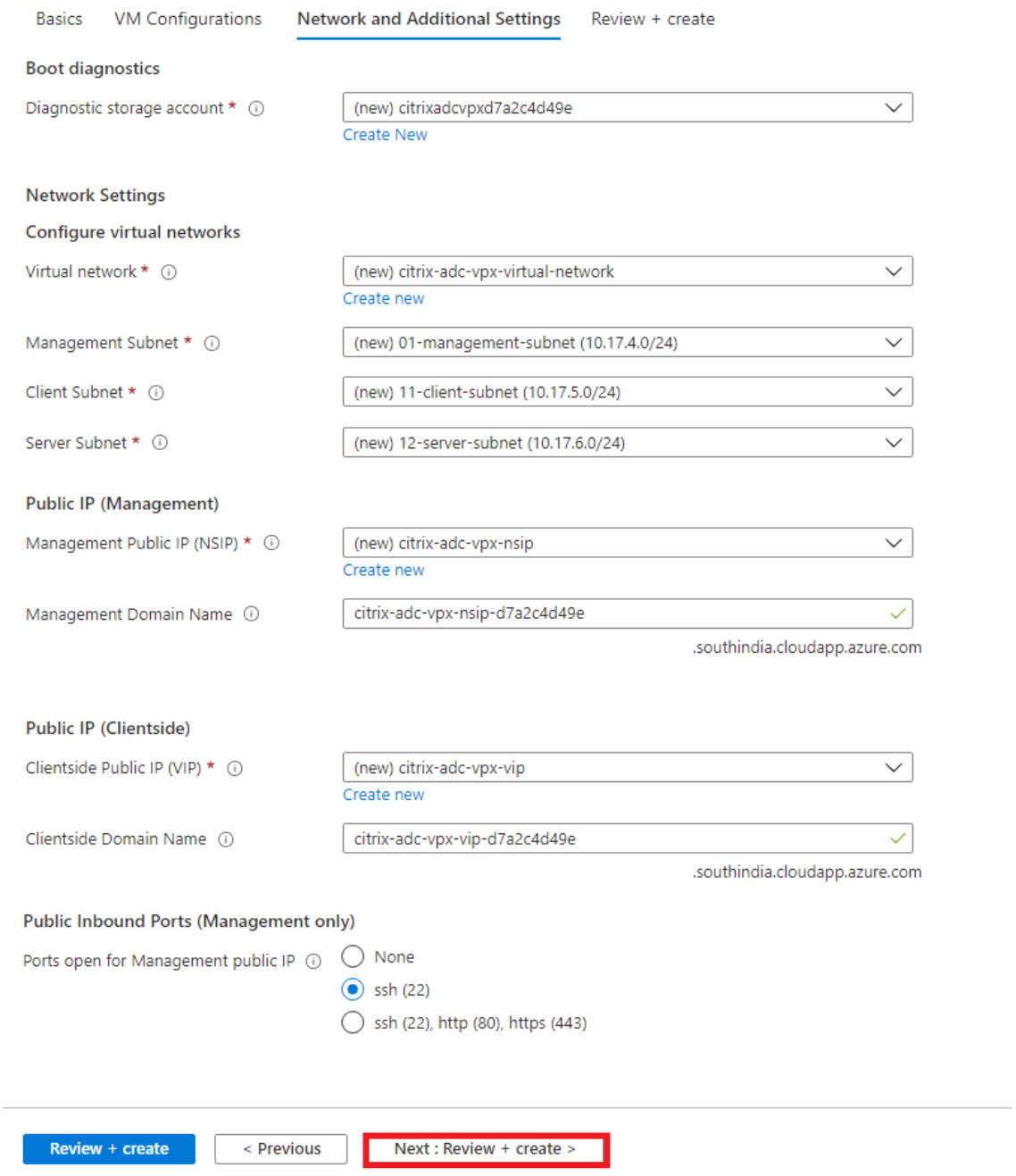
6. [**VM** 構成] ページで、次の手順を実行します。

   - パブリック IP ドメイン名サフィックスの設定
   - Azure 監視メトリクスを有効または無効にする
   - バックエンド **Autoscale** を有効または無効にする

7. [次へ**:** ネットワークとその他の設定] をクリックします。



8. [ネットワークとその他の設定] ページで、ブート診断アカウントを作成し、ネットワーク設定を行います。

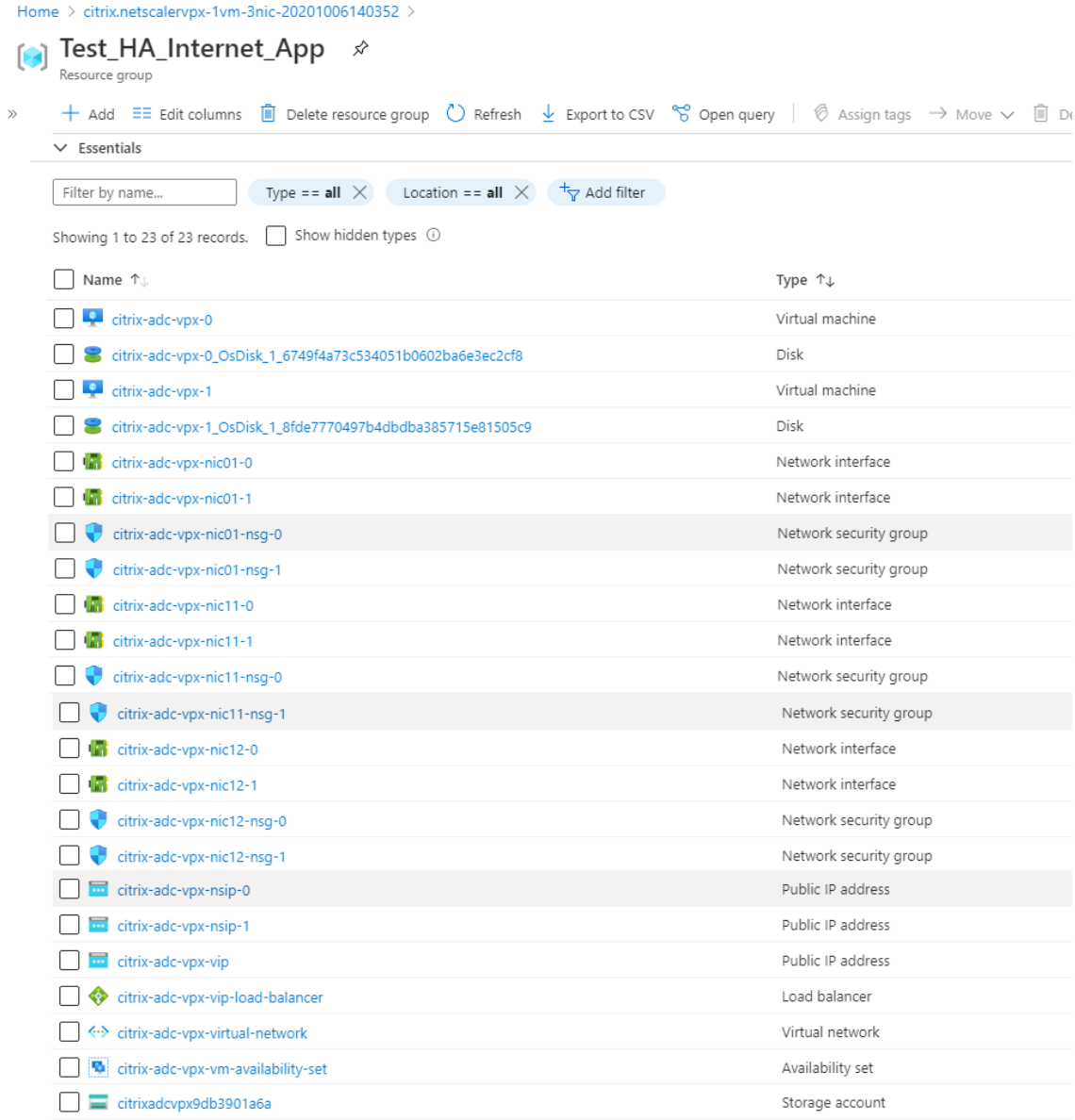


9. [次へ] をクリックします: レビュー + 作成する > 。
10. 基本設定、VM 構成、ネットワーク、その他の設定を確認して、[作成] をクリックします。

    必要な構成で Azure リソースグループが作成されるまで時間がかかることがあります。完了したら、Azure ポータルでリソースグループを選択すると、LB ルール、バックエンドプール、ヘルスプローブなどの構成の詳細が表示されます。高可用性ペアは、**citrix-adc-vpx-0** と **citrix-adc-vpx-1** として表示されます。

追加のセキュリティルールやポートを作成するなど、HA セットアップでさらに変更が必要な場合は、Azure Portal から実行できます。

必要な構成が完了すると、次のリソースが作成されます。



11. 次の構成を検証するには、citrix-adc-vpx-0 ノードと citrix-adc-vpx-1 ノードにログオンする必要があります **。</p>

    - 両方のノードの NSIP アドレスは管理サブネットに存在する必要があります。
    - プライマリ (citrix-adc-vpx-0) ノードとセカンダリ (citrix-adc-vpx-1) ノードには、2 つの SNIP アドレスが必要です。1 つの SNIP (クライアントサブネット) は ALB プローブへの応答に使用され、もう 1 つの SNIP (サーバーサブネット) はバックエンドサーバー通信に使用されます。

> 注
>
> HA-INC モードでは、citrix-adc-vpx-0 と citrix-adc-vpx-1 VM の SNIP アドレスは異なります。これは、両方が同じである従来のオンプレミス ADC 高可用性導入環境とは異なります。

プライマリノード (citrix-adc-vpx-0) で

```
> sh ip
        Ipaddress        Traffic Domain  Type            Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----            ----     ---      ----     -------  ------
1)      10.18.0.4        0               NetScaler IP    Active   Enabled  Enabled  NA       Enabled
2)      10.18.1.5        0               SNIP            Active   Enabled  Enabled  NA       Enabled
3)      10.18.2.4        0               SNIP            Active   Enabled  Enabled  NA       Enabled
 Done
```

```
> sh ha node
1)       Node ID:       0
         IP:        10.18.0.4 (ns-vpx0)
         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 msecs
         Dead Interval: 3 secs
         Node in this Master State for: 0:3:34:21 (days:hrs:min:sec)
2)       Node ID:       1
         IP:        10.18.0.5
         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
```

セカンダリノード (citrix-adc-vpx-1) 上

```
> show ip
        Ipaddress        Traffic Domain  Type            Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----            ----     ---      ----     -------  ------
1)      10.18.0.5        0               NetScaler IP    Active   Enabled  Enabled  NA       Enabled
2)      10.18.1.4        0               SNIP            Active   Enabled  Enabled  NA       Enabled
3)      10.18.2.5        0               SNIP            Active   Enabled  Enabled  NA       Enabled
 Done
>
```

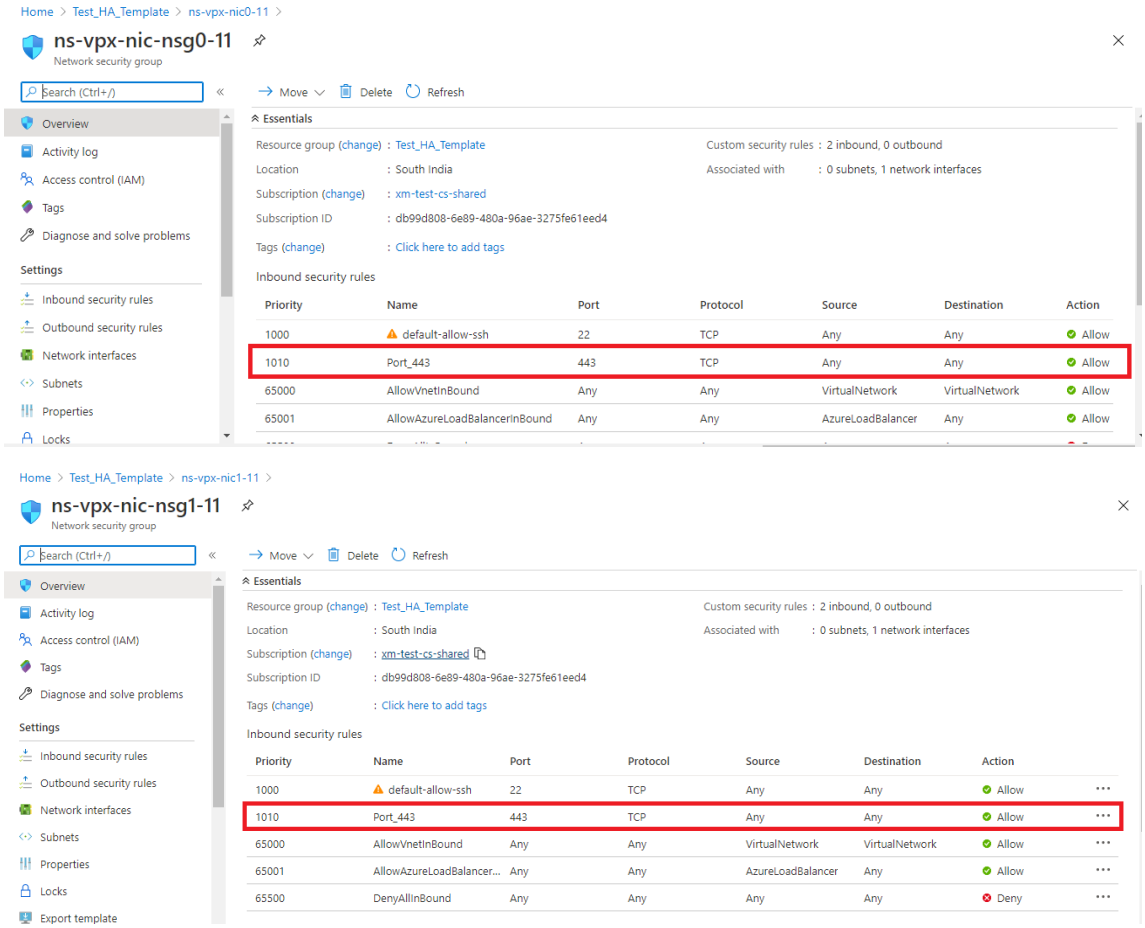


```
> sh ha node
1)      Node ID:      0
        IP:       10.18.0.5 (ns-vpx1)
        Node State: UP
        Master State: Secondary
        Fail-Safe Mode: OFF
        INC State: ENABLED
        Sync State: SUCCESS
        Propagation: ENABLED
        Enabled Interfaces : 0/1 1/1 1/2
        Disabled 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 msecs
        Dead Interval: 3 secs
        Node in this Master State for: 0:3:23:51 (days:hrs:min:sec)
2)      Node ID:      1
        IP:       10.18.0.4
        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
>
```

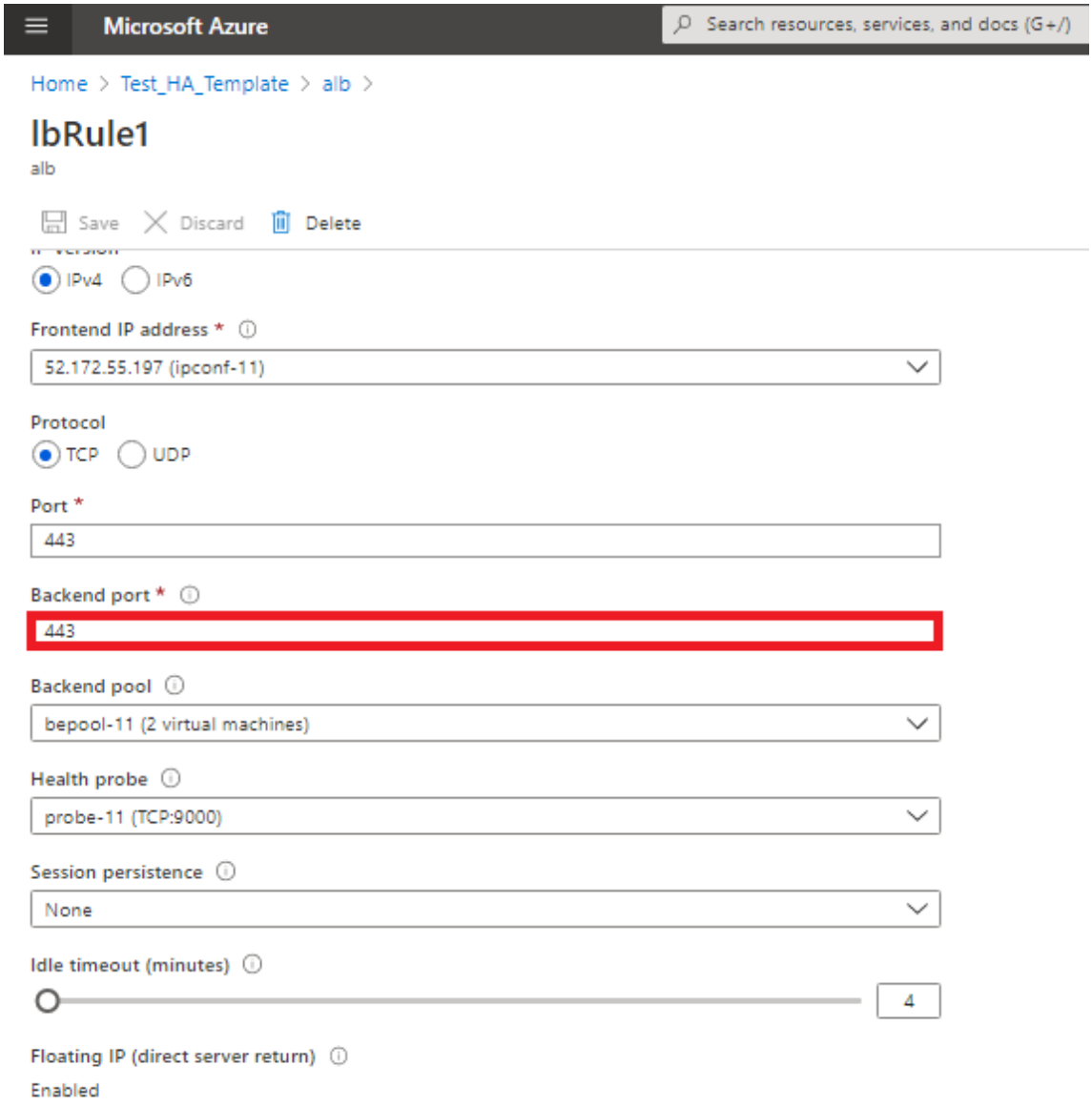
12. プライマリノードとセカンダリノードが UP になり、同期ステータスが **SUCCESS** になったら、ALB 仮想のパブリック IP アドレスを使用して、プライマリノード（citrix-adc-vpx-0）の負荷分散仮想サーバーまたはゲートウェイ仮想サーバーを構成する必要があります。詳細については、「サンプル設定」セクションを参照してください。

13. ALB 仮想サーバーのパブリック IP アドレスを見つけるには、**Azure portal > Azure Load Balancer > Frontend IP configuration** に移動します。

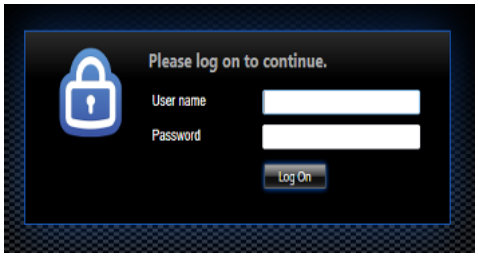
14. 仮想サーバーポート 443 のインバウンドセキュリティルールを、両方のクライアントインターフェイスのネットワークセキュリティグループに追加します。



15. アクセスする ALB ポートを設定し、指定したポートのインバウンドセキュリティルールを作成します。バックエンドポートは、負荷分散仮想サーバーポートまたは VPN 仮想サーバーポートです。



16. これで、ALB パブリック IP アドレスに関連付けられた完全修飾ドメイン名 (FQDN) を使用して、負荷分散仮想サーバーまたは VPN 仮想サーバーにアクセスできます。



設定例

ゲートウェイ VPN 仮想サーバーと負荷分散仮想サーバーを設定するには、プライマリノード (ADC-VPX-0) で次のコマンドを実行します。設定はセカンダリノード（ADC-VPX-1）に自動的に同期されます。

ゲートウェイのサンプル構成

```
enable feature aaa LB SSL SSLVPN
add ip 52.172.55.197 255.255.255.0 -type VIP
add vpn vserver vpn_ssl SSL 52.172.55.197 443
add ssl certKey ckp -cert cgwsanity.cer -key cgwsanity.key
bind ssl vserver vpn_ssl -certkeyName ckp
```

負荷分散のサンプル構成

```
enable feature LB SSL
enable ns mode MBF
add lb vserver lb_vs1 SSL 52.172.55.197 443
bind ssl vserver lb_vs1 -certkeyName ckp
```

ALB のパブリック IP アドレスに関連付けられた FQDN を使用して、負荷分散または VPN 仮想サーバーにアクセスできるようになりました。

負荷分散仮想サーバーを構成する方法の詳細については、「リソース」セクションを参照してください。

リソース:

次のリンクには、HA の導入と仮想サーバの設定に関する追加情報が表示されます。

- 仮想サーバーを作成する
- 基本的な負荷分散を設定する

Azure 外部および内部ロードバランサーで同時に高可用性セットアップを構成する

October 17, 2024

Azure の高可用性ペアは、外部ロードバランサーと内部ロードバランサーの両方を同時にサポートします。

Azure 外部ロードバランサーと内部ロードバランサーの両方を使用して高可用性ペアを構成するには、次の 2 つのオプションがあります。

- NetScaler ADC アプライアンス上で 2 つの LB 仮想サーバーを使用する。
- 1 つの LB 仮想サーバーと IP セットを使用する。単一の LB 仮想サーバは、IPSet によって定義された複数の IP にトラフィックを処理します。

外部ロードバランサーと内部ロードバランサーを同時に使用して Azure で高可用性ペアを構成するには、次の手順を実行します。

手順 1 と 2 については、Azure ポータルを使用します。手順 3 および 4 では、NetScaler VPX GUI または CLI を使用します。

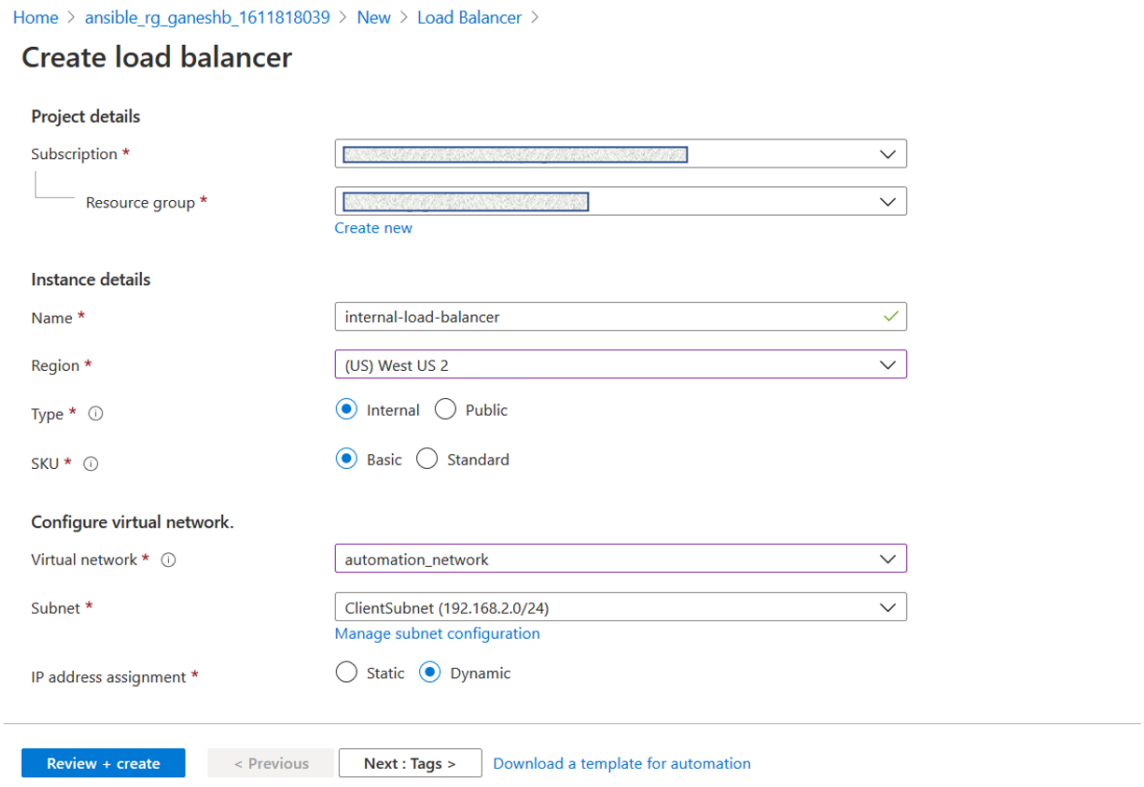
ステップ **1**。外部ロード バランサーまたは内部ロード バランサーのいずれかの Azure ロード バランサーを構成します。

Azure 外部ロード バランサーを使用した高可用性設定の構成の詳細については、「複数の IP アドレスと NIC を使用して高可用性設定を構成する」を参照してください。

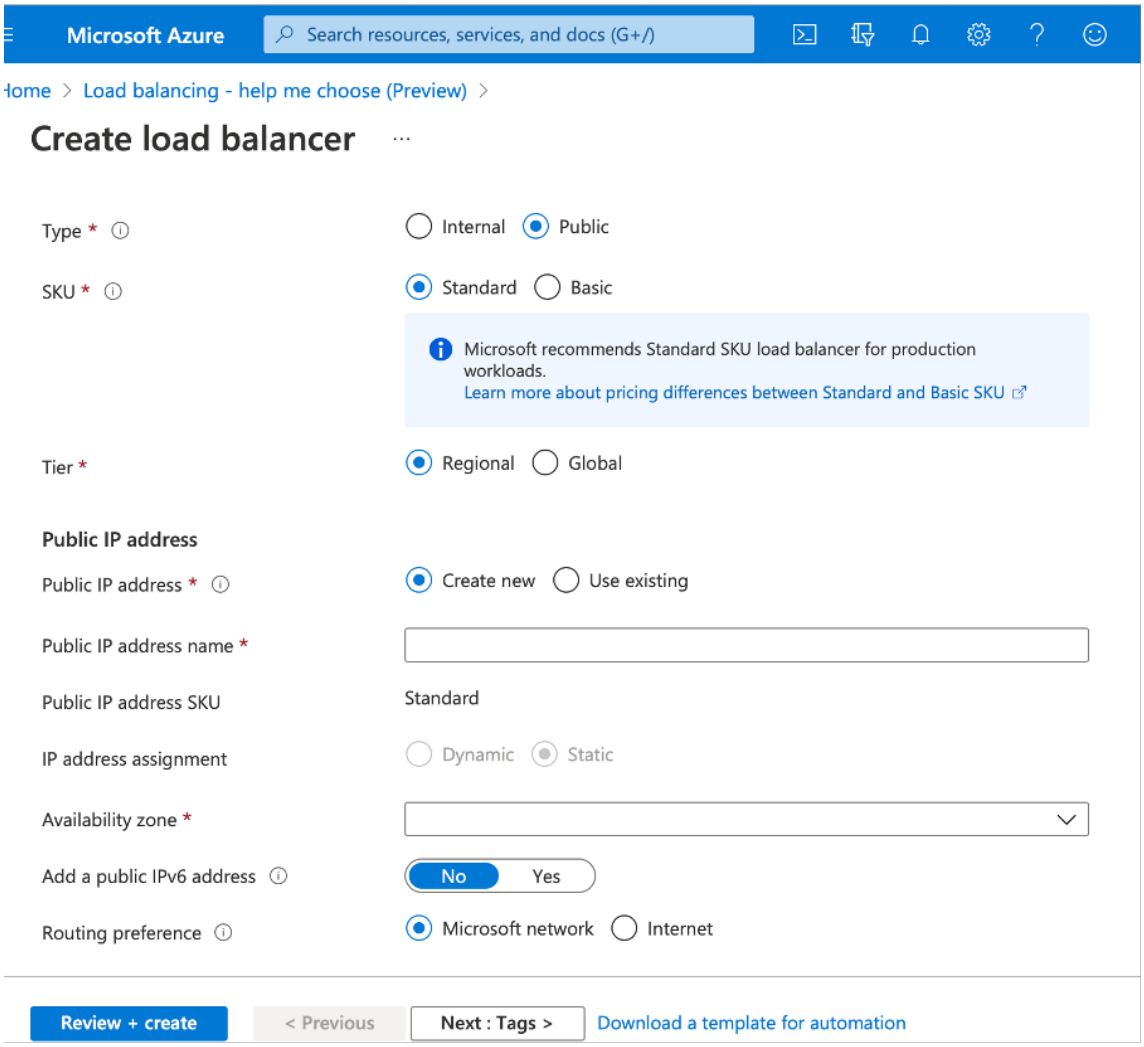
Azure 内部ロードバランサーを使用した高可用性セットアップの構成の詳細については、以下を参照してください。NetScaler 高可用性テンプレートと Azure ILB を使用した HA-INC ノードの構成.

ステップ **2**。リソース グループに追加のロード バランサー (ILB) を作成します。ステップ 1 では、外部ロードバランサーを作成した場合は、内部ロードバランサーを作成し、逆に作成します。

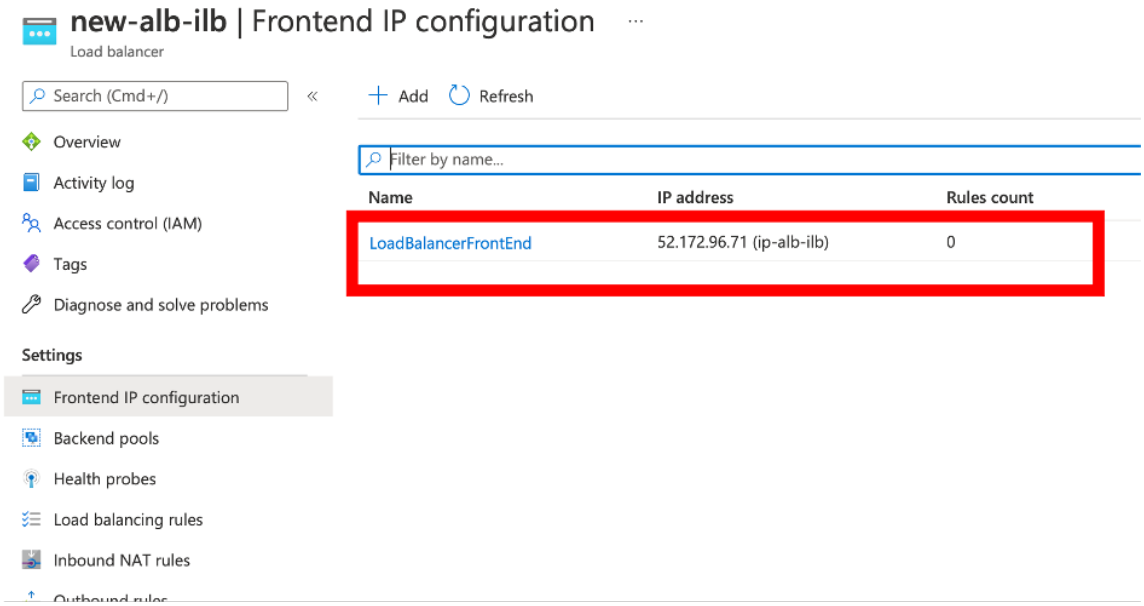
- 内部ロードバランサーを作成するには、ロードバランサのタイプを [内部] として選択します。[サブネット] フィールドで、NetScaler ADC クライアントサブネットを選択する必要があります。競合がない限り、そのサブネットに静的 IP アドレスを指定することもできます。それ以外の場合は、ダイナミック IP アドレスを選択します。



- 外部ロードバランサーを作成するには、ロードバランサの種類を [パブリック] として選択し、ここにパブリック IP アドレスを作成します。



1. Azure Load Balancer を作成したら、フロントエンド **IP** 設定に移動し、ここに示す IP アドレスを書き留めます。ステップ 3 のように ADC 負荷分散仮想サーバーを作成するときは、この IP アドレスを使用する必要があります。



2. **Azure Load Balancer** の設定ページで、ARM テンプレートのデプロイは、LB ルール、バックエンドプール、およびヘルスプローブの作成に役立ちます。
3. 高可用性ペアのクライアント NIC を ILB のバックエンドプールに追加します。
4. ヘルスプローブの作成（TCP、9000 ポート）
5. 次の 2 つの負荷分散ルールを作成します。
   - ポート 80 の HTTP トラフィック（Webapp ユースケース）の 1 つの LB ルール。ルールでは、バックエンドポート 80 も使用する必要があります。作成したバックエンドプールとヘルスプローブを選択します。フローティング IP を有効にする必要があります。
   - ポート 443 の HTTPS または CVAD トラフィックに対する別の LB ルール。プロセスは HTTP トラフィックと同じです。

ステップ **3**。 NetScaler アプライアンスのプライマリ ノードで、ILB 用の負荷分散仮想サーバーを作成します。

1. 負荷分散仮想サーバーを追加します。

```
add lb vserver <name> <serviceType> [<ILB Frontend IP address>]
    [<port>]
```

例

```
add lb vserver vserver_name HTTP 52.172.96.71 80
```

> 注
>
> ステップ 2 で作成した追加のロードバランサーに関連付けられた、ロードバランサーのフロントエンド IP アドレスを使用します。

2. サービスを負荷分散仮想サーバーにバインドします。


```
bind lb vserver <name> <serviceName>
```

例

```
bind lb vserver Vserver-LB-1 Service-HTTP-1
```

詳細については、「基本的な負荷分散の設定」を参照してください。

ステップ **4:** ステップ 3 の代わりに、IPSet を使用して ILB の負荷分散仮想サーバーを作成できます。

1. 仮想サーバー IP (VIP) タイプの IP アドレスを追加します。

```
add nsip <ILB Frontend IP address> -type <type>
```

例

```
add nsip 52.172.96.71 -type vip
```

2. プライマリノードとセカンダリノードの両方に IPSet を追加します。

```
add ipset <name>
```

例

```
add ipset ipset1
```

3. IP アドレスを IP セットにバインドします。

```
bind ipset <name> <ILB Frontend IP address>
```

例

```
bind ipset ipset1 52.172.96.71
```

4. 既存の LB 仮想サーバーを IPSet を使用するように設定します。

```
set lb vserver <vserver name> -ipset <ipset name>
```

例

```
set lb vserver vserver_name -ipset ipset1
```

詳細については、「マルチ IP 仮想サーバーの構成」を参照してください。

Azure VMware ソリューションに NetScaler VPX インスタンスをインストールする

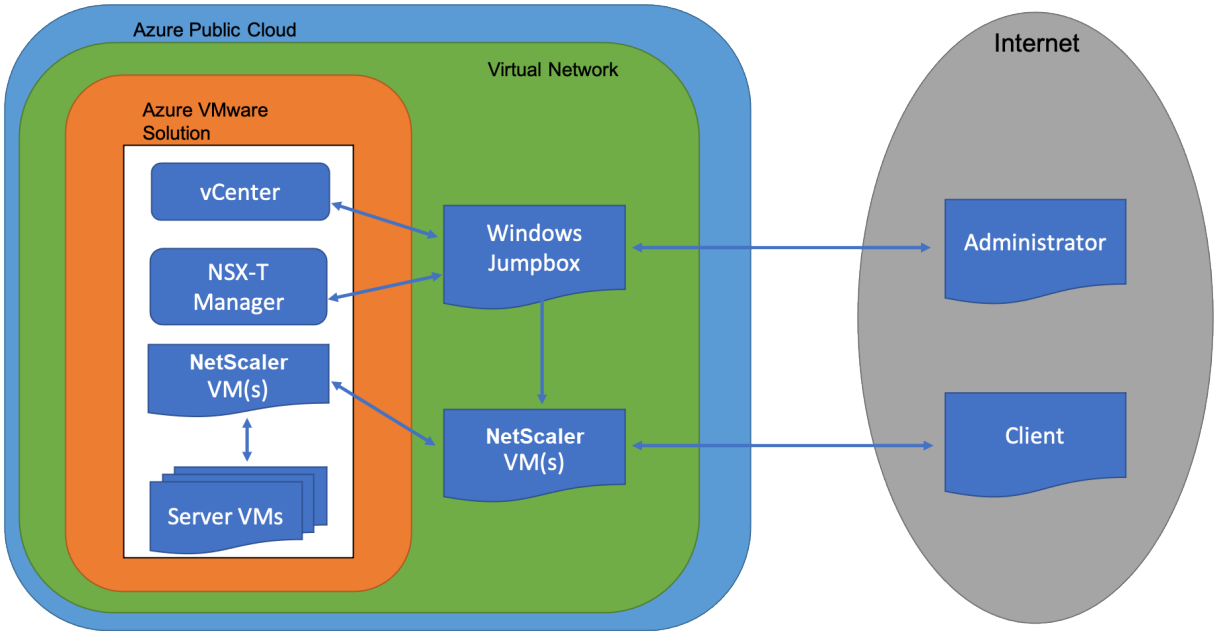
October 17, 2024

Azure VMware ソリューション (AVS) は、専用のベアメタル Azure インフラストラクチャから構築された vSphere クラスタを含むプライベートクラウドを提供します。最初のデプロイメントは最小で 3 台のホストですが、追加ホストは一度に 1 つずつ追加でき、クラスタごとに最大 16 台のホストを追加できます。プロビジョニングされたすべてのプライベートクラウドには、vCenter Server、vSAN、vSphere、NSX-T があります。

Azure 上の VMware クラウド (VMC) を使用すると、必要な数の ESX ホストを使用して Azure 上にクラウドソフトウェア定義データセンター (SDDC) を作成できます。Azure 上の VMC は、NetScaler VPX デプロイメントをサポートしています。VMC は、オンプレミスの vCenter と同じユーザー・インタフェースを提供します。これは、ESX ベースの NetScaler VPX 展開と同様に機能します。

次の図は、管理者またはクライアントがインターネット経由でアクセスできる Azure パブリッククラウド上の Azure VMware ソリューションを示しています。管理者は、Azure VMware ソリューションを使用して、ワークロードまたはサーバー仮想マシンを作成、管理、および構成できます。管理者は、Windows ジャンプボックスから AVS の Web ベースの vCenter および NSX-T マネージャにアクセスできます。vCenter を使用して Azure VMware Solution 内に NetScaler VPX インスタンス（スタンドアロンまたは高可用性ペア）とサーバー仮想マシンを作成し、NSX-T Manager を使用して対応するネットワークを管理できます。AVS 上の NetScaler VPX インスタンスは、オンプレミスの VMware ホストのクラスタと同様に機能します。AVS は、同じ仮想ネットワーク内に作成された Windows ジャンプボックスから管理されます。

クライアントは、ADC の VIP に接続することによってのみ AVS サービスにアクセスできます。Azure VMware ソリューション外の別の NetScaler VPX インスタンスは、同じ Azure 仮想ネットワーク内にある別の NetScaler VPX インスタンスは、Azure VMware ソリューション内の NetScaler VPX インスタンスの VIP をサービスとして追加するのに役立ちます。要件に応じて、インターネット上でサービスを提供するように NetScaler VPX インスタンスを構成できます。



前提条件

仮想アプライアンスのインストールを開始する前に、次の操作を行います。

- Azure VMware ソリューションとその前提条件の詳細については、Azure VMware ソリューションのドキュメントを参照してください。
- Azure VMware ソリューションのデプロイの詳細については、「Azure VMware ソリューションのプライベートクラウドをデプロイする」を参照してください。
- Azure VMware ソリューションにアクセスして管理するための Windows ジャンプボックス仮想マシンの作成の詳細については、「Azure VMware ソリューションのプライベートクラウドにアクセスする」を参照してください。
- Windows ジャンプボックス仮想マシンで、NetScaler VPX アプライアンスセットアップファイルをダウンロードします。
- 仮想マシンが接続する VMware SDDC 上に、適切な NSX-T ネットワークセグメントを作成します。詳細については、「Azure VMware ソリューションでのネットワークセグメントの追加」を参照してください。
- VPX ライセンスファイルを入手します。
- Azure VMware Solution プライベートクラウドに作成または移行された仮想マシン (VM) は、ネットワークセグメントに接続する必要があります。

VMware クラウドのハードウェア要件

次の表に、VMware SDDC が各 VPX nCore 仮想アプライアンスに対して提供する必要がある仮想コンピューティングリソースを示します。

表 1. NetScaler VPX インスタンスの実行に必要な最小限の仮想コンピューティングリソース

コンポーネント	条件
メモリ	2 GB
仮想 CPU（VCPU）	2
仮想ネットワークインターフェイス	VMware SDDC では、VPX ハードウェアをバージョン 7 以上にアップグレードすると、最大 10 個の仮想ネットワーク インターフェイスをインストールできます。
ディスク領域	20GB

> 注
>
> これは、ハイパーバイザーのディスク要件に加えて必要になります。

VPX 仮想アプライアンスを実稼働で使用するには、フルメモリ割り当てを予約する必要があります。

OVF ツール 1.0 のシステム要件

Open Virtualization Format Tool（OVF Tool）は、Windows および Linux システムで実行できるクライアントアプリケーションです。次の表に、OVF ツールをインストールするためのシステム要件を示します。

表 2. OVF ツールのインストールに関するシステム要件

コンポーネント	条件
オペレーティングシステム	VMware からの詳細な要件については、http://kb.vmware.com/で『OVF ツールユーザーガイド』の PDF ファイルを検索してください。
CPU	最低 750MHz、1GHz 以上推奨
RAM	最小 1 GB、推奨 2 GB
NIC	100Mbps 以上の NIC。

OVF のインストールについては、http://kb.vmware.com/で『OVF ツールユーザーガイド』の PDF ファイルを検索してください。

NetScaler VPX セットアップファイルのダウンロード

VMware ESX 用の NetScaler VPX インスタンスセットアップパッケージは、オープン仮想マシン（OVF）フォーマット標準に準拠しています。これらのファイルは、Citrix Web サイトからダウンロードできます。ログオンするには、Citrix アカウントが必要です。Citrix アカウントをお持ちでない場合は、http://www.citrix.comのホームページにアクセスしてください。［新しいユーザー］リンクをクリックし、指示に従って新しい Citrix アカウントを作成します。

ログオンしたら、Citrix のホームページから次のパスをナビゲートします。

Citrix.com > ダウンロード **> NetScaler >** 仮想アプライアンス。

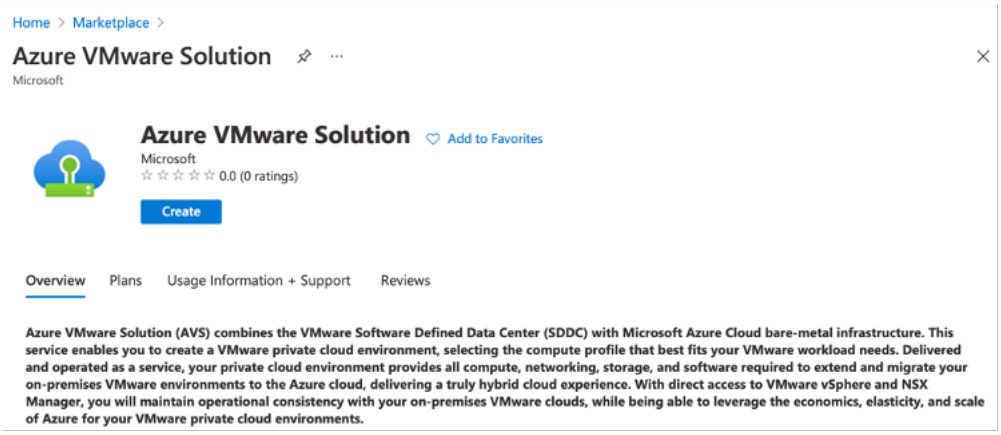
次のファイルを、ESX サーバーと同じネットワーク上のワークステーションにコピーします。3 つのファイルをすべて同じフォルダーにコピーします。

- NSVPX-ESX-<release number>-<build number>-disk1.vmdk（例えば、NSVPX-ESX-13.0-79.64-disk1.vmdk）
- NSVPX-ESX-<release number>-<build number>.ovf（例えば、NSVPX-ESX-13.0-79.64.ovf）
- NSVPX-ESX-<release number>-<build number>.mf（例えば、NSVPX-ESX-13.0-79.64.mf）

Azure VMware ソリューションをデプロイする

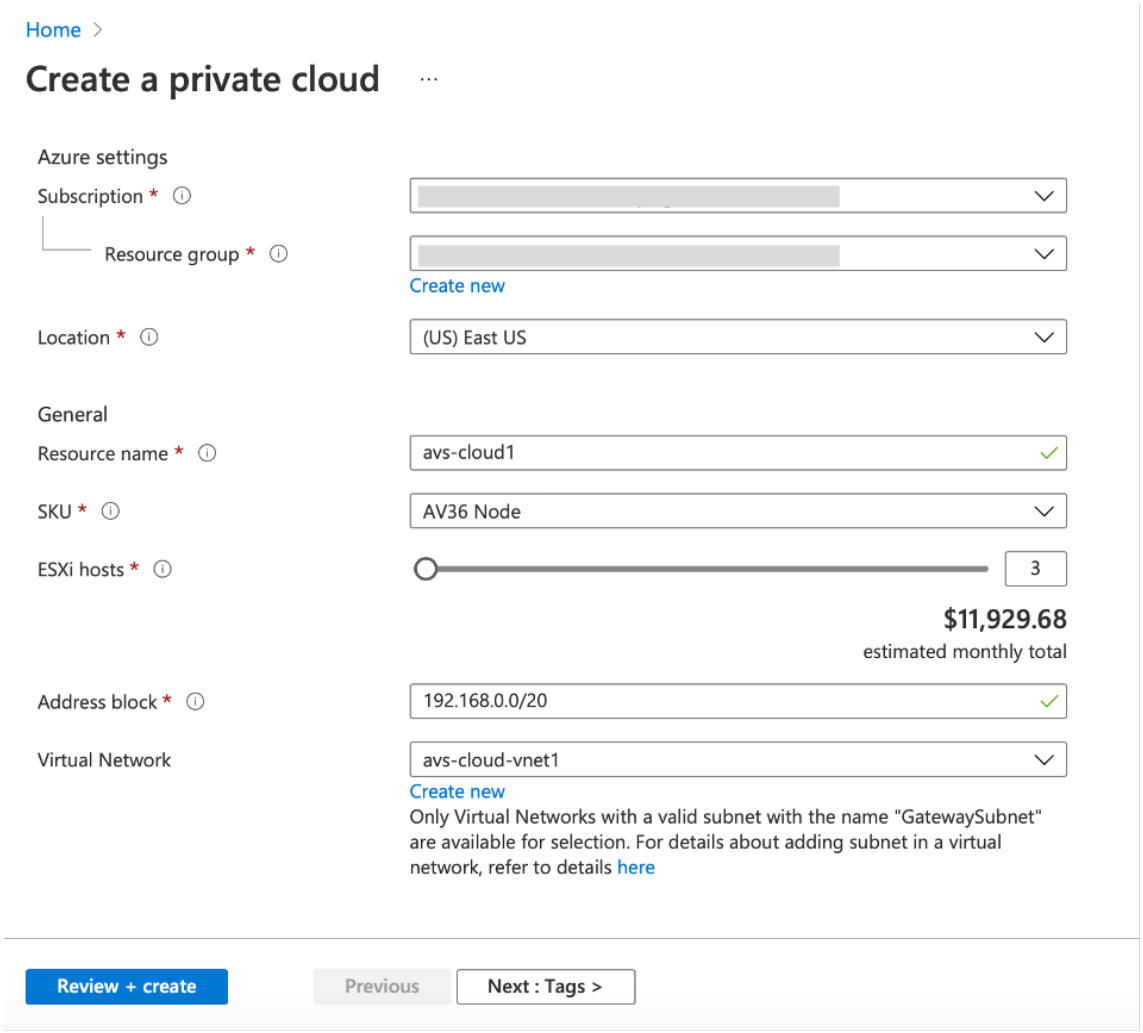
1. Microsoft Azure ポータルにログインし、Azureマーケットプレイスに移動します。

2. **Azure** マーケットプレイスから **AzureVMware** ソリューションを検索し、[作成] をクリックします。

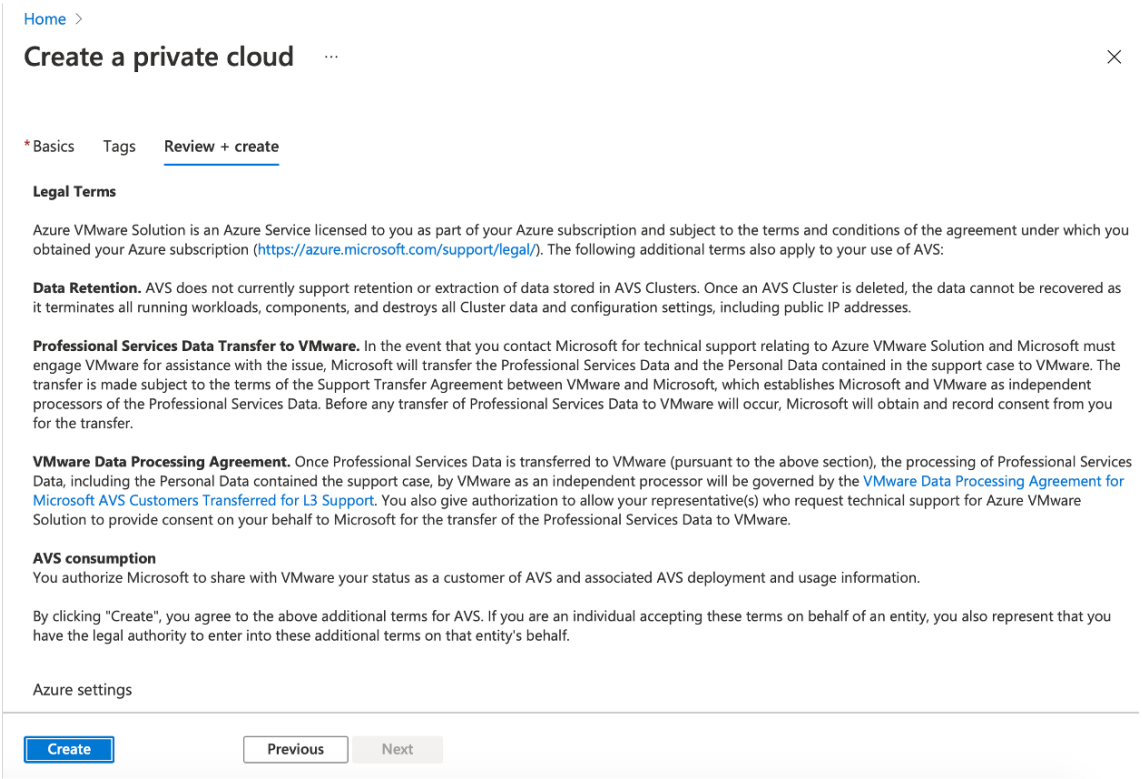


3. [プライベートクラウドの作成] ページで、次の詳細を入力します。

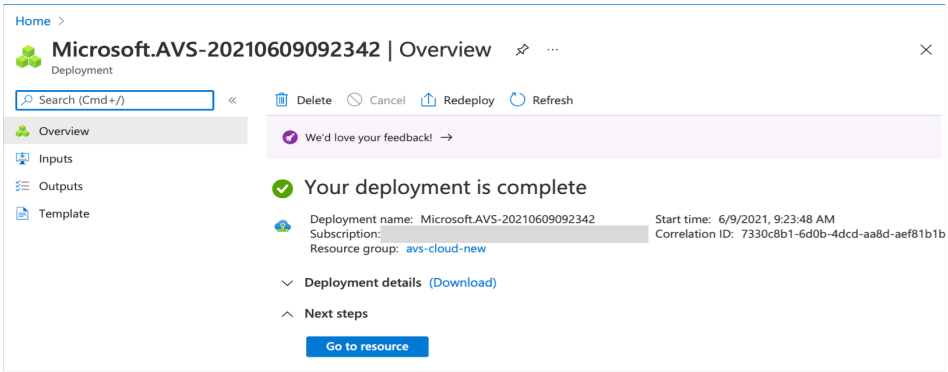
   - プライベートクラウドのデフォルトクラスタを作成するには、最低 3 つの ESXi ホストを選択します。
   - [**Address** ブロック] フィールドには、**/22** アドレス空間を使用します。
   - 仮想ネットワークの場合、CIDR 範囲が、オンプレミスまたはその他の Azure サブネット (仮想ネットワーク) またはゲートウェイサブネットと重複していないことを確認します。
   - ゲートウェイサブネットは、プライベートクラウドとの接続のルーティングを表現するために使用されます。



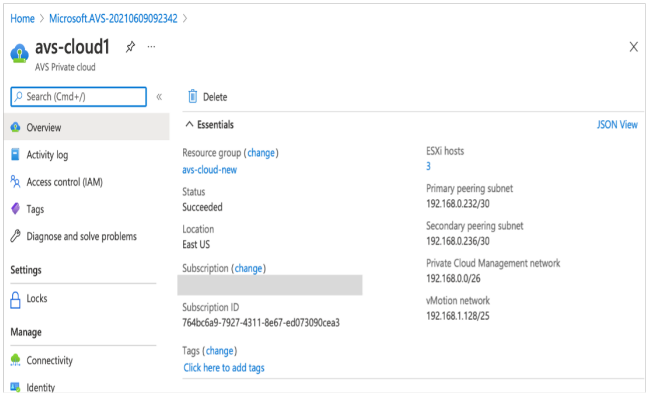
4. [レビュー**] + [**作成] をクリックします。
5. 設定を確認します。設定を変更する必要がある場合は、[前へ] をクリックします。



6. [**Create**] をクリックします。プライベートクラウドのプロビジョニングプロセスが開始されます。プライベートクラウドのプロビジョニングには最大 2 時間かかることがあります。



7. [リソースに移動] をクリックして、作成されたプライベートクラウドを確認します。



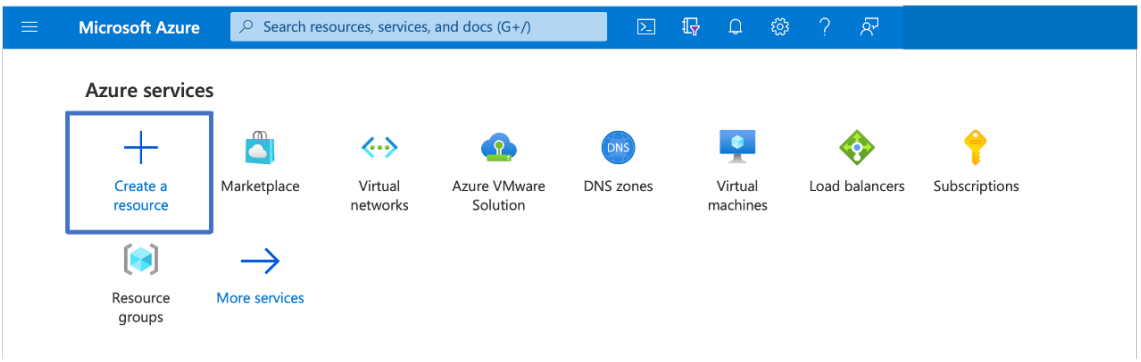
> 注
>
> このリソースにアクセスするには、Windows でジャンプボックスとして機能する仮想マシンが必要です。

Windows を実行している Azure 仮想マシンに接続する

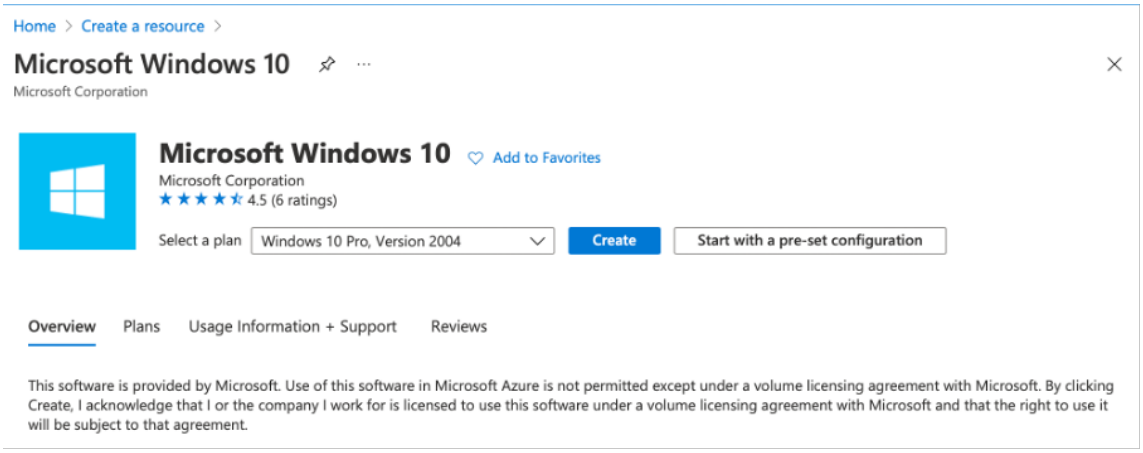
この手順では、Azure ポータルを使用して、Windows Server 2019 を実行する仮想マシン (VM) を Azure にデプロイする方法について説明します。VM の動作を確認するには、仮想マシンに RDP し、IIS Web サーバーをインストールします。

作成したプライベートクラウドにアクセスするには、同じ仮想ネットワーク内に Windows ジャンプボックスを作成する必要があります。

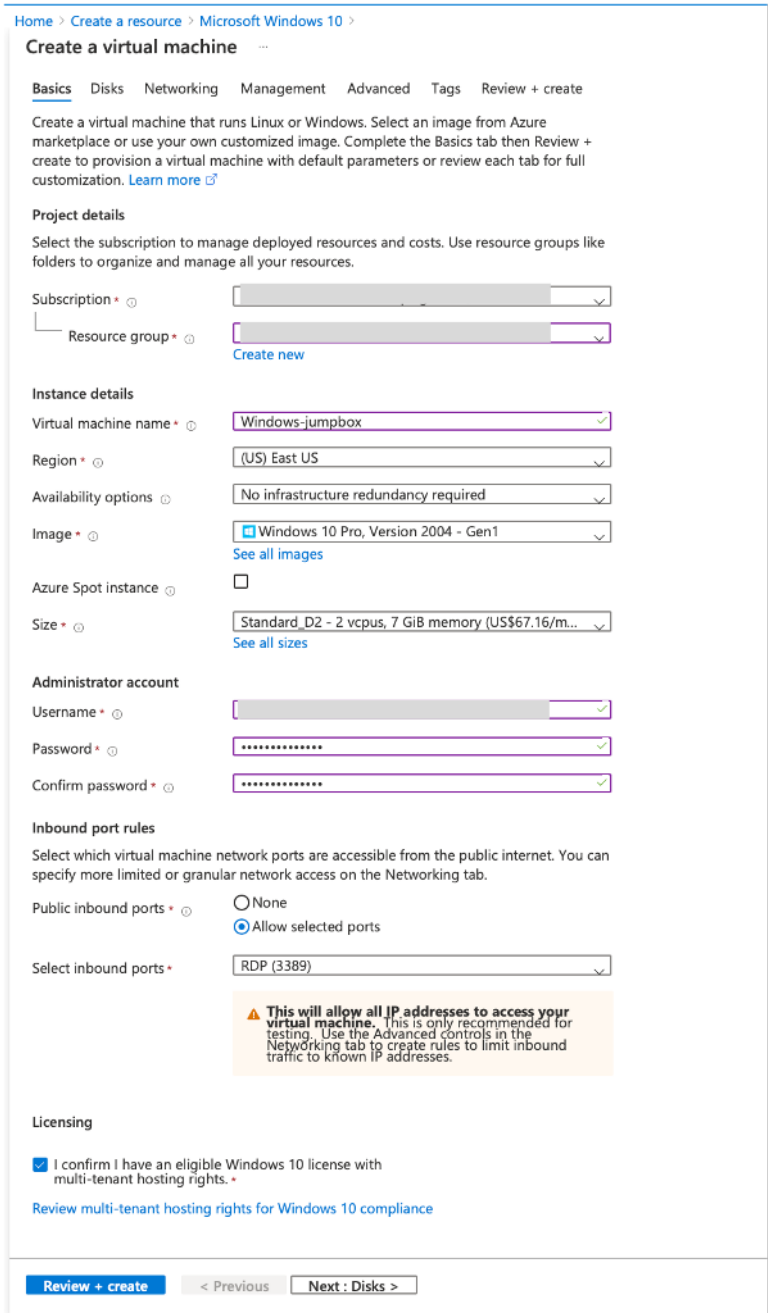
1. **Azure** ポータルに移動し、[リソースの作成] をクリックします。



2. **Microsoft Windows 10** を検索し、[作成] をクリックします。



3. Windows Server 2019 を実行する仮想マシン (VM) を作成します。[仮想マシンの作成] ページが表示されます。[基本] タブにすべての詳細を入力し、[ライセンス] チェックボックスをオンにします。残りのデフォルトのままにして、ページの下部にある [**Review + create**] ボタンを選択します。



4. 検証の実行後、ページの下部にある [作成] ボタンを選択します。
5. デプロイが完了したら、[リソースに移動] を選択します。
6. 作成した Windows 仮想マシンに移動します。Windows 仮想マシンのパブリック IP アドレスを使用し、RDP を使用して接続します。

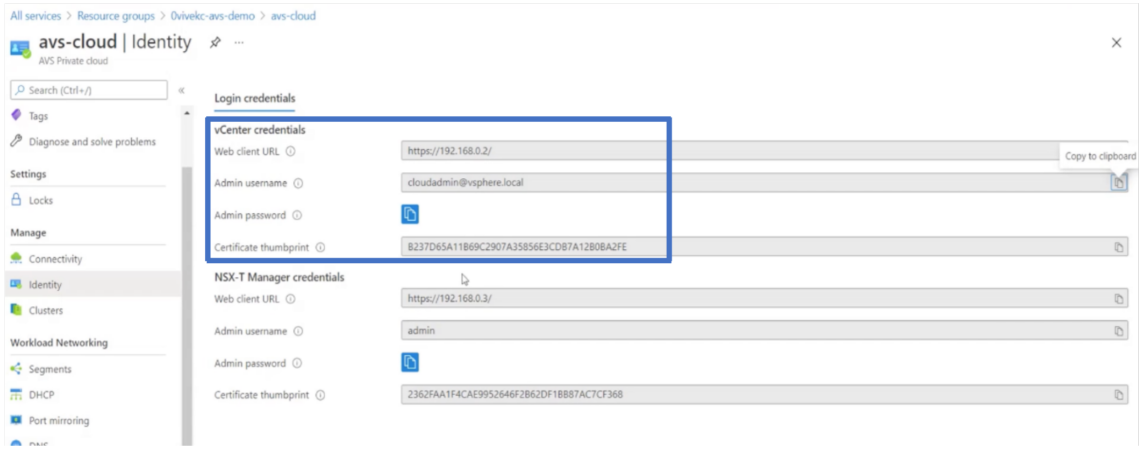
   Azure ポータルの [接続] ボタンを使用して、Windows デスクトップからリモートデスクトップ (RDP) セッションを開始します。まず仮想マシンに接続し、次にサインオンします。

   Mac から Windows 仮想マシンに接続するには、Microsoft リモートデスクトップなどの Mac 用 RDP クラ

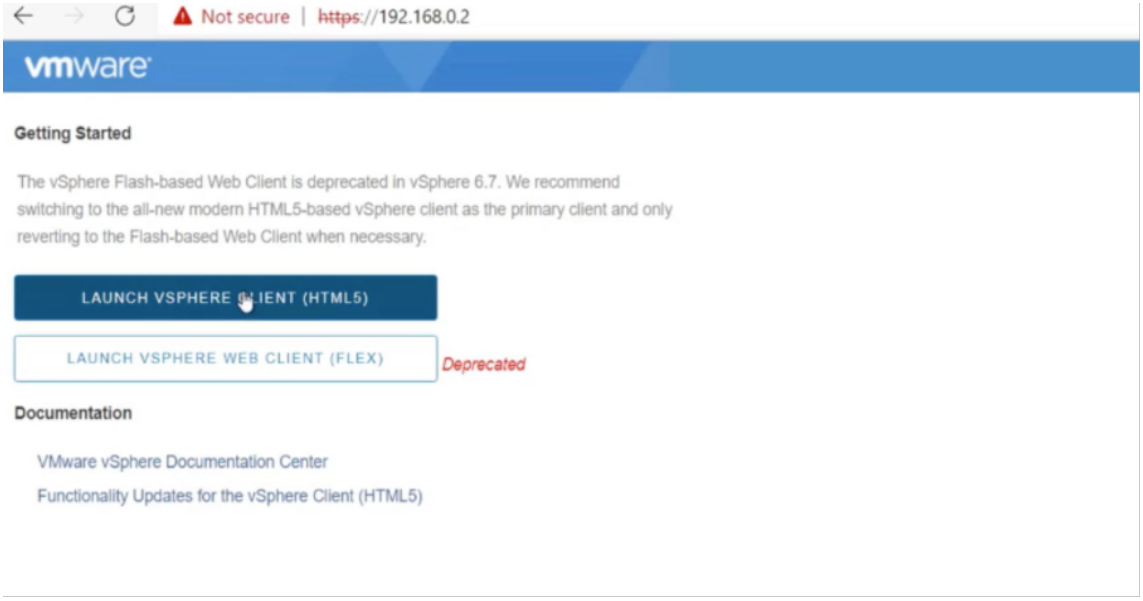
イアントをインストールする必要があります。詳細については、「Windows を実行する Azure 仮想マシンに接続してサインオンする方法」を参照してください。

プライベートクラウド **vCenter** ポータルにアクセスする

1. Azure VMware ソリューションのプライベートクラウドで、[管理] で [アイデンティティ] を選択します。vCenter の認証情報を書き留めます。

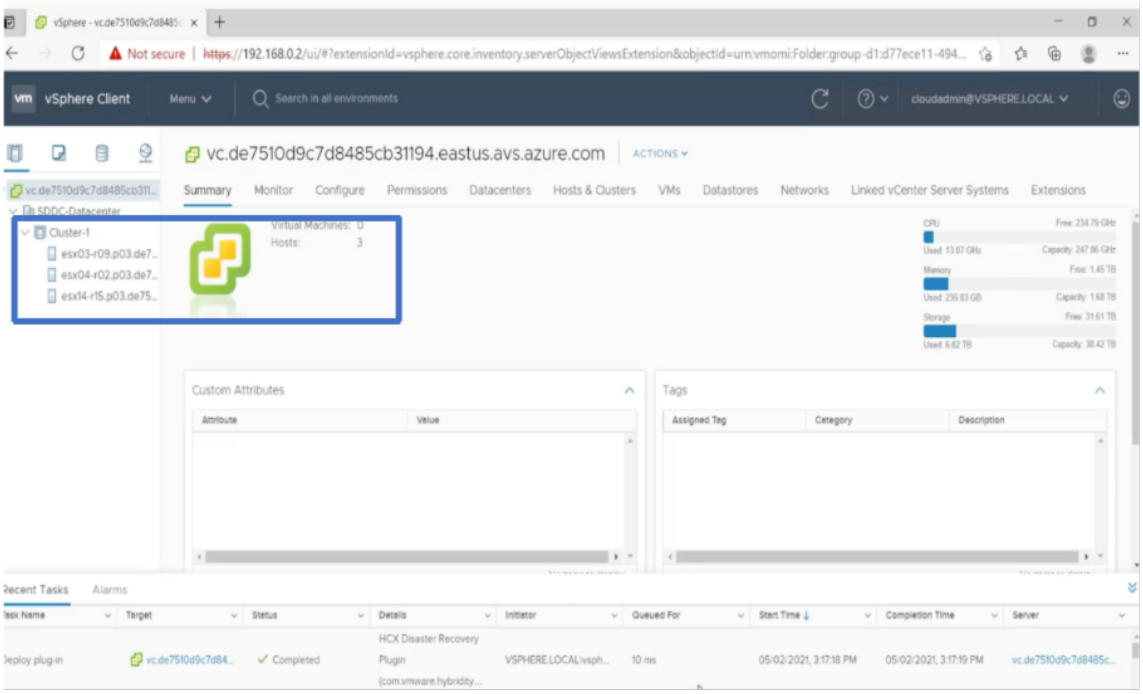


2. vCenter Web クライアントの URL を入力して、vSphere クライアントを起動します。



3. Azure VMware ソリューションプライベートクラウドの vCenter 認証情報を使用して VMware vSphere にログインします。

4. vSphere クライアントでは、Azure ポータルで作成した ESXi ホストを確認できます。



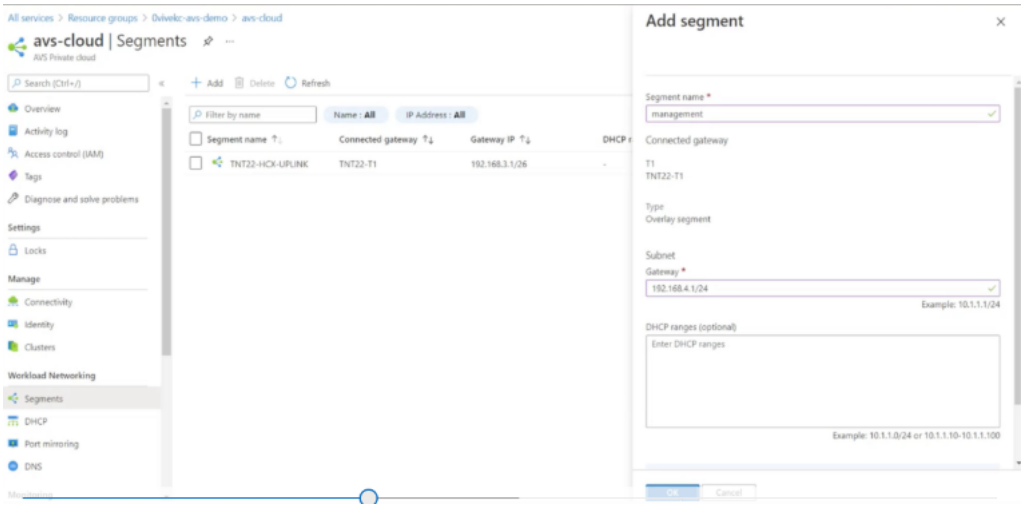
詳細については、「プライベートクラウド vCenter ポータルへのアクセス」を参照してください。

Azure ポータルで **NSX-T** セグメントを作成します

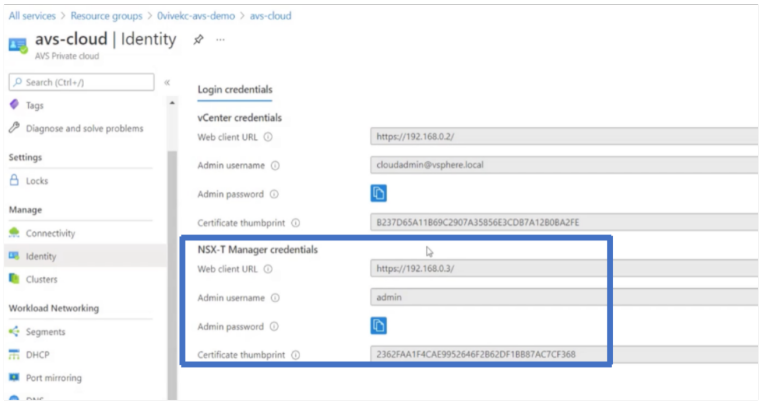
NSX-T セグメントは、Azure ポータルの Azure VMware ソリューションコンソールから作成および構成できます。これらのセグメントはデフォルトの Tier-1 ゲートウェイに接続され、これらのセグメントのワークロードは

East-West および North-South 接続を取得します。セグメントを作成すると、NSX-T Manager および vCenter に表示されます。

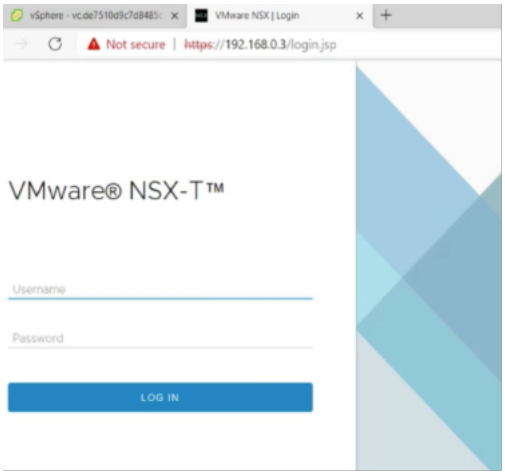
1. Azure VMware ソリューションのプライベートクラウドで、[ワークロードネットワーキング] で、[セグメント] **> [**追加] の順に選択します。新しい論理セグメントの詳細を入力し、「**OK**」を選択します。クライアント、管理、およびサーバーインターフェイスに対して 3 つの別々のセグメントを作成できます。



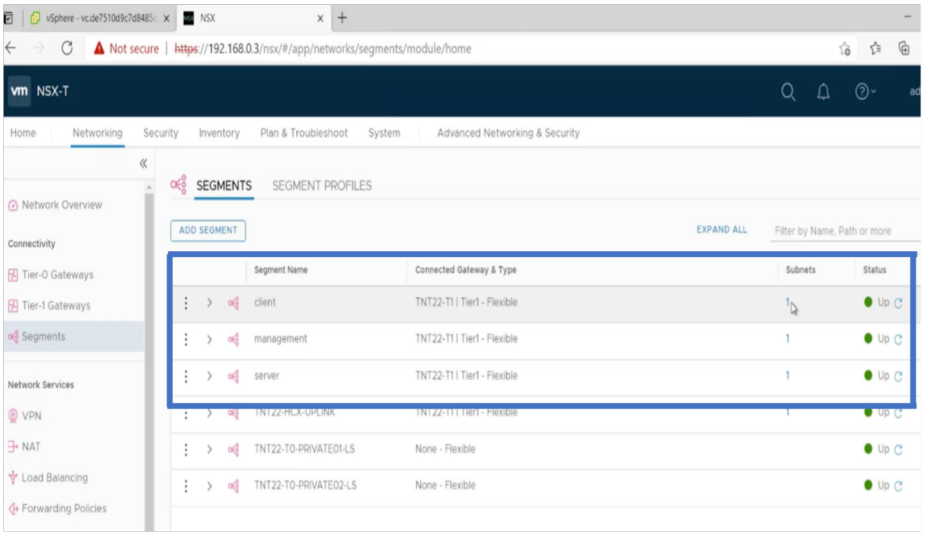
2. Azure VMware ソリューションのプライベートクラウドで、[管理] で [アイデンティティ] を選択します。NSX-T マネージャのクレデンシャルを書き留めます。



3. NSX-T Web クライアント URL を入力して VMware NSX-T マネージャを起動します。



4. NSX-T マネージャの [ネットワーク] **>** [セグメント] の下に、作成したすべてのセグメントが表示されます。サブネットを確認することもできます。



詳細については、「Azure ポータルで NSX-T セグメントを作成する」を参照してください。

VMware クラウドへの **Citrix ADC VPX** インスタンスのインストール

VMware ソフトウェア定義データセンター（SDDC）をインストールして構成したら、SDDC を使用して VMware クラウドに仮想アプライアンスをインストールできます。インストールできる仮想アプライアンスの数は、SDDC で使用可能なメモリの量によって異なります。

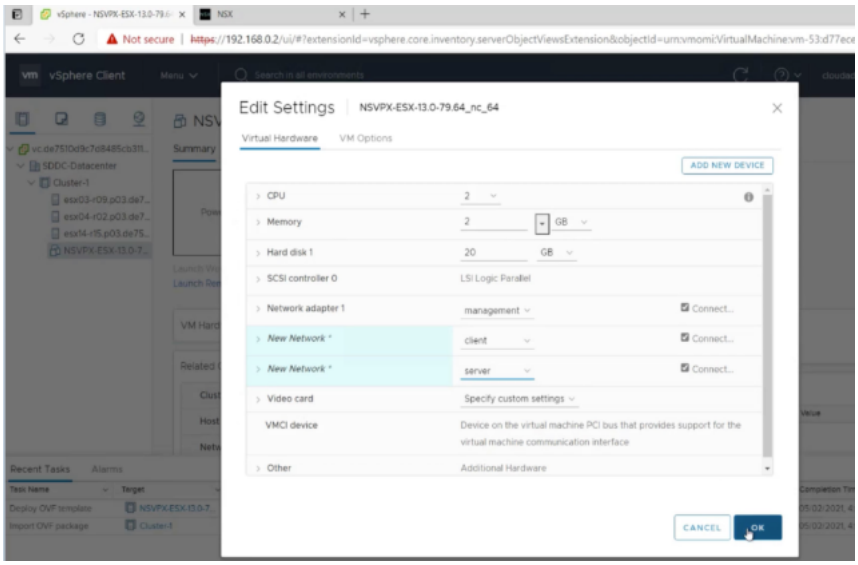
VMware クラウドに NetScaler VPX インスタンスをインストールするには、Windows ジャンプボックス仮想マシンで次の手順を実行します。

1. ESXi ホスト用の NetScaler VPX インスタンスセットアップファイルを、NetScaler ダウンロードサイトからダウンロードします。
2. Windows のジャンプボックスで VMware SDDC を開きます。

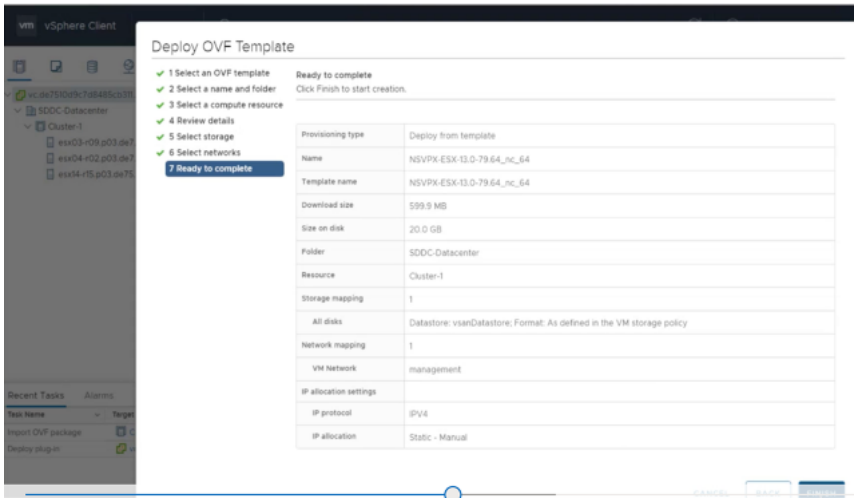
3. [ユーザー名] フィールドと [パスワード] フィールドに管理者の資格情報を入力し、[ログイン] をクリックします。
4. [**File**] メニューの [**Deploy OVF Template**] を選択します。
5. [**OVF** テンプレートのデプロイ] ダイアログボックスの [ファイルからの展開] フィールドで、NetScaler VPX インスタンスセットアップファイルを保存した場所を参照し、.ovf ファイルを選択し、[次へ] をクリックします。

   > 注
   >
   > デフォルトでは、NetScaler VPX インスタンスは E1000 ネットワークインターフェイスを使用します。VMXNET3 インターフェイスで ADC を展開するには、E1000 ではなく VMXNET3 インターフェイスを使用するように OVF を変更します。VMXNET3 インターフェイスの可用性は Azure インフラストラクチャによって制限され、Azure VMware ソリューションでは利用できない場合があります。

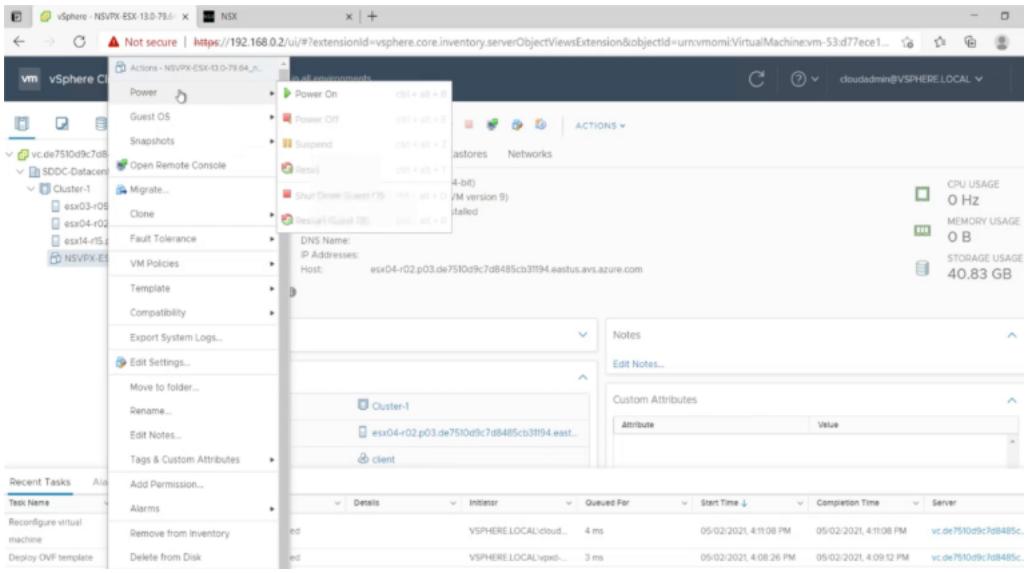
6. 仮想アプライアンス OVF テンプレートに表示されるネットワークを、VMware SDDC で設定したネットワークにマッピングします。[作成] または [**OK]** をクリックします。



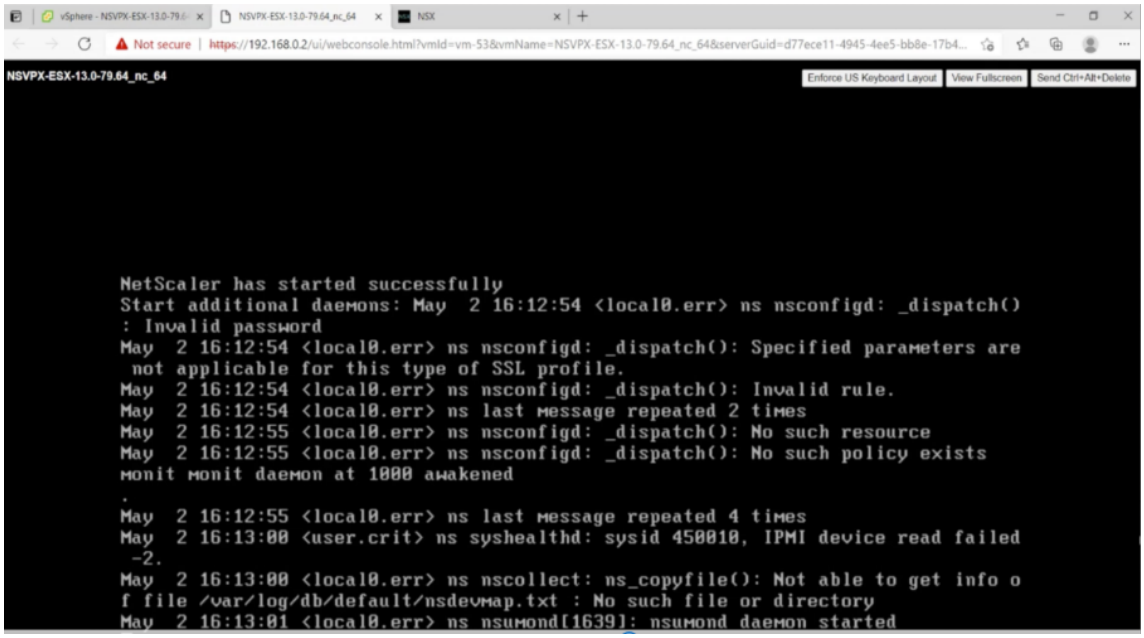
7. [完了] をクリックして VMware SDDC への仮想アプライアンスのインストールを開始します。



8. これで、NetScaler VPX インスタンスを起動する準備ができました。ナビゲーションペインで、インストールした Citrix ADC VPX インスタンスを選択し、右クリックメニューから［パワーオン］を選択します。コンソールポートをエミュレートするには、［**Console**］タブをクリックします。



9. これで、vSphere クライアントから NetScaler 仮想マシンに接続されています。



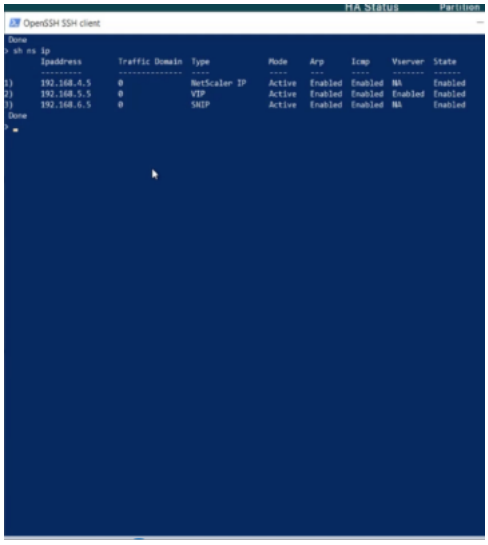
10. SSH キーを使用して NetScaler アプライアンスにアクセスするには、CLI で次のコマンドを入力します。

```
ssh nsroot@<management IP address>
```

例

```
ssh nsroot@192.168.4.5
```

11. ADC の設定は、`show ns ip`コマンドを使用して確認できます。

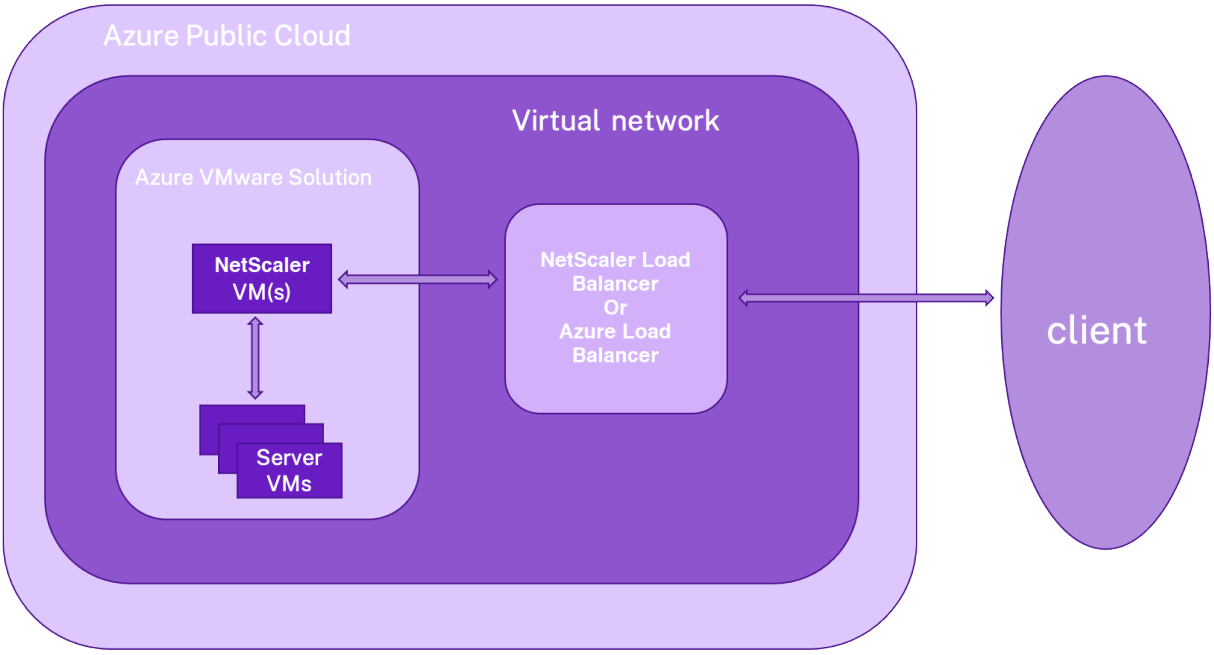


Azure VMware ソリューションでスタンドアロンの NetScaler ADC VPX インスタンスを構成する

October 17, 2024

インターネット向けアプリケーション用の Azure VMware ソリューション（AVS）上の NetScaler VPX スタンドアロンインスタンスを構成できます。

次の図は、Azure VMware ソリューション上の NetScaler VPX スタンドアロンインスタンスを示しています。クライアントは、AVS 内の NetScaler の仮想 IP（VIP）アドレスに接続することで AVS サービスにアクセスできます。これを実現するには、NetScaler ロードバランサーまたは Azure ロードバランサーインスタンスを AVS の外部で同じ Azure 仮想ネットワーク内にプロビジョニングします。AVS サービス内の NetScaler VPX インスタンスの VIP にアクセスするようにロードバランサーを構成します。



前提条件

仮想アプライアンスのインストールを開始する前に、次の Azure の前提条件をお読みください。

- Azure VMware ソリューションとその前提条件の詳細については、Azure VMware ソリューションのドキュメントを参照してください。
- Azure VMware ソリューションのデプロイの詳細については、「Azure VMware ソリューションのプライベートクラウドをデプロイする」を参照してください。
- Azure VMware ソリューションにアクセスして管理するための Windows ジャンプボックス仮想マシンの作成の詳細については、「Azure VMware ソリューションのプライベートクラウドへのアクセス」を参照してください。

- Windows ジャンプボックス仮想マシンで、NetScaler VPX アプライアンスセットアップファイルをダウンロードします。
- 仮想マシンが接続する VMware SDDC 上に、適切な NSX-T ネットワークセグメントを作成します。詳細については、「Azure VMware ソリューションでのネットワークセグメントの追加」を参照してください。
- VMware クラウドに NetScaler VPX インスタンスをインストールする方法の詳細については、「VMware クラウドに NetScaler VPX インスタンスをインストールする」を参照してください。

NetScaler ロードバランサーを使用して **AVS** 上の **NetScaler VPX** スタンドアロンインスタンスを構成します

次の手順に従って、NetScaler ロードバランサーを使用するインターネット向けアプリケーション用に AVS 上の NetScaler VPX スタンドアロンインスタンスを構成します。

1. NetScaler VPX インスタンスを Azure クラウドにデプロイします。詳細については、「NetScaler VPX スタンドアロン インスタンスの構成」を参照してください。

   > 注
   >
   > Azure VMware Cloud と同じ仮想ネットワークにデプロイされていることを確認します。

2. AVS にデプロイされた NetScaler VPX の VIP アドレスにアクセスするように NetScaler VPX インスタンスを構成します。

   a) 負荷分散仮想サーバーを追加します。

   ```
   add lb vserver <name> <serviceType> [<vip>] [<port>]
   ```

   例

   ```
   add lb vserver lb1 HTTPS 172.31.0.6 443
   ```

   b) AVS にデプロイされた NetScaler VPX IP に接続するサービスを追加します。

   ```
   add service <name> <ip> <serviceType> <port>
   ```

   例

   ```
   add service webserver1 192.168.4.10 HTTP 80
   ```

   c) サービスを負荷分散仮想サーバーにバインドします。

   ```
   bind lb vserver <name> <serviceName>
   ```

   例

   ```
   bind lb vserver lb1 webserver1
   ```


Azure ロードバランサーを使用して **AVS** 上の **NetScaler VPX** スタンドアロンインスタンスを構成する

以下の手順に従って、Azure ロードバランサーを使用するインターネット向けアプリケーション用に AVS 上の NetScaler VPX スタンドアロンインスタンスを構成します。

1. Azure クラウドで Azure ロードバランサーインスタンスを構成します。詳しくは、ロードバランサーの作成に関する Azure ドキュメントを参照してください。
2. AVS にデプロイされている NetScaler VPX インスタンスの VIP アドレスをバックエンドプールに追加します。

   次の Azure コマンドは、1 つのバックエンド IP アドレスを負荷分散バックエンドアドレスプールに追加します。

```
az network lb address-pool address add
                                    --resource-group <Azure VMC
                                        Resource Group>
                                    --lb-name <LB Name>
                                    --pool-name <Backend pool
                                        name>
                                    --vnet <Azure VMC Vnet>
                                    --name <IP Address name>
                                    --ip-address <VIP of ADC in
                                        VMC>
```

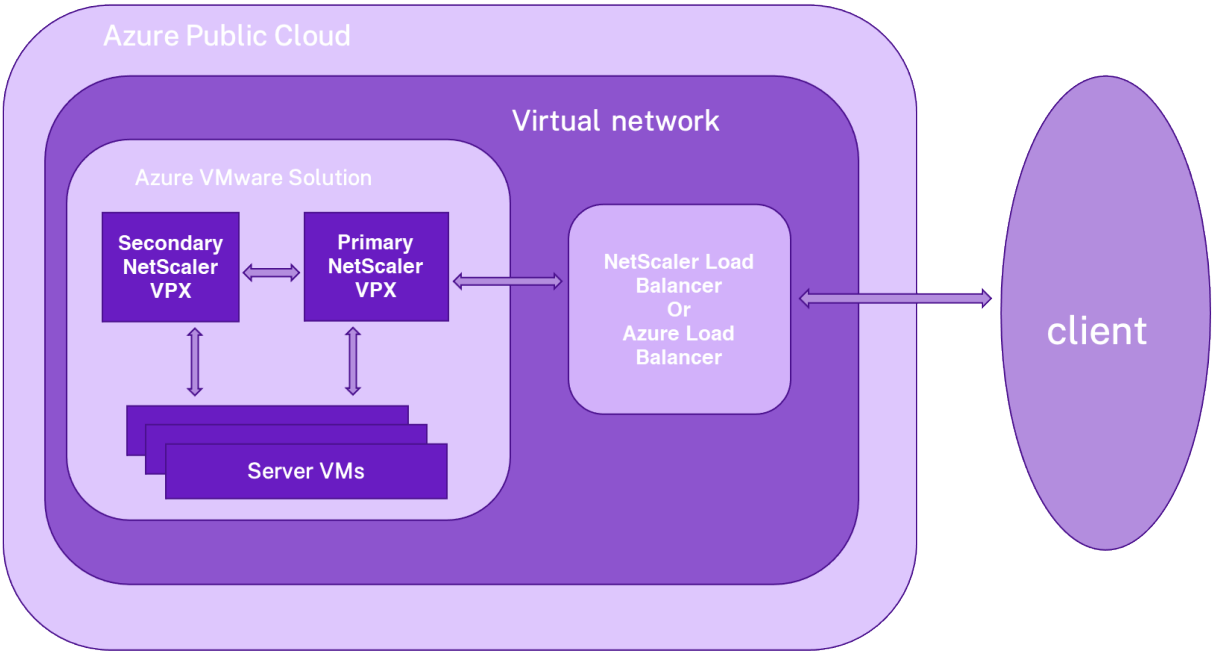
   > 注
   >
   > Azure ロードバランサーが Azure VMware クラウドと同じ仮想ネットワークにデプロイされていることを確認します。

Azure VMware ソリューションで **Citrix ADC VPX** の高可用性セットアップを構成する

October 17, 2024

インターネットに接続するアプリケーション用の Azure VMware ソリューション（AVS）で NetScaler VPX HA セットアップを構成できます。

次の図は、AVS 上の NetScaler VPX HA ペアを示しています。クライアントは、AVS 内のプライマリ ADC ノードの VIP に接続することで、AVS サービスにアクセスできます。これを実現するには、NetScaler ロードバランサーまたは Azure ロードバランサーインスタンスを AVS の外部で同じ Azure 仮想ネットワーク内にプロビジョニングします。AVS サービス内のプライマリ ADC ノードの VIP にアクセスするようにロードバランサを設定します。



前提条件

仮想アプライアンスのインストールを開始する前に、次の Azure の前提条件をお読みください。

- Azure VMware ソリューションとその前提条件の詳細については、Azure VMware ソリューションのドキュメントを参照してください。
- Azure VMware ソリューションのデプロイの詳細については、「Azure VMware ソリューションのプライベートクラウドをデプロイする」を参照してください。
- Azure VMware ソリューションにアクセスして管理するための Windows ジャンプボックス仮想マシンの作成の詳細については、「Azure VMware ソリューションのプライベートクラウドへのアクセス」を参照してください。
- Windows ジャンプボックス仮想マシンで、NetScaler VPX アプライアンスセットアップファイルをダウンロードします。
- 仮想マシンが接続する VMware SDDC 上に、適切な NSX-T ネットワークセグメントを作成します。詳細については、「Azure VMware Solution でのネットワークセグメントの追加」を参照してください。

構成の手順

以下の手順に従って、インターネット向けアプリケーションの AVS で NetScaler VPX 高可用性セットアップを構成します。

1. VMware クラウド上に 2 つの NetScaler VPX インスタンスを作成します。詳細については、「VMware クラウドに NetScaler VPX インスタンスをインストールする」を参照してください。

2. NetScaler HA のセットアップを構成します。詳細については、「高可用性の構成」を参照してください。
3. インターネットに接続されたアプリケーションにアクセスできるように NetScaler HA セットアップを構成します。

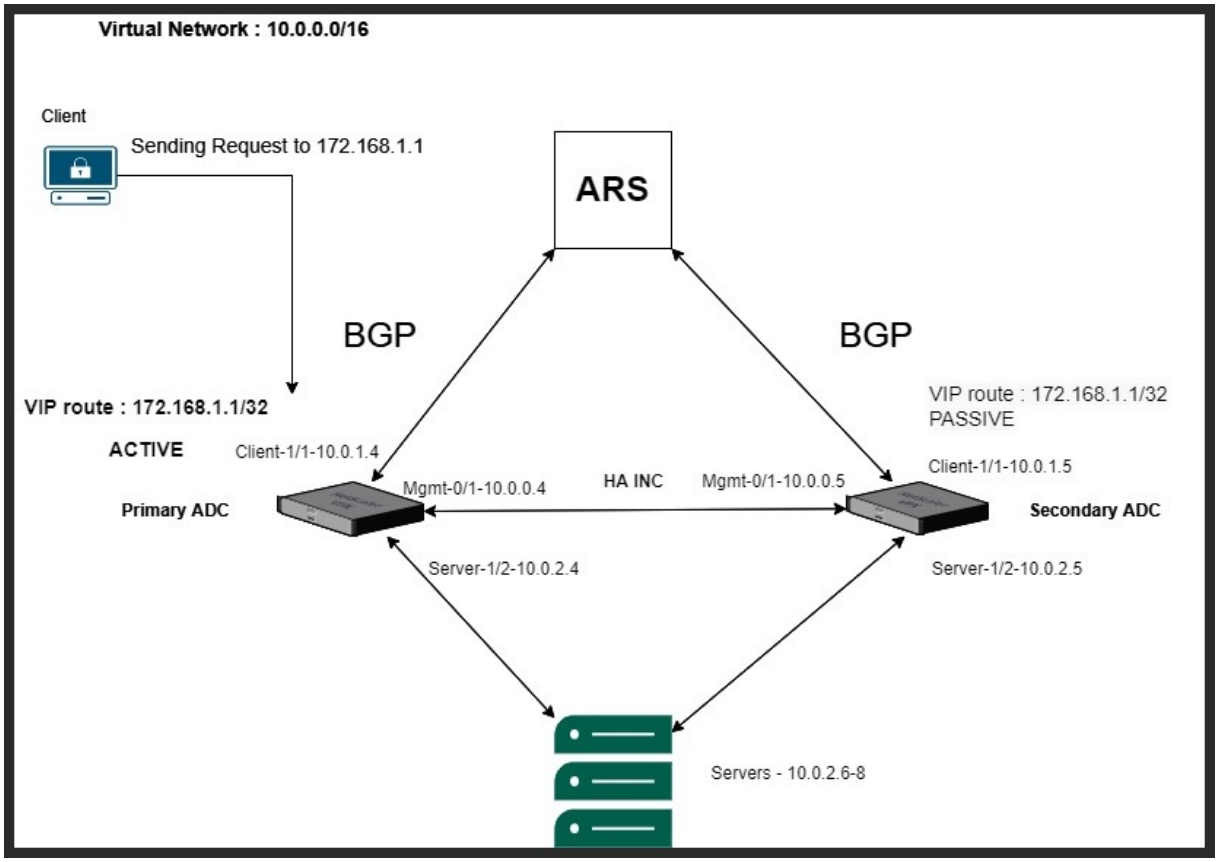
    - NetScaler ロードバランサーを使用して NetScaler VPX インスタンスを構成するには、「NetScaler ロードバランサーを使用して AVS 上に NetScaler VPX スタンドアロンインスタンスを構成する」を参照してください。
    - Azure ロード バランサーを使用して NetScaler VPX インスタンスを構成するには、「Azure ロード バランサーを使用して AVS で NetScaler VPX スタンドアロン インスタンスを構成する」を参照してください。

NetScaler VPX HA ペアで Azure ルートサーバーを構成する

October 17, 2024

NetScaler VPX インスタンスを使用して Azure ルートサーバーを構成し、BGP プロトコルを使用して仮想ネットワークで構成された VIP ルートを交換できます。Citrix ADC は、スタンドアロンまたは HA-INC モードで展開し、BGP で構成できます。この展開では、ADC HA ペアの前に Azure ロードバランサー (ALB) は必要ありません。

次の図は、VPX HA トポロジが Azure ルートサーバーとどのように統合されるかを示しています。各 ADC インスタンスには、管理用、クライアントトラフィック用、サーバートラフィック用の 3 つのインターフェイスがあります。



トポロジ図では、次の IP アドレスを使用します。

プライマリ **ADC** インスタンスの **IP** 設定の例**:**

```
NSIP: 10.0.0.4/24
SNIP on 1/1: 10.0.1.4/24
SNIP on 1/2: 10.0.2.4/24
VIP: 172.168.1.1/32
```

セカンダリ **ADC** インスタンスの **IP** 設定の例**:**

```
NSIP: 10.0.0.5/24
SNIP on 1/1: 10.0.1.5/24
SNIP on 1/2: 10.0.2.5/24
VIP: 172.168.1.1/32
```

前提条件

NetScaler VPX インスタンスを Azure に展開する前に、次の情報を理解している必要があります。

- Azure の用語とネットワークの詳細。詳細については、「Azure 用語」を参照してください。
- Azure Portal にルートサーバーを作成します。詳細については、「Azure portal を使用したルートサーバーの作成と構成」を参照してください。

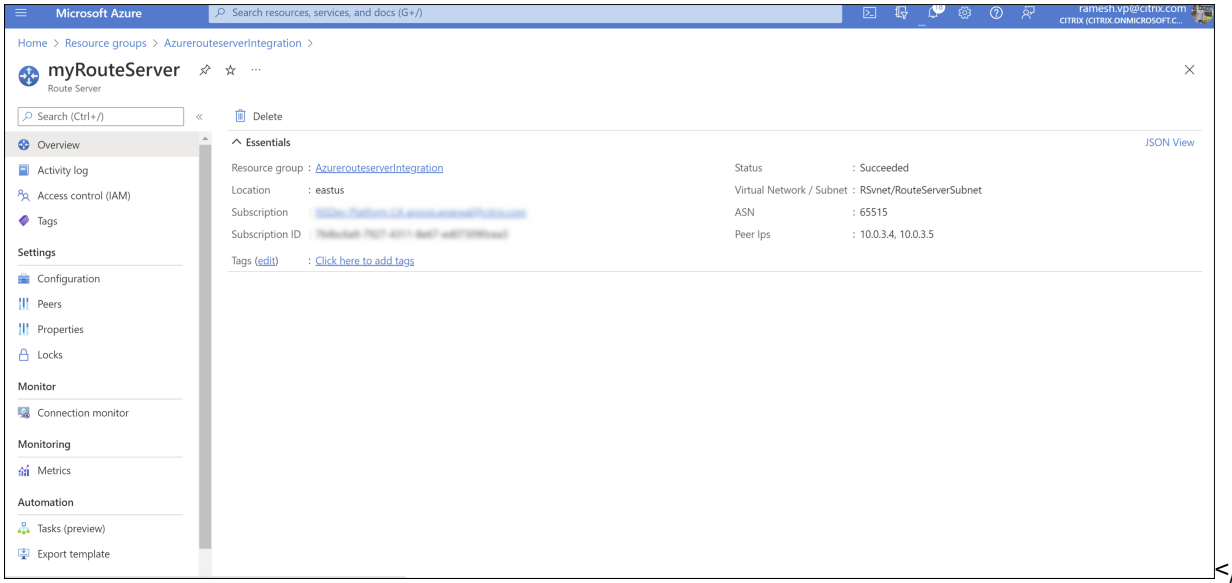
- NetScaler アプライアンスの動作。詳しくは、NetScaler のドキュメントを参照してください。
- NetScaler ネットワーキング。詳細については、ADC ネットワークを参照してください。

NetScaler VPX HA ペアで Azure ルートサーバーを構成する方法

1. Azure ポータルでルート サーバーを作成します。詳細については、「

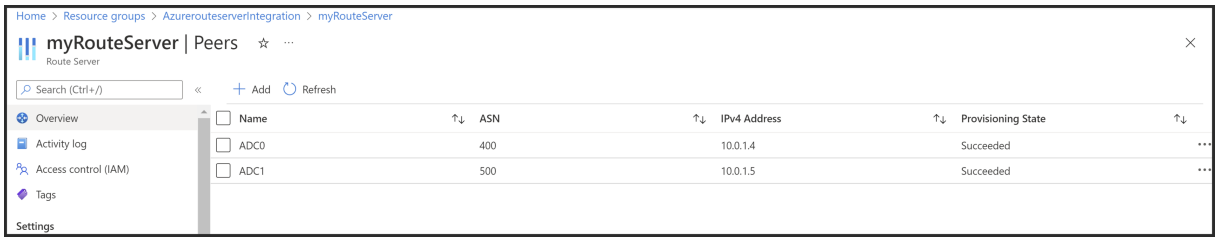
   Azure Route Server とは」を参照してください。</p>

次の例では、サブネット 10.0.3.0/24 が Azure サーバーのデプロイに使用されます。ルートサーバーが作成されたら、ルートサーバーの IP アドレスを取得します (例:10.0.3.4、10.0.3.5)。



1 Azure Portal でネットワーク仮想アプライアンス (NVA) とのピアリングを設定します。NetScaler VPX インスタンスを NVA として追加します。詳細については、「NVA とのピアリングの設定」を参照してください。

次の例では、1/1 インターフェイスの ADC SNIP（10.0.1.4 と 10.0.1.5）と ASN: 400 と 500 がピアの追加時に使用されます。



1 高可用性構成用に 2 つの NetScaler ADC VPX インスタンスを追加します。

次の手順を実行します:

```
1.  Azureに2つのVPXインスタンス（プライマリインスタンスとセカンダリインスタンス）をデプロイします。
1.  両方のインスタンスにクライアントとサーバーの NIC を追加します。
```


1. NetScaler GUIを使用して、両方のインスタンスで高可用性設定を構成します。 1 プライマリ ADC インスタンスで動的ルーティングを設定します。

設定例:

```
  enable ns mode L3 MBF USNIP SRADV DRADV  PMTUD
  enable ns feature LB BGP
  add ns ip 10.0.1.4 255.255.255.0 -vServer DISABLED -dynamicRouting
     ENABLED
  VTYSH
  configure terminal
  router BGP 400
  timers bgp 1 3
  neighbor 10.0.3.4 remote-as 65515
  neighbor 10.0.3.4 advertisement-interval 3
  neighbor 10.0.3.4 fall-over  bfd
  neighbor 10.0.3.5 remote-as 65515
  neighbor 10.0.3.5 advertisement-interval 3
  neighbor 10.0.3.5 fall-over  bfd
  address-family ipv4
  redistribute kernel
  redistribute static
```

1 セカンダリ ADC インスタンスで動的ルーティングを設定します。

設定例:

```
  enable ns mode L3 MBF USNIP SRADV DRADV  PMTUD
  enable ns feature LB BGP
  add ns ip 10.0.1.5 255.255.255.0 -vServer DISABLED -dynamicRouting
     ENABLED
  VTYSH
  configure terminal
  router BGP 500
  timers bgp 1 3
  neighbor 10.0.3.4 remote-as 65515
  neighbor 10.0.3.4 advertisement-interval 3
  neighbor 10.0.3.4 fall-over  bfd
  neighbor 10.0.3.5 remote-as 65515
  neighbor 10.0.3.5 advertisement-interval 3
  neighbor 10.0.3.5 fall-over  bfd
  address-family ipv4
  redistribute kernel
  redistribute static
```

1 VTY シェルインターフェイスで BGP コマンドを使用して確立された BGP ピアを確認します。詳細については、「BGP 設定の確認」を参照してください。

```
```



```
  show ip bgp neighbors
```

1 プライマリ ADC インスタンスで LB 仮想サーバーを設定します。

設定例**:**

```
  add ns ip 172.16.1.1 255.255.255.255 -type VIP -hostRoute ENABLED
  add lbvserver v1 HTTP 172.16.1.1 80
  add service s1 10.0.2.6 HTTP 80
  bind lbvserver v1 s1
  enable ns feature lb
```

NetScaler VPX インスタンスと同じ仮想ネットワーク内のクライアントが、LB 仮想サーバーにアクセスできるようになりました。この場合、NetScaler VPX インスタンスは VIP ルートを Azure ルートサーバーにアドバタイズします。

バックエンドの **Azure** 自動スケーリングサービスを追加

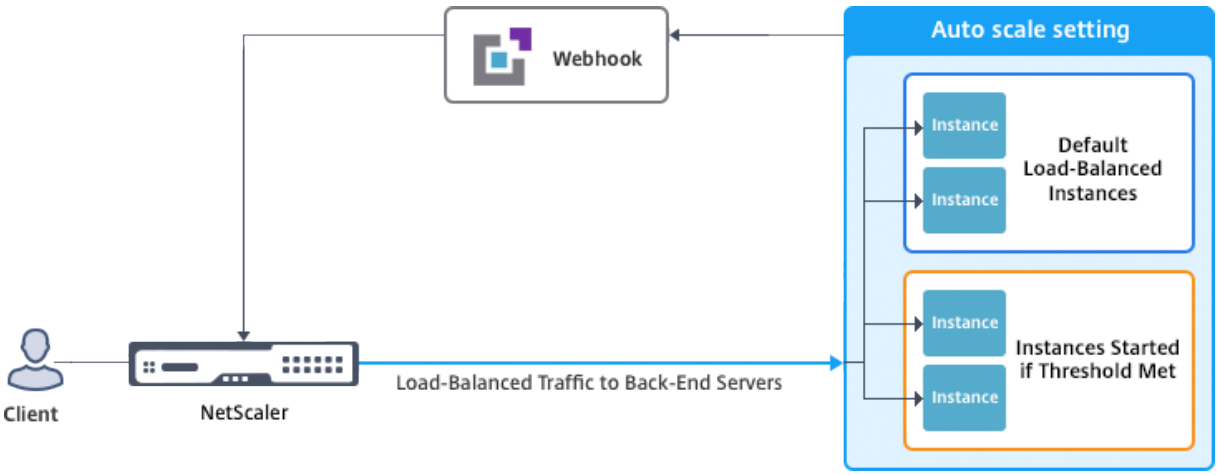
October 17, 2024

クラウドでアプリケーションを効率的にホストすると、アプリケーションの必要に応じて、リソースを簡単にコスト効率よく管理できます。増加する需要に対応するには、ネットワークリソースをスケールアップする必要があります。需要が収まるかどうかにかかわらず、アイドル状態のリソースの不必要なコストを避けるためにスケールダウンする必要があります。アプリケーションの実行コストを最小限に抑えるには、トラフィック、メモリ、CPU の使用などを常に監視する必要があります。しかし、トラフィックを手動で監視するのは煩雑です。アプリケーション環境を動的にスケールアップまたはスケールダウンするには、トラフィックを監視し、必要に応じてリソースをスケールアップまたはスケールダウンするプロセスを自動化する必要があります。

Azure での VPX マルチ IP スタンドアロンおよび高可用性のデプロイには、Azure 仮想マシンスケールセット (VMSS) で Autoscale を使用できます。

Azure VMSS および AAutoscale 機能と統合された NetScaler VPX インスタンスには、次の利点があります。

- 負荷分散と管理: 需要に応じてサーバーをスケールアップおよびスケールダウンするように自動構成します。NetScaler VPX インスタンスは、VPX インスタンスが展開されているのと同じ仮想ネットワーク、または同じ Azure サブスクリプション内のピアリングされた仮想ネットワーク内の VMS AAutoscale e 設定を自動検出します。VMSS Autoscale 設定を選択して、負荷を分散できます。これは、VPX インスタンスで NetScaler 仮想 IP アドレスとサブネット IP アドレスを自動構成することによって行われます。
- 高可用性:Autoscale グループを検出し、サーバーの負荷を分散します。
- ネットワークの可用性の向上：VPX インスタンスは、異なる仮想ネットワーク（VNet）上のバックエンドサーバーをサポートします。



詳細については、次の Azure トピックを参照してください。

- 仮想マシンのスケールセットのドキュメント
- Microsoft Azure 仮想マシン、クラウドサービス、および Web アプリケーションの Autoscale の概要

はじめに

- Azure 関連の使用に関するガイドラインを参照してください。詳細については、「Microsoft Azure に NetScaler VPX インスタンスをデプロイする」を参照してください。
- 要件（スタンドアロンまたは高可用性デプロイ）に応じて、Azure 上に 3 つのネットワークインターフェイスを使用して 1 つまたは複数の NetScaler VPX インスタンスを作成します。
- VPX インスタンスの 0/1 インターフェイスのネットワークセキュリティグループで TCP 9001 ポートを開きます。VPX インスタンスは、このポートを使用してスケールアウトおよびスケールイン通知を受け取ります。
- NetScaler VPX インスタンスが展開されている同じ仮想ネットワークに Azure VMSS を作成します。VMSS と NetScaler VPX インスタンスが異なる Azure 仮想ネットワークに展開されている場合、次の条件を満たす必要があります。

  - 両方の仮想ネットワークが同じ Azure サブスクリプションに含まれている必要があります。
  - 2 つの仮想ネットワークは、Azure の仮想ネットワークピアリング機能を使用して接続する必要があります。

  既存の VMSS 設定がない場合は、次のタスクを完了します：

  a）VMSS の作成

  b) VMSS でオートスケールを有効にする

  c) VMSS Autoscale 設定でスケールインポリシーとスケールアウトポリシーを作成する

  詳細については、「Azure 仮想マシンのスケールセットを使用した Autoscale の概要」を参照してください。

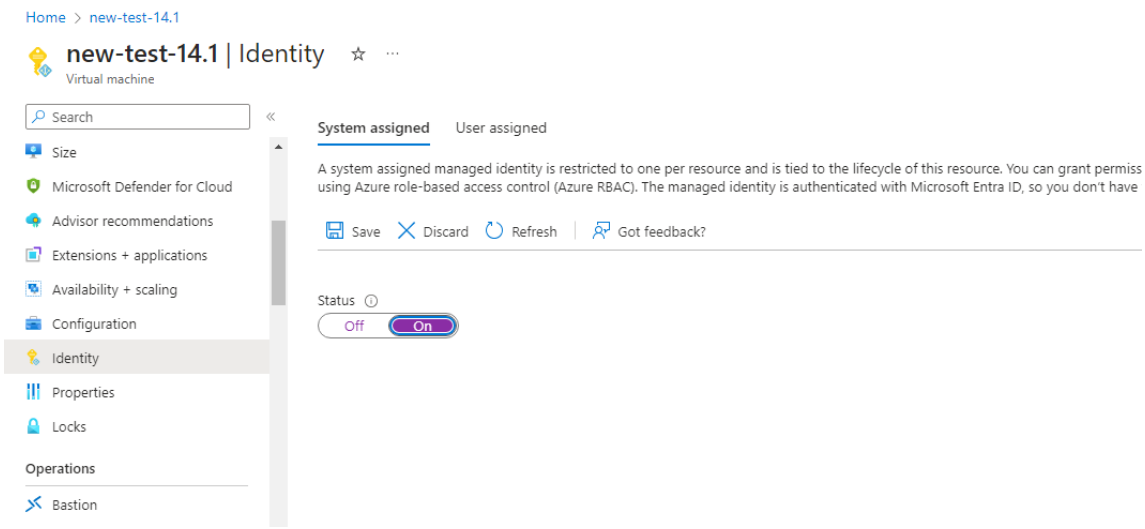
- NetScaler VPX は、ユニフォームオーケストレーションを使用する VMSS のみをサポートしています。フレキシブルオーケストレーション機能を備えた VMSS はサポートされていません。詳細については、「Azure の仮想マシンスケールセットのオーケストレーションモード」を参照してください。
- NetScaler リリース 14.1-12.x 以降、NetScaler VPX は Azure クラウドのマネージド ID をサポートしています。マネージド ID は、サービスプリンシパルを仮想マシンなどの Azure リソースにリンクします。マネージド ID では、クラウド認証情報 (アプリケーション ID、アプリケーションシークレット、テナント ID) を管理する必要がないため、セキュリティリスクを回避できます。現在、NetScaler VPX は、システムによって割り当てられた管理対象 ID と単一ユーザーが割り当てた管理対象 ID のみをサポートしています。複数ユーザーに割り当てられたマネージド ID はサポートされていません。

  14.1-12.x より前の NetScaler リリースでは、Azure Active Directory（AAD）を介して NetScaler VPX クラウド認証情報を手動で管理する必要があります。新しく作成した AAD アプリケーションに貢献者の役割を割り当てます。クラウド認証情報は、有効期限が切れた後、定期的に再作成する必要があります。詳細については、「Azure Active Directory アプリケーションとサービスプリンシパルの作成」を参照してください。

  Azure コンソールでマネージド ID を構成し、NetScaler でクラウド認証情報を構成すると、マネージド ID がクラウド認証情報よりも優先されます。

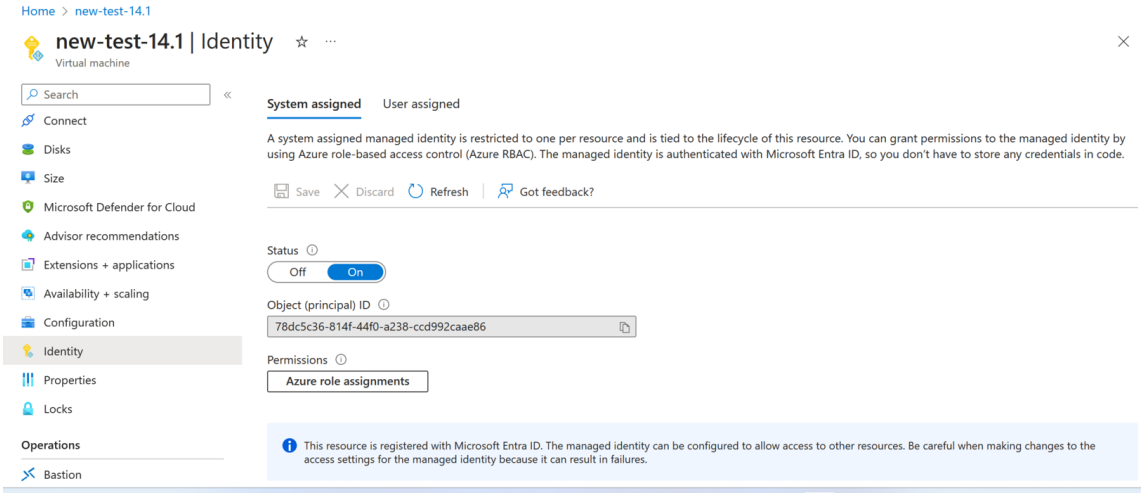
仮想マシンでのマネージド **ID** の設定

1. Azure Portal にサインインします。
2. 仮想マシンに移動し、[**ID**] を選択します。
3. 要件に応じて、[システム割り当て] または [ユーザー割り当て] のいずれかを選択します。
4. [ステータス] で [オン] を選択し、[保存] をクリックします。



ステータスが保存されると、サービスプリンシパルオブジェクトが作成され、VM に割り当てられていることがわかります。

5.「**Azure** ロール割り当て」をクリックします。



6.「ロール割り当ての追加」ウィンドウで、スコープを選択します。次のオプションから選択できます:

- サブスクリプション

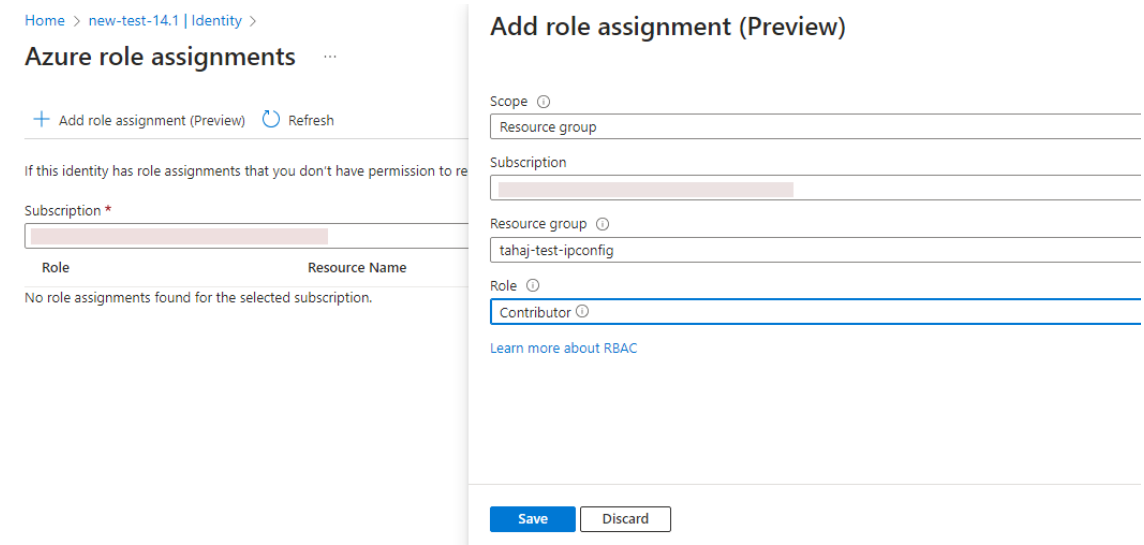
  VMSS と VM が異なるリソースグループにある場合は、スコープとして **Subscription** を使用してください。

- リソースグループ

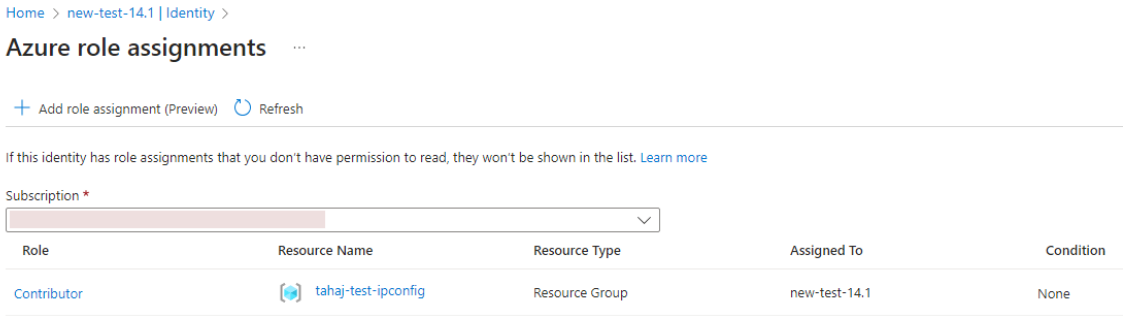
  VMSS が VM と同じリソースグループにある場合は、リソースグループをスコープとして使用します。

- Key Vault
- ストレージ
- SQL

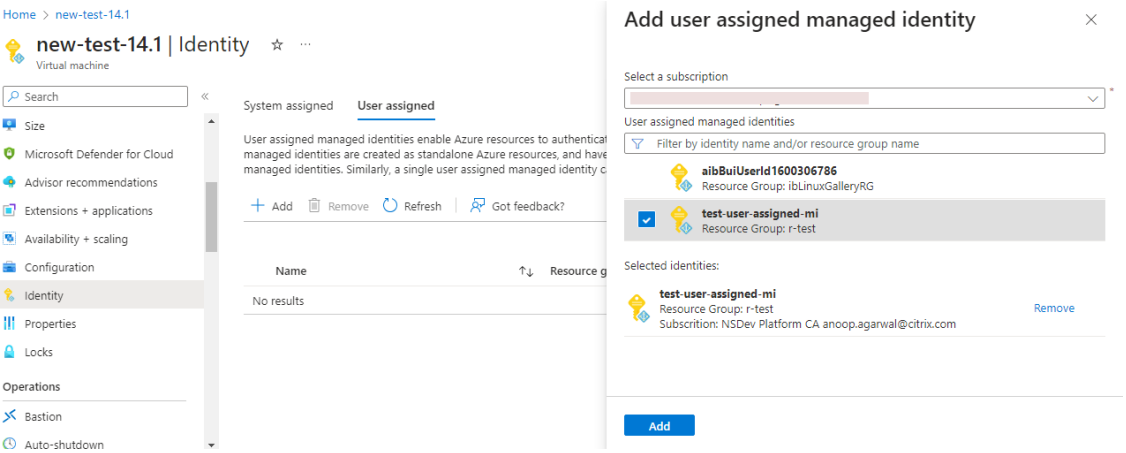
選択した範囲に基づいて、他のフィールドの詳細を入力します。寄稿者の役割を割り当て、構成を保存します。



Azure ロール割り当てページには、作成したマネージド ID が表示されます。



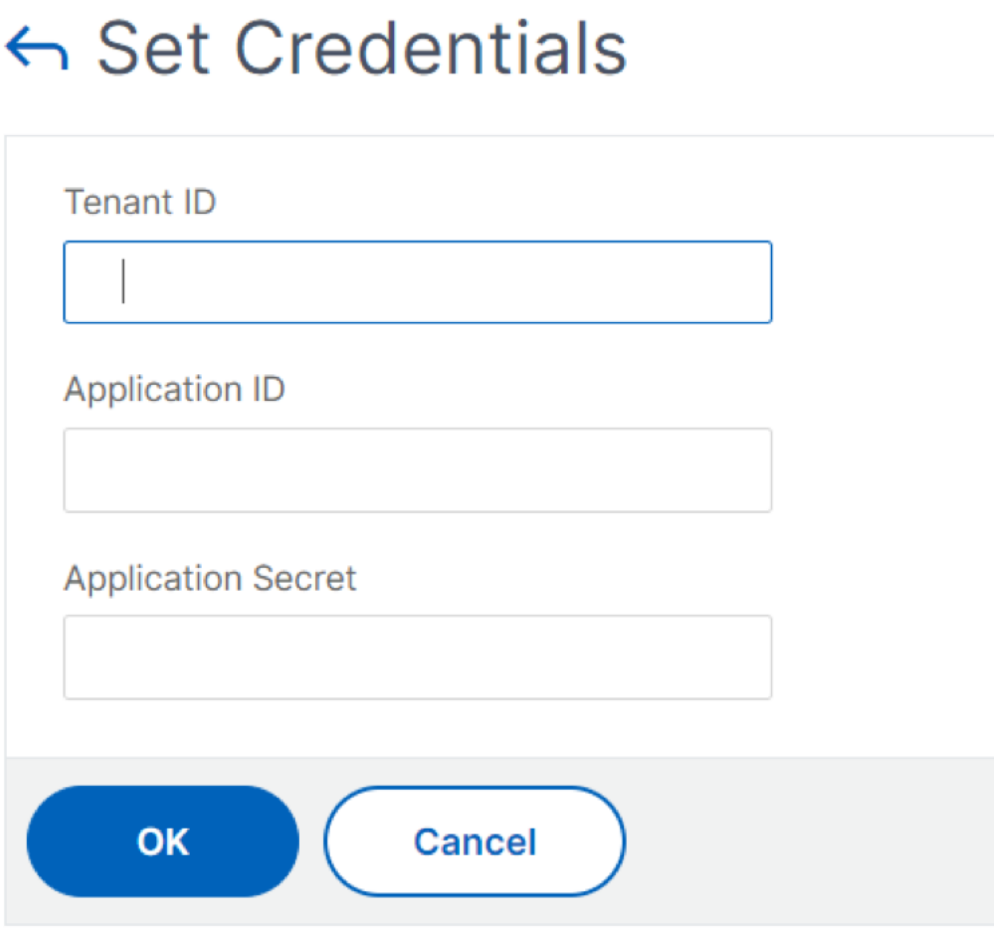
7. ユーザー割り当てマネージド ID を作成するには、サブスクリプションを選択し、ユーザー割り当てマネージド ID を選択して、「追加」をクリックします。



VMSS を **NetScaler VPX** インスタンスに追加する

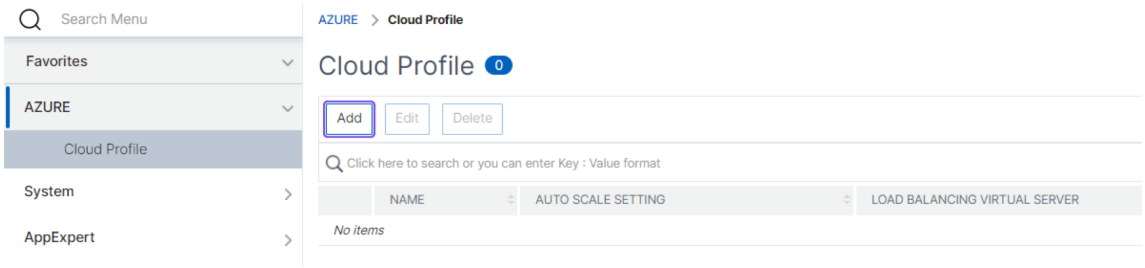
次のステップを実行して、VPX インスタンスに Autoscale e 設定を追加します:

1. VPX インスタンスにログオンします。
2. [構成] **> [Azure] > [**認証情報の設定] に移動します。Autoscale 機能を機能させるために必要な Azure 認証情報を追加します。

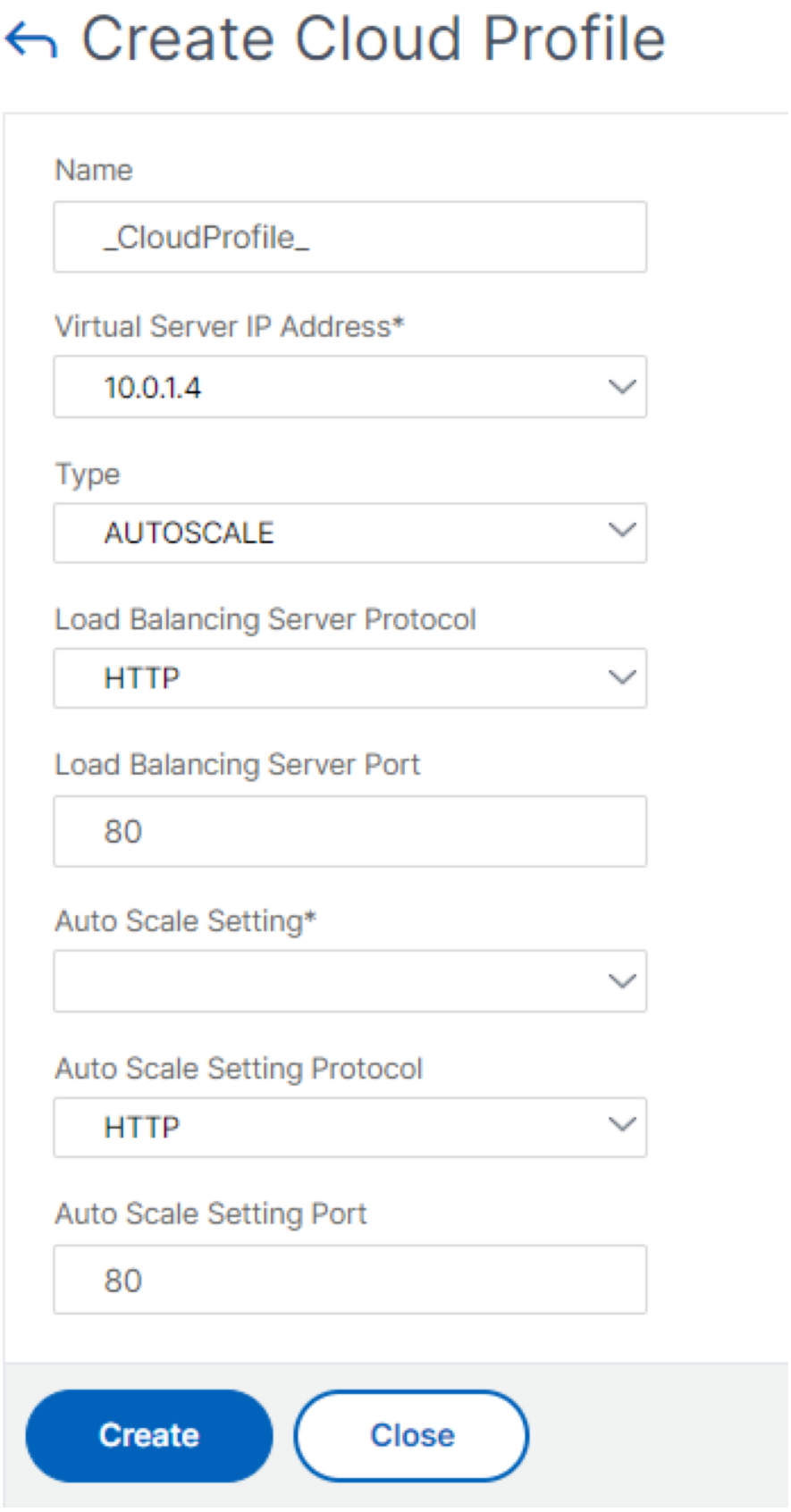


> 注
>
> Azure マネージド ID を使用している場合は、認証情報を設定する必要はありません。

3. [システム**] > [Azure] > [**クラウドプロファイル] に移動し、[追加] をクリックしてクラウドプロファイルを作成します。



クラウドプロファイルの作成設定ページが表示されます。



クラウドプロファイルは、NetScaler 負荷分散仮想サーバーと、Auto Scaling グループのサーバーとしてメンバー（サーバー）を持つサービスグループを作成します。バックエンドサーバーは、VPX インスタンスで構成された SNIP を介して到達可能である必要があります。

クラウドプロファイルの作成時に留意すべきポイント

- 仮想サーバーの IP アドレスは、VPX インスタンスで使用可能な空き IP アドレスから自動的に設定されます。詳細については、「Azure ポータルを使用して仮想マシンに複数の IP アドレスを割り当てる」を参照してください。
- オートスケール設定は、同じ仮想ネットワークまたはピアリングされた仮想ネットワーク内の NetScaler VPX インスタンスに接続されている VMSS インスタンスから事前入力されます。詳細については、「Azure 仮想マシンのスケールセットを使用した Autoscale の概要」を参照してください。
- **Auto Scale Setting Protocol** と **Auto Scale Setting Port** を選択する際は、サーバーがプロトコルとポートをリッスンしていることを確認し、サービスグループに正しいモニターをバインドしてください。デフォルトでは、TCP モニターが使用されます。
- SSL プロトコルタイプの自動スケーリングでは、クラウドプロファイルを作成した後、証明書がないために負荷分散仮想サーバーまたはサービスグループがダウンします。証明書は、仮想サーバまたはサービスグループに手動でバインドできます。

> 注
>
> NetScaler リリース 13.1-42.x 以降では、Azure の同じ VMSS を使用して、（異なるポートを使用して）サービスごとに異なるクラウドプロファイルを作成できます。したがって、NetScaler VPX インスタンスは、パブリッククラウド内の同じ自動スケーリンググループを持つ複数のサービスをサポートします。

Azure portal でオートスケール関連の情報を表示するには、**[**仮想マシンスケールセット**]** に移動し、**[**** 仮想マシンスケールセット] > [スケーリング]** を選択します。

参照ドキュメント

NetScaler Application Delivery and Management を使用した Microsoft Azure での NetScaler VPX の自動スケーリングの詳細については、「Azure Autoscale using NetScaler ADM」を参照してください。

NetScaler VPX 展開用の Azure タグ

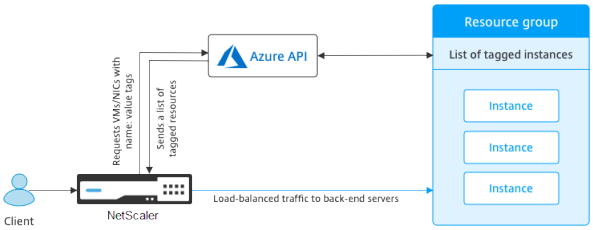
October 17, 2024

Azure クラウドポータルでは、名前: 値のペア (Dept: Finance など) でリソースにタグを付けて、リソースグループ間、およびポータル内でサブスクリプション間でリソースを分類して表示できます。タグ付けは、課金、管理、または自動化のためにリソースを整理する必要がある場合に役立ちます。

VPX デプロイにおける Azure タグの仕組み

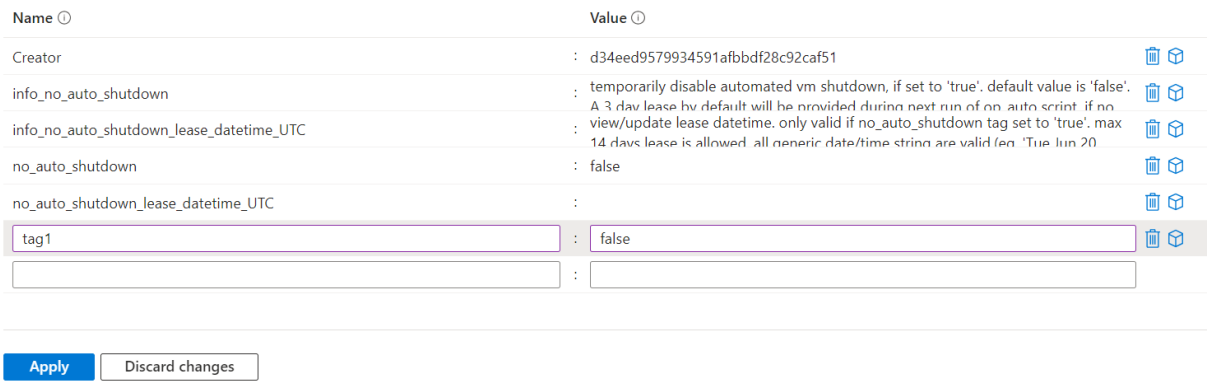
Azure Cloud にデプロイされた NetScaler VPX スタンドアロンおよび高可用性インスタンスの場合、Azure タグに関連付けられた負荷分散サービスグループを作成できるようになりました。VPX インスタンスは、Azure 仮想マシン（バックエンドサーバー）とネットワークインターフェイス（NIC）、またはその両方をそれぞれのタグで常に監視し、それに応じてサービスグループを更新します。

VPX インスタンスは、タグを使用してバックエンドサーバーの負荷分散を行うサービスグループを作成します。インスタンスは、特定のタグ名とタグ値でタグ付けされたすべてのリソースについて Azure API にクエリします。割り当てられたポーリング期間（デフォルトでは 60 秒）に応じて、VPX インスタンスは定期的に Azure API をポーリングし、VPX GUI で割り当てられたタグ名とタグ値を使用して利用可能なリソースを取得します。適切なタグが付いた VM または NIC が追加または削除されると、ADC はそれぞれの変更を検出し、VM または NIC の IP アドレスをサービスグループに自動的に追加または削除します。



はじめに

Citrix ADC 負荷分散サービスグループを作成する前に、Azure のサーバーにタグを追加します。タグは、仮想マシンまたは NIC に割り当てることができます。



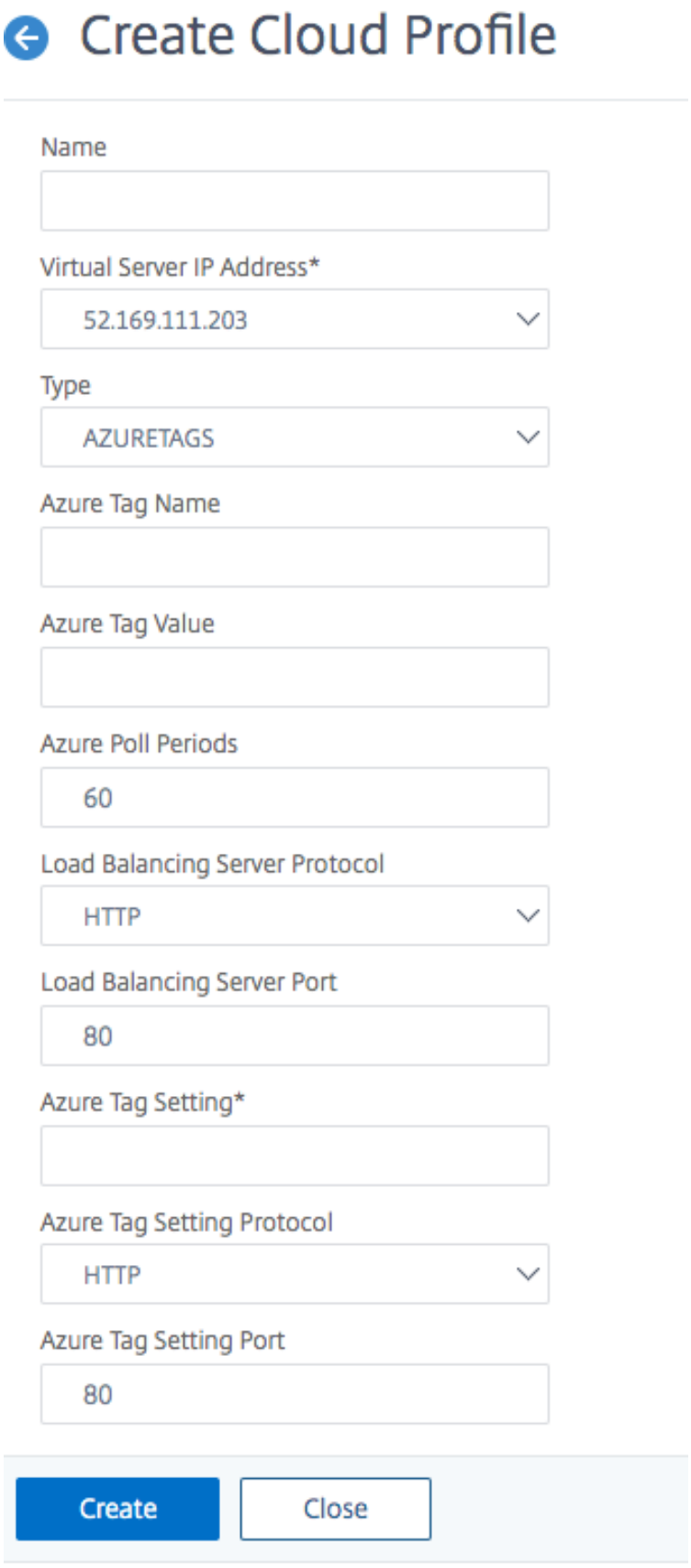
Azure タグの追加の詳細については、Microsoft ドキュメント「タグを使用して Azure リソースを整理する」を参照してください。

> 注
>
> Azure タグ設定を追加する ADC CLI コマンドは、数字またはアルファベットのみで始まり、他のキーボード文字で始まらないタグ名とタグ値をサポートします。

VPX GUI を使用して **Azure** タグ設定を追加する方法

VPX GUI を使用して Azure タグクラウドプロファイルを VPX インスタンスに追加すると、インスタンスは指定されたタグを使用してバックエンドサーバーの負荷を分散できます。以下の手順を実行します:

1. VPX GUI から、[構成**] > [Azure] > [**クラウドプロファイル**]** に移動します。
2. [追加] をクリックしてクラウドプロファイルを作成します。クラウドプロファイルウィンドウが開きます。



1. 次のフィールドに値を入力します。

   - 名前: プロフィールの名前を追加します
   - 仮想サーバーの IP アドレス: 仮想サーバーの IP アドレスは、VPX インスタンスで使用可能な空き IP アドレスから自動的に設定されます。詳細については、「Azure ポータルを使用して仮想マシンに複数の IP アドレスを割り当てる」を参照してください。
   - タイプ: メニューから「AZURETAGS」を選択します。
   - Azure タグ名:Azure ポータルで仮想マシンまたは NIC に割り当てた名前を入力します。
   - Azure タグ値:Azure ポータルの VM または NIC に割り当てた値を入力します。
   - Azure ポーリング期間: デフォルトでは、ポーリング間隔は最小値の 60 秒です。必要に応じて変更できます。
   - 負荷分散サーバープロトコル: ロードバランサーがリッスンするプロトコルを選択します。
   - 負荷分散サーバーポート: ロードバランサーが受信するポートを選択します。
   - Azure タグ設定: このクラウドプロファイル用に作成されるサービスグループの名前。
   - Azure タグ設定プロトコル: バックエンドサーバーがリッスンするプロトコルを選択します。
   - Azure タグ設定ポート: バックエンドサーバーがリッスンするポートを選択します。

2. [**Create**] をクリックします。

タグ付けされた仮想マシンまたは NIC に対して、ロードバランサー仮想サーバーとサービスグループが作成されます。ロードバランサー仮想サーバーを確認するには、VPX GUI から [** トラフィック管理] > [負荷分散] **[仮想サーバー] に移動します。

VPX CLI を使用して **Azure** タグ設定を追加する方法

NetScaler CLI で次のコマンドを入力して、Azure タグのクラウドプロファイルを作成します。

```
add cloud profile `<profile name>` -type azuretags -vServerName `<
   vserver name>` -serviceType HTTP -IPAddress `<vserver IP address>`
    -port 80 -serviceGroupName `<service group name>` -
   boundServiceGroupSvcType HTTP -vsvrbindsvcport 80 -azureTagName `<
   Azure tag specified on Azure portal>` -azureTagValue `<Azure value
    specified on the Azure portal>` -azurePollPeriod 60
```

> 重要:
>
> すべての構成を保存する必要があります。そうしないと、インスタンスを再起動した後に構成が失われます。「`save config`」と入力します。

例 **1**:「mytagName/MyTagValue」ペアでタグ付けされたすべての Azure VM/NIC の HTTP トラフィックのクラウドプロファイルのサンプルコマンドを次に示します。

```
add cloud profile MyTagCloudProfile -type azuretags -vServerName
   MyTagVServer -serviceType HTTP -IPAddress 40.115.116.57 -port 80 -
   serviceGroupName MyTagsServiceGroup -boundServiceGroupSvcType HTTP
```



```
          -vsvrbindsvcport 80 -azureTagName myTagName -azureTagValue
        myTagValue -azurePollPeriod 60
2   Done
```

クラウドプロファイルを表示するには、次のように入力します `show cloudprofile`。

例 **2**: 次の CLI コマンドは、例 1 で新しく追加されたクラウドプロファイルに関する情報を出力します。

```
1   show cloudprofile
2   1)    Name:  MyTagCloudProfile Type:  azuretags         VServerName:
        MyTagVServer ServiceType:  HTTP       IPAddress:  52.178.209.133
        Port:  80               ServiceGroupName:  MyTagsServiceGroup
        BoundServiceGroupSvcType:  HTTP
3         Vsvrbindsvcport:  80     AzureTagName:  myTagName AzureTagValue
              :  myTagValue AzurePollPeriod:  60     GraceFul:  NO
                        Delay:  60
```

クラウドプロファイルを削除するには、「rm cloud profile <cloud profile name>」と入力します。

例 **3**: 次のコマンドは、例 1 で作成したクラウドプロファイルを削除します。

```
1   > rm cloudprofile MyTagCloudProfile
2     Done
```

トラブルシューティング

問題: ごくまれに、「rm cloud profile」CLI コマンドで、削除されたクラウドプロファイルに関連付けられているサービスグループおよびサーバーの削除に失敗することがあります。これは、削除されるクラウドプロファイルのポーリング期間が経過する秒前にコマンドが発行された場合に発生します。

解決方法: 残りのサービスグループごとに次の CLI コマンドを入力して、残りのサービスグループを手動で削除します。

```
1   #> rm servicegroup <serviceGroupName>
```

残りの各サーバに対して次の CLI コマンドを入力して、残りのサーバもそれぞれ削除します。

```
1   #> rm server <name>
```

問題:CLI を使用して VPX インスタンスに Azure タグ設定を追加すると、ウォームリブート後も HA ペアノードで rain_tags プロセスが実行され続けます。

解決方法: ウォームリブート後に、セカンダリノードでプロセスを手動で終了します。セカンダリ HA ノードの CLI からシェルプロンプトに出ます。

```
1   #> shell
```

rain_tags プロセスを強制終了するには、次のコマンドを使用します。


```
 # PID=`ps -aux | grep rain_tags | awk '{
print $2 }
'`; kill -9 $PID
```

問題: バックエンドサーバーは正常であるにもかかわらず、VPX インスタンスから到達できず、DOWN として報告されることがあります。解決方法: VPX インスタンスが、バックエンドサーバーに対応するタグ付き IP アドレスに到達できることを確認します。タグ付きの NIC の場合、これは NIC の IP アドレスです。タグ付きの VM の場合、これは仮想マシンのプライマリ IP アドレスです。VM/NIC が別の Azure VNet 上に存在する場合は、VNet ピアリングが有効になっていることを確認します。

NetScaler VPX インスタンスで GSLB を構成する

October 17, 2024

グローバルサーバー負荷分散（GSLB）用に構成された NetScaler ADC アプライアンスは、WAN の障害点から保護することにより、ディザスタリカバリとアプリケーションの継続的な可用性を提供します。GSLB は、クライアント要求を最も近い、または最もパフォーマンスの高いデータセンター、または停止が発生した場合に存続しているデータセンターに送信することにより、データセンター間で負荷を分散できます。

このセクションでは、Windows PowerShell コマンドを使用して、Microsoft Azure 環境の 2 つのサイトの VPX インスタンスで GSLB を有効にする方法について説明します。

> 注
>
> GSLB の詳細については、「グローバルサーバーの負荷分散」を参照してください。

Azure 上の NetScaler VPX インスタンスで GSLB を構成するには、次の 2 つのステップがあります。

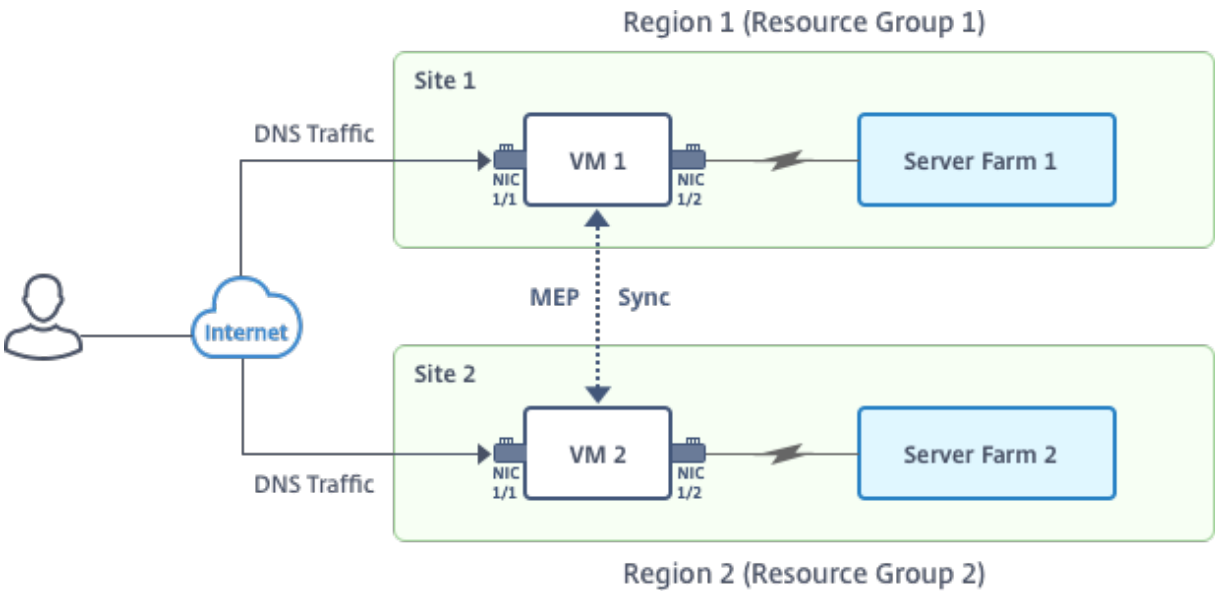
1. 各サイトに、複数の NIC と複数の IP アドレスを持つ VPX インスタンスを作成します。
2. VPX インスタンスで GSLB を有効にします。

> 注
>
> 複数の NIC および IP アドレスの構成の詳細については、「PowerShell コマンドを使用してスタンドアロンモードで Citrix ADC VPX インスタンスの複数の IP アドレスを構成する」を参照してください。

シナリオ

このシナリオには、2 つのサイト（Site 1 と Site 2）が含まれています。各サイトの VM（VM1 と VM2）には、複数の NIC、複数の IP アドレス、および GSLB が構成されています。

図. GSLB セットアップは、サイト 1 とサイト 2 の 2 つのサイトに実装されています。



このシナリオでは、各 VM には 3 つの NIC（NIC 0/1、1/1、1/2）が設定されています。各 NIC に複数のプライベートおよびパブリック IP アドレスを設定できます。これらの NIC は次の目的で構成されています。

- NIC 0/1：管理トラフィックを提供する
- NIC 1/1：クライアント側のトラフィックを提供する
- NIC 1/2：バックエンドサーバーと通信する

このシナリオで各 NIC に設定された IP アドレスの詳細については、「IP 構成の詳細」セクションを参照してください。

パラメーター

このドキュメントのこのシナリオのサンプルパラメーター設定は、次のとおりです。必要な場合は、異なる設定を使用できます。

```
$location="West Central US"

$vnetName="NSVPX-vnet"

$RGName="multiIP-RG"

$prmStorageAccountName="multiipstorageaccnt"

$avSetName="MultiIP-avset"

$vmSize="Standard\_DS3\_V2"
```

> 注
>
> VPX インスタンスの最小要件は、2 つの vCPU と 2 GB の RAM です。


```
$publisher="citrix"

$offer="netscalervpx111"

$sku="netscalerbyol"

$version="latest"

$vmNamePrefix="MultiIPVPX"

$nicNamePrefix="MultiipVPX"

$osDiskSuffix="osdiskdb"

$numberOfVMs=1

$ipAddressPrefix="10.0.0."

$ipAddressPrefix1="10.0.1."

$ipAddressPrefix2="10.0.2."

$pubIPName1="MultiIP-pip1"

$pubIPName2="MultiIP-pip2"

$IpConfigName1="IPConfig1"

$IPConfigName2="IPConfig-2"

$IPConfigName3="IPConfig-3"

$IPConfigName4="IPConfig-4"

$frontendSubnetName="default"

$backendSubnetName1="subnet\_1"

$backendSubnetName2="subnet\_2"

$suffixNumber=10
```

仮想マシンの作成

PowerShell コマンドを使用して、ステップ 1 ～10 に従って、複数の NIC と複数の IP アドレスを使用して VM1 を作成します。

1. リソースグループの作成
2. ストレージアカウントの作成

3. アベイラビリティセットの作成
4. 仮想ネットワークの作成
5. パブリック IP アドレスの作成
6. NIC の作成
7. VM 設定オブジェクトの作成
8. 認証情報を取得し、VM の OS プロパティを設定します
9. NIC の追加
10. OS ディスクの指定と VM の作成

すべての手順とコマンドを完了して VM1 を作成した後で、これらの手順を繰り返して VM2 固有のパラメーターで VM2 を作成します。

リソースグループの作成

```
New-AzureRMResourceGroup -Name $RGName -Location $location
```

ストレージアカウントの作成

```
$prmStorageAccount=New-AzureRMStorageAccount -Name
   $prmStorageAccountName -ResourceGroupName $RGName -Type
   Standard_LRS -Location $location
```

アベイラビリティセットの作成

```
$avSet=New-AzureRMAvailabilitySet -Name $avSetName -ResourceGroupName
    $RGName -Location $location
```

仮想ネットワークの作成

1. サブネットを追加します。

```
$subnet1=New-AzureRmVirtualNetworkSubnetConfig -Name
   $frontendSubnetName -AddressPrefix "10.0.0.0/24"
$subnet2=New-AzureRmVirtualNetworkSubnetConfig -Name
   $backendSubnetName1 -AddressPrefix "10.0.1.0/24"
$subnet3=New-AzureRmVirtualNetworkSubnetConfig -Name
   $backendSubnetName2 -AddressPrefix "10.0.2.0/24"
```

2. 仮想ネットワークオブジェクトを追加します。


```
$vnet=New-AzureRmVirtualNetwork -Name $vnetName -
    ResourceGroupName $RGName -Location $location -AddressPrefix
    10.0.0.0/16 -Subnet $subnet1, $subnet2, $subnet3
```

3. サブネットを取得します。

```
$frontendSubnet=$vnet.Subnets|?{
$_.Name -eq $frontendSubnetName }

$backendSubnet1=$vnet.Subnets|?{
$_.Name -eq $backendSubnetName1 }

$backendSubnet2=$vnet.Subnets|?{
$_.Name -eq $backendSubnetName2 }
```

パブリック **IP** アドレスの作成

```
$pip1=New-AzureRmPublicIpAddress -Name $pubIPName1 -ResourceGroupName
    $RGName -Location $location -AllocationMethod Dynamic
$pip2=New-AzureRmPublicIpAddress -Name $pubIPName2 -ResourceGroupName
    $RGName -Location $location -AllocationMethod Dynamic
```

NIC の作成

NIC 0/1 の作成

```
$nic1Name=$nicNamePrefix + $suffixNumber + "-Mgmnt"
$ipAddress1=$ipAddressPrefix + $suffixNumber
$IPConfig1=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName1 -
    SubnetId $frontendSubnet.Id -PublicIpAddress $pip1 -
    PrivateIpAddress $ipAddress1 -Primary
$nic1=New-AzureRMNetworkInterface -Name $nic1Name -ResourceGroupName
    $RGName -Location $location -IpConfiguration $IpConfig1
```

NIC 1/1 の作成

```
$nic2Name $nicNamePrefix + $suffixNumber + "-frontend"
$ipAddress2=$ipAddressPrefix1 + ($suffixNumber)
$ipAddress3=$ipAddressPrefix1 + ($suffixNumber + 1)
$IPConfig2=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName2 -
    PublicIpAddress $pip2 -SubnetId $backendSubnet1.Id  -
    PrivateIpAddress $ipAddress2  -Primary
$IPConfig3=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName3 -
    SubnetId $backendSubnet1.Id  -PrivateIpAddress $ipAddress3
nic2=New-AzureRMNetworkInterface -Name $nic2Name -ResourceGroupName
    $RGName -Location $location -IpConfiguration $IpConfig2,
    $IpConfig3
```


NIC 1/2 の作成

```
$nic3Name=$nicNamePrefix + $suffixNumber + "-backend"
$ipAddress4=$ipAddressPrefix2 + ($suffixNumber)
$IPConfig4=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName4 -
    SubnetId $backendSubnet2.Id -PrivateIpAddress $ipAddress4 -Primary
$nic3=New-AzureRMNetworkInterface -Name $nic3Name -ResourceGroupName
    $RGName -Location $location -IpConfiguration $IpConfig4
```

VM 設定オブジェクトの作成

```
$vmName=$vmNamePrefix
$vmConfig=New-AzureRMVMConfig -VMName $vmName -VMSize $vmSize -
    AvailabilitySetId $avSet.Id
```

認証情報の取得と **OS** プロパティの設定

```
$cred=Get-Credential -Message "Type the name and password for VPX
    login."
$vmConfig=Set-AzureRMVMOperatingSystem -VM $vmConfig -Linux -
    ComputerName $vmName -Credential $cred
$vmConfig=Set-AzureRMVMSourceImage -VM $vmConfig -PublisherName
    $publisher -Offer $offer -Skus $sku -Version $version
```

NIC の追加

```
$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic1.Id -
    Primary
$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic2.Id
$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic3.Id
```

OS ディスクの指定と **VM** の作成

```
$osDiskName=$vmName + "-" + $osDiskSuffix
$osVhdUri=$prmStorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds
    /" +$osDiskName + ".vhd"
$vmConfig=Set-AzureRMVMOSDisk -VM $vmConfig -Name $osDiskName -VhdUri
     $osVhdUri -CreateOption fromImage
Set-AzureRmVMPlan -VM $vmConfig -Publisher $publisher -Product $offer
     -Name $sku
New-AzureRMVM -VM $vmConfig -ResourceGroupName $RGName -Location
    $location
```


> 注
>
> 「PowerShell コマンドを使用したマルチ NIC 仮想マシンの作成」に記載されている手順 1～10 を繰り返して、VM2 に固有のパラメータを使用して VM2 を作成します。

IP 構成の詳細

次の IP アドレスを使用します。

テーブル **1**. VM1 で使用する IP アドレス

NIC	プライベート IP	パブリック IP（PIP）	説明
0/1	10.0.0.10	PIP1	NSIP（管理 IP）として構成
1/1	10.0.1.10	PIP2	SNIP/GSLB サイト IP として設定されています
-	10.0.1.11	-	LB サーバ IP として設定されています。パブリック IP アドレスは必須ではありません
1/2	10.0.2.10	-	モニタプローブをサービスに送信するための SNIP として設定。パブリック IP は必須ではありません。

表 **2.** VM2 で使用される IP アドレス

NIC	内部 IP	パブリック IP（PIP）	説明
0/1	20.0.0.10	PIP4	NSIP（管理 IP）として構成
1/1	20.0.1.10	PIP5	SNIP/GSLB サイト IP として設定されています
-	20.0.1.11	-	LB サーバ IP として設定されています。パブリック IP アドレスは必須ではありません

NIC	内部 IP	パブリック IP（PIP）	説明
1/2	20.0.2.10	-	モニタプローブをサービスに送信するための SNIP として設定。パブリック IP は必須ではありません。

このシナリオの構成例を次に示します。VM1 と VM2 の NetScaler VPX CLI で作成された IP アドレスと初期 LB 構成を示しています。

VM1 の設定例を次に示します。

```
add ns ip 10.0.1.10 255.255.255.0 -mgmtAccess ENABLED
Add nsip 10.0.2.10 255.255.255.0
add service svc1 10.0.1.10 ADNS 53
add lb vserver v1 HTTP 10.0.1.11 80
add service s1 10.0.2.120 http 80
Add service s2 10.0.2.121 http 80
Bind lb vs v1 s[1-2]
```

VM2 の設定例を次に示します。

```
add ns ip 20.0.1.10 255.255.255.0  -mgmtAccess ENABLED
Add nsip 20.0.2.10 255.255.255.0
add service svc1 20.0.1.10 ADNS 53
add lb vserver v1 HTTP 20.0.1.11 80
Add service s1 20.0.2.90 http 80
Add service s2 20.0.2.91 http 80
Bind lb vs v1 s[1-2]
```

GSLB サイトおよびその他の設定を構成する

次のトピックで説明するタスクを実行して、2 つの GSLB サイトとその他の必要な設定を構成します。

Global Server Load Balancing

VM1 および VM2 での GSLB 設定の例を次に示します。

```
enable ns feature LB GSLB
add gslb site site1 10.0.1.10 -publicIP PIP2
add gslb site site2 20.0.1.10 -publicIP PIP5
add gslb service site1_gslb_http_svc1 10.0.1.11 HTTP 80 -publicIP
   PIP3 -publicPort 80 -siteName site1
add gslb service site2_gslb_http_svc1 20.0.1.11 HTTP 80 -publicIP
   PIP6 -publicPort 80 -siteName site2
add gslb vserver gslb_http_vip1 HTTP
bind gslb vserver gslb_http_vip1 -serviceName site2_gslb_http_svc1
bind gslb vserver gslb_http_vip1 -serviceName site1_gslb_http_svc1
bind gslb vserver gslb_http_vip1 -domainName www.gslbindia.com -TTL 5
```


Azure で実行されている NetScaler VPX インスタンスに GSLB を構成しました。

障害回復

災害（さいがん）とは、自然の災害、または人為的な出来事によって引き起こされる事業機能の突然の混乱である。災害はデータセンターの運用に影響を及ぼします。その後、災害現場で失われたリソースとデータを完全に再構築して復元する必要があります。データ消失やデータセンターのダウンタイムは重要であり、ビジネス継続性が低下します。

お客様が今日直面している課題の 1 つは、DR サイトをどこに置くかを決めることです。企業は、基盤となるインフラストラクチャやネットワーク障害に関係なく、一貫性とパフォーマンスを求めています。

多くの組織がクラウドへの移行を決定している理由として考えられるのは、次のとおりです。

- オンプレミスのデータセンターを持つことは非常に高価です。クラウドを使用することで、企業は自社のシステムを拡張する時間とリソースを解放できます。
- 自動オーケストレーションの多くは、より迅速なリカバリを可能にします
- 継続的なデータ保護や継続的なスナップショットを提供してデータを複製し、システム停止や攻撃から保護します。
- パブリッククラウドにすでに存在しているさまざまな種類のコンプライアンスやセキュリティ制御を顧客が必要とするユースケースをサポートします。これらにより、独自に構築するよりも、必要なコンプライアンスを簡単に達成できます。

GSLB 用に構成された NetScaler ADC は、トラフィックを最も負荷の少ないデータセンターまたは最もパフォーマンスの高いデータセンターに転送します。この構成は、アクティブ-アクティブ設定と呼ばれ、パフォーマンスが向上するだけでなく、セットアップの一部であるデータセンターがダウンした場合に、トラフィックを他のデータセンターにルーティングすることで、ディザスタリカバリを即座に実行できます。これにより、NetScaler はお客様の貴重な時間と費用を節約できます。

災害復旧のためのマルチ **NIC** マルチ **IP (3** つの **NIC)** の導入

お客様は、セキュリティ、冗長性、可用性、容量、およびスケーラビリティが重要な本番環境に導入する場合、3 つの NIC 導入を使用して導入する可能性があります。この展開方法では、複雑さと管理の容易さはユーザーにとって重大な問題ではありません。

ディザスタリカバリ用の単一 **NIC** マルチ **IP**（**1NIC**）導入

お客様は、以下の理由で非実稼働環境に導入する場合、NIC を 1 つにまとめて導入する可能性があります。

- テスト用の環境をセットアップするか、本番環境への導入前に新しい環境をステージングします。

- クラウドに迅速かつ効率的に直接デプロイします。
- 単一のサブネット構成のシンプルさを求めながら。

アクティブ/スタンバイの高可用性セットアップで **GSLB** を構成する

October 17, 2024

Azure のアクティブ/スタンバイ HA 展開には、次の 3 つの手順でグローバルサーバー負荷分散（GSLB）を構成できます。

1. 各 GSLB サイトで VPX HA ペアを作成します。HA ペアの作成方法については、「複数の IP アドレスと NIC を使用して高可用性セットアップを構成する 」を参照してください。
2. Azure Load Balancer（ALB）をフロントエンド IP アドレスと、GSLB および DNS トラフィックを許可する規則で構成します。

   この手順には、次の下位手順が含まれています。これらの下位手順の完了に使用する PowerShell コマンドについては、このセクションのシナリオを参照してください。

   a. GSLB サイトのフロントエンド`IPconfig`を作成します。

   b. HA 内のノードの NIC 1/1 の IP アドレスを持つバックエンドアドレスプールを作成します。

   c. 次のような負荷分散規則を作成します。

   ```
   TCP/3009  -  gslb communication
   TCP/3008 - gslb communication
   UDP/53 - DNS communication
   ```

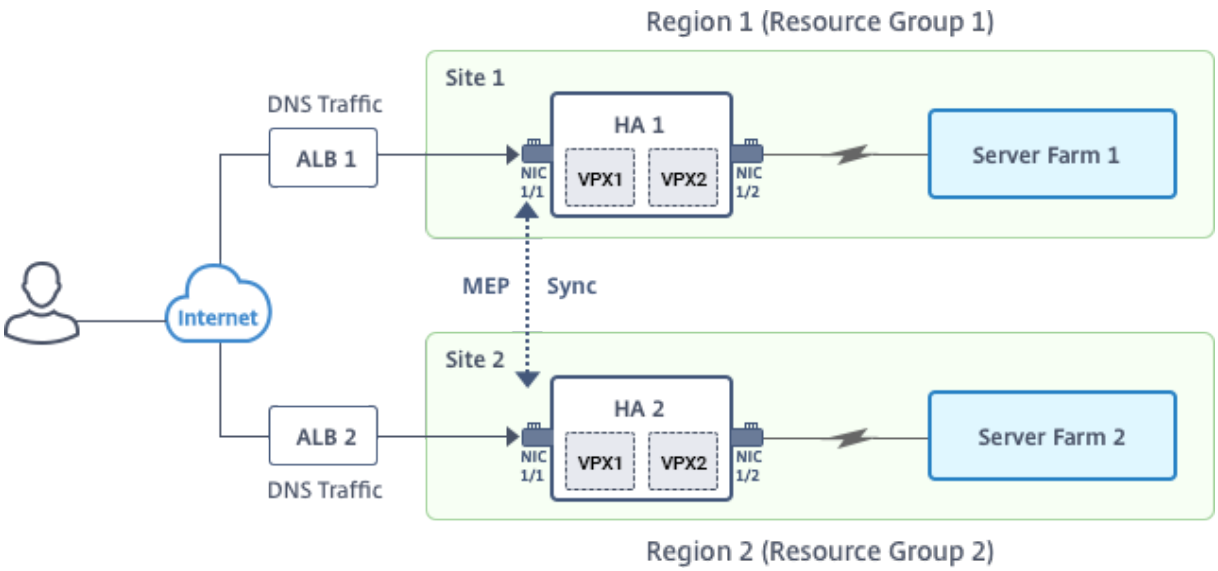
   d. バックエンドアドレスプールと手順 c で作成した LB 規則を関連付けます。

   e. 両方の HA ペアのノードの NIC 1/1 のネットワークセキュリティグループを更新して、TCP 3008、TCP 3009、および UDP 53 ポートのトラフィックを許可します。
3. 各 HA ペアで GSLB を有効にします。

シナリオ

このシナリオには、2 つのサイト（Site 1 と Site 2）が含まれています。各サイトの HA ペア（HA1 と HA2）には、複数の NIC、複数の IP アドレス、および GSLB が構成されています。

図: Azure でのアクティブ-スタンバイ HA デプロイメントでの GLSB



このシナリオでは、各 VM には 3 つの NIC（NIC 0/1、1/1、1/2）が設定されています。これらの NIC は次の目的で構成されています。

NIC 0/1：管理トラフィックを提供する

NIC 1/1：クライアント側のトラフィックを提供する

NIC 1/2：バックエンドサーバーと通信する

パラメーター設定

ALB のサンプルパラメーター設定は、次のとおりです。必要な場合は、異なる設定を使用できます。

```
$locName="South east Asia"

$rgName="MulitIP-MultiNIC-RG"

$pubIPName4="PIPFORGSLB1"

$domName4="vpxgslbdns"

$lbName="MultiIPALB"

$frontEndConfigName2="FrontEndIP2"

$backendPoolName1="BackendPoolHttp"

$lbRuleName2="LBRuleGSLB1"

$lbRuleName3="LBRuleGSLB2"

$lbRuleName4="LBRuleDNS"

$healthProbeName="HealthProbe"
```


フロントエンド **IP** アドレスとルールを使用して **ALB** を構成し、**GSLB** と **DNS** トラフィックを許可する

手順 1. 手順 1: **GSLB** サイト **IP** 用のパブリック **IP** を作成する

```
$pip4=New-AzureRmPublicIpAddress -Name $pubIPName4 -ResourceGroupName
    $rgName -DomainNameLabel $domName4 -Location $locName -
   AllocationMethod Dynamic


Get-AzureRmLoadBalancer -Name \$lbName -ResourceGroupName \$rgName |
   Add-AzureRmLoadBalancerFrontendIpConfig -Name \
   $frontEndConfigName2 -PublicIpAddress \$pip4 | Set-
   AzureRmLoadBalancer
```

手順 3. 手順 2: **LB** ルールを作成し、既存の **ALB** を更新します。

```
$alb = get-AzureRmLoadBalancer -Name $lbName -ResourceGroupName
   $rgName


$frontendipconfig2=Get-AzureRmLoadBalancerFrontendIpConfig -
   LoadBalancer $alb -Name $frontEndConfigName2


$backendPool=Get-AzureRmLoadBalancerBackendAddressPoolConfig -
   LoadBalancer $alb -Name $backendPoolName1


$healthprobe=Get-AzureRmLoadBalancerProbeConfig -LoadBalancer $alb -
   Name $healthProbeName


\$alb | Add-AzureRmLoadBalancerRuleConfig -Name \$lbRuleName2 -
   BackendAddressPool \$backendPool -FrontendIPConfiguration  \
   $frontendipconfig2 -Protocol \"Tcp\" -FrontendPort 3009 -
   BackendPort 3009 -Probe \$healthprobe -EnableFloatingIP  | Set-
   AzureRmLoadBalancer


\$alb | Add-AzureRmLoadBalancerRuleConfig -Name \$lbRuleName3 -
   BackendAddressPool \$backendPool -FrontendIPConfiguration  \
   $frontendipconfig2 -Protocol \"Tcp\" -FrontendPort 3008 -
   BackendPort 3008 -Probe \$healthprobe -EnableFloatingIP  | Set-
   AzureRmLoadBalancer


\$alb | Add-AzureRmLoadBalancerRuleConfig -Name \$lbRuleName4 -
   BackendAddressPool \$backendPool -FrontendIPConfiguration  \
   $frontendipconfig2 -Protocol \"Udp\" -FrontendPort 53 -BackendPort
    53 -Probe \$healthprobe -EnableFloatingIP  | Set-
   AzureRmLoadBalancer
```


各高可用性ペアで **GSLB** を有効にします

各 ALB（ALB 1 と ALB 2）で 2 つのフロントエンド IP アドレスを設定しました。1 つめの IP アドレスは LB 仮想サーバー、もう 1 つは GSLB サイトの IP です。

HA 1 には次のフロントエンド IP アドレスがあります。

- frontendiPOFalb1 (LB 仮想サーバー用)
- PIPFORGSLB1（GSLB IP）

HA 2 には次のフロントエンド IP アドレスがあります。

- frontendiPOFALB2 (LB 仮想サーバー用)
- PIPFORGSLB2（GSLB IP）

このシナリオでは、次のコマンドを使用します。

```
enable ns feature LB GSLB

add service dnssvc PIPFORGSLB1 ADNS 53

add gslb site site1 PIPFORGSLB1 -publicIP PIPFORGSLB1

add gslb site site2 PIPFORGSLB2 -publicIP PIPFORGSLB2

add gslb service site1_gslb_http_svc1 FrontEndIPofALB1 HTTP 80 -
   publicIP FrontEndIPofALB1 -publicPort 80 -siteName site1

add gslb service site2_gslb_http_svc1 FrontEndIPofALB2 HTTP 80 -
   publicIP FrontEndIPofALB2 -publicPort 80 -siteName site2

add gslb vserver gslb_http_vip1 HTTP

bind gslb vserver gslb_http_vip1 -serviceName site2_gslb_http_svc1

bind gslb vserver gslb_http_vip1 -serviceName site1_gslb_http_svc1

bind gslb vserver gslb_http_vip1 -domainName www.gslbindia.com -TTL 5
```

関連リソース:

NetScaler VPX インスタンスで GSLB を構成する

Global Server Load Balancing

Azure に NetScaler GSLB を展開

October 17, 2024

需要が高まる中、地域の顧客にサービスを提供するオンプレミスデータセンターを運営している企業は、Azure クラウドを使用して世界中に規模を拡大してデプロイしたいと考えています。NetScaler をネットワーク管理者側で使用すると、GSLB StyleBook を使用してオンプレミスとクラウドの両方でアプリケーションを構成できます。NetScaler ADM を使用して同じ構成をクラウドに転送できます。GSLB との距離に応じて、オンプレミスリソースとクラウドリソースのいずれかにアクセスできます。これにより、世界のどこにいても、シームレスなエクスペリエンスを実現できます。

DBS の概要

NetScaler GSLB は、クラウドロードバランサーでのドメインベースサービス（DBS）の使用をサポートしています。これにより、クラウドロードバランサーソリューションを使用して動的クラウドサービスを自動検出できます。この構成により、NetScaler はアクティブ-アクティブ環境に GSLB DBS を実装できます。DBS では、DNS 検出から Microsoft Azure 環境のバックエンドリソースを拡張できます。このセクションでは、AzureAutoscale 環境における NetScaler 間の統合について説明します。

Azure ロードバランサー (ALB) を使用するドメイン名ベースのサービス

GLSB DBS は、ユーザー ALB の FQDN を利用して、Azure 内で作成および削除されるバックエンドサーバーを含むように GSLB サービスグループを動的に更新します。この機能を設定するには、ユーザーは NetScaler を ALB にポイントして、Azure 内のさまざまなサーバーに動的にルーティングします。これは、Azure 内でインスタンスが作成および削除されるたびに NetScaler を手動で更新しなくても実行できます。GSLB サービスグループ向けの NetScaler DBS 機能は、DNS 対応のサービス検出を使用して、Autoscale グループで識別される DBS 名前空間のメンバーサービスリソースを特定します。

次の図は、クラウドロードバランサーを使用する NetScaler GSLB DBS オートスケールコンポーネントを示しています。

Azure GSLB の前提条件

NetScaler GSLB サービスグループの前提条件には、セキュリティグループ、Linux ウェブサーバー、AWS 内の NetScaler アプライアンス、Elastic IP、および Elastic ロードバランサー（ELB）を設定するための知識と能力を備えた、正常に機能する Microsoft Azure 環境が含まれます。

- GSLB DBS サービスの統合には、Microsoft Azure ロードバランサーインスタンス用の NetScaler バージョン 12.0.57 が必要です。
- GSLB サービスグループエンティティ:NetScaler バージョン 12.0.57
- DBS 動的検出を使用した自動スケーリングをサポートする GSLB サービスグループが導入されました。
- DBS 機能コンポーネント (ドメインベースのサービス) は GSLB サービスグループにバインドする必要があります。

例

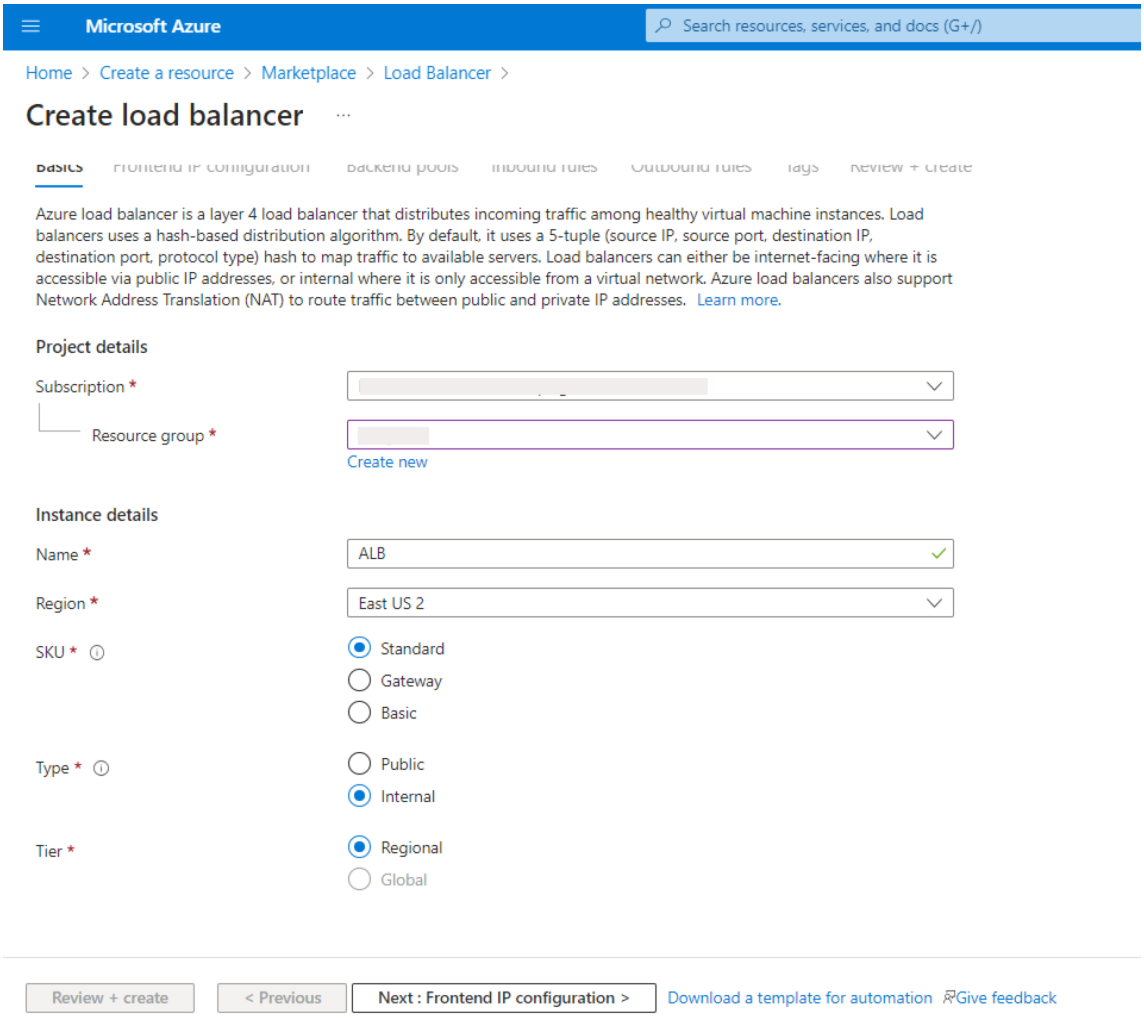
```
```

    > add server sydney_server LB-Sydney-xxxxxxxxxx.ap-southeast-2.elb.amazonaws.com
    > add gslb serviceGroup sydney_sg HTTP -autoscale DNS -siteName sydney
    > bind gslb serviceGroup sydney_sg sydney_server 80

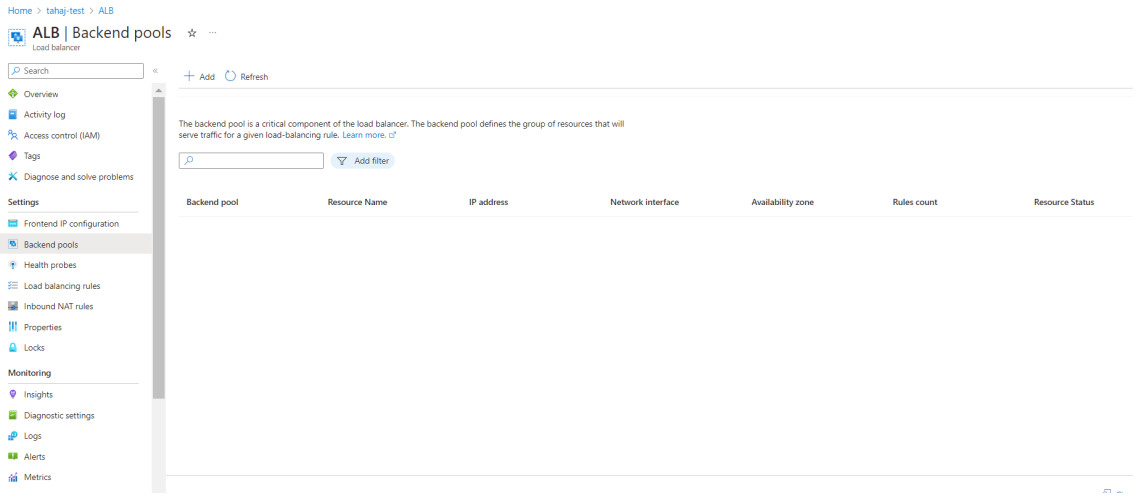
```
```

Azure コンポーネントの設定

1. ユーザー Azure Portal にログインし、NetScaler テンプレートから新しい仮想マシンを作成します。
2. Azure Load Balancer を作成します。



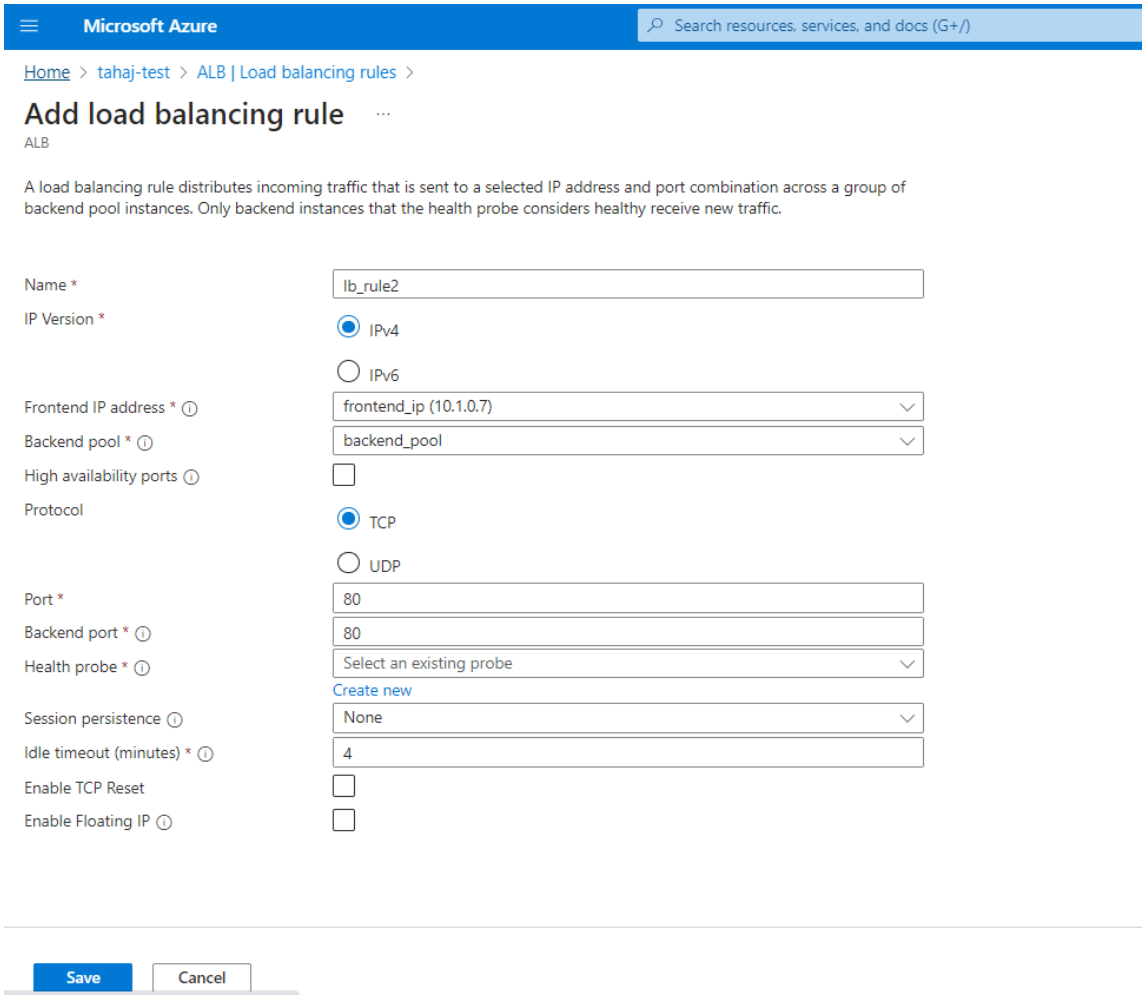
3. 作成した NetScaler バックエンドプールを追加します。



4. ポート 80 のヘルスプローブを作成します。

   ロードバランサーから作成されたフロントエンド IP を利用して負荷分散ルールを作成します。

- プロトコル:TCP
- バックエンドポート:80
- バックエンドプール: 手順 1 で作成した NetScaler
- ヘルスプローブ: ステップ 4 で作成
- セッションの永続性: なし



NetScaler GSLB ドメインベースのサービスの設定

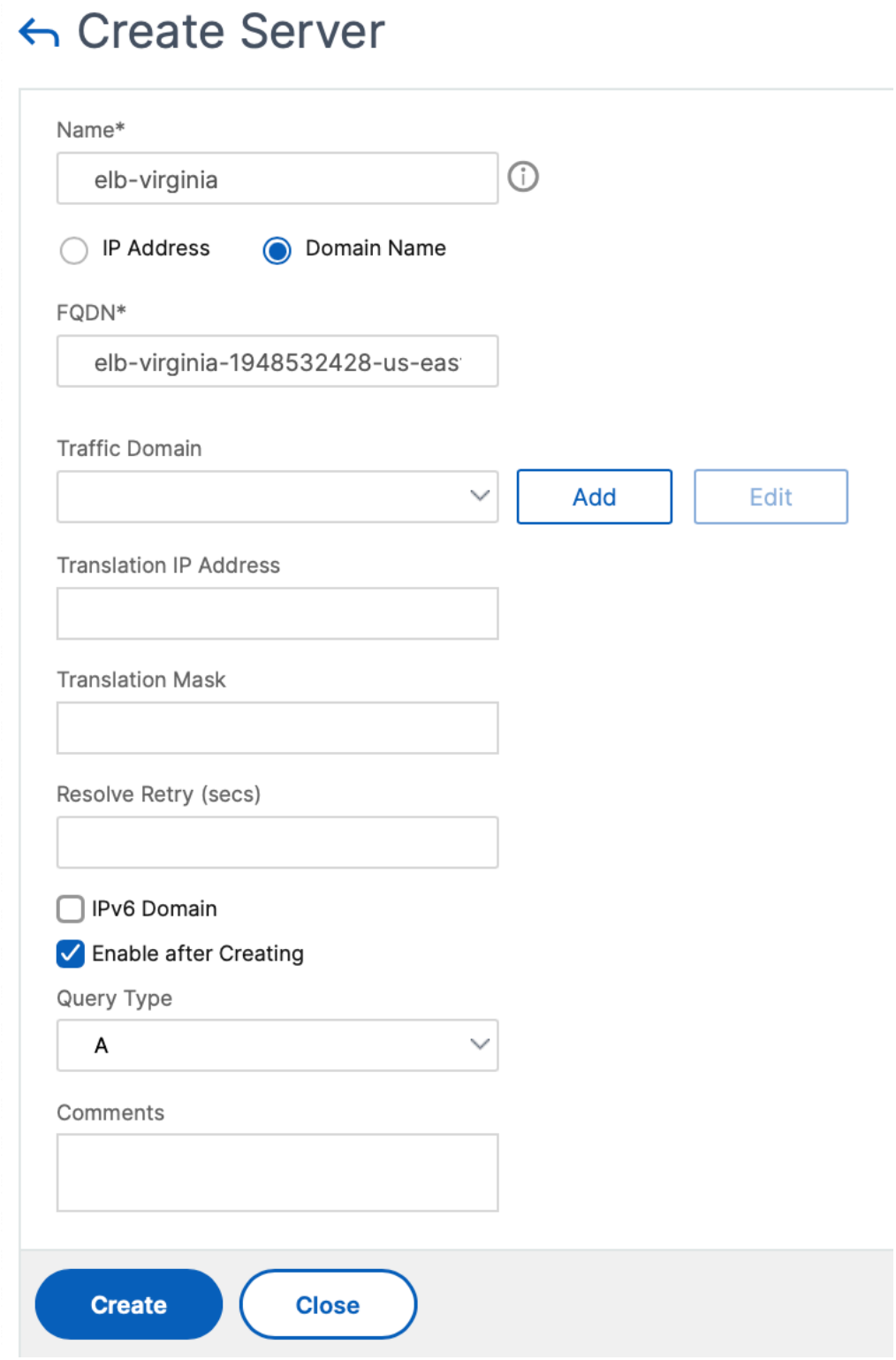
次の構成は、GSLB 対応環境で ADC を自動スケーリングするためのドメインベースのサービスを有効にするために必要なものをまとめたものです。

- トラフィック管理の設定
- GSLB 構成

トラフィック管理の設定

> 注
>
> NetScaler をネームサーバーまたは DBS サービスグループの ELB/ALB ドメインの解決に使用する DNS 仮想サーバーのいずれかで構成する必要があります。ネームサーバーまたは DNS 仮想サーバーの詳細については、「DNS ネームサーバー」を参照してください。

1. [トラフィック管理] **>** [負荷分散] **>** [サーバー] に移動し ます。
2. [追加] をクリックしてサーバーを作成し、ALB の Azure の A レコード (ドメイン名) に対応する名前と FQDN を指定します。



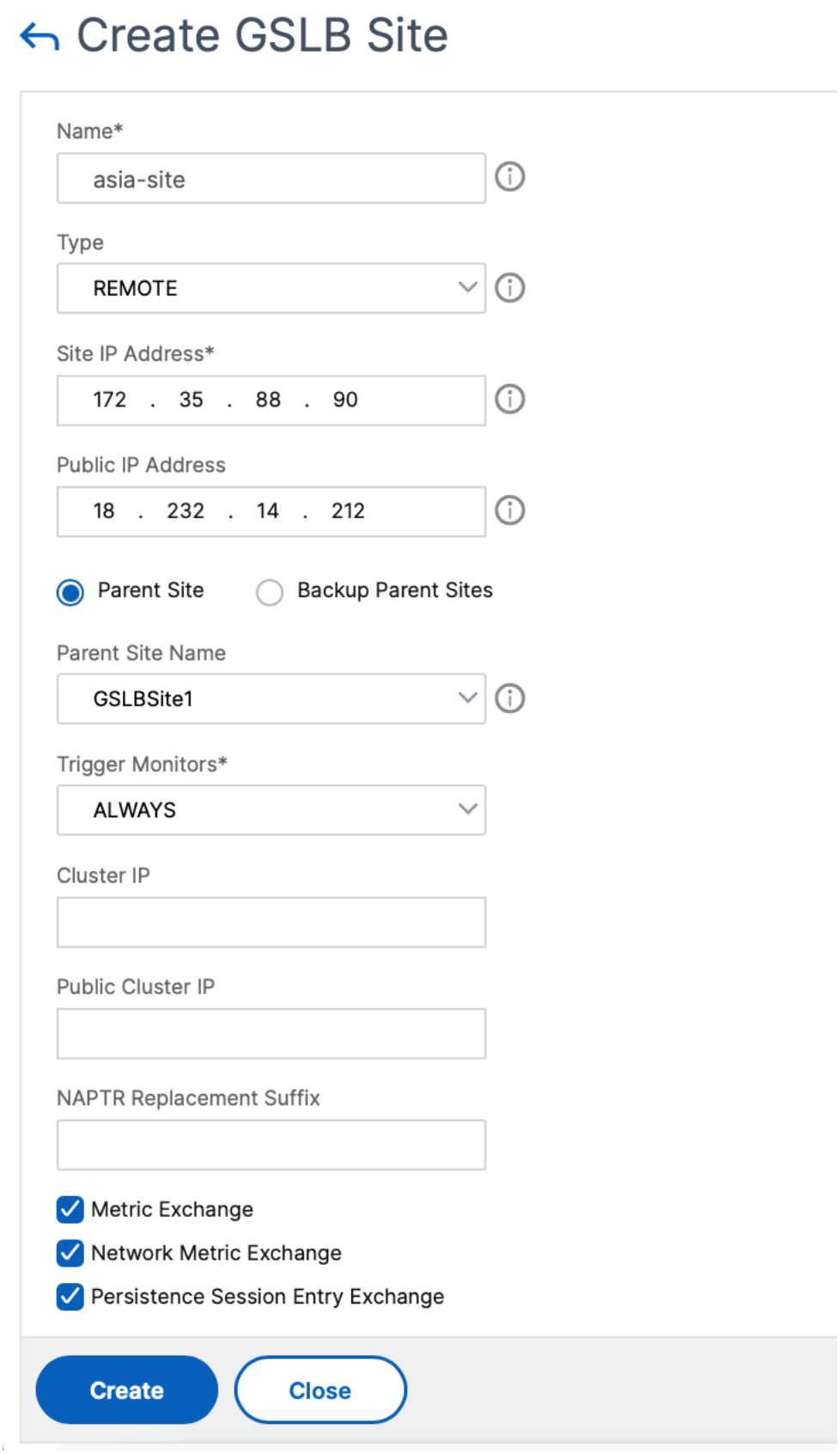
3. 手順 2 を繰り返して、Azure の 2 番目のリソースから 2 番目の ALB を追加します。

GSLB 構成

1. GSLB サイトを構成するには、[追加] をクリックします。
2. GSLB サイトを構成するための詳細を指定します。

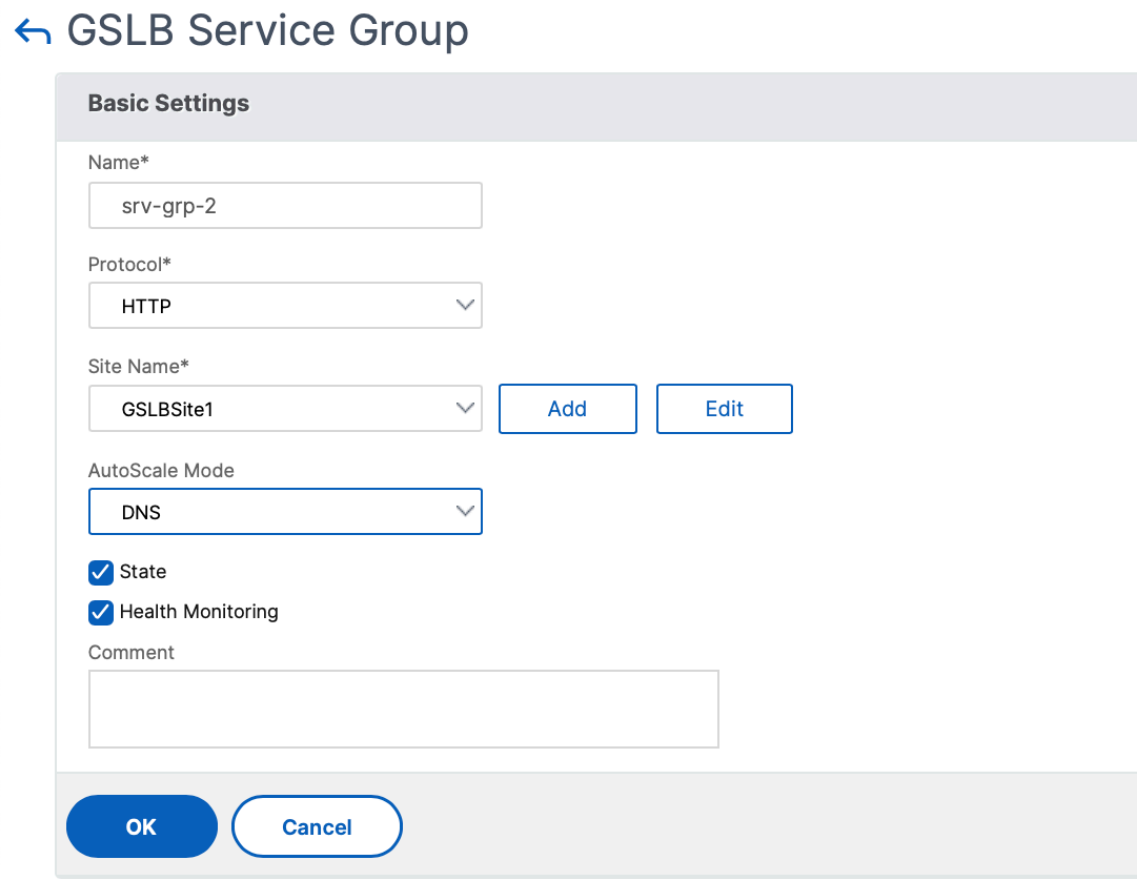
   サイトに名前を付けます。タイプは、サイトを構成する NetScaler に基づいてリモートまたはローカルとして構成されます。サイト IP アドレスは GSLB サイトの IP アドレスです。GSLB サイトは、この IP アドレスを使用して他の GSLB サイトと通信します。パブリック IP アドレスは、特定の IP アドレスが外部のファイアウォールまたは NAT デバイスでホストされているクラウドサービスを使用する場合に必要です。サイトは親サイトとして構成する必要があります。トリガーモニターが **ALWAYS** に設定されていることを確認します。** また、下部にある **[**メトリック交換**]**、**[**** ネットワークメトリック交換**]**、および **[**** パーシスタンスセッションエントリ交換 **] の 3 つのボックスを必ずオンにしてください。

   トリガーモニターを **MEPDOWN** に設定することをお勧めします。詳細については、「GSLB サービス グループの構成」を参照してください。

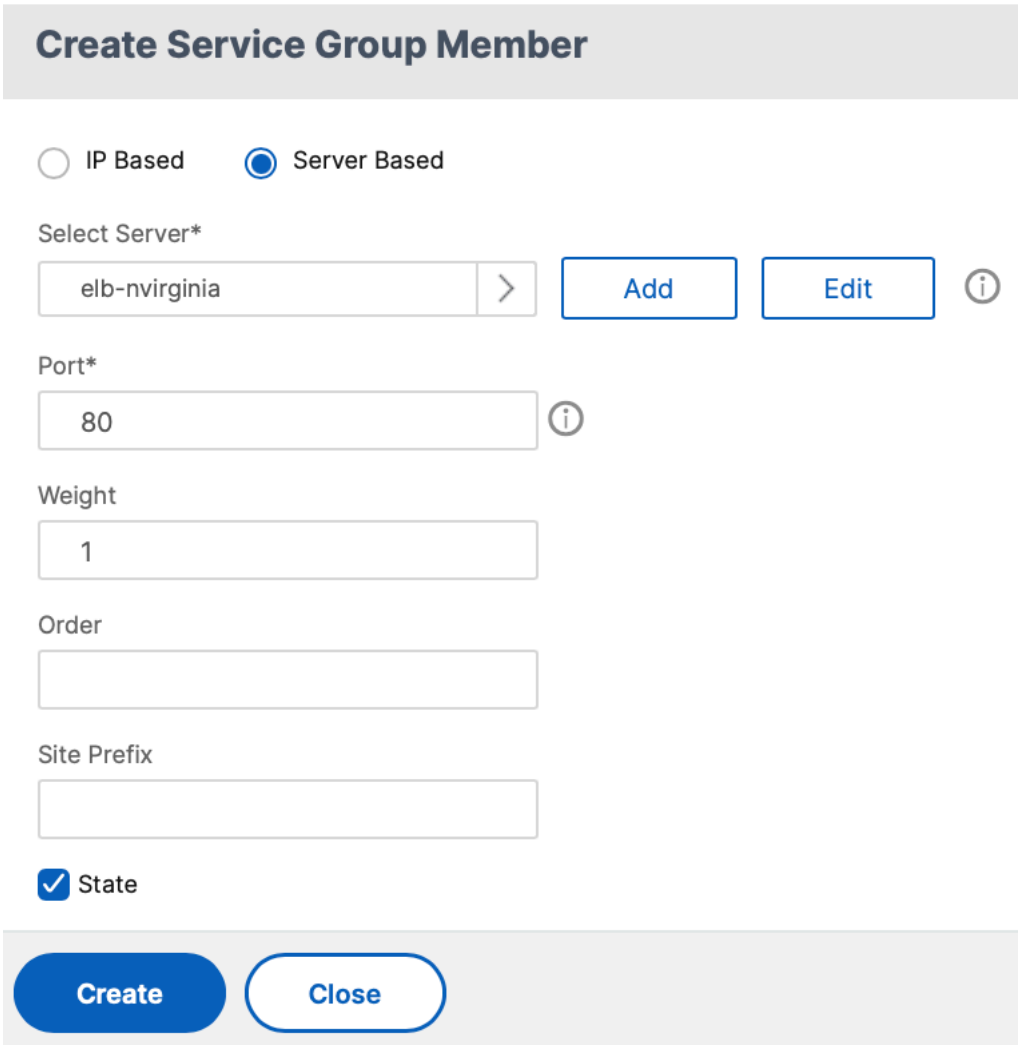


3. 作成をクリックします。
4. [トラフィック管理] **> [GSLB] > [**サービスグループ] に移動します。
5. [追加] をクリックしてサービスグループを追加します。
6. 詳細を指定してサービスグループを設定します

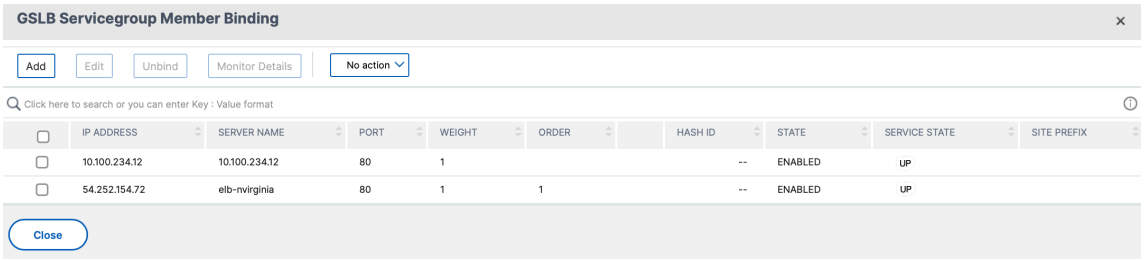
   サービスグループに名前を付け、HTTP プロトコルを使用します。[サイト名] で、作成した各サイトを選択します。必ず自動スケールモードを DNS として設定し、状態およびヘルスモニタリングのチェックボックスをオフにします。**OK** をクリックしてサービスグループを作成します。



7. [サービスグループメンバー] をクリックし、[サーバーベース] を選択します。実行ガイドの開始時に設定した各 ELB を選択します。トラフィックがポート 80 を通過するように設定します。作成をクリックします。

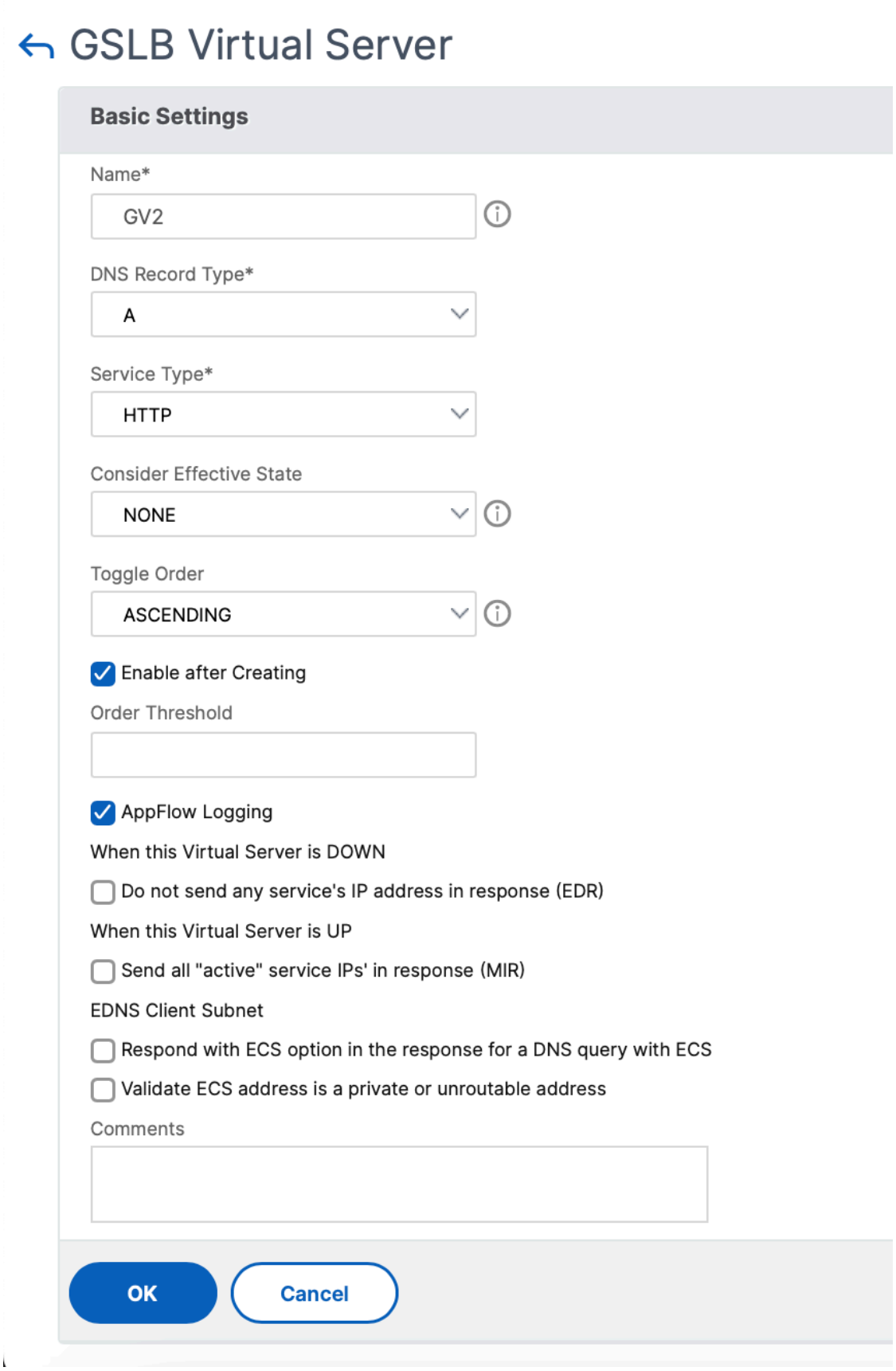


サービスグループメンバーバインディングには、ELB から受信する 2 つのインスタンスが入力されている必要があります。

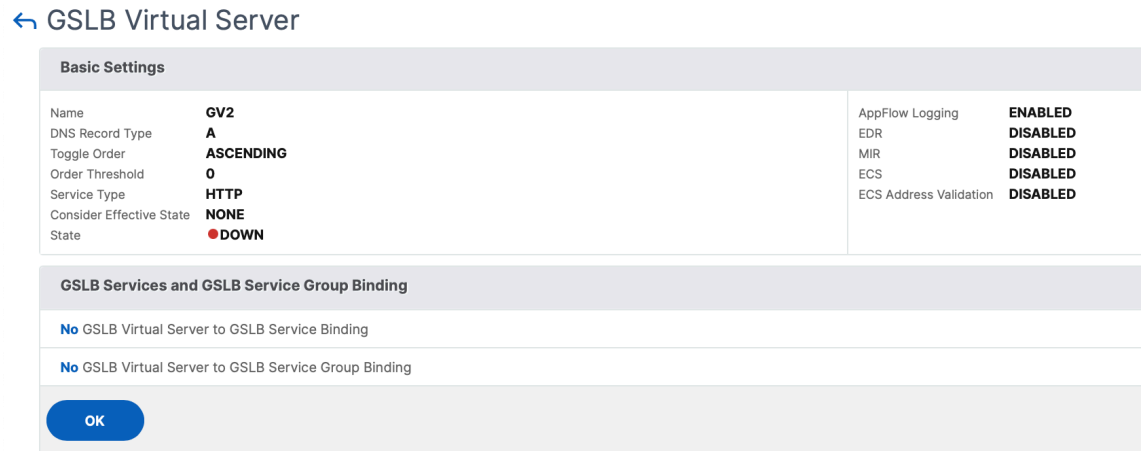


8. 手順 5 と 6 を繰り返して、Azure の 2 番目のリソースロケーションのサービスグループを設定します。（これは同じ NetScaler GUI から実行できます）。
9. GSLB 仮想サーバーをセットアップします。[トラフィック管理] **> [GSLB] > [**仮想サーバー] に移動します。
10. [追加] をクリックして仮想サーバーを作成します。
11. 詳細を指定して GSLB 仮想サーバーを構成します。

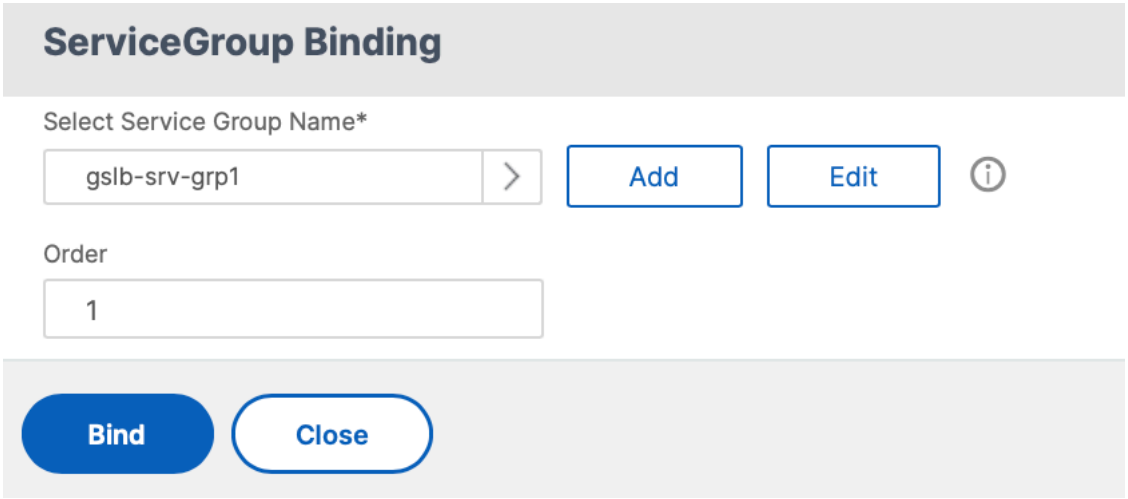
サーバーの名前を指定し、DNS レコードタイプが A に設定され、サービスタイプが HTTP に設定され、AppFlow ロギングの作成後に有効にするチェックボックスをオンにします。**OK** をクリックして GSLB 仮想サーバーを作成します。



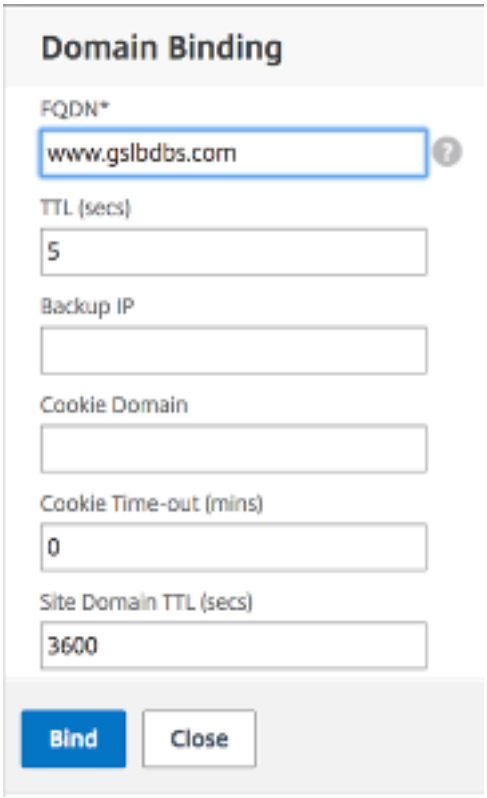
12. GSLB 仮想サーバーを作成したら、「**GSLB** 仮想サーバーサービスグループバインディングなし」をクリックします。



13. 「サービスグループバインディング」で 、「サービスグループ名 の選択」を使用して、前のステップで作成したサービスグループを選択して追加します。

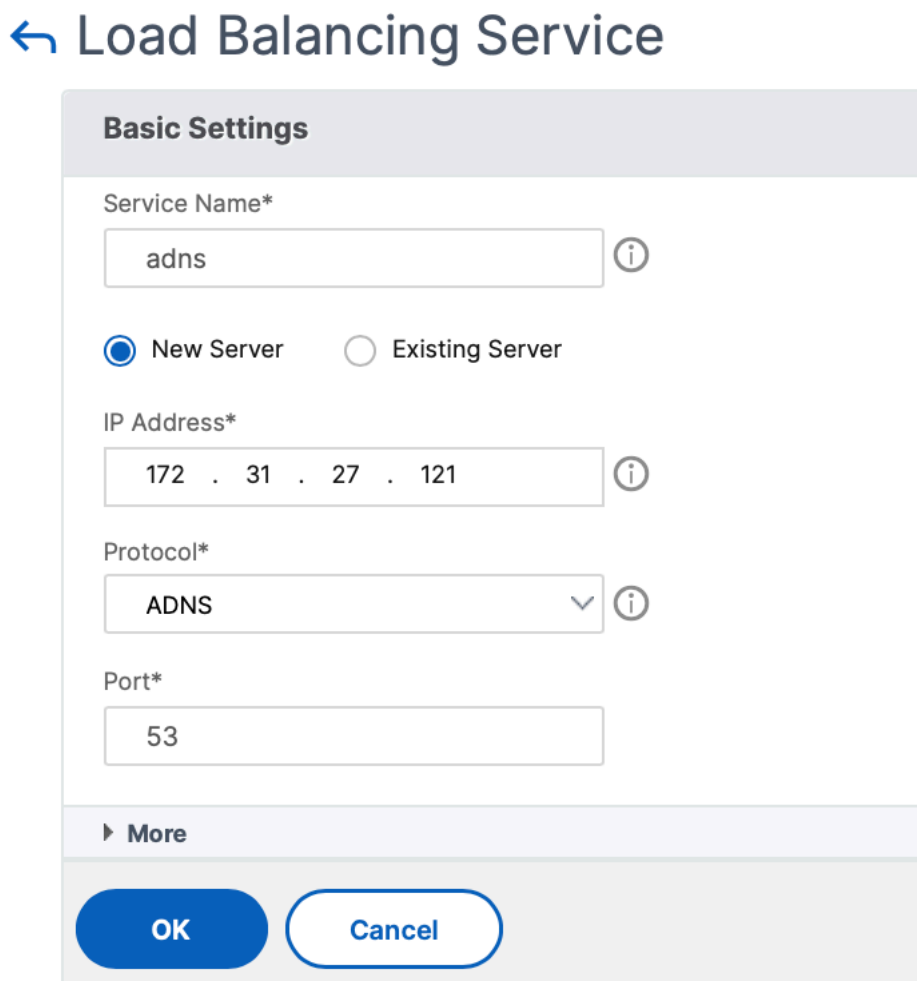


14. GSLB 仮想サーバードメインバインディングを設定するには、「**GSLB** 仮想サーバードメインバインディングなし」をクリックします。FQDN とバインドを設定します。他のパラメータはデフォルト設定のままにします。



15. [サービスなし] をクリックして **ADNS** サービスを設定します。
16. 詳細を指定して負荷分散サービスを設定します。

    サービス名を追加し、[新規サーバー] をクリックして、ADNS サーバーの **IP** アドレスを入力します。ユーザ ADNS がすでに設定されている場合、ユーザは [既存のサーバ] を選択し、ドロップダウンメニューからユーザ ADNS を選択できます。プロトコルが ADNS で、トラフィックがポート 53 を経由するように設定されていることを確認します。



17. 方法を [最小接続] に、[バックアップ方法] を [ラウンドロビン] に設定します。
18. 「完了」をクリックし、ユーザーの GSLB 仮想サーバーが「Up」と表示されていることを確認します。

そのほかの参照先

ハイブリッドおよびマルチクラウド環境向けの NetScaler グローバル負荷分散

NetScaler Gateway アプライアンスのアドレスプールのイントラネット IP を構成する

October 17, 2024

場合によっては、NetScaler Gateway プラグインを使用して接続するユーザーは、NetScaler Gateway アプライアンス用に一意の IP アドレスが必要です。グループのアドレスプール（IP プーリングとも呼ばれる）を有効にすると、NetScaler Gateway アプライアンスは一意の IP アドレスエイリアスを各ユーザーに割り当てることができます。アドレスプールは、イントラネット IP (IIP) アドレスを使用して構成します。

Azure にデプロイされた NetScaler Gateway アプライアンスでアドレスプールを構成するには、次の 2 ステップの手順に従います。

- アドレスプールで使用されるプライベート IP アドレスを Azure に登録する
- NetScaler Gateway アプライアンスでのアドレスプールの構成

Azure ポータルにプライベート IP アドレスを登録する

Azure では、複数の IP アドレスを持つ NetScaler ADC VPX インスタンスを展開できます。次の 2 つの方法で IP アドレスを VPX インスタンスに追加できます。

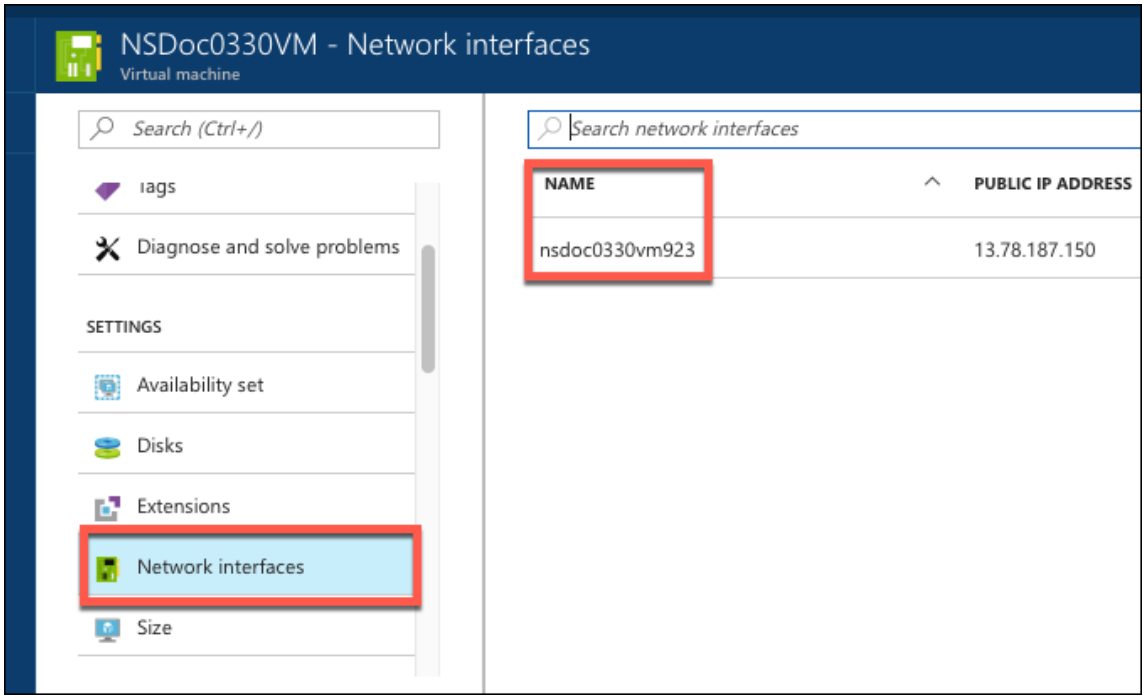
a. VPX インスタンスの Provisioning 中

VPX インスタンスのプロビジョニング中に複数の IP アドレスを追加する方法の詳細については、「NetScaler スタンドアロン インスタンスに複数の IP アドレスを構成する」を参照してください。VPX インスタンスのプロビジョニング中に PowerShell コマンドを使用して IP アドレスを追加するには、「PowerShell コマンドを使用してスタンドアロン モードで NetScaler VPX インスタンスに複数の IP アドレスを構成する」を参照してください。

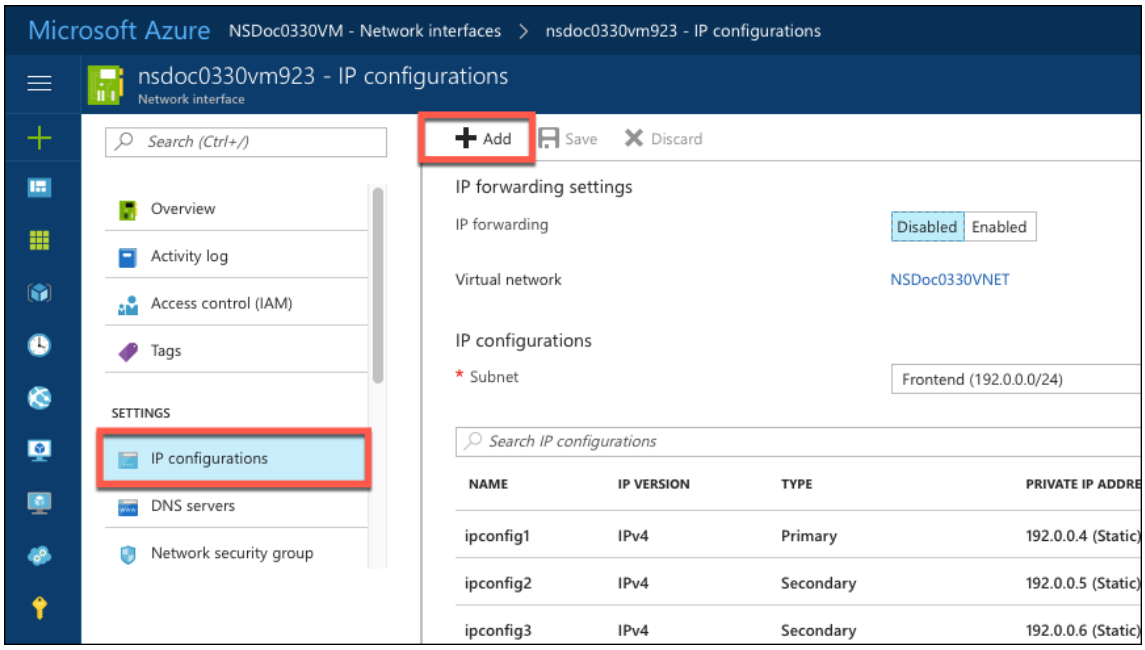
b. VPX インスタンスをプロビジョニング後

VPX インスタンスをプロビジョニングしたら、次の手順に従って Azure ポータルにプライベート IP アドレスを登録します。この IP アドレスは、NetScaler Gateway アプライアンスでアドレスプールとして構成します。

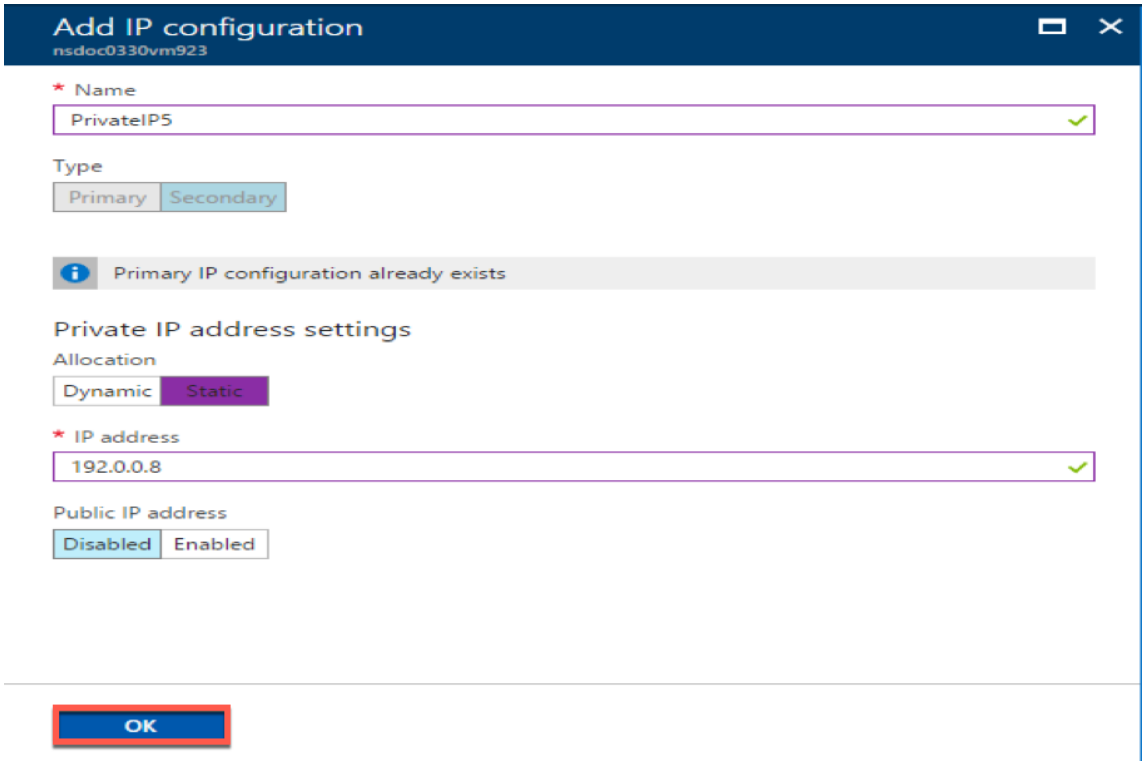
1. Azure Resource Manager（ARM）から、すでに作成されている NetScaler VPX インスタンス > ネットワークインターフェイスに移動します。登録する IIP が属しているサブネットにバインドされているネットワークインターフェイスを選択します。



2. [**IP** 構成] をクリックし、[追加] をクリックします。



3. 以下の例のように必要な詳細を入力し、[**OK**] をクリックします。



NetScaler Gateway アプライアンスでアドレスプールを構成する

NetScaler Gateway でアドレス プールを構成する方法の詳細については、「アドレス プールの構成」を参照してください。

> 制限事項:
>
> IIP アドレスの範囲をユーザーにバインドすることはできません。アドレスプールで使用されるすべての IIP アドレスを登録する必要があります。

PowerShell コマンドを使用して、**NetScaler VPX** スタンドアロンインスタンスに複数の **IP** アドレスを構成する

October 17, 2024

Azure 環境では、複数の NIC を設定して NetScaler VPX 仮想アプライアンスを展開できます。各 NIC に複数の IP アドレスを設定できます。このセクションでは、PowerShell コマンドを使用して、単一の NIC と複数の IP アドレスを使用して Citrix ADC VPX インスタンスを展開する方法について説明します。複数 NIC と複数 IP の展開にも同じスクリプトを使用できます。

> 注
>
> このドキュメントでは、IP-config は、個々の NIC に関連付けられている IP アドレス、パブリック IP、プライベート IP のペアを指します。詳細については、「Azure 用語」セクションを参照してください。

使用例

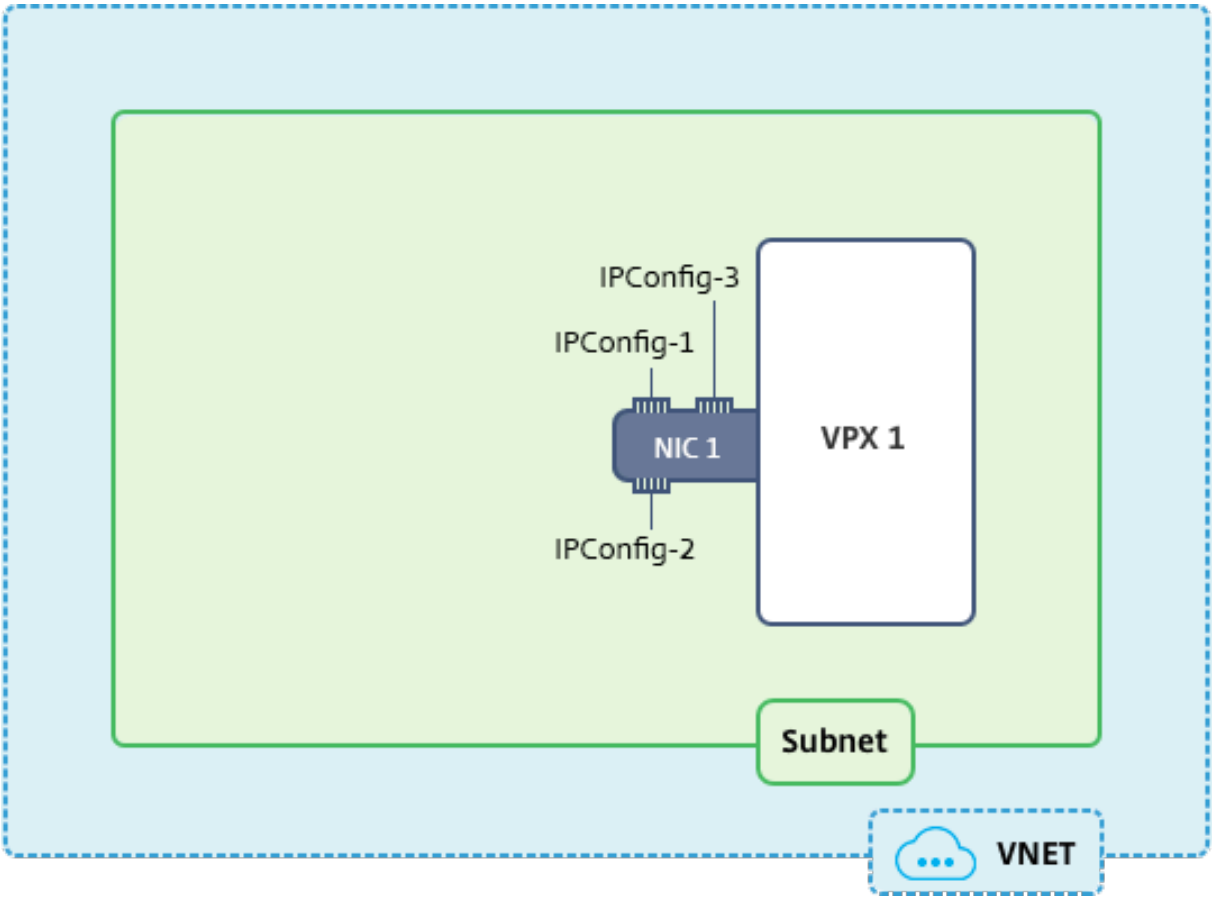
この使用例では、1 つの NIC が仮想ネットワーク（VNET）に接続されています。この NIC には、次の表に示す 3 つの IP 構成が関連付けられています。

IP コンフィグ	関連付けられている
IPConfig-1	静的パブリック IP アドレス; 静的プライベート IP アドレス
IPConfig-2	静的パブリック IP アドレス; 静的プライベートアドレス
IPConfig-3	静的プライベート IP アドレス

> 注
>
> IPConfig-3 は、パブリック IP アドレスに関連付けられていません。

図: トポロジ

次の図はこの使用例を視覚的に示しています。



> 注
>
> マルチ NIC、マルチ IP Azure NetScaler VPX 展開では、プライマリ（最初の）NIC のプライマリ（最初）`IPConfig`に関連付けられたプライベート IP アドレスが、アプライアンスの管理 NSIP アドレスとして自動的に追加されます。`IPConfigs`に関連付けられた残りのプライベート IP アドレスは、要件に応じて`add ns ip`コマンドを使用して、VPX インスタンスに VIP または SNIP として追加する必要があります。

スタンドアロンモードで NetScaler VPX 仮想アプライアンスに対して複数 IP アドレスを構成する場合に必要な手順の概要は次のとおりです。

1. リソースグループの作成
2. ストレージアカウントの作成
3. 可用性セットの作成
4. ネットワークサービスグループの作成
5. 仮想ネットワークの作成
6. パブリック IP アドレスの作成
7. IP 構成の割り当て
8. NIC の作成
9. NetScaler VPX インスタンスの作成
10. NIC 構成のチェック

11. VPX 側の構成のチェック

スクリプト

パラメーター

このドキュメントのこの使用例のサンプルパラメーター設定は、次のとおりです。必要な場合は、異なる設定を使用できます。

$locName=”westcentralus”

$rgName=”Azure-MultiIP”

$nicName1=”VM1-NIC1”

$vNetName=”Azure-MultiIP-vnet”

$vNetAddressRange=”11.6.0.0/16”

$frontEndSubnetName=”frontEndSubnet”

$frontEndSubnetRange=”11.6.1.0/24”

$prmStorageAccountName=”multiipstorage”

$avSetName=”multiip-avSet”

$vmsize= “Standard_DS4_v2”(このパラメータは最大 4 つの NIC を持つ仮想マシンを作成します。)

注**:** VPX インスタンスの最小要件は、2 つの vCPU と 2 GB RAM です。

$ パブリッシャー =「Citrix」

$offer=”netscalervpx110-6531”（異なる offer を使用できます）

$sku=”netscalerbyol”（offer によって、異なる SKU にすることができます）

$version=”latest”

$pubIPName1=”PIP1”

$pubIPName2=”PIP2”

$domName1=”multiipvpx1”

$domName2=”multiipvpx2”

$vmNamePrefix=”VPXMultiIP”

$osDiskSuffix=”osmultiipalbdiskdb1”

ネットワークセキュリティグループ **(NSG)** 関連情報:

$nsgName=”NSG-MultiIP”

$rule1Name=” Inbound-HTTP”

$rule2Name=” Inbound-HTTPS”

$rule3Name=” Inbound-SSH”

$IpConfigName1=” IPConfig1”

$IPConfigName2=” IPConfig-2”

$IPConfigName3=” IPConfig-3”

1. リソースグループを作成する

```
New-AzureRmResourceGroup -Name $rgName -Location $locName
```

2. ストレージアカウントを作成する

```
$prmStorageAccount = New-AzureRMStorageAccount -Name $prmStorageAccountName
 -ResourceGroupName $rgName -Type Standard_LRS -Location $locName
```

3. 可用性セットを作成する

```
$avSet = New-AzureRMAvailabilitySet -Name $avSetName -ResourceGroupName
 $rgName -Location $locName
```

4. ネットワークセキュリティグループを作成する

1. 規則を追加します。トラフィックを処理するポートのネットワークセキュリティグループにルールを追加する必要があります。

   ```
   $rule1=New-AzureRmNetworkSecurityRuleConfig -Name $rule1Name -
   Description "Allow HTTP"-Access Allow -Protocol Tcp -Direction
   Inbound -Priority 101 -SourceAddressPrefix Internet -SourcePortRange
    * -DestinationAddressPrefix * -DestinationPortRange 80   $rule2=
   New-AzureRmNetworkSecurityRuleConfig -Name $rule2Name -Description
    "HTTPS を許可する"-Access Allow -Protocol Tcp -Direction Inbound -
   Priority 110 -SourceAddressPrefix Internet -SourcePortRange * -
   DestinationAddressPrefix * -DestinationPortRange 443   $rule3=New
   -AzureRmNetworkSecurityRuleConfig -Name $rule3Name -Description
    "SSH を許可する"-Access Allow -Protocol Tcp -Direction Inbound -
   Priority 120 -SourceAddressPrefix Internet -SourcePortRange * -
   DestinationAddressPrefix * -DestinationPortRange 22
   ```


2. ネットワークセキュリティグループオブジェクトを作成します。

```
$nsg=New-AzureRmNetworkSecurityGroup -ResourceGroupName $rgName
-Location $locName -Name $nsgName -SecurityRules $rule1,$rule2,
$rule3
```

5. 仮想ネットワークを作成する

1. サブネットを追加します。

```
$frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
$frontEndSubnetName -AddressPrefix $frontEndSubnetRange
```

2. 仮想ネットワークオブジェクトを追加します。

```
$vnet=New-AzureRmVirtualNetwork -Name $vNetName -ResourceGroupName
 $rgName -Location $locName -AddressPrefix $vNetAddressRange -
Subnet $frontendSubnet
```

3. サブネットを取得します。

```
$subnetName="フロントエンドサブネット" $subnet1=$vnet.サブネット|?{ $_.Name -
eq $subnetName }
```

6. パブリック **IP** アドレスを作成する

```
$pip1=New-AzureRmPublicIpAddress -Name $pubIPName1 -ResourceGroupName
 $rgName -DomainNameLabel $domName1 -Location $locName -AllocationMethod
 Static$pip2=New-AzureRmPublicIpAddress -Name $pubIPName2 -ResourceGroupName
 $rgName -DomainNameLabel $domName2 -Location $locName -AllocationMethod
 Static
```

> 注
>
> 使用する前にドメイン名の可用性をチェックします。
>
> IP アドレスの割り当て方法は動的または静的にできます。

7. IP 設定を割り当てる

この使用例では、IP アドレスを割り当てる前に次の点を検討します。

- IPConfig-1 が VPX1 の subnet1 に属していること
- IPConfig-2 が VPX1 の subnet 1 に属していること
- IPConfig-3 が VPX1 の subnet 1 に属していること

> 注
>
> 複数の IP 構成を 1 つの NIC に割り当てるときには、1 つの構成をプライマリとして割り当てる必要があります。

```
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
```

サブネットの要件に合う有効な IP アドレスを使用して、その可用性をチェックします。

8. NIC を作成する

```
$nic1=New-AzureRmNetworkInterface -Name $nicName1 -ResourceGroupName
 $rgName -Location $locName -IpConfiguration $IpConfig1,$IpConfig2,
$IPConfig3 -NetworkSecurityGroupId $nsg.Id
```

9. NetScaler VPX インスタンスを作成する

1. 変数を初期化します。

   ```
   $suffixNumber = 1 $vmName = $vmNamePrefix + $suffixNumber
   ```

2. VM Config オブジェクトを作成します。

   ```
   $vmConfig=New-AzureRMVMConfig -VMName $vmName -VMSize $vmSize -
   AvailabilitySetId $avSet.Id
   ```

3. 資格情報、OS、イメージを設定します。

   ```
   $cred=Get-Credential -Message "VPXログインの名前とパスワードを入力してくだ
   さい。"       $vmConfig=Set-AzureRMVMOperatingSystem -VM $vmConfig -
   Linux -ComputerName $vmName -Credential $cred       $vmConfig=Set-
   AzureRMVMSourceImage -VM $vmConfig -PublisherName $publisher -
   Offer $offer -SKU $sku -バージョン $version
   ```

4. NIC を追加します。

   ```
   $vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic1.
   Id -Primary
   ```


> 注
>
> マルチ NIC NetScaler VPX 展開では、1 つの NIC がプライマリである必要があります。したがって、その NIC を NetScaler VPX インスタンスに追加するときに、「-Primary」を追加する必要があります。

5. OS ディスクを指定して、VM を作成します。

```
$osDiskName=$vmName + "-"+ $osDiskSuffix1$osVhdUri=$prmStorageAccount
.PrimaryEndpoints.Blob.ToString()+ "vhds/"+ $osDiskName + ".vhd"
$vmConfig=Set-AzureRMVMOSDisk -VM $vmConfig -名前 $osDiskName -
VhdUri $osVhdUri -CreateOption fromImage   Set-AzureRmVMPlan -VM
$vmConfig -発行元 $publisher -製品 $offer -名前 $sku New-AzureRMVM -VM
 $vmConfig -リソースグループ名 $rgName -場所 $locName
```

10. NIC 設定を確認する

NetScaler VPX インスタンスが起動したら、以下のコマンドを使用して、NetScaler VPX NIC の `IPConfigs` に割り当てられている IP アドレスを確認できます。

```
$nic.IPConfig
```

11. VPX 側の設定を確認する

NetScaler VPX インスタンスが起動すると、`IPconfig` プライマリ NIC のプライマリに関連付けられたプライベート IP アドレスが NSIP アドレスとして追加されます。残りのプライベート IP アドレスは、要件に従って、VIP または SNIP アドレスとして追加する必要があります。次のコマンドを使用します。

```
add nsip <Private IPAddress><netmask> -type VIP/SNIP
```

スタンドアロンモードの NetScaler VPX インスタンスに対して複数 IP アドレスを構成しました。

Azure 展開の追加の PowerShell スクリプト

October 17, 2024

このセクションでは、Azure PowerShell で次の構成を実行できる PowerShell コマンドレットについて説明します。

- NetScaler VPX スタンドアロンインスタンスのプロビジョニング
- Azure 外部ロードバランサーを使用した高可用性セットアップで NetScaler VPX ペアをプロビジョニングします

- Azure 内部ロードバランサーを使用した高可用性セットアップで NetScaler VPX ペアをプロビジョニングします

PowerShell コマンドを使用して実行できる構成については、次のトピックも参照してください。

- PowerShell コマンドを使用して複数の IP アドレスと NIC を使用して高可用性セットアップを構成する
- NetScaler VPX インスタンスで GSLB を構成する
- NetScaler のアクティブ/スタンバイ高可用性セットアップで GSLB を構成する
- PowerShell コマンドを使用して、スタンドアロンモードの NetScaler ADC VPX インスタンスで複数の IP アドレスを構成する

NetScaler VPX スタンドアロンインスタンスのプロビジョニング

1. リソースグループの作成

   リソースグループには、ソリューションのすべてのリソースを含めることも、グループとして管理するリソースのみを含めることもできます。ここで指定した場所は、そのリソースグループ内のリソースのデフォルトの場所です。ロードバランサーを作成する場合、すべてのコマンドで同じリソースグループを使用してください。

   ```
   $rgName="&lt;resource group name&gt;" $locName="&lt;location name, such as West US&gt;  New-AzureRmResourceGroup -名前 $rgName -場所 $locName
   ```

   例えば:

   ```
   1    $rgName = "ARM-VPX"
   2    $locName = "West US"
   3    New-AzureRmResourceGroup -Name $rgName -Location $locName
   ```

2. ストレージアカウントの作成

   ストレージアカウントには、小文字と数字のみを含む一意の名前を選択する必要があります。

   ```
   $saName="&lt;storage account name&gt;"      $saType="&lt;storage account type&gt;"、1 つ指定してください: Standard_LRS、Standard_GRS、Standard_RAGRS 、 または Premium_LRS New-AzureRmStorageAccount -名前 $saName -リソースグループ名 $rgName -タイプ $saType -場所 $locName
   ```

   例えば:

   ```
   1    $saName="vpxstorage"
   2    $saType="Standard\_LRS"
   3    New-AzureRmStorageAccount -Name $saName -ResourceGroupName
           $rgName -Type $saType -Location $locName
   ```

3. アベイラビリティセットの作成

可用性セットにより、メンテナンス時などのダウンタイム中でも仮想マシンを使用し続けることができます。可用性セットが構成されたロードバランサーでは、アプリケーションをいつでも使用できます。

```
$avName="<availability set name>"

New-AzureRmAvailabilitySet -Name $avName -ResourceGroupName
$rgName -Location $locName
```

4. 仮想ネットワークの作成

以前に作成されたサブネットがない場合、少なくとも 1 つのサブネットを持つ新しい仮想ネットワークを追加します。

```
$FrontendAddressPrefix="10.0.1.0/24"      $BackendAddressPrefix="
10.0.2.0/24"   $vnetAddressPrefix="10.0.0.0/16"   $frontendSubnet
=New-AzureRmVirtualNetworkSubnetConfig -Name frontendSubnet -
AddressPrefix $FrontendAddressPrefix$backendSubnet=New-AzureRmVirtualNetwo
 -Name backendSubnet -AddressPrefix $BackendAddressPrefix    New-
AzureRmVirtualNetwork -Name TestNet -ResourceGroupName $rgName
 -Location $locName -AddressPrefix $vnetAddressPrefix -Subnet
$frontendSubnet、$backendSubnet
```

例えば:

```
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. **NIC** を作成する

NIC を作成し、それを NetScaler VPX インスタンスに関連付けます。上記の手順で作成されたフロントエンドサブネットは 0 でインデックス付けされ、バックエンドサブネットは 1 でインデックス付けされます。次の 3 つのいずれかの方法で NIC を作成します。

a) パブリック *IP* アドレスを持つ *NIC*

```
$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
```


b) パブリック *IP* アドレスと *DNS* ラベルが付けられた *NIC*

```
$nicName="<name of the NIC of the VM>"

$domName="<domain name label>"

$pip = New-AzureRmPublicIpAddress -Name $nicName -ResourceGroupName
 $rgName -DomainNameLabel $domName -Location $locName -AllocationMethod
 Dynamic
```

$domName を割り当てる前に、次のコマンドを使用して、それが利用できるかどうかを確認します。

```
Test-AzureRmDnsAvailability -DomainQualifiedName $domName -
Location $locName

$nic = New-AzureRmNetworkInterface -Name $nicName -ResourceGroupName
 $rgName -Location $locName -SubnetId $vnet.Subnets[$subnetIndex
].Id -PublicIpAddressId $pip.Id
```

例えば:

```
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
```

c) 動的パブリックアドレスと静的プライベート *IP* アドレスを持つ *NIC*

仮想マシンに追加するプライベート（静的）IP アドレスが、指定したサブネットのアドレスと同じ範囲である必要があります。

```
$nicName="<name of the NIC of the VM>"

$staticIP="<available static IP address on the subnet>"

$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 -PrivateIpAddress $staticIP
```

6. 仮想オブジェクトの作成

```
$vmName="<VM name>"
```



```
$vmSize="<VM size string>"

$avSet=Get-AzureRmAvailabilitySet -Name $avName -ResourceGroupName
 $rgName

$vm=New-AzureRmVMConfig -VMName $vmName -VMSize $vmSize -AvailabilitySetId
 $avset.Id
```

7. **NetScaler VPX** イメージを取得

```
$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."
```

VPX へのログインに使用する資格情報を入力してください

```
$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
```

例えば:

```
$pubName="citrix"
```

次のコマンドを使用すると、Citrix からのすべてのオファーが表示されます。

```
1   Get-AzureRMVMImageOffer -Location $locName -Publisher $pubName |
        Select Offer
2
3   $offerName="netscalervpx110-6531"
```

次のコマンドは、特定のオファー名について発行元から提供される SKU を知るために使用します。

```
Get-AzureRMVMImageSku -Location $locName -Publisher $pubName -
Offer $offerName | Select Skus
```

8. 仮想マシンの作成

```
$diskName="<name identifier for the disk in Azure storage, such
as OSDisk>"
```

例えば:

```
1   $diskName="dynamic"
2
```



```
$pubName="citrix"

$offerName="netscalervpx110-6531"

$skuName="netscalerbyol"

$storageAcc=Get-AzureRmStorageAccount -ResourceGroupName $rgName
    -Name $saName

$osDiskUri=$storageAcc.PrimaryEndpoints.Blob.ToString() + "vhds/
   " + $diskName  + ".vhd"

$vm=Set-AzureRmVMOSDisk -VM $vm -Name $diskName -VhdUri
   $osDiskUri -CreateOption fromImage
```

MarketPlace に存在するイメージから VM を作成する場合、次のコマンドを使用して VM プランを指定します。

```
Set-AzureRmVMPlan -VM $vm -Publisher $pubName -Product $offerName
 -Name $skuName

New-AzureRmVM -ResourceGroupName $rgName -Location $locName -VM
$vm
```

Azure 外部ロードバランサーを使用した高可用性セットアップで **NetScaler VPX** ペアをプロビジョニングします

Azure のユーザー資格情報を使用して AzureRmAccount にログオンします。

1. リソースグループの作成

   ここで指定した場所は、そのリソースグループ内のリソースのデフォルトの場所です。ロードバランサーを作成する場合、すべてのコマンドで同じリソースグループを使用してください。

   ```
   $rgName="<resource group name>"

   $locName="<location name, such as West US>"

   New-AzureRmResourceGroup -Name $rgName -Location $locName
   ```

   例えば:

   ```
   $rgName = "ARM-LB-NS"

   $locName = "West US"

   New-AzureRmResourceGroup -Name $rgName -Location $locName
   ```

2. ストレージアカウントの作成

   ストレージアカウントには、小文字と数字のみを含む一意の名前を選択する必要があります。


```
$saName="<storage account name>"
```

```
$saType="&lt;storage account type&gt;"
```
、1 つ指定してください: Standard_LRS、Standard_GRS、Standard_RAGRS 、または Premium_LRS

```
New-AzureRmStorageAccount -Name $saName -ResourceGroupName
$rgName -Type $saType -Location $locName
```

例えば:

```
$saName="vpxstorage"

$saType="Standard_LRS"

New-AzureRmStorageAccount -Name $saName -ResourceGroupName
    $rgName -Type $saType -Location $locName
```

3. アベイラビリティセットの作成

   可用性セットが構成されたロードバランサーでは、アプリケーションをいつでも使用できます。

   ```
   $avName="<availability set name>"
   ```

   ```
   New-AzureRmAvailabilitySet -Name $avName -ResourceGroupName
   $rgName -Location $locName
   ```

4. 仮想ネットワークの作成

   以前に作成されたサブネットがない場合、少なくとも 1 つのサブネットを持つ新しい仮想ネットワークを追加します。

   ```
   $vnetName = "LBVnet"

   $FrontendAddressPrefix="10.0.1.0/24"

   $BackendAddressPrefix="10.0.2.0/24"

   $vnetAddressPrefix="10.0.0.0/16"

   $frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
       frontendSubnet -AddressPrefix $FrontendAddressPrefix

   $backendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
       backendSubnet -AddressPrefix $BackendAddressPrefix

   $vnet=New-AzureRmVirtualNetwork -Name $vnetName -
       ResourceGroupName $rgName -Location $locName -AddressPrefix
       $vnetAddressPrefix -Subnet $frontendSubnet,$backendSubnet
   ```

   > 注
   >
   > 要件に応じて AddressPrefix パラメータ値を選択します。

フロントエンドおよびバックエンドサブネットを、前の手順で作成した仮想ネットワークに割り当てます。

フロントエンドサブネットが配列 VNet の最初の要素である場合、subnetId は $vnet.subnets [0] .Id でなければなりません。

フロントエンドサブネットが配列の 2 番目の要素である場合、subnetId は $vnet.subnets [1] .Id というようにする必要があります。

5. フロントエンド **IP** アドレスを構成し、バックエンドアドレスプールを作成する

   受信ロードバランサーネットワークトラフィック用のフロントエンド IP アドレスを構成し、負荷分散トラフィックを受信するバックエンドアドレスプールを作成します。

```
$pubName="PublicIp1"

$publicIP1 = New-AzureRmPublicIpAddress -Name $pubName -
    ResourceGroupName $rgName -Location $locName -
    AllocationMethod Static -DomainNameLabel nsvpx
```

   注

   DomainNameLabel の値が使用可能かどうかを確認します。

```
        $FIPName = "ELBFIP"

        $frontendIP1 = New-AzureRmLoadBalancerFrontendIpConfig -
            Name $FIPName -PublicIpAddress $publicIP1

        $BEPool = "LB-backend-Pool"

        $beaddresspool1= New-
            AzureRmLoadBalancerBackendAddressPoolConfig -Name
            $BEPool
```

6. ヘルスプローブの作成

   ポート 9000、間隔 5 秒で TCP ヘルププローブを作成します。

```
$healthProbe = New-AzureRmLoadBalancerProbeConfig -Name
    HealthProbe -Protocol Tcp -Port 9000 -IntervalInSeconds 5 -
    ProbeCount 2
```

7. 負荷分散ルールを作成する

   負荷分散するサービスごとに LB ルールを作成します。

   例えば:

   次の例を使用して、HTTP サービスの負荷分散を行うことができます。

```
$lbrule1 = New-AzureRmLoadBalancerRuleConfig -Name "HTTP-LB" -
    FrontendIpConfiguration $frontendIP1 -BackendAddressPool
    $beAddressPool1 -Probe $healthProbe -Protocol Tcp -
    FrontendPort 80 -BackendPort 80
```


8. インバウンド **NAT** ルールの作成

   負荷分散していないサービスに対する NAT 規則を作成します。

   たとえば、NetScaler VPX インスタンスへの SSH アクセスを作成する場合などです。

   > 注
   >
   > 2 つの NAT ルールでは、Protocol-FrontEndPort-BackendPort トリプレットが同じであってはなりません。

   ```
   $inboundNATRule1= New-
       AzureRmLoadBalancerInboundNatRuleConfig -Name SSH1
           -FrontendIpConfiguration $frontendIP1 -Protocol
       TCP -FrontendPort 22 -BackendPort 22

   $inboundNATRule2= New-
       AzureRmLoadBalancerInboundNatRuleConfig -Name SSH2 -
       FrontendIpConfiguration $frontendIP1 -Protocol TCP -
       FrontendPort 10022 -BackendPort 22
   ```

9. ロードバランサーエンティティの作成

   すべてのオブジェクト（NAT 規則、ロードバランサー規則、プローブ構成）を一度に追加してロードバランサーを作成します。

   ```
   $lbName="ELB"

   $NRPLB = New-AzureRmLoadBalancer -ResourceGroupName $rgName -
       Name $lbName -Location $locName -InboundNatRule
       $inboundNATRule1, $inboundNATRule2 -FrontendIpConfiguration
       $frontendIP1 -LoadBalancingRule $lbrule1 -BackendAddressPool
       $beAddressPool1 -Probe $healthProbe
   ```

10. **NIC** を作成する

    2 つの NIC を作成し、各 NIC を各 VPX インスタンスに関連付けます

    a）NIC1 を VPX1 に

    例えば:

    ```
    $nicName="NIC1"

    $lbName="ELB"

    $bePoolIndex=0

    \* Rule indexes starts from 0.

    $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 を VPX2 に

例えば:

```
1    $nicName="NIC2"
2
3    $lbName="ELB"
4
5    $bePoolIndex=0
6
7    $natRuleIndex=1
8
9    \* Second Inbound NAT (SSH) rule we need to use
10
11   `$subnetIndex=0
12
13   \* Frontend subnet index
14
15   $lb=Get-AzureRmLoadBalancer -Name $lbName -ResourceGroupName
         $rgName
16
17   $nic2=New-AzureRmNetworkInterface -Name $nicName -
         ResourceGroupName $rgName -Location $locName -Subnet $vnet.
         Subnets\[$subnetIndex\] -LoadBalancerBackendAddressPool $lb.
         BackendAddressPools\[$bePoolIndex\] -
         LoadBalancerInboundNatRule  $lb.InboundNatRules\[
         $natRuleIndex\]
```

11. **NetScaler VPX** インスタンスの作成

    2 つの NetScaler VPX インスタンスを、同じリソースグループおよび可用性セットの一部として作成し、外部ロードバランサーに割り当てます。

    a）NetScaler VPX インスタンス 1

    例えば:

```
1    $vmName="VPX1"
2
```



```
$vmSize="Standard\_A3"

$pubName="citrix"

$offerName="netscalervpx110-6531"

$skuName="netscalerbyol"

$avSet=Get-AzureRmAvailabilitySet -Name $avName -
    ResourceGroupName $rgName

$vm1=New-AzureRmVMConfig -VMName $vmName -VMSize $vmSize -
    AvailabilitySetId $avset.Id

$cred=Get-Credential -Message "Type Credentials which will be
    used to login to VPX instance"

$vm1=Set-AzureRmVMOperatingSystem -VM $vm1 -Linux -ComputerName
    $vmName -Credential $cred -Verbose

$vm1=Set-AzureRmVMSourceImage -VM $vm1 -PublisherName $pubName -
    Offer $offerName -Skus $skuName -Version "latest"

$vm1=Add-AzureRmVMNetworkInterface -VM $vm1 -Id $nic1.Id

$diskName="dynamic"

$storageAcc=Get-AzureRmStorageAccount -ResourceGroupName $rgName
     -Name $saName

$osDiskUri1=$storageAcc.PrimaryEndpoints.Blob.ToString() + "
    vhds1/" + $diskName  + ".vhd"

$vm1=Set-AzureRmVMOSDisk -VM $vm1 -Name $diskName -VhdUri
    $osDiskUri1 -CreateOption fromImage

Set-AzureRmVMPlan -VM $vm1 -Publisher $pubName -Product
    $offerName -Name $skuName

New-AzureRmVM -ResourceGroupName $rgName -Location $locName -VM
    $vm1
```

b）NetScaler VPX インスタンス 2

例えば:

```
$vmName="VPX2"

$vmSize="Standard\_A3"

$avSet=Get-AzureRmAvailabilitySet -Name $avName -
    ResourceGroupName $rgName

$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  $osDiskUri1=$storageAcc.PrimaryEndpoints.Blob.ToString() + "
        vhds2/" + $diskName  + ".vhd"
22
23  $vm2=Set-AzureRmVMOSDisk -VM $vm2 -Name $diskName -VhdUri
        $osDiskUri1 -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. 仮想マシンを構成する

    両方の NetScaler VPX インスタンスが開始された場合、SSH プロトコル経由で両方の VPX インスタンスに接続して仮想マシンを構成します。

    a) アクティブ-アクティブ: 両方の NetScaler VPX インスタンスのコマンドラインで同じ構成コマンドセットを実行します。

    b) アクティブ/パッシブ: このコマンドを両方の NetScaler VPX インスタンスのコマンドラインで実行します。

    ```
    add ha node #nodeID <nsip of other NetScaler VPX>
    ```

    アクティブ-パッシブモードでは、プライマリノードでのみ構成コマンドを実行します。

Azure 内部ロードバランサーを使用した高可用性セットアップで **NetScaler VPX** ペアをプロビジョニングします

Azure のユーザー資格情報を使用して AzureRmAccount にログオンします。

1. リソースグループの作成

ここで指定した場所は、そのリソースグループ内のリソースのデフォルトの場所です。ロードバランサーを作成する場合、すべてのコマンドで同じリソースグループを使用してください。

```
$rgName="\\&#060;resource group name\\&#062;"

$locName="\\&#060;location name, such as West US\\&#062;"

New-AzureRmResourceGroup -Name $rgName -Location $locName
```

例えば:

```
1   $rgName = "ARM-LB-NS"
2
3   $locName = "West US"
4
5   New-AzureRmResourceGroup -Name $rgName -Location $locName
```

2. ストレージアカウントの作成

   ストレージアカウントには、小文字と数字のみを含む一意の名前を選択する必要があります。

   ```
   $saName="<storage account name>"

   $saType="&lt;storage account type&gt;"、1 つ指定してください: Standard_LRS、Standard_GRS、Standard_RAGRS 、または Premium_LRS

   New-AzureRmStorageAccount -Name $saName -ResourceGroupName $rgName -Type $saType -Location $locName
   ```

   例えば:

   ```
   1   $saName="vpxstorage"
   2
   3   $saType="Standard_LRS"
   4
   5   New-AzureRmStorageAccount -Name $saName -ResourceGroupName
           $rgName -Type $saType -Location $locName
   ```

3. アベイラビリティセットの作成

   可用性セットが構成されたロードバランサーでは、アプリケーションをいつでも使用できます。

   ```
   $avName="<availability set name>"

   New-AzureRmAvailabilitySet -Name $avName -ResourceGroupName $rgName -Location $locName
   ```

4. 仮想ネットワークの作成

   以前に作成されたサブネットがない場合、少なくとも 1 つのサブネットを持つ新しい仮想ネットワークを追加します。

   ```
   1   $vnetName = "LBVnet"
   2
   ```



```
$vnetAddressPrefix="10.0.0.0/16"

$FrontendAddressPrefix="10.0.1.0/24"

$BackendAddressPrefix="10.0.2.0/24"

$vnet=New-AzureRmVirtualNetwork -Name $vnetName -
    ResourceGroupName $rgName -Location $locName -AddressPrefix
    $vnetAddressPrefix -Subnet $frontendSubnet,$backendSubnet\`

$frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
    frontendSubnet -AddressPrefix $FrontendAddressPrefix

$backendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
    backendSubnet -AddressPrefix $BackendAddressPrefix
```

> 注
>
> 要件に応じて AddressPrefix パラメータ値を選択します。

フロントエンドおよびバックエンドサブネットを、前の手順で作成した仮想ネットワークに割り当てます。

フロントエンドサブネットが配列 VNet の最初の要素である場合、subnetId は $vnet.subnets [0] .Id でなければなりません。

フロントエンドサブネットが配列の 2 番目の要素である場合、subnetId は $vnet.subnets [1] .Id というようにする必要があります。

5. バックエンドアドレスプールを作成する

```
$beaddresspool= New-AzureRmLoadBalancerBackendAddressPoolConfig -
Name "LB-backend"
```

6. **NAT** ルールを作成する

負荷分散していないサービスに対する NAT 規則を作成します。

```
$inboundNATRule1= New-AzureRmLoadBalancerInboundNatRuleConfig -
    Name "Inboundnatrule1" -FrontendIpConfiguration $frontendIP -
    Protocol TCP -FrontendPort 3441 -BackendPort 3389

$inboundNATRule2= New-AzureRmLoadBalancerInboundNatRuleConfig -
    Name "RDP2" -FrontendIpConfiguration $frontendIP -Protocol
    TCP -FrontendPort 3442 -BackendPort 3389
```

要件に応じて、フロントエンドポートおよびバックエンドポートを使用します。

7. ヘルスプローブの作成

ポート 9000、間隔 5 秒で TCP ヘルププローブを作成します。

```
$healthProbe = New-AzureRmLoadBalancerProbeConfig -Name "
    HealthProbe"  " -Protocol tcp -Port 9000 -IntervalInSeconds 5
     -ProbeCount 2
```


8. 負荷分散ルールを作成する

   負荷分散するサービスごとに LB ルールを作成します。

   例えば:

   次の例を使用して、HTTP サービスの負荷分散を行うことができます。

```
$lbrule = New-AzureRmLoadBalancerRuleConfig -Name "lbrule1" -
    FrontendIpConfiguration $frontendIP -BackendAddressPool
    $beAddressPool -Probe $healthProbe -Protocol Tcp -
    FrontendPort 80 -BackendPort 80
```

   要件に応じて、フロントエンドポートおよびバックエンドポートを使用します。

9. ロードバランサーエンティティの作成

   すべてのオブジェクト（NAT 規則、ロードバランサー規則、プローブ構成）を一度に追加してロードバランサーを作成します。

```
$NRPLB = New-AzureRmLoadBalancer -ResourceGroupName $rgname -
    Name "InternalLB" -Location $locName -FrontendIpConfiguration
     $frontendIP -InboundNatRule $inboundNATRule1,
    $inboundNatRule2 -LoadBalancingRule $lbrule -
    BackendAddressPool $beAddressPool -Probe $healthProbe
```

10. **NIC** を作成する

    2 つの NIC を作成し、各 NIC を各 NetScaler VPX インスタンスに関連付けます

```
$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\[0\]
```

    この NIC は NetScaler VPX 1 用です。プライベート IP は、追加されたサブネットと同じサブネット内に存在する必要があります。

```
$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
    \[1\].
```

    この NIC は NetScaler ADC VPX 用です 2. `Private IPAddress`パラメーターには、要件に応じて任意のプライベート IP を設定できます。

11. **NetScaler VPX** インスタンスの作成

    同じリソースグループと可用性セットの一部である 2 つの VPX インスタンスを作成し、それを内部ロードバランサーにアタッチします。

a）NetScaler VPX インスタンス 1

例えば:

```
$vmName="VPX1"

$vmSize="Standard\_A3"

$avSet=Get-AzureRmAvailabilitySet -Name $avName -
    ResourceGroupName $rgName

$vm1=New-AzureRmVMConfig -VMName $vmName -VMSize $vmSize -
    AvailabilitySetId $avset.Id

$cred=Get-Credential -Message "Type Credentials which will be
    used to login to VPX instance"

$vm1=Set-AzureRmVMOperatingSystem -VM $vm1 -Linux -ComputerName
    $vmName -Credential $cred -Verbose

$vm1=Set-AzureRmVMSourceImage -VM $vm1 -PublisherName $pubName -
    Offer $offerName -Skus $skuName -Version "latest"

$vm1=Add-AzureRmVMNetworkInterface -VM $vm1 -Id $backendnic1.Id

$diskName="dynamic"

$storageAcc=Get-AzureRmStorageAccount -ResourceGroupName $rgName
     -Name $saName

$osDiskUri1=$storageAcc.PrimaryEndpoints.Blob.ToString() + "
    vhds1/" + $diskName  + ".vhd"

$vm1=Set-AzureRmVMOSDisk -VM $vm1 -Name $diskName -VhdUri
    $osDiskUri1 -CreateOption fromImage

Set-AzureRmVMPlan -VM $vm1 -Publisher $pubName -Product
    $offerName -Name $skuName

New-AzureRmVM -ResourceGroupName $rgName -Location $locName -VM
    $vm1
```

b）NetScaler VPX インスタンス 2

例えば:

```
$vmName="VPX2"

$vmSize="Standard\_A3"

$avSet=Get-AzureRmAvailabilitySet -Name $avName -
    ResourceGroupName $rgName

$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  $osDiskUri1=$storageAcc.PrimaryEndpoints.Blob.ToString() + "
        vhds2/" + $diskName  + ".vhd"
22
23  $vm2=Set-AzureRmVMOSDisk -VM $vm2 -Name $diskName -VhdUri
        $osDiskUri1 -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. 仮想マシンを構成する

    両方の NetScaler VPX インスタンスが開始された場合、SSH プロトコル経由で両方の VPX インスタンスに接続して仮想マシンを構成します。

    a) アクティブ-アクティブ: 両方の NetScaler VPX インスタンスのコマンドラインで同じ構成コマンドセットを実行します。

    b) アクティブ/パッシブ: このコマンドを両方の NetScaler VPX インスタンスのコマンドラインで実行します。

    ```
    add ha node #nodeID <nsip of other NetScaler VPX>
    ```

    アクティブ-パッシブモードでは、プライマリノードでのみ構成コマンドを実行します。

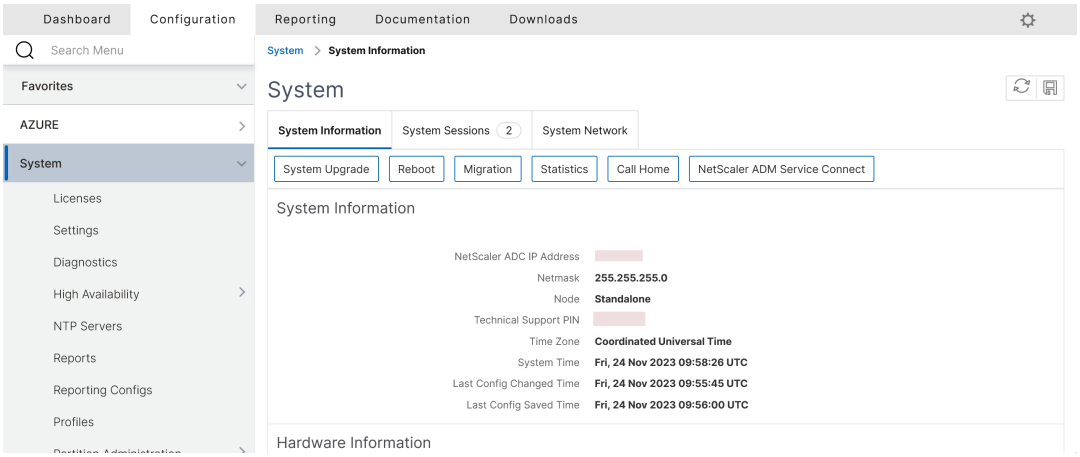
Create a support ticket for the VPX instance on Azure

April 23, 2024

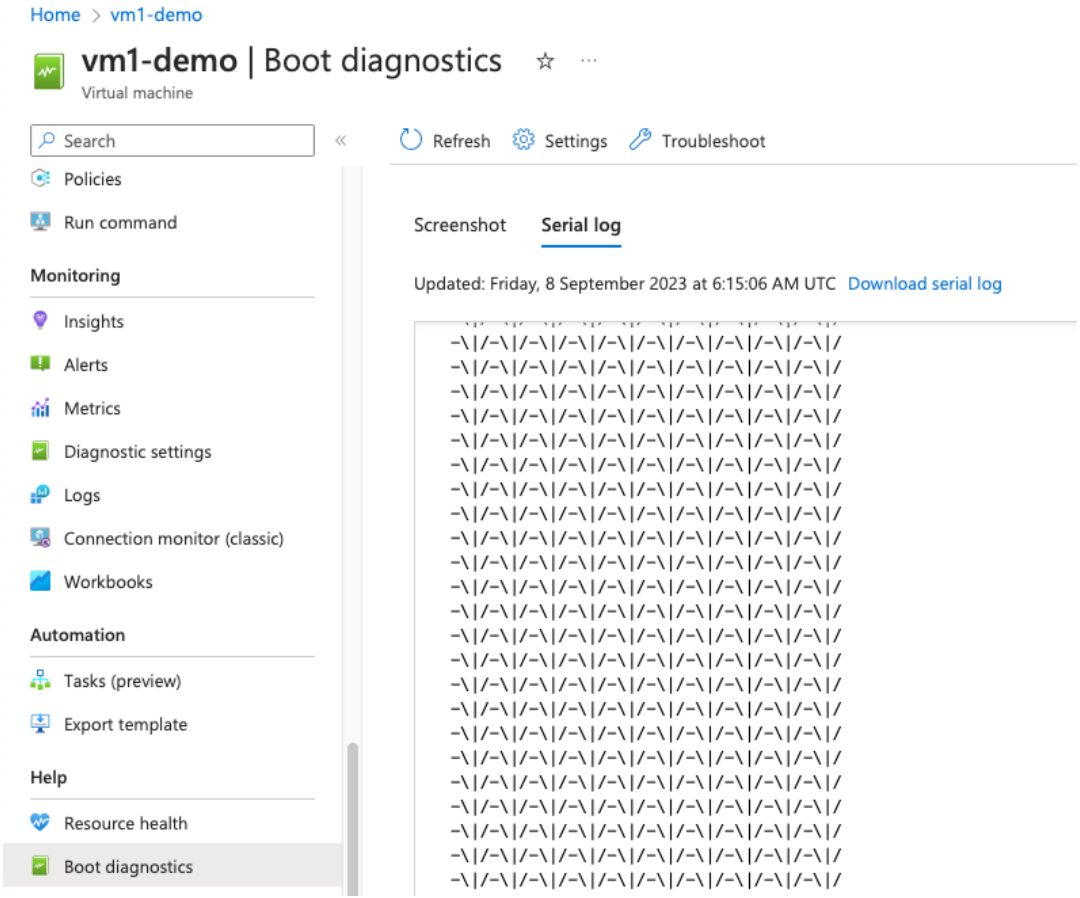
If you’ re experiencing issues with your NetScaler VPX instance on Azure, for troubleshooting, you can create a support ticket in the NetScaler support portal.

To file a support ticket, make sure the following:

- Your network is connected.
- You have your Azure account number, the support PIN code of the NetScaler subscription-based offering that you have deployed on Azure, and the Azure serial log handy.
  - You can find the support PIN code on the **Systems page** in the VPX GUI.



  - You can find the serial log in the Azure portal (**Boot diagnostics** section of your VM).



> **Note:**
>
> NetScaler supports subscription-based offerings on Azure (subscription license with hourly price).

Once you have all the information ready, call NetScaler support. You’re asked to provide your name and email address.

Azure に関するよくある質問

October 17, 2024

- **Azure Marketplace** からインストールされた **NetScaler VPX** インスタンスのアップグレード手順は、オンプレミスアップグレード手順とは異なりますか?

  いいえ。標準の NetScaler VPX アップグレード手順を使用して、Microsoft Azure クラウド内の NetScaler VPX インスタンスを NetScaler VPX リリース 11.1 以降にアップグレードできます。GUI または CLI の手

順を使用してアップグレードできます。新規インストールの場合は、Microsoft Azure クラウド用の Citrix ADC VPX イメージを使用します。

NetScaler VPX アップグレードビルドをダウンロードするには、「**NetScaler** ダウンロード」**>**「NetScaler ファームウェア ** 」に移動します。

- **Azure** でホストされている **Citrix ADC VPX** インスタンスで観察される **MAC** 移動とインターフェイスミュートを修正するにはどうすればよいですか?

  Azure マルチ NIC 環境では、デフォルトでは、すべてのデータインターフェイスに MAC 移動とインターフェイスのミュートが表示されることがあります。Azure 環境で MAC が移動したりインターフェイスがミュートされたりしないように、NetScaler VPX インスタンスのデータインターフェイス（タグなし）ごとに VLAN を作成し、NIC のプライマリ IP を Azure にバインドすることを Citrix では推奨しています。

  詳細については、CTX224626 の記事を参照してください。

Google Cloud Platform への NetScaler ADC VPX インスタンスのデプロイ

October 17, 2024

NetScaler VPX インスタンスを Google Cloud Platform（GCP）にデプロイできます。GCP の VPX インスタンスを使用すると、GCP クラウドコンピューティング機能を活用し、ビジネスニーズに合わせて Citrix の負荷分散機能とトラフィック管理機能を使用できます。VPX インスタンスを GCP にスタンドアロンインスタンスとしてデプロイできます。シングル NIC 構成とマルチ NIC 構成の両方がサポートされています。

サポートされる機能

Premium、Advanced、Standard のすべての機能は、使用されているライセンス/バージョンタイプに基づいて GCP でサポートされます。

制限事項

- IPv6 はサポートされていません。

ハードウェア要件

GCP の VPX インスタンスには、最低 2 つの vCPU と 4 GB の RAM が必要です。

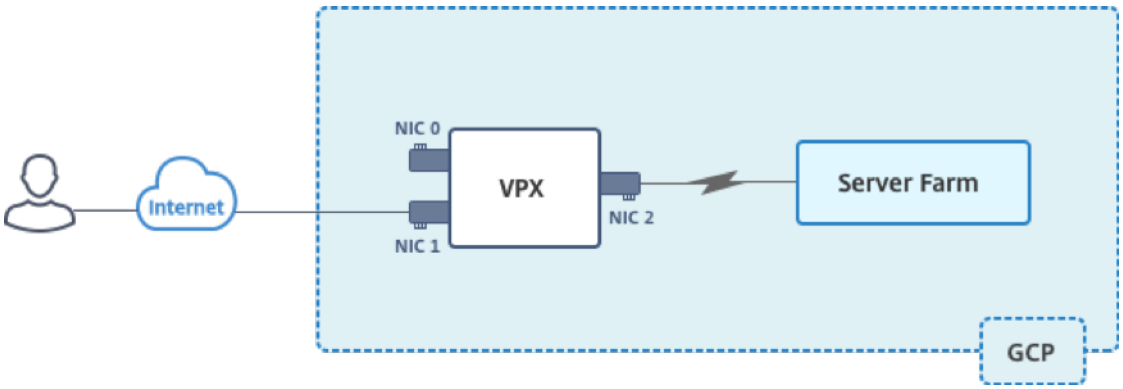
注意事項

デプロイを開始する前に、次の GCP 固有の点を考慮してください。

- インスタンスの作成後は、ネットワークインターフェイスの追加や削除はできません。
- マルチ NIC デプロイの場合は、NIC ごとに個別の VPC ネットワークを作成します。1 つの NIC を関連付けることができるネットワークは 1 つだけです。
- シングル NIC インスタンスの場合、GCP コンソールはデフォルトでネットワークを作成します。
- 2 つ以上のネットワークインターフェースを持つインスタンスには、最低 4 つの vCPU が必要です。
- IP 転送が必要な場合は、インスタンスの作成と NIC の設定中に IP 転送を有効にする必要があります。

シナリオ: マルチ NIC、マルチ IP のスタンドアロン NetScaler VPX インスタンスを展開する

このシナリオでは、NetScaler VPX スタンドアロンインスタンスを GCP にデプロイする方法を示しています。このシナリオでは、多数の NIC を持つスタンドアロン VPX インスタンスを作成します。インスタンスはバックエンドサーバー (サーバーファーム) と通信します。



次の目的に応える NIC を 3 つ作成します。

NIC	目的	VPC ネットワークに関連付けられている
NIC 0	管理トラフィック（NetScaler IP）にサービスを提供する	管理ネットワーク
NIC 1	クライアント側のトラフィック（VIP）をサービスする	クライアントネットワーク
NIC 2	バックエンド・サーバ（SNIP）との通信	バックエンドサーバーネットワーク

次の間の必要な通信ルートを設定します。

- NetScaler VPX インスタンスとバックエンド サーバー。
- NetScaler VPX インスタンスとパブリック インターネット上の外部ホスト。

導入手順の概要

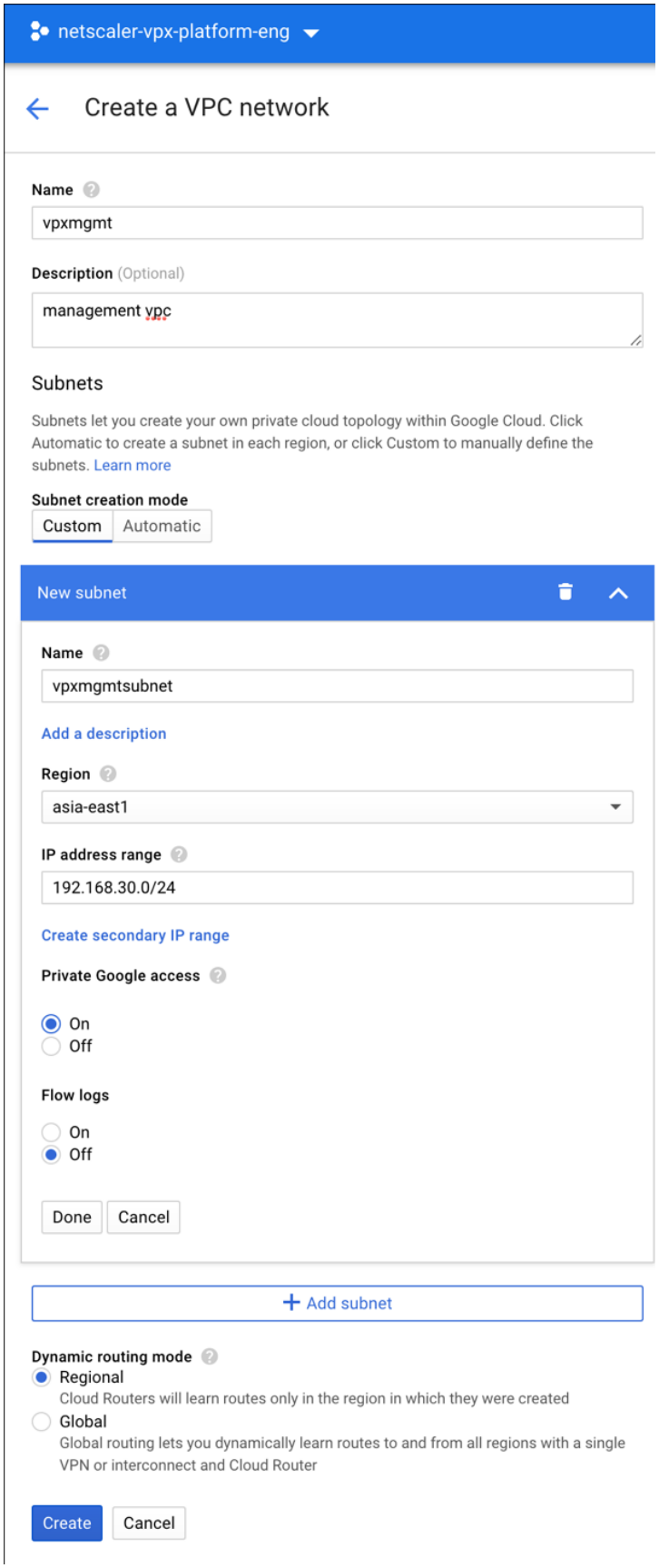
1. 3 つの異なる NIC に対して 3 つの VPC ネットワークを作成します。
2. ポート 22、80、および 443 のファイアウォールルールを作成します。
3. 3 つの NIC を持つインスタンスを作成します。

GCP マーケットプレイスから NetScaler VPX インスタンスを選択します。

> 注
>
> VPC ネットワークを作成したのと同じリージョンにインスタンスを作成します。

手順 **1.** 手順 **1**: **VPC** ネットワークを作成する**.**

管理 NIC、クライアント NIC、およびサーバー NIC に関連付けられた 3 つの VPC ネットワークを作成します。VPC ネットワークを作成するには、**Google** コンソール **>** ネットワーク **> VPC** ネットワーク **> VPC** ネットワークの作成にログオンします。スクリーン・キャプチャに示されている必須フィールドに入力し、「作成」をクリックします。

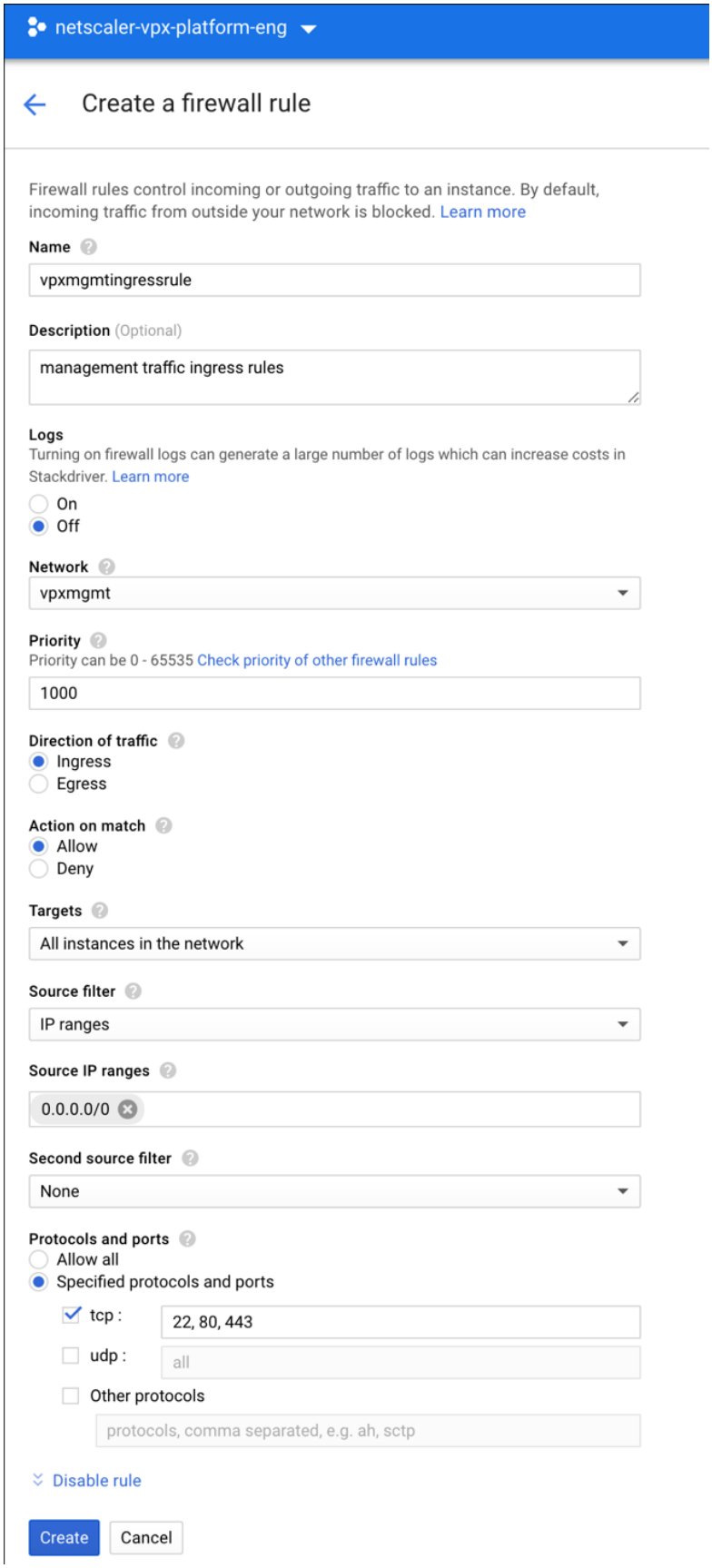


同様に、クライアント側およびサーバー側 NIC 用の VPC ネットワークを作成します。

> 注
>
> 3 つの VPC ネットワークはすべて同じリージョン (このシナリオでは asia-east1) にある必要があります。

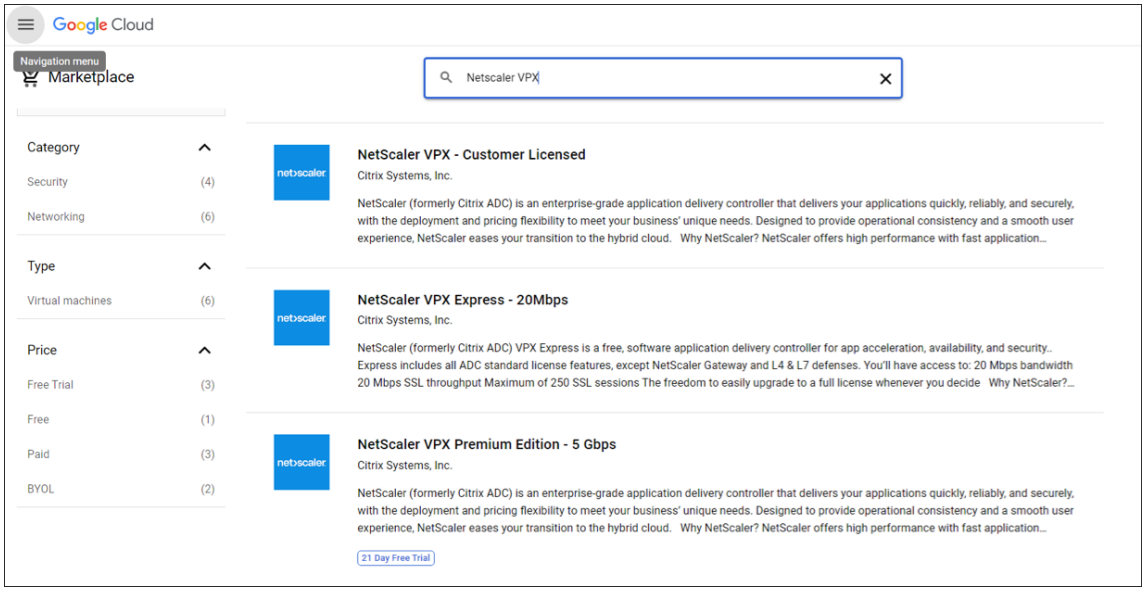
手順 **3.** ポート **22**、**80**、および **443** のファイアウォールルールを作成します。

VPC ネットワークごとに SSH（ポート 22)、HTTP（ポート 80)、HTTPS（ポート 443）のルールを作成します。ファイアウォールルールの詳細については、「ファイアウォールルールの概要」を参照してください。

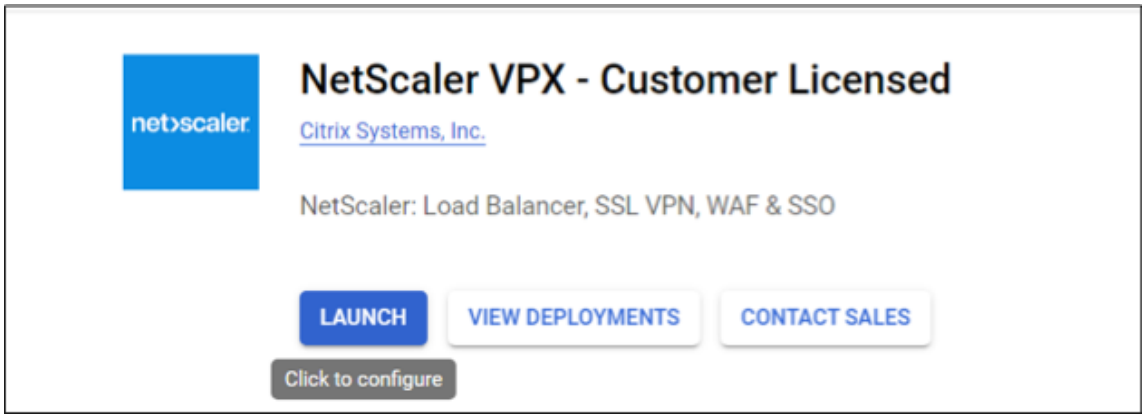


手順 **3. VPX** インスタンスにサービスまたはサービスグループを追加します。

1. GCP コンソールにログインします。
2. GCP マーケットプレイスに移動します。
3. 要件に基づいてサブスクリプションを選択してください。



4. 選択したサブスクリプションで [起動] をクリックします。



5. デプロイフォームに必要事項を記入し、「デプロイ」をクリックします。

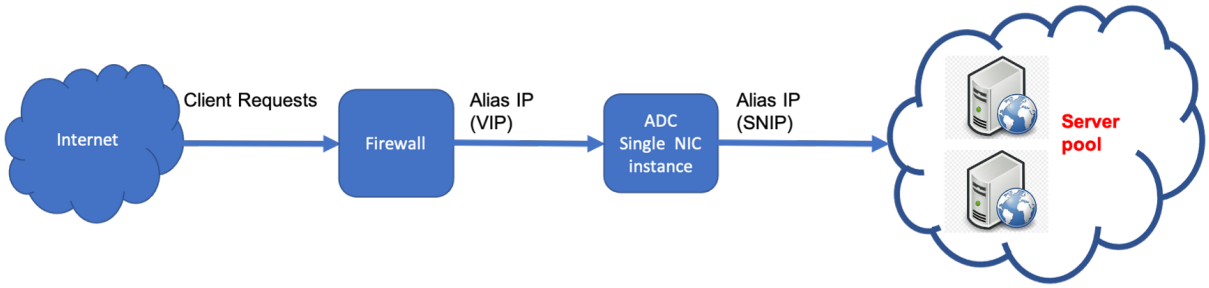
> 注
>
> ステップ **1** で作成した VPC ネットワークを使用します。

6. デプロイされたインスタンスは **Compute Engine > VM** インスタンスの下に表示されます。

   GCP SSH またはシリアルコンソールを使用して VPX インスタンスを構成および管理します。

シナリオ**:** シングル **NIC** のスタンドアロン **VPX** インスタンスをデプロイする

このシナリオでは、NetScaler VPX スタンドアロンインスタンスを単一の NIC で GCP にデプロイする方法を示しています。エイリアス IP アドレスは、この展開を実現するために使用されます。



1 つの NIC (NIC0) を作成して、次の目的を果たします。

- 管理ネットワーク内の管理トラフィック（NetScaler IP）を処理します。
- クライアントネットワーク内のクライアント側トラフィック (VIP) を処理します。
- バックエンドサーバーネットワーク内のバックエンドサーバー (SNIP) と通信します。

次の間の必要な通信ルートを設定します。

- インスタンスとバックエンドサーバー。
- パブリックインターネット上のインスタンスと外部ホスト。

導入手順の概要

1. NIC0 用の VPC ネットワークを作成します。
2. ポート 22、80、および 443 のファイアウォールルールを作成します。
3. 1 つの NIC でインスタンスを作成します。
4. VPX にエイリアス IP アドレスを追加します。
5. VPX に VIP と SNIP を追加します。
6. 負荷分散仮想サーバーを追加します。
7. インスタンスにサービスまたはサービスグループを追加します。
8. サービスまたはサービスグループをインスタンス上の負荷分散仮想サーバーにバインドします。

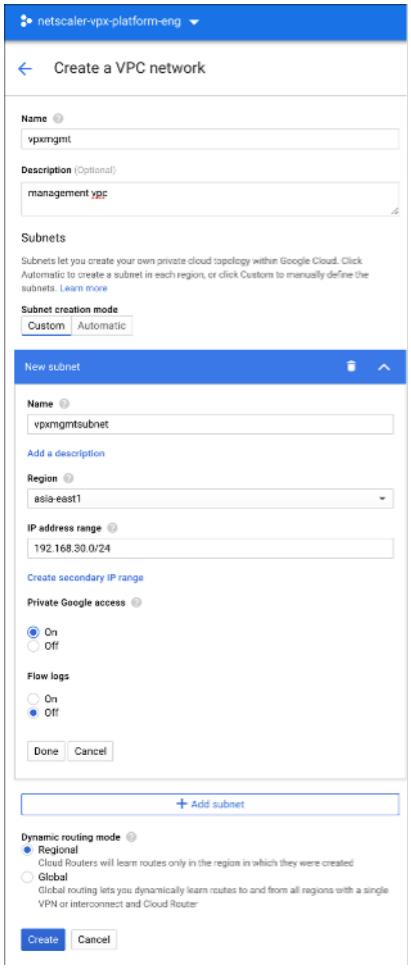
> 注
>
> VPC ネットワークを作成したのと同じリージョンにインスタンスを作成します。

手順 **1.** 手順 **1**: **1** つの **VPC** ネットワークを作成します。

NIC0 に関連付ける VPC ネットワークを 1 つ作成します。

VPC ネットワークを作成するには、次の手順を実行します。

1. **GCP** コンソール **>** ネットワーク **> VPC** ネットワーク **> VPC** ネットワークの作成にログオンします。
2. 必須フィールドに入力し、[**Create**] をクリックします。



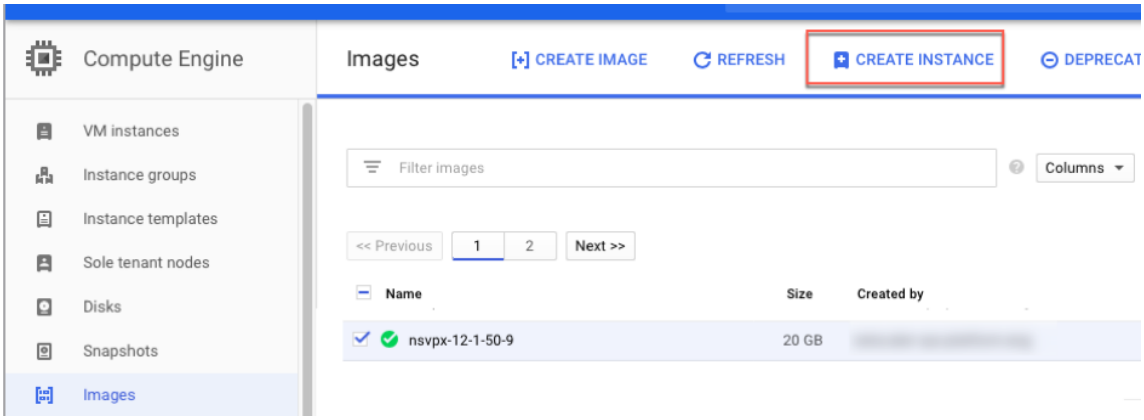
手順 **3.** ポート **22**、**80**、および **443** のファイアウォールルールを作成します。

VPC ネットワークの SSH（ポート 22）、HTTP（ポート 80）、HTTPS（ポート 443）のルールを作成します。ファイアウォールルールの詳細については、「ファイアウォールルールの概要」を参照してください。

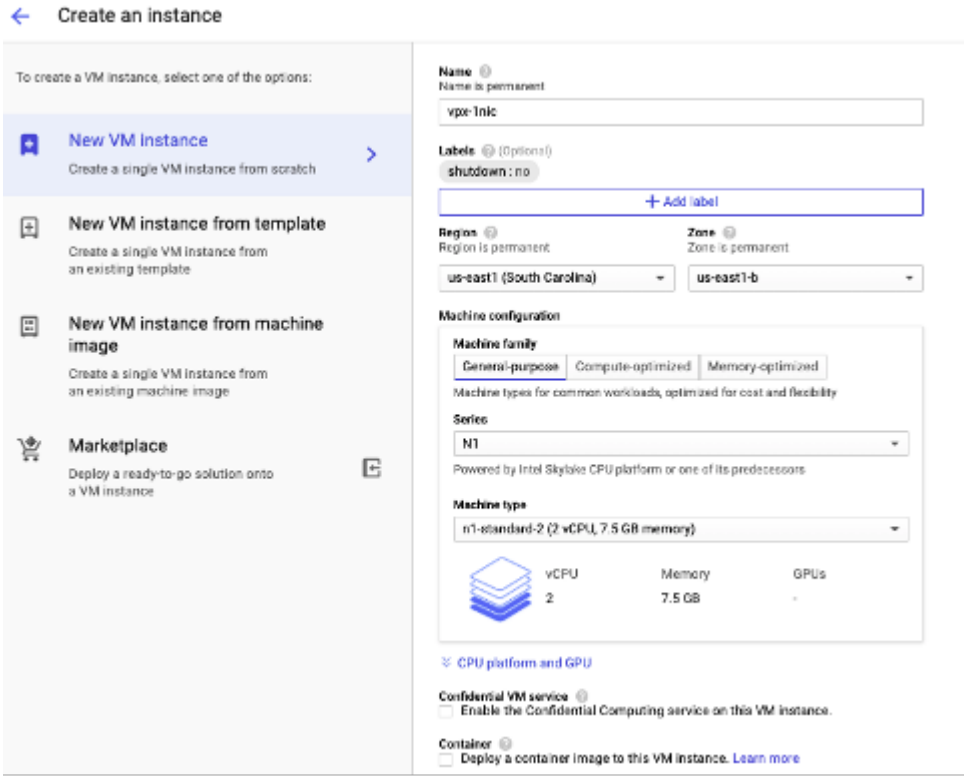
手順 **3.** 手順 **3**: **1** つの **NIC** でインスタンスを作成します。

単一の NIC でインスタンスを作成するには、次の手順を実行します。

1. **GCP** コンソールにログインします。
2. [コンピュート] で **[**** コンピュートエンジン] にカーソルを合わせ、**[**** イメージ] を選択します。
3. イメージを選択し、[インスタンスの作成] をクリックします。



4. 2 つの vCPU を持つインスタンスタイプを選択します（ADC の最小要件）。



5. [管理、セキュリティ、ディスク、**** ネットワーク] ウィンドウから [ネットワーク **] タブをクリックします。
6. [ネットワークインターフェイス] で、[編集] アイコンをクリックして、デフォルトの NIC を編集します。
7. [ネットワークインターフェイス] ウィンドウの [ネットワーク] で、作成した VPC ネットワークを選択します。
8. 静的外部 IP アドレスを作成できます。[外部 **IP** アドレス] で、**[**IP** アドレスの作成 **] をクリックします。
9. [静的アドレスを予約] ウィンドウで、名前と説明を追加し、[予約] をクリックします。
10. [作成] をクリックして VPX インスタンスを作成します。新しいインスタンスが [VM インスタンス] の下に表示されます。新しいインスタンスが VM インスタンスの下に表示されます。

手順 **4**: **VPX** インスタンスに **VIP** と **SNIP** を追加します。

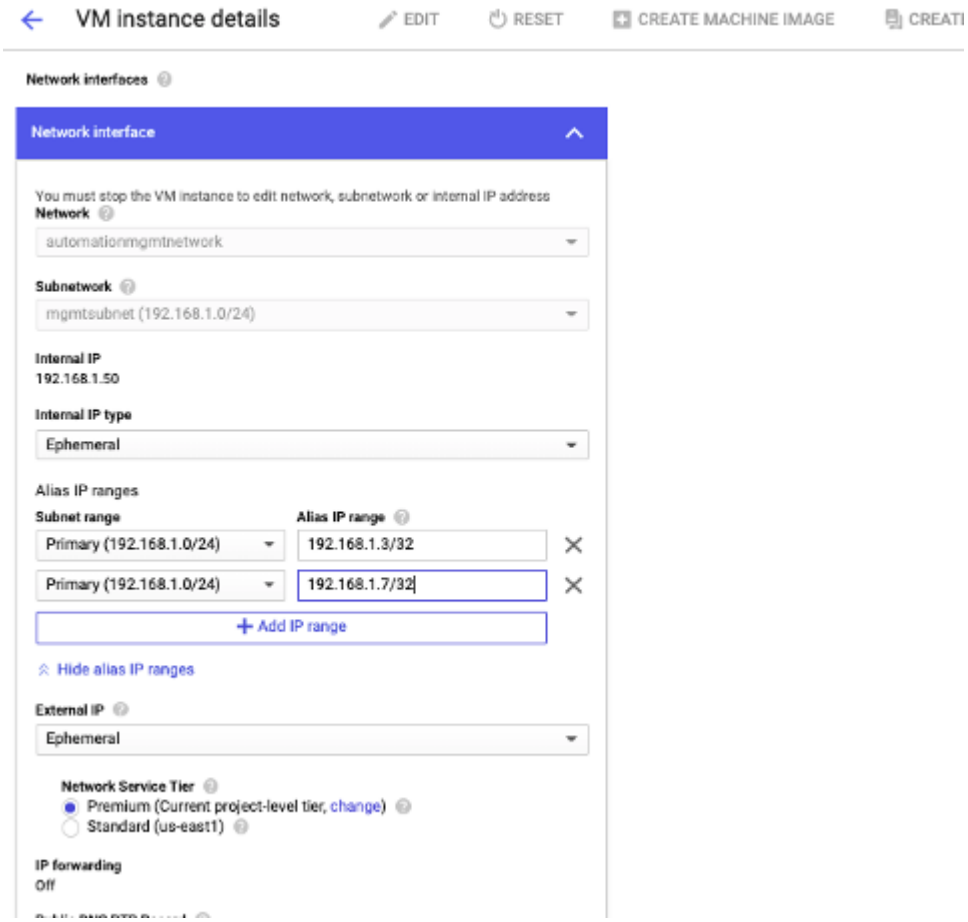
VIP アドレスと SNIP アドレスとして使用する VPX インスタンスに 2 つのエイリアス IP アドレスを割り当てます。

> 注
>
> VPX インスタンスのプライマリ内部 IP アドレスを使用して VIP または SNIP を構成しないでください。

エイリアス IP アドレスを作成するには、次の手順を実行します。

1. VM インスタンスに移動し、[編集] をクリックします。

2. [ネットワークインターフェイス] ウィンドウで、NIC0 インターフェイスを編集します。
3. [エイリアス **IP** 範囲] フィールドに、エイリアス IP アドレスを入力します。

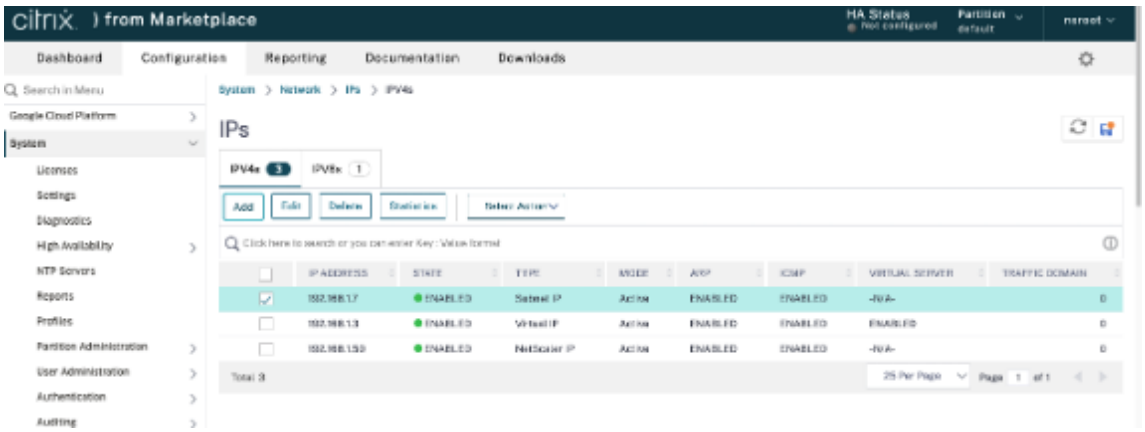


4. [完了]、[保存] の順にクリックします。
5. **VM** インスタンスの詳細ページでエイリアス IP アドレスを確認します。

手順 **5**：**VPX** インスタンスに **VIP** と **SNIP** を追加します。

VPX インスタンスで、クライアントエイリアス IP アドレスとサーバーエイリアス IP アドレスを追加します。

1. NetScaler GUI で、[システム] **>** [ネットワーク] **>** [**IP**] **>** [**IPv4**] に移動し、[追加] をクリックします。



2. クライアントエイリアス IP（VIP）アドレスを作成するには、次の手順を実行します。

   - VM インスタンスで VPC サブネットに設定されたクライアントエイリアス IP アドレスとネットマスクを入力します。
   - [**IP Type**] フィールドで、ドロップダウンメニューから [**Virtual IP**] を選択します。
   - [**Create**] をクリックします。

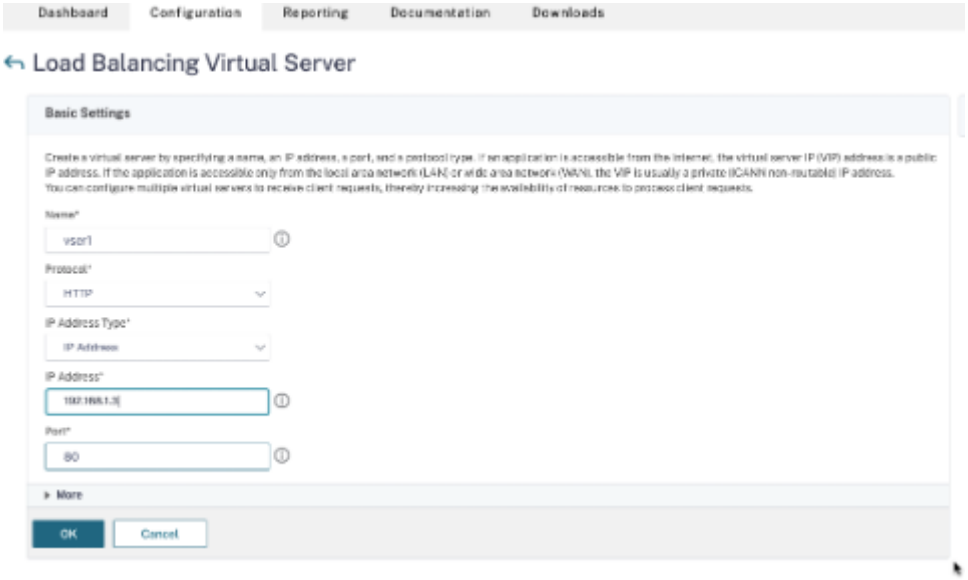
3. サーバーエイリアス IP (SNIP) アドレスを作成するには、次の手順を実行します。

   - VM インスタンスの VPC サブネットに設定されたサーバーエイリアス IP アドレスとネットマスクを入力します。
   - [**IP Type**] フィールドで、ドロップダウンメニューから [**Subnet IP**] を選択します。

- ［**Create**］をクリックします。

手順 **6**：サービス/サービスグループをインスタンス上の負荷分散仮想サーバーにバインドします。

1. NetScaler GUI で、［構成］**>**［トラフィック管理］**>**［負荷分散］**>**［仮想サーバー］に移動し、［追加］をクリックします。
2. [名前]、[プロトコル]、[IP アドレスタイプ (IP アドレス)]、[IP アドレス] (クライアントエイリアス IP)、および [ポート] に必要な値を追加します。
3. **OK** をクリックして、負荷分散仮想サーバーを作成します。



手順 **7**：**VPX** インスタンスにサービスまたはサービス グループを追加します。

1. NetScaler GUI から、［構成］**>**［トラフィック管理］**>**［負荷分散］**>**［サービス］に移動し、［追加］をクリックします。
2. サービス名、IP アドレス、プロトコル、およびポートに必要な値を追加し、「**OK**」をクリックします。

手順 **8**：サービス/サービス グループをインスタンス上の負荷分散仮想サーバーにバインドします。

1. GUI から、[設定**] > [**トラフィック管理**] > [**負荷分散**] > [**仮想サーバ] に移動します。
2. 手順 **6** で構成した負荷分散仮想サーバーを選択し、[編集] をクリックします。
3. [サービスとサービスグループ] ウィンドウで、[負荷分散仮想サーバーサービスのバインドなし] をクリックします。
4. ステップ **7** で設定したサービスを選択し、[バインド（Bind）] をクリックします。

VPX インスタンスを **GCP** にデプロイした後の注意点

- ユーザー名`nsroot`とインスタンス ID をパスワードとして VPX にログオンします。プロンプトで、パスワードを変更し、設定を保存します。

- テクニカルサポートバンドルを収集するには、慣例`show techsupport`ではなくコマンド`shell /netscaler/showtech_cloud.pl`を実行します。
- GCP コンソールから NetScaler ADC VM を削除した後、関連する NetScaler ADC 内部ターゲットインスタンスも削除します。これを行うには、gcloud CLI に移動し、次のコマンドを入力します。

```
1    gcloud compute -q  target-instances delete <instance-name>-
        adcinternal --zone <zone>
```

> 注
>
> <instance-name>-adcinternal は、削除する必要があるターゲットインスタンスの名前です。

NetScaler VPX ライセンス

GCP 上の NetScaler ADC VPX インスタンスにはライセンスが必要です。GCP で実行されている NetScaler ADC VPX インスタンスでは、以下のライセンスオプションを使用できます。

- サブスクリプションベースのライセンス：NetScaler VPX アプライアンスは、GCP マーケットプレイスで有料インスタンスとして利用できます。サブスクリプションベースのライセンスは、従量課金制のオプションです。ユーザーは時間単位で課金されます。GCP マーケットプレイスでは、以下の VPX モデルとライセンスエディションが利用可能です。

サポートされている VPX パフォーマンス
NetScaler VPX アドバンスド-200 Mbps
NetScaler VPX プレミアム - 1 Gbps
NetScaler VPX プレミアム - 5 Gbps
NetScaler VPX エクスプレス - 20 Mbps
NetScaler VPX - 顧客ライセンス
NetScaler VPX FIPS - 顧客ライセンス

- 自分のライセンスを持参 **(BYOL)**: 自分のライセンス (BYOL) を持ち込む場合は、http://support.citrix.com/article/CTX122426にある VPX ライセンスガイドを参照してください。次の操作を実行する必要があります：
  - Citrix Web サイト内のライセンスポータルを使用して、有効なライセンスを生成します。
  - ライセンスをインスタンスにアップロードします。
- **NetScaler VPX** チェックイン/チェックアウトライセンス：詳細については、「NetScalerVPX チェックイン/チェックアウトライセンス」を参照してください。

オンプレミスおよびクラウド展開用の VPX Express では、ライセンスファイルは不要です。NetScaler VPX Express の詳細については、Citrix ADC ライセンスの概要の「NetScaler VPX Express ライセンス」セクションを参照してください。

NetScaler VPX インスタンスを展開するための GDM テンプレート

NetScaler VPX Google デプロイメントマネージャー（GDM）テンプレートを使用して、GCP に VPX インスタンスを展開できます。詳細については、NetScaler GDM テンプレートを参照してください。

NetScaler マーケットプレイスのイメージ

GDM テンプレート内のイメージを使用して、NetScaler ADC アプライアンスを起動できます。

次の表は、GCP マーケットプレイスで利用可能な画像の一覧です。

解除	イメージ名	イメージの場所
14.1	citrix-adc-vpx-express-14-1-21-57	projects/citrix-master-project/global/images/citrix-adc-vpx-express-14-1-21-57
14.1	citrix-adc-vpx-200-Enterprise-14-1-21-57	projects/citrix-master-project/global/images/citrix-adc-vpx-200-enterprise-14-1-21-57
14.1	citrix-adc-vpx-1000-platinum-14-1-21-57	projects/citrix-master-project/global/images/citrix-adc-vpx-1000-platinum-14-1-21-57
14.1	citrix-adc-vpx-5000-platinum-14-1-21-57	projects/citrix-master-project/global/images/citrix-adc-vpx-5000-platinum-14-1-21-57
14.1	citrix-adc-vpx-byol-14-1-21-57	projects/citrix-master-project/global/images/citrix-adc-vpx-byol-14-1-21-57

リソース

- 複数のネットワークインターフェースを持つインスタンスの作成
- VM インスタンスの作成と起動

関連情報

- VPX の高可用性ペアを Google Cloud Platform に展開する

VPX の高可用性ペアを Google Cloud Platform に展開する

October 17, 2024

Google Cloud Platform（GCP）上の 2 つの NetScaler ADC VPX インスタンスを、高可用性（HA）アクティブ/パッシブペアとして構成できます。1 つのインスタンスをプライマリノードとして構成し、もう 1 つをセカンダリノードとして設定すると、プライマリノードは接続を受け入れ、サーバーを管理します。セカンダリノードはプライマリを監視します。何らかの理由で 1 次ノードが接続を受け入れることができない場合、2 次ノードが引き継ぎます。

HA の詳細については、「高可用性」を参照してください。

ノードは同じリージョンにある必要がありますが、同じゾーンまたは異なるゾーンにある可能性があります。詳細については、「リージョンとゾーン」を参照してください。

各 VPX インスタンスには、少なくとも 3 つの IP サブネット（Google VPC ネットワーク）が必要です。

- 管理サブネット
- クライアント側サブネット（VIP）
- バックエンド向けサブネット (SNIP、MIP など)

Citrix では、標準の VPX インスタンスには 3 つのネットワークインターフェイスを推奨しています。

VPX 高可用性ペアは、次の方法でデプロイできます。

- 外部静的 IP アドレスの使用
- プライベート IP アドレスの使用
- プライベート IP アドレスを持つシングル NIC 仮想マシンの使用

VPX 高可用性ペアを GCP にデプロイするための GDM テンプレート

NetScaler Google デプロイメントマネージャー（GDM）テンプレートを使用して、GCP に VPX 高可用性ペアを展開できます。詳細については、NetScaler GDM テンプレートを参照してください。

GCP での VPX 高可用性ペアの転送ルールのサポート

転送ルールを使用して、GCP に VPX 高可用性ペアをデプロイできます。

転送ルールの詳細については、「転送ルールの概要」を参照してください。

前提条件

- 転送ルールは、VPX インスタンスと同じリージョンにある必要があります。
- ターゲットインスタンスは、VPX インスタンスと同じゾーンにある必要があります。
- プライマリノードとセカンダリノードの両方のターゲットインスタンスの数が一致する必要があります。

例

`us-east1` リージョンに高可用性ペアがあり、プライマリ VPX が `us-east1-b` ゾーンにあり、セカンダリ VPX が `us-east1-c` ゾーンにあります。`us-east1-b` ゾーンにターゲットインスタンスがあるプライマリ VPX に対して転送ルールが設定されます。`us-east1-c` ゾーンでセカンダリ VPX のターゲットインスタンスを構成して、フェイルオーバー時に転送ルールを更新します。

制限事項

VPX 高可用性デプロイメントでは、バックエンドでターゲットインスタンスを使用して構成された転送ルールのみがサポートされます。

Google Cloud Platform に外部の静的 **IP** アドレスを指定した **VPX** 高可用性ペアをデプロイする

October 17, 2024

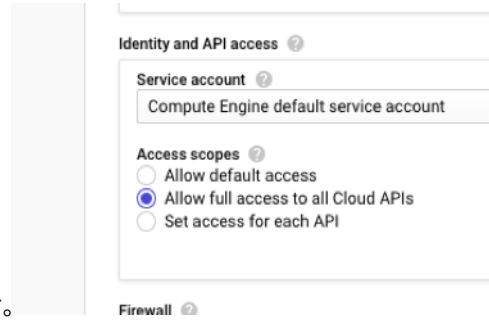
VPX ハイアベイラビリティペアは、外部の静的 IP アドレスを使用して GCP にデプロイできます。プライマリノードのクライアント IP アドレスは、外部の静的 IP アドレスにバインドする必要があります。フェールオーバー時に、外部静的 IP アドレスがセカンダリノードに移動され、トラフィックが再開されます。

静的外部 IP アドレスは、プロジェクトを解放するまでプロジェクト用に予約されている外部 IP アドレスです。IP アドレスを使用してサービスにアクセスする場合、その IP アドレスを予約して、プロジェクトのみが使用できるようにすることができます。詳細については、「静的外部 IP アドレスの予約」を参照してください。

HA の詳細については、「高可用性」を参照してください。

はじめに

- Google Cloud Platform に NetScaler VPX インスタンスをデプロイするに記載されている制限事項、ハードウェア要件、注意事項をお読みください。この情報は、HA 配置にも適用されます。
- GCP プロジェクトで クラウドリソースマネージャー **API** を有効にします。



- インスタンスの作成中に、すべての Cloud API へのフルアクセスを許可します。
- GCP サービスアカウントに関連付けられた IAM ロールに次の IAM 権限があることを確認します。

```
REQUIRED_INSTANCE_IAM_PERMS = [

"compute.addresses.use",
"compute.forwardingRules.list",
"compute.forwardingRules.setTarget",
"compute.instances.setMetadata"
"compute.instances.addAccessConfig",
"compute.instances.deleteAccessConfig",
"compute.instances.get",
"Compute.instances.list",
"compute.networks.useExternalIp",
"compute.subnetworks.useExternalIp",
"compute.targetInstances.list",
"compute.targetInstances.use",
"compute.targetInstances.create",
"compute.zones.list",
"compute.zoneOperations.get",
]
```

- 管理インターフェイス以外のインターフェイスでエイリアス IP アドレスを設定している場合は、GCP サービスアカウントに次の追加の IAM 権限があることを確認してください。

```
"compute.instances.updateNetworkInterface"
```

- プライマリ ノードで GCP 転送ルールを構成している場合は、GCP 上の VPX 高可用性ペアの転送ルールのサポート に記載されている制限と要件を読んで、フェイルオーバー時に新しいプライマリに更新してください。

Google Cloud Platform に VPX HA ペアを展開する方法

HA 展開手順の概要を次に示します。

1. 同じリージョンに VPC ネットワークを作成します。たとえば、アジア東です。
2. 同じリージョンに 2 つの VPX インスタンス（プライマリノードとセカンダリノード）を作成します。それらは、同じゾーンまたは異なるゾーンに存在することができます。たとえば、アジア東-1a、アジア東-Ib。
3. NetScaler GUI または ADC CLI コマンドを使用して、両方のインスタンスで高可用性設定を構成します。

手順 **1.** 手順 **1**: **VPC** ネットワークを作成する

要件に基づいて VPC ネットワークを作成します。管理 NIC、クライアント NIC、サーバー NIC に関連付けるために 3 つの VPC ネットワークを作成することをお勧めします。

VPC ネットワークを作成するには、次の手順を実行します。

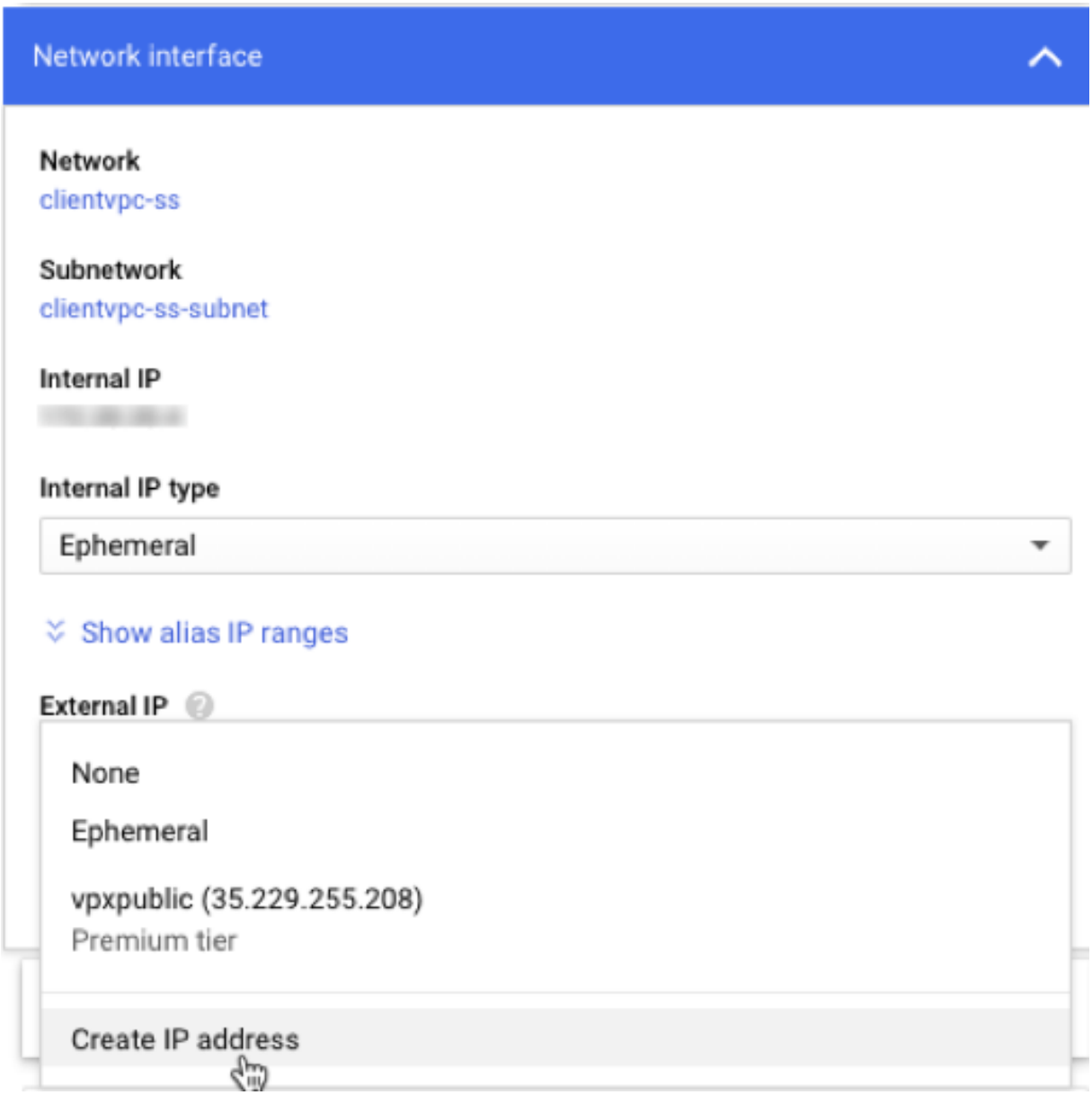
1. **Google** コンソールにログインし、**[**ネットワーク**] > [VPC** ネットワーク**] > [VPC** ネットワークの作成**]** をクリックします。
2. 必須フィールドに入力し、[**Create**] をクリックします。

詳細については、「Google Cloud Platform に NetScaler VPX インスタンスをデプロイする」の「**VPC** ネットワークの作成」セクションを参照してください。

手順 **3.** 手順 **2**: **2** つの **VPX** インスタンスを作成する

シナリオ: マルチ NIC、マルチ IP のスタンドアロン VPX インスタンスをデプロイするに記載されている手順に従って、2 つの VPX インスタンスを作成します。

> 重要**:**
>
> プライマリノードのクライアント IP アドレス (VIP) に静的外部 IP アドレスを割り当てます。既存の予約済み IP アドレスを使用するか、新しい予約済み IP アドレスを作成できます。静的外部 IP アドレスを作成するには、[ネットワークインターフェイス**] > [**外部 **IP**] に移動し、[**IP** アドレスの作成] をクリックします。



フェールオーバー後、古いプライマリが新しいセカンダリになると、スタティック外部 IP アドレスは古いプライマリから移動し、新しいプライマリに接続されます。詳細については、Google Cloud ドキュメント「静的外部 IP アドレスを予約する」を参照してください。

VPX インスタンスを構成したら、VIP アドレスと SNIP アドレスを構成できます。詳細については、「Citrix ADC 所有の IP アドレスの構成」を参照してください。

手順 **3.** 高可用性の構成

Google Cloud Platform でインスタンスを作成した後、CLI 用 Citrix ADC GUI を使用して HA を構成できます。

GUI を使用した **HA** の設定　ステップ **1.** 両方のインスタンスで INC モードで高可用性をセットアップします。

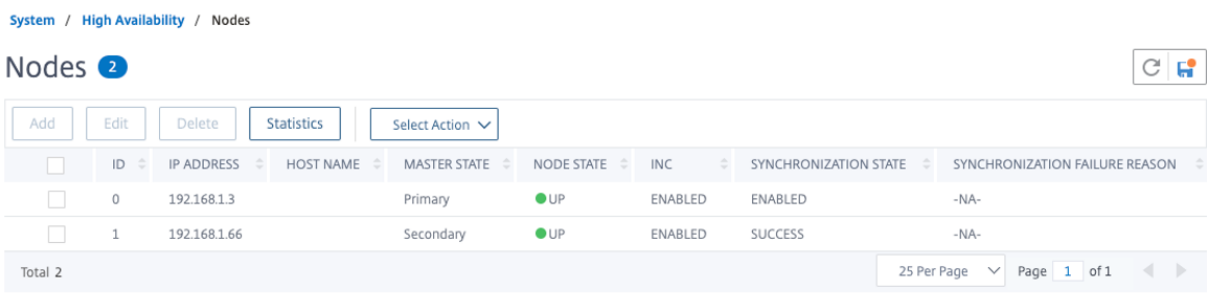
プライマリノードで、次の手順を実行します。

1. GCP Console `nsroot` からノードのユーザー名とインスタンス ID をパスワードとしてインスタンスにログオンします。
2. 構成 **>** システム **>** 高可用性 **>** ノードに移動し、追加をクリックします。
3. [リモートノードの **IP** アドレス] フィールドに、セカンダリノードの管理 NIC のプライベート IP アドレスを入力します。
4. [セルフノードで **INC (**独立ネットワーク構成**)** モードをオンにする] チェックボックスをオンにします。
5. [**Create**] をクリックします。

先に進む前に、[**Nodes**] ページにセカンダリノードの同期状態が **SUCCESS** と表示されていることを確認してください。

1. GCP Console `nsroot` からノードのユーザー名とインスタンス ID をパスワードとしてインスタンスにログオンします。
2. 構成 **>** システム **>** 高可用性 **>** ノードに移動し、追加をクリックします。
3. [リモートノード **IP** アドレス] フィールドに、プライマリノードの管理 NIC のプライベート IP アドレスを入力します。
4. [セルフノードで **INC (**独立ネットワーク構成**)** モードをオンにする] チェックボックスをオンにします。
5. [**Create**] をクリックします。

セカンダリノードで、次の手順を実行します。



> 注
>
> これで、セカンダリノードは、プライマリノードと同じログオン資格情報を持ちます。

ステップ **2.** 両方のノードに仮想 IP アドレスとサブネット IP アドレスを追加します。

プライマリノードで、次の手順を実行します。

1. **System > Network > IP Sets > Add** に移動します。
2. 次の手順に従って、プライマリ VIP アドレスを追加します。

    a) セカンダリインスタンスのクライアント側インターフェイスの内部 IP アドレスと、VM インスタンスのクライアントサブネットに設定されたネットマスクを入力します。

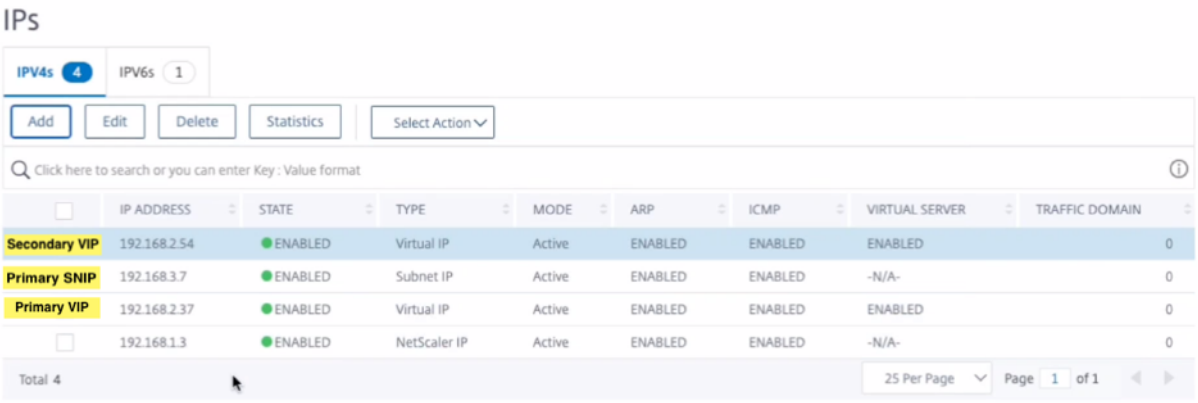
b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Virtual IP**] を選択します。
c) [**Create**] をクリックします。

3. 次の手順に従って、プライマリ SNIP アドレスを追加します。

a) セカンダリインスタンスのサーバ側インターフェイスの内部 IP アドレスと、セカンダリインスタンスのサーバサブネットに設定されたネットマスクを入力します。
b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Subnet IP**] を選択します。
c) [**Create**] をクリックします。

4. 次の手順に従って、セカンダリ VIP アドレスを追加します。

a) プライマリ・インスタンスのクライアント側インターフェイスの内部 IP アドレスと、VM インスタンスのクライアント・サブネットに対して構成されたネットマスクを入力します。
b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Virtual IP**] を選択します。
c) [**Create**] をクリックします。



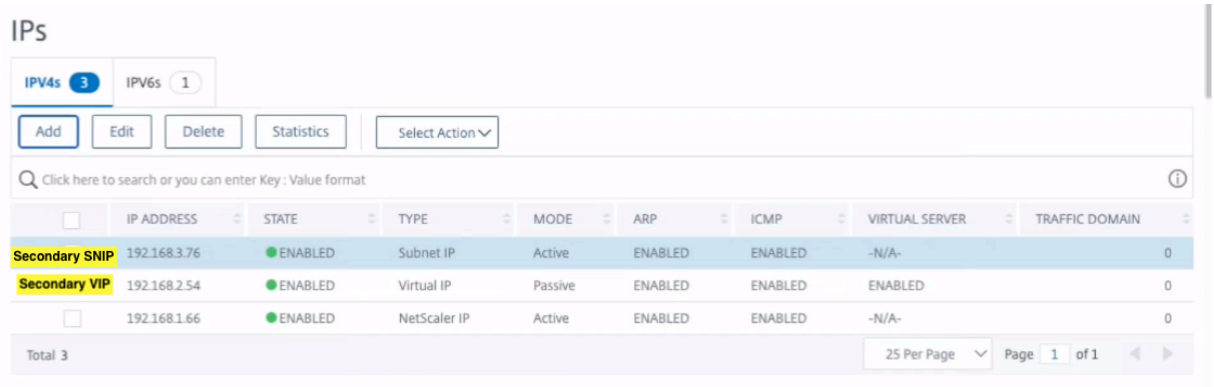
先に進む前に、[**Nodes**] ページにセカンダリノードの同期状態が **SUCCESS** と表示されていることを確認してください。

1. **System > Network > IP Sets > Add** に移動します。
2. 次の手順に従って、セカンダリ VIP アドレスを追加します。

a) プライマリ・インスタンスのクライアント側インターフェイスの内部 IP アドレスと、VM インスタンスのクライアント・サブネットに対して構成されたネットマスクを入力します。
b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Virtual IP**] を選択します。

3. 次の手順に従って、セカンダリ SNIP アドレスを追加します。

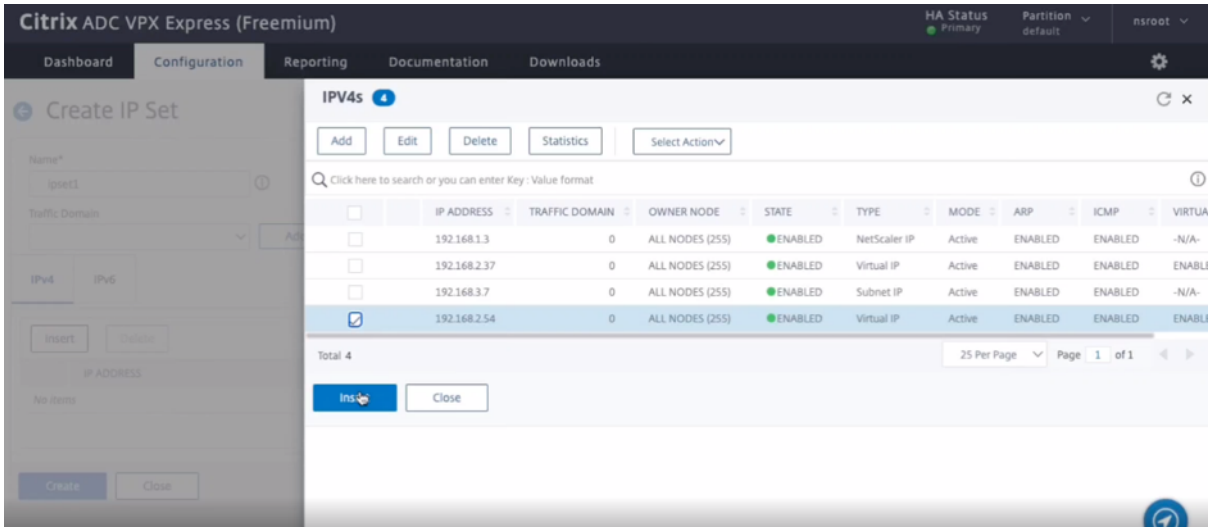
a) プライマリ・インスタンスのサーバ側インターフェイスの内部 IP アドレスと、プライマリ・インスタンスのサーバ・サブネットに対して構成されたネットマスクを入力します。
b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Subnet IP**] を選択します。
c) [**Create**] をクリックします。



ステップ **3:** プライマリ・インスタンスに仮想サーバを追加します。IP セットを追加し、両方のインスタンスで IP セットをセカンダリ VIP にバインドします。

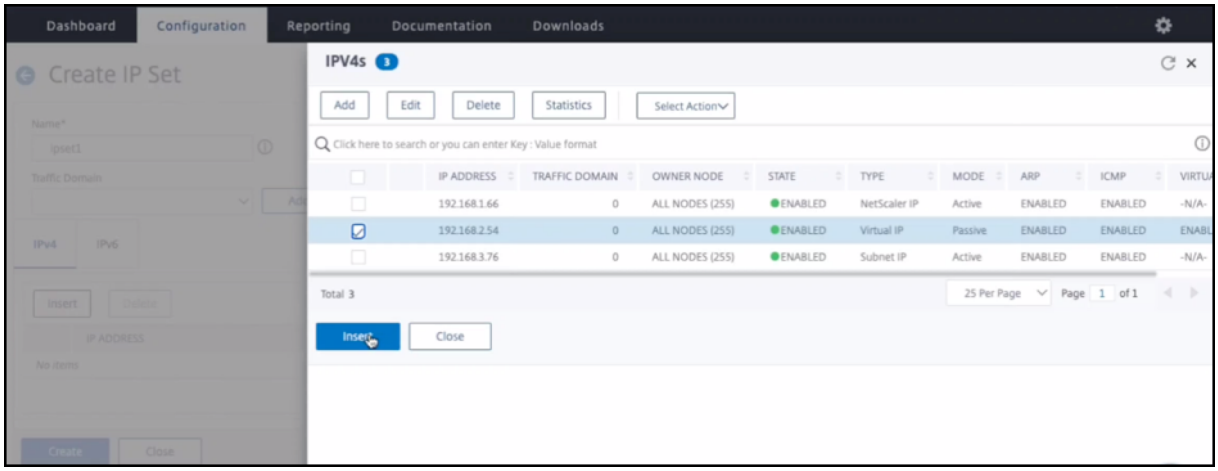
プライマリノードで、次の手順を実行します。

1. ステップ **2:** 両方のインスタンスに IP セットを追加します。
2. IP セット名を追加し、[**Insert**] をクリックします。
3. [**IPv4**] ページで、仮想 IP（セカンダリ VIP）を選択し、[挿入] をクリックします。
4. [**Create**] をクリックして IP セットを作成します。



先に進む前に、[**Nodes**] ページにセカンダリノードの同期状態が **SUCCESS** と表示されていることを確認してください。

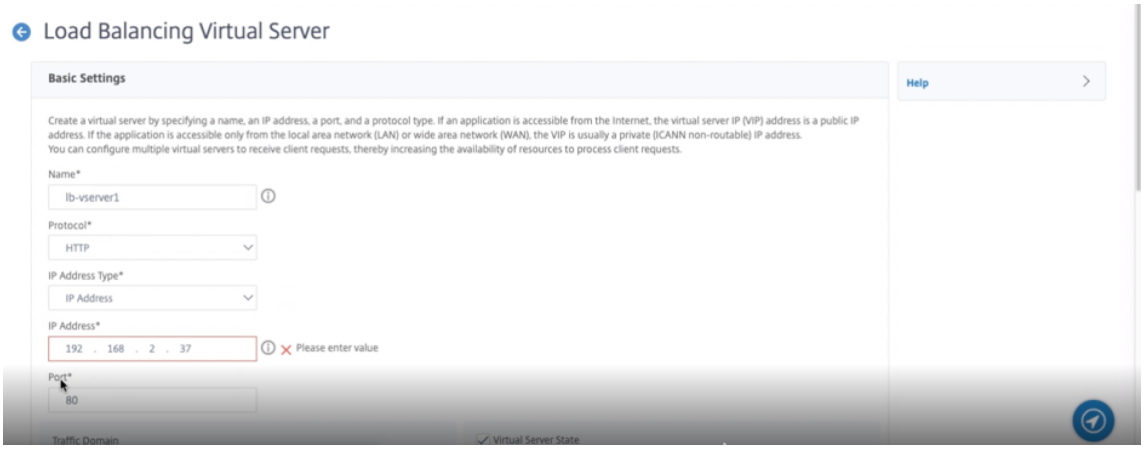
1. ステップ **2:** 両方のインスタンスに IP セットを追加します。
2. IP セット名を追加し、[**Insert**] をクリックします。
3. [**IPv4**] ページで、仮想 IP（セカンダリ VIP）を選択し、[挿入] をクリックします。
4. [**Create**] をクリックして IP セットを作成します。



> 注
>
> IP セット名は、両方のインスタンスで同じである必要があります。

ステップ **4:** プライマリ・インスタンスに仮想サーバを追加します。プライマリインスタンスにサービスまたはサービスグループを追加します。

1. [設定] **>** [トラフィック管理] **>** [負荷分散] **>** [仮想サーバー] **>** [追加] に移動します。
2. [名前]、[プロトコル]、[IP アドレスタイプ (IP アドレス)]、[IP アドレス] (プライマリ VIP)、および [ポート] に必要な値を追加します。



3. [詳細] クリックします。[**IP** 範囲 **IP** セット設定] に移動し、ドロップダウンメニューから [**IPSet**] を選択し、ステップ **3** で作成した IPSet を指定します。
4. **OK** をクリックして、負荷分散仮想サーバーを作成します。

ステップ **5**. プライマリノードにサービスまたはサービスグループを追加します。

1. **[**設定**] > [**トラフィック管理**] > [**負荷分散**] > [**サービス**] >[**追加**]** に移動します。
2. サービス名、IP アドレス、プロトコル、およびポートに必要な値を追加し、[**OK**] をクリックします。

ステップ **6**. サービスまたはサービスグループをプライマリノードの負荷分散仮想サーバーにバインドします。

1. **[設定] > [**トラフィック管理**] > [**負荷分散**] > [**仮想サーバー**]** に移動します。
2. 手順 **4** で構成した負荷分散仮想サーバーを選択し、[編集] をクリックします。
3. [サービスとサービスグループ] タブで、[負荷分散仮想サーバーサービスバインドなし] をクリックします。
4. 手順 **5** で構成したサービスを選択し、[バインド] をクリックします。

構成を保存します。強制フェールオーバーの後、セカンダリは新しいプライマリになります。古いプライマリ VIP の外部スタティック IP は、新しいセカンダリ VIP に移動します。

CLI を使用した高可用性の設定　ステップ **1.** 両方のインスタンスで INC モードで高可用性をセットアップします。

セカンダリノードで、次のコマンドを入力します。

```
1    add ha node 1 <sec_ip> -inc ENABLED
```

プライマリノードで、次のコマンドを入力します。

```
1    add ha node 1 <prim_ip> -inc ENABLED
```

prim_ipは、プライマリノードの管理 NIC の内部 IP アドレスを指します。

sec_ipは、セカンダリノードの管理 NIC の内部 IP アドレスを指します。

ステップ **2.** 両方のノードに仮想 IP とサブネット IP を追加します。

セカンダリノードで、次のコマンドを入力します。

```
1    add ns ip <primary_vip> <subnet> -type VIP
2
3    add ns ip <secondary_vip> <subnet> -type VIP
4
5    add ns ip <primary_snip> <subnet> -type SNIP
```

secondary_vipは、セカンダリインスタンスのクライアント側インターフェイスの内部 IP アドレスを指します。

primary_snipは、プライマリ・インスタンスのサーバ側インターフェイスの内部 IP アドレスを指します。

primary_vipは、プライマリ・インスタンスのクライアント側インターフェイスの内部 IP アドレスを指します。

プライマリノードで、次のコマンドを入力します。

```
1    add ns ip <secondary_vip> <subnet> -type VIP
2
3    add ns ip <secondary_snip> <subnet> -type SNIP
```

primary_snipは、プライマリ・インスタンスのサーバ側インターフェイスの内部 IP アドレスを指します。

secondary_vipは、セカンダリインスタンスのクライアント側インターフェイスの内部 IP アドレスを指します。

ステップ **3:** プライマリ・インスタンスに仮想サーバを追加します。IP セットを追加し、両方のインスタンスで IP セットをセカンダリ VIP にバインドします。

セカンダリノードで、次のコマンドを入力します。

```
1    add ipset <ipsetname>
2    bind ipset <ipsetname> <secondary VIP>
```

セカンダリノードで、次のコマンドを入力します。

```
1    add ipset <ipsetname>
2    bind ipset <ipsetname> <secondary VIP>
```

> 注
>
> IP セット名は、両方のインスタンスで同じである必要があります。

ステップ **4:** プライマリ・インスタンスに仮想サーバを追加します。プライマリ・インスタンスに仮想サーバを追加します。

次のコマンドを入力します:

```
1    add <server_type> vserver <vserver_name> <protocol> <primary_vip> <
        port> -ipset <ipset_name>
```

ステップ **5**. プライマリインスタンスにサービスまたはサービスグループを追加します。

次のコマンドを入力します:

```
1    add service <service_name> <service_ip_address> <protocol> <port>
```

ステップ **6**. サービス/サービスグループをプライマリインスタンス上の負荷分散仮想サーバーにバインドします。

次のコマンドを入力します:

```
1    bind <server_type> vserver <vserver_name> <service_name>
```

> 注
>
> 設定を保存するには、コマンド`save config`を入力します。そうしないと、インスタンスの再起動後に設定が失われます。

ステップ **7**. 設定を確認します。

プライマリクライアント NIC に接続されている外部 IP アドレスが、フェールオーバー時にセカンダリに移動することを確認します。

1. 外部 IP アドレスに cURL 要求を行い、それが到達可能であることを確認します。
2. プライマリインスタンスで、フェイルオーバーを実行します:

   GUI から、[設定] **>** [システム] **>** [高可用性] **>** [アクション] **>** [強制フェールオーバー] に移動します。

CLI から、次のコマンドを入力します。

```
force ha failover -f
```

GCP コンソールで、セカンダリインスタンスに移動します。外部 IP アドレスは、フェールオーバー後にセカンダリのクライアント NIC に移動されている必要があります。

3. 外部 IP に cURL 要求を発行し、再び到達可能であることを確認します。

Google Cloud Platform にプライベート IP アドレスを指定した 1 つの NIC VPX 高可用性ペアをデプロイします

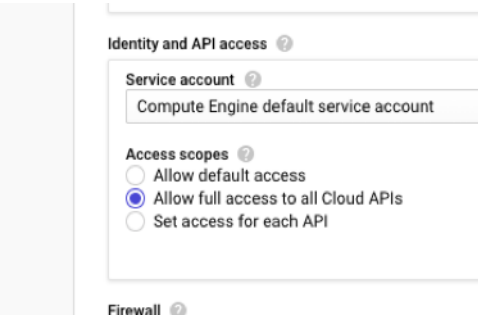
October 17, 2024

プライベート IP アドレスを使用して、単一の NIC VPX 高可用性ペアを GCP にデプロイできます。クライアント IP (VIP) アドレスは、プライマリノードのエイリアス IP アドレスとして設定する必要があります。フェールオーバー時に、クライアント IP アドレスがセカンダリノードに移動され、トラフィックが再開されます。各ノードのサブネット IP (SNiP) アドレスもエイリアス IP 範囲として設定する必要があります。

高可用性の詳細については、「高可用性」を参照してください。

はじめに

- Google Cloud Platform に NetScaler VPX インスタンスをデプロイするに記載されている制限事項、ハードウェア要件、注意事項をお読みください。この情報は、高可用性展開にも適用されます。
- GCP プロジェクトで クラウドリソースマネージャー **API** を有効にします。
- インスタンスの作成中に、すべての Cloud API へのフルアクセスを許可します。



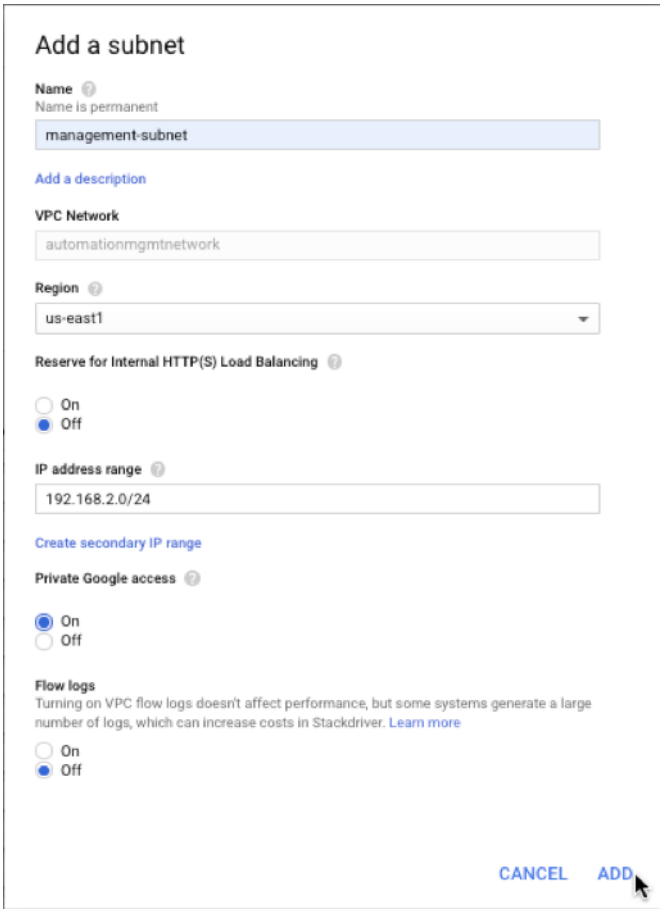
- GCP サービスアカウントに次の IAM 権限があることを確認します。

```
REQUIRED_INSTANCE_IAM_PERMS = [
"compute.forwardingRules.list",
"compute.forwardingRules.setTarget",
"compute.instances.setMetadata",
"compute.instances.get",
```



```
    "compute.instances.list",
    "compute.instances.updateNetworkInterface",
    "compute.targetInstances.list",
    "compute.targetInstances.use",
    "compute.targetInstances.create",
    "compute.zones.list",
    "compute.zoneOperations.get",
    ]
```

- VM がインターネットにアクセスできない場合は、VPC サブネットでプライベート **Google** アクセスを有効にする必要があります。



- プライマリ ノードで GCP 転送ルールを構成している場合は、GCP 上の VPX 高可用性ペアの転送ルールのサポート に記載されている制限と要件を読んで、フェイルオーバー時に新しいプライマリに更新してください。

VPX 高可用性ペアを **Google Cloud Platform** にデプロイする方法

NIC が 1 つの HA ペアを導入する手順の概要は次のとおりです。

1. 手順 1: 1 つの VPC ネットワークを作成します。
2. 同じリージョンに 2 つの VPX インスタンス（プライマリノードとセカンダリノード）を作成します。それらは、同じゾーンまたは異なるゾーンに存在することができます。たとえば、アジア東-1a、アジア東-Ib。
3. NetScaler GUI または ADC CLI コマンドを使用して、両方のインスタンスで高可用性設定を構成します。

手順 **1. VPC** ネットワークを **1** つ作成

VPC ネットワークを作成するには、次の手順を実行します。

1. **Google** コンソール **> [**ネットワーク**] > [VPC** ネットワーク**] > [VPC** ネットワークの作成**]** にログインします。
2. 必須フィールドに入力し、[**Create**] をクリックします。

詳細については、「Google Cloud Platform に NetScaler VPX インスタンスをデプロイする」の「**VPC** ネットワークの作成」セクションを参照してください。

手順 **3.** 手順 **2**: **2** つの **VPX** インスタンスを作成する

シナリオ: 単一 NIC のスタンドアロン VPX インスタンスをデプロイするの手順 1 から手順 3 に従って、2 つの VPX インスタンスを作成します。

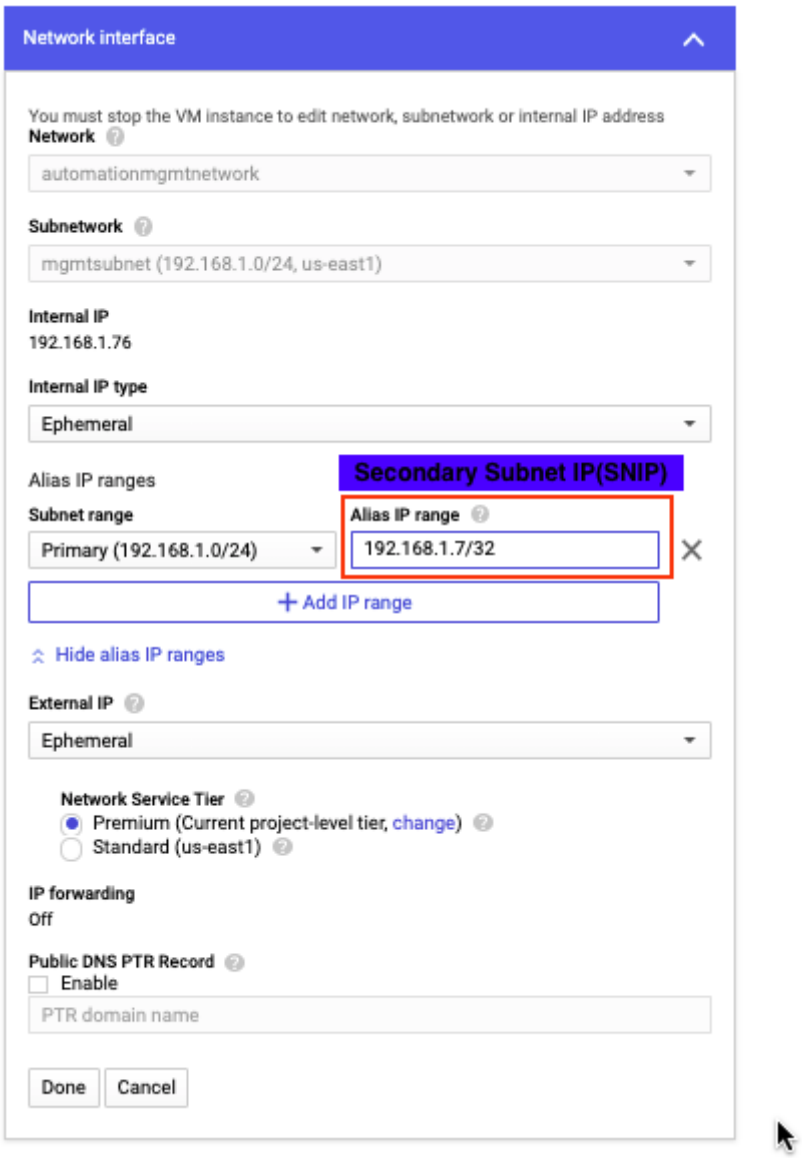
> 重要**:**
>
> クライアントエイリアス IP アドレスをプライマリノードにのみ割り当て、サーバエイリアス IP アドレスをプライマリノードとセカンダリノードに割り当てます。VPX インスタンスの内部 IP アドレスを使用して VIP または SNIP を構成しないでください。

クライアントとサーバーのエイリアス IP アドレスを作成するには、プライマリノードで次の手順を実行します。

1. VM インスタンスに移動し、[編集] をクリックします。
2. ネットワークインターフェースウィンドウで、クライアント (NIC0) インターフェースを編集します。
3. [エイリアス **IP** 範囲（Alias IP range）] フィールドに、クライアントエイリアス IP アドレスを入力します。
4. 「**IP** 範囲を追加」をクリックし、サーバーのエイリアス IP アドレスを入力します。

サーバエイリアス IP アドレスを作成するには、セカンダリノードで次の手順を実行します。

1. VM インスタンスに移動し、[編集] をクリックします。
2. ネットワークインターフェースウィンドウで、クライアント (NIC0) インターフェースを編集します。
3.「エイリアス **IP** 範囲」フィールドに、サーバーのエイリアス IP アドレスを入力します。



フェイルオーバー後、古いプライマリが新しいセカンダリになると、クライアントのエイリアス IP アドレスが古いプライマリから移動され、新しいプライマリに接続されます。

VPX インスタンスを構成したら、仮想 (VIP) アドレスとサブネット IP (SNIP) アドレスを構成できます。詳細については、「Citrix ADC 所有の IP アドレスの構成」を参照してください。

手順 **3.** 高可用性の構成

Google Cloud Platform でインスタンスを作成した後、NetScaler GUI または CLI を使用して高可用性を構成できます。

GUI を使用した高可用性の構成

ステップ **1.** 両方のノードで INC Enabled モードで高可用性を設定します。

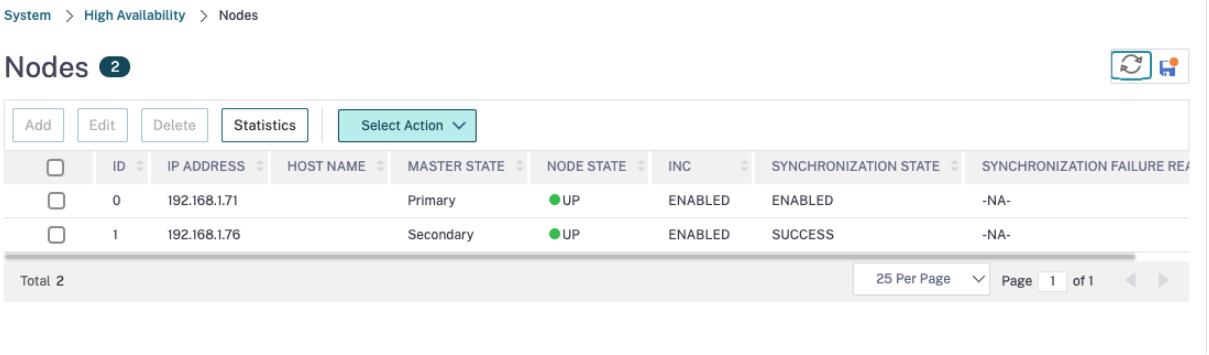
プライマリノードで、次の手順を実行します。

1. GCP Console `nsroot` からノードのユーザー名とインスタンス ID をパスワードとしてインスタンスにログオンします。
2. 構成 **>** システム **>** 高可用性 **>** ノードに移動し、追加をクリックします。
3. [リモートノードの **IP** アドレス] フィールドに、セカンダリノードの管理 NIC のプライベート IP アドレスを入力します。
4. [セルフノードで **INC (**独立ネットワーク構成**)** モードをオンにする] チェックボックスをオンにします。
5. [**Create**] をクリックします。

先に進む前に、[**Nodes**] ページにセカンダリノードの同期状態が **SUCCESS** と表示されていることを確認してください。

1. GCP Console `nsroot` からノードのユーザー名とインスタンス ID をパスワードとしてインスタンスにログオンします。
2. 構成 **>** システム **>** 高可用性 **>** ノードに移動し、追加をクリックします。
3. [リモートノード **IP** アドレス] フィールドに、プライマリノードの管理 NIC のプライベート IP アドレスを入力します。
4. [セルフノードで **INC (**独立ネットワーク構成**)** モードをオンにする] チェックボックスをオンにします。
5. [**Create**] をクリックします。

セカンダリノードで、次の手順を実行します。



> 注
>
> セカンダリノードがプライマリノードと同期されると、セカンダリノードにはプライマリノードと同じログオン認証情報が割り当てられます。

ステップ **2.** 両方のノードに仮想 IP アドレスとサブネット IP アドレスを追加します。

プライマリノードで、次の手順を実行します。

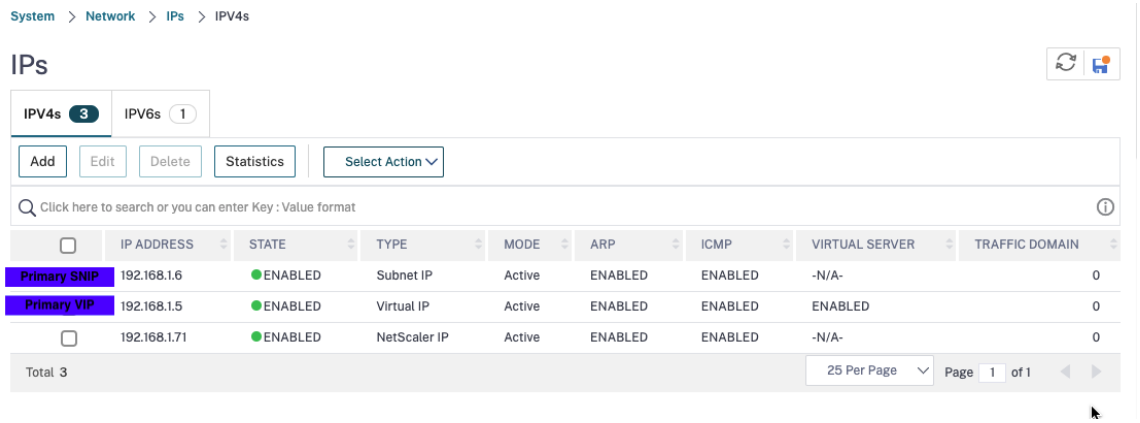
1. **System > Network > IP Sets > Add** に移動します。

2. クライアントエイリアス IP（VIP）アドレスを作成するには、次の手順を実行します。

   a) プライマリ VM インスタンスの VPC サブネットに設定されているクライアントエイリアス IP アドレスとネットマスクを入力します。
   b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Virtual IP**] を選択します。
   c) [**Create**] をクリックします。

3. サーバーエイリアス IP (SNIP) アドレスを作成するには、次の手順を実行します。

   a) プライマリ VM インスタンスの VPC サブネットに設定されているサーバーエイリアス IP アドレスとネットマスクを入力します。
   b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Subnet IP**] を選択します。
   c) [**Create**] をクリックします。



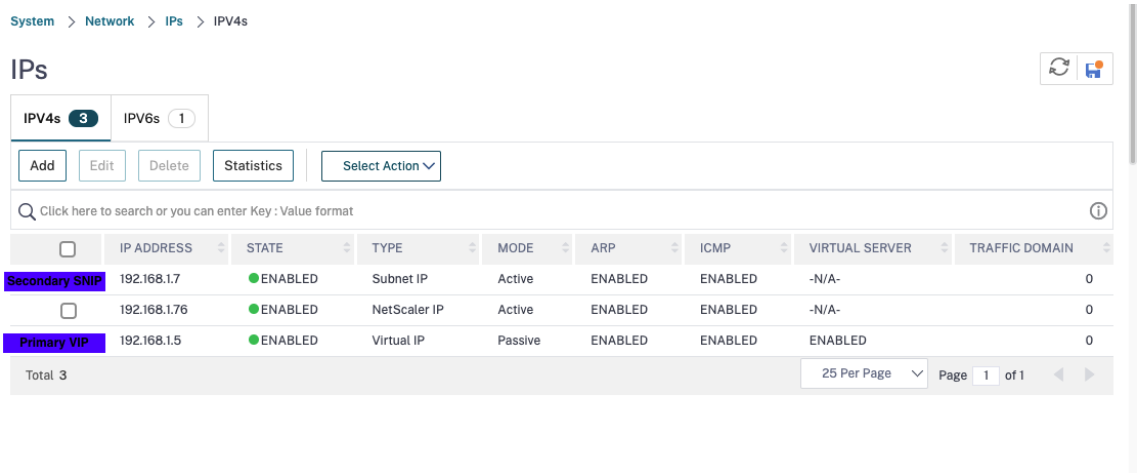
セカンダリノードで、次の手順を実行します。

1. **System > Network > IP Sets > Add** に移動します。
2. クライアントエイリアス IP（VIP）アドレスを作成するには、次の手順を実行します。

   a) プライマリ VM インスタンスの VPC サブネットに設定されているクライアントエイリアス IP アドレスとネットマスクを入力します。
   b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Virtual IP**] を選択します。
   c) [**Create**] をクリックします。

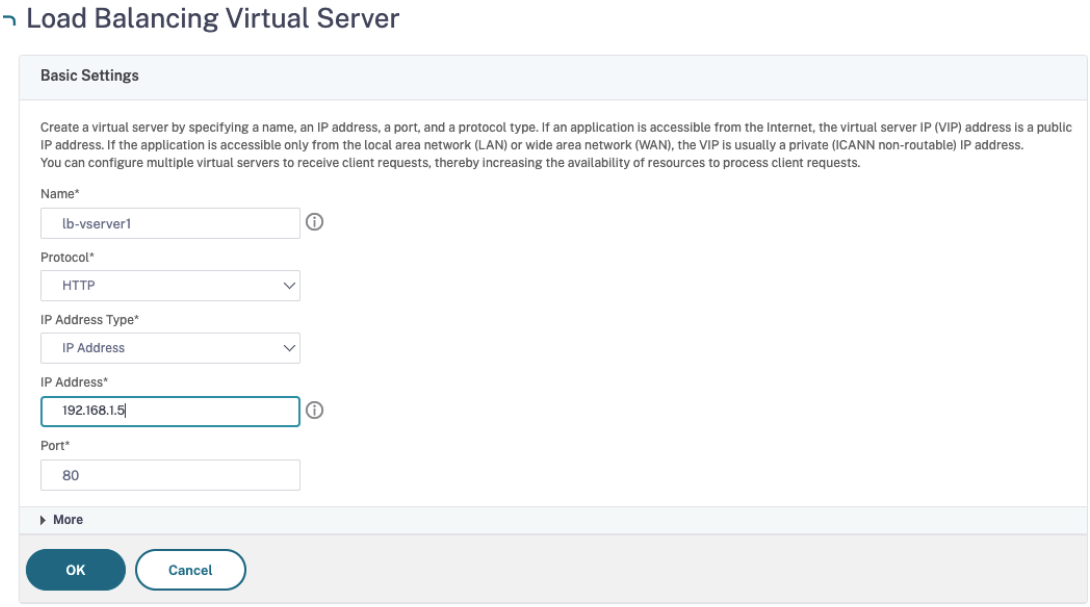
3. サーバーエイリアス IP (SNIP) アドレスを作成するには、次の手順を実行します。

   a) セカンダリ VM インスタンスの VPC サブネットに設定されているサーバエイリアス IP アドレスとネットマスクを入力します。
   b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Subnet IP**] を選択します。
   c) [**Create**] をクリックします。



ステップ **3:** プライマリ・インスタンスに仮想サーバを追加します。ステップ **3:** プライマリノードに負荷分散仮想サーバーを追加します。

1. [設定] **>** [トラフィック管理] **>** [負荷分散] **>** [仮想サーバー] **>** [追加] に移動します。
2. 名前、プロトコル、IP アドレスタイプ（IP アドレス）、IP アドレス（プライマリクライアントエイリアス IP アドレス）、およびポートに必要な値を追加し、「**OK**」をクリックします。



ステップ **4:** プライマリ・インスタンスに仮想サーバを追加します。プライマリノードにサービスまたはサービスグループを追加します。

1. **[**設定**] > [**トラフィック管理**] > [**負荷分散**] > [**サービス**] >[**追加**]** に移動します。
2. サービス名、IP アドレス、プロトコル、およびポートに必要な値を追加し、[**OK**] をクリックします。

ステップ **5**. サービスまたはサービスグループをプライマリノードの負荷分散仮想サーバーにバインドします。

1. **[設定] > [**トラフィック管理**] > [**負荷分散**] > [**仮想サーバー**]** に移動します。
2. 手順 **3** で構成した負荷分散仮想サーバーを選択し、[編集] をクリックします。
3. [サービスとサービスグループ] タブで、[負荷分散仮想サーバーサービスバインドなし] をクリックします。
4. 手順 **4** で構成したサービスを選択し、[バインド] をクリックします。

ステップ **6**. 構成を保存します。

強制フェールオーバーの後、セカンダリは新しいプライマリになります。古いプライマリのクライアントエイリアス IP (VIP) が新しいプライマリに移動します。

CLI を使用した高可用性の設定

ステップ **1**. ステップ **1:** NetScaler CLI を使用して、両方のインスタンスで **INC** 対応モードで高可用性を設定します。

セカンダリノードで、次のコマンドを入力します。

```
add ha node 1 <sec_ip> -inc ENABLED
```

プライマリノードで、次のコマンドを入力します。

```
add ha node 1 <prim_ip> -inc ENABLED
```

sec_ipは、セカンダリノードの管理 NIC の内部 IP アドレスを指します。

prim_ipは、プライマリノードの管理 NIC の内部 IP アドレスを指します。

ステップ **2**. ステップ **2:** プライマリノードとセカンダリノードの両方に VIP と SNIP を追加します。

プライマリノードで次のコマンドを入力します。

```
add ns ip <primary_client_alias_ip> <subnet> -type VIP
```

> 注
>
> VM インスタンスのクライアントサブネットに設定されているエイリアス IP アドレスとネットマスクを入力します。

```
add ns ip <primary_server_alias_ip> <subnet> -type SNIP
```

セカンダリノードで次のコマンドを入力します。

```
add ns ip <primary_client_alias_ip> <subnet> -type VIP
```

> 注
>
> VM インスタンスのクライアントサブネットに設定されているエイリアス IP アドレスとネットマスクを入力します。


```
add ns ip <secondary_server_alias_ip> <subnet> -type SNIP
```

> 注
>
> VM インスタンスのサーバサブネットに設定されているエイリアス IP アドレスとネットマスクを入力します。

ステップ **3:** プライマリ・インスタンスに仮想サーバを追加します。ステップ **3:** プライマリノードに仮想サーバを追加します。

次のコマンドを入力します:

```
add <server_type> vserver <vserver_name> <protocol> <
   primary_client_alias_ip> <port>
```

ステップ **4:** プライマリ・インスタンスに仮想サーバを追加します。プライマリノードにサービスまたはサービスグループを追加します。

次のコマンドを入力します:

```
add service <service_name> <service_ip_address> <protocol> <port>
```

ステップ **5**. サービスまたはサービスグループをプライマリノードの負荷分散仮想サーバーにバインドします。

次のコマンドを入力します:

```
bind <server_type> vserver <vserver_name> <service_name>
```

> 注
>
> 設定を保存するには、コマンド`save config`を入力します。そうしないと、インスタンスの再起動後に設定が失われます。

プライベート **IP** アドレスを持つ **VPX** 高可用性ペアを **Google Cloud Platform** にデプロイする

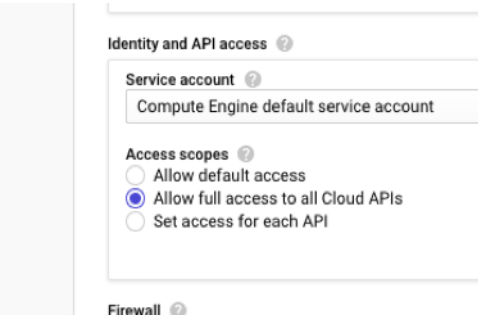
October 17, 2024

プライベート IP アドレスを使用して、VPX 高可用性ペアを GCP にデプロイできます。クライアント IP（VIP）は、プライマリノードのエイリアス IP アドレスとして設定する必要があります。フェールオーバー時に、クライアント IP アドレスがセカンダリノードに移動され、トラフィックが再開されます。

高可用性の詳細については、「高可用性」を参照してください。

はじめに

- Google Cloud Platform に NetScaler VPX インスタンスをデプロイするに記載されている制限事項、ハードウェア要件、注意事項をお読みください。この情報は、高可用性展開にも適用されます。
- GCP プロジェクトで クラウドリソースマネージャー **API** を有効にします。
- インスタンスの作成中に、すべての Cloud API へのフルアクセスを許可します。



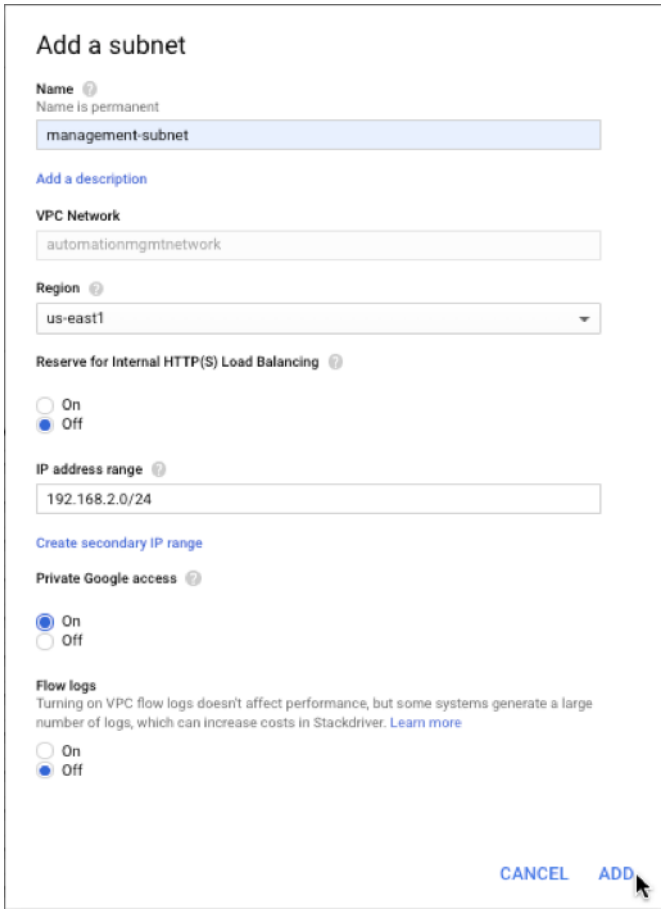
- GCP サービスアカウントに次の IAM 権限があることを確認します。

```
REQUIRED_INSTANCE_IAM_PERMS = [
"compute.forwardingRules.list",
"compute.forwardingRules.setTarget",
"compute.instances.setMetadata",
"compute.instances.get",
"compute.instances.list",
"compute.instances.updateNetworkInterface",
"compute.targetInstances.list",
"compute.targetInstances.use",
"compute.targetInstances.create",
"compute.zones.list",
"compute.zoneOperations.get",
]
```

- 管理インターフェイス以外のインターフェイスに外部 IP アドレスを設定している場合は、GCP サービスアカウントに次の追加の IAM 権限があることを確認します。

```
REQUIRED_INSTANCE_IAM_PERMS = [
"compute.addresses.use"
"compute.instances.addAccessConfig",
"compute.instances.deleteAccessConfig",
"compute.networks.useExternalIp",
"compute.subnetworks.useExternalIp",
]
```

- 仮想マシンにインターネットアクセスがない場合は、管理サブネットでプライベート **Google Access** を有効



にする必要があります。

- プライマリ ノードで GCP 転送ルールを構成している場合は、GCP 上の VPX 高可用性ペアの転送ルールのサポート に記載されている制限と要件を読んで、フェイルオーバー時に新しいプライマリに更新してください。

VPX 高可用性ペアを Google Cloud Platform にデプロイする方法

ここでは、高可用性展開手順の概要を示します。

1. 同じリージョンに VPC ネットワークを作成します。たとえば、アジア東です。
2. 同じリージョンに 2 つの VPX インスタンス（プライマリノードとセカンダリノード）を作成します。それらは、同じゾーンまたは異なるゾーンに存在することができます。たとえば、アジア東-1a、アジア東-Ib。
3. NetScaler GUI または ADC CLI コマンドを使用して、両方のインスタンスで高可用性設定を構成します。

手順 1. 手順 1: VPC ネットワークを作成する

要件に基づいて VPC ネットワークを作成します。管理 NIC、クライアント NIC、サーバー NIC に関連付けるために 3 つの VPC ネットワークを作成することをお勧めします。

VPC ネットワークを作成するには、次の手順を実行します。

1. **Google** コンソールにログインし、**[**ネットワーク**] > [VPC** ネットワーク**] > [VPC** ネットワークの作成**]** をクリックします。
2. 必須フィールドに入力し、[**Create**] をクリックします。

詳細については、「Google Cloud Platform に NetScaler VPX インスタンスをデプロイする」の「**VPC** ネットワークの作成」セクションを参照してください。

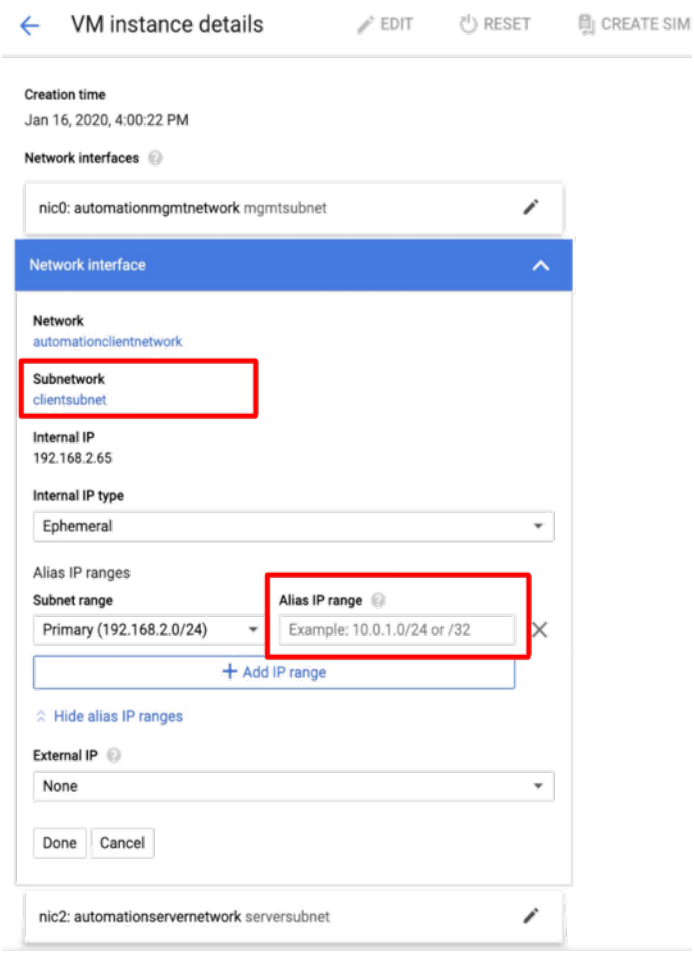
手順 **3.** 手順 **2**: **2** つの **VPX** インスタンスを作成する

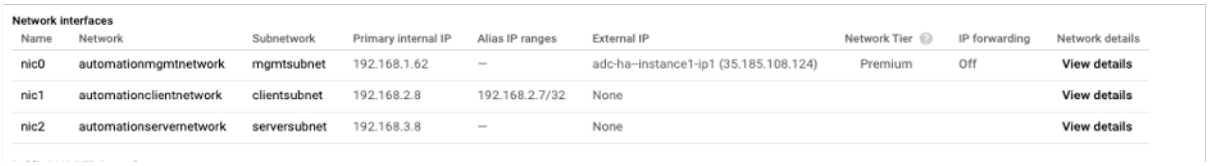
シナリオ: マルチ NIC、マルチ IP のスタンドアロン VPX インスタンスをデプロイするに記載されている手順に従って、2 つの VPX インスタンスを作成します。

> 重要**:**
>
> クライアントエイリアス IP アドレスをプライマリノードに割り当てます。VPX インスタンスの内部 IP アドレスを使用して VIP を構成しないでください。

クライアントエイリアス IP アドレスを作成するには、次の手順を実行します。

1. VM インスタンスに移動し、[編集] をクリックします。
2. [ネットワークインターフェイス（Network Interface）] ウィンドウで、クライアントインターフェイスを編集します。
3. [エイリアス **IP** 範囲（Alias IP range）] フィールドに、クライアントエイリアス IP アドレスを入力します。





Network interfaces

Name	Network	Subnetwork	Primary internal IP	Alias IP ranges	External IP	Network Tier	IP forwarding	Network details
nic0	automationmgmtnetwork	mgmtsubnet	192.168.1.62	–	adc-ha--instance1-ip1 (35.185.108.124)	Premium	Off	View details
nic1	automationclientnetwork	clientsubnet	192.168.2.8	192.168.2.7/32	None			View details
nic2	automationservernetwork	serversubnet	192.168.3.8	–	None			View details

フェールオーバー後、古いプライマリが新しいセカンダリになると、エイリアス IP アドレスは古いプライマリから移動し、新しいプライマリに接続されます。

VPX インスタンスを構成したら、仮想 (VIP) アドレスとサブネット IP (SNIP) アドレスを構成できます。詳細については、「Citrix ADC 所有の IP アドレスの構成」を参照してください。

手順 **3.** 高可用性の構成

Google Cloud Platform でインスタンスを作成した後、NetScaler GUI または CLI を使用して高可用性を構成できます。

GUI を使用した高可用性の構成

ステップ **1.** 両方のノードで INC Enabled モードで高可用性を設定します。

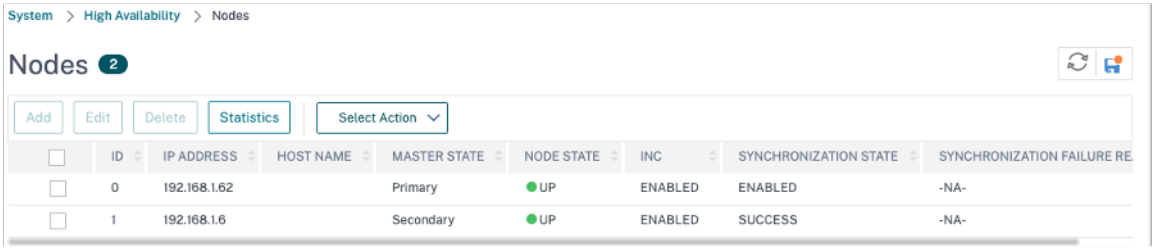
プライマリノードで、次の手順を実行します。

1. GCP Console `nsroot` からノードのユーザー名とインスタンス ID をパスワードとしてインスタンスにログオンします。
2. 構成 **>** システム **>** 高可用性 **>** ノードに移動し、追加をクリックします。
3. [リモートノードの **IP** アドレス] フィールドに、セカンダリノードの管理 NIC のプライベート IP アドレスを入力します。
4. [セルフノードで **INC (**独立ネットワーク構成**)** モードをオンにする] チェックボックスをオンにします。
5. [**Create**] をクリックします。

先に進む前に、[**Nodes**] ページにセカンダリノードの同期状態が **SUCCESS** と表示されていることを確認してください。

1. GCP Console `nsroot` からノードのユーザー名とインスタンス ID をパスワードとしてインスタンスにログオンします。
2. 構成 **>** システム **>** 高可用性 **>** ノードに移動し、追加をクリックします。
3. [リモートノード **IP** アドレス] フィールドに、プライマリノードの管理 NIC のプライベート IP アドレスを入力します。
4. [セルフノードで **INC (**独立ネットワーク構成**)** モードをオンにする] チェックボックスをオンにします。
5. [**Create**] をクリックします。

セカンダリノードで、次の手順を実行します。



> 注
>
> セカンダリノードがプライマリノードと同期されると、セカンダリノードにはプライマリノードと同じログオン認証情報が割り当てられます。

ステップ **2.** 両方のノードに仮想 IP アドレスとサブネット IP アドレスを追加します。

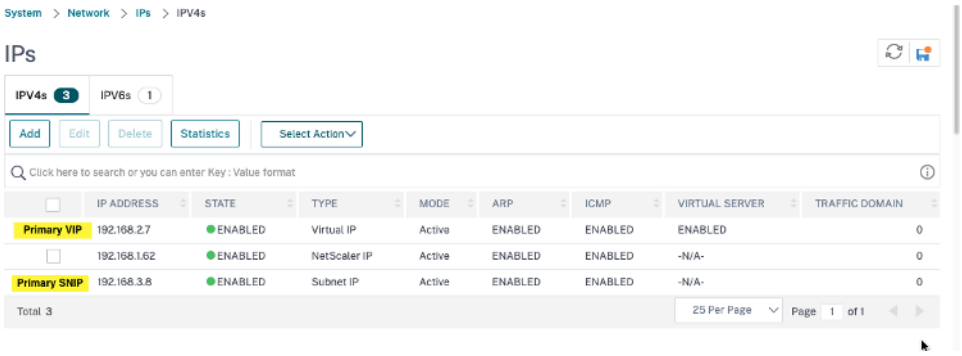
プライマリノードで、次の手順を実行します。

1. **System > Network > IP Sets > Add** に移動します。
2. クライアントエイリアス IP（VIP）アドレスを作成するには、次の手順を実行します。

a) 仮想マシンインスタンスのクライアントサブネットに設定されたエイリアス IP アドレスとネットマスクを入力します。
b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Virtual IP**] を選択します。
c) [**Create**] をクリックします。

3. サーバ IP (SNIP) アドレスを作成するには、次の手順を実行します。

a) プライマリ・インスタンスのサーバ側インターフェイスの内部 IP アドレスと、サーバ・サブネットに設定されたネットマスクを入力します。
b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Subnet IP**] を選択します。
c) [**Create**] をクリックします。



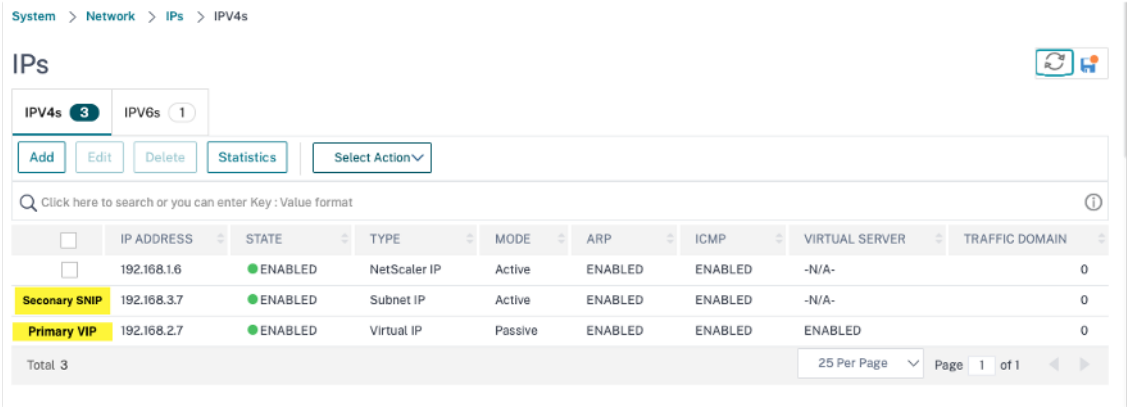
セカンダリノードで、次の手順を実行します。

1. **System > Network > IP Sets > Add** に移動します。
2. クライアントエイリアス IP（VIP）アドレスを作成するには、次の手順を実行します。

a) プライマリ VM インスタンスのクライアントサブネットに対して構成された Alias IP アドレスとネットマスクを入力します。
b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Subnet IP**] を選択します。
c) [**Create**] をクリックします。

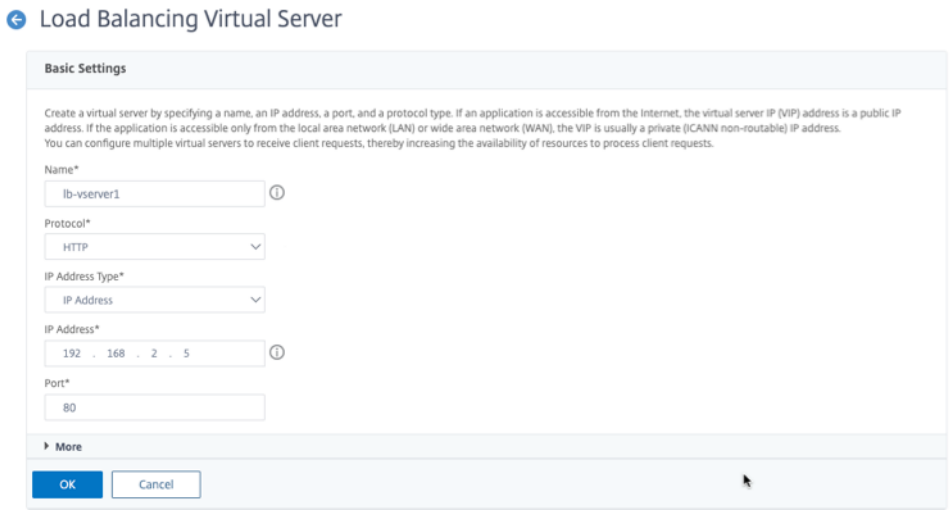
3. サーバ IP (SNIP) アドレスを作成するには、次の手順を実行します。

a) セカンダリインスタンスのサーバ側インターフェイスの内部 IP アドレスと、サーバサブネットに設定されたネットマスクを入力します。
b) [**IP Type**] フィールドで、ドロップダウンメニューから [**Subnet IP**] を選択します。
c) [**Create**] をクリックします。



ステップ **3:** プライマリ・インスタンスに仮想サーバを追加します。ステップ **3:** プライマリノードに負荷分散仮想サーバーを追加します。

1. [設定] **>** [トラフィック管理] **>** [負荷分散] **>** [仮想サーバー] **>** [追加] に移動します。
2. 名前、プロトコル、IP アドレスタイプ（IP アドレス）、IP アドレス（プライマリクライアントエイリアス IP アドレス）、およびポートに必要な値を追加し、「**OK**」をクリックします。



ステップ **4:** プライマリ・インスタンスに仮想サーバを追加します。プライマリノードにサービスまたはサービスグループを追加します。

1. **[**設定**] > [**トラフィック管理**] > [**負荷分散**] > [**サービス**] >[**追加**]** に移動します。
2. サービス名、IP アドレス、プロトコル、およびポートに必要な値を追加し、[**OK**] をクリックします。

ステップ **5**. サービスまたはサービスグループをプライマリノードの負荷分散仮想サーバーにバインドします。

1. **[**設定**] > [**トラフィック管理**] > [**負荷分散**] > [**仮想サーバー**]** に移動します。
2. 手順 **3** で構成した負荷分散仮想サーバーを選択し、[編集] をクリックします。
3. [サービスとサービスグループ] タブで、[負荷分散仮想サーバーサービスバインドなし] をクリックします。
4. 手順 **4** で構成したサービスを選択し、[バインド] をクリックします。

ステップ **5.** 構成を保存します。

強制フェールオーバーの後、セカンダリは新しいプライマリになります。古いプライマリからのクライアントエイリアス IP (VIP) とサーバエイリアス IP (SNIP) が新しいプライマリに移動します。

CLI を使用した高可用性の設定

ステップ **1.** ステップ **1:** NetScaler CLI を使用して、両方のインスタンスで **INC** 対応モードで高可用性を設定します。

セカンダリノードで、次のコマンドを入力します。

```
add ha node 1 <sec_ip> -inc ENABLED
```

プライマリノードで、次のコマンドを入力します。

```
add ha node 1 <prim_ip> -inc ENABLED
```

sec_ipは、セカンダリノードの管理 NIC の内部 IP アドレスを指します。

prim_ipは、プライマリノードの管理 NIC の内部 IP アドレスを指します。

ステップ **2.** 両方のノードに VIP と SNIP を追加します。

プライマリノードで次のコマンドを入力します。

```
add ns ip <primary_client_alias_ip> <subnet> -type VIP
```

> 注
>
> 仮想マシンインスタンスのクライアントサブネットに設定されたエイリアス IP アドレスとネットマスクを入力します。

```
add ns ip <primary_snip> <subnet> -type SNIP
```

primary_snipは、プライマリインスタンスのサーバ側インターフェイスの内部 IP アドレスを指します。

セカンダリノードで次のコマンドを入力します。

```
add ns ip <primary_client_alias_ip> <subnet> -type VIP
```

> 注
>
> プライマリ VM インスタンスのクライアントサブネットに対して構成された Alias IP アドレスとネットマスクを入力します。

```
add ns ip <secondary_snip> <subnet> -type SNIP
```

secondary_snipは、セカンダリインスタンスのサーバ側インターフェイスの内部 IP アドレスを指します。

> 注
>
> VM インスタンスのサーバサブネットに設定された IP アドレスとネットマスクを入力します。

ステップ **3:** プライマリ・インスタンスに仮想サーバを追加します。ステップ **3:** プライマリノードに仮想サーバを追加します。

次のコマンドを入力します:

```
1   add <server_type> vserver <vserver_name> <protocol> <
       primary_client_alias_ip> <port>
```

ステップ **4:** プライマリ・インスタンスに仮想サーバを追加します。プライマリノードにサービスまたはサービスグループを追加します。

次のコマンドを入力します:

```
1   add service <service_name> <service_ip_address> <protocol> <port>
```

ステップ **5**. サービスまたはサービスグループをプライマリノードの負荷分散仮想サーバーにバインドします。

次のコマンドを入力します:

```
1   bind <server_type> vserver <vserver_name> <service_name>
```

> 注
>
> 設定を保存するには、コマンド`save config`を入力します。そうしないと、インスタンスの再起動後に設定が失われます。

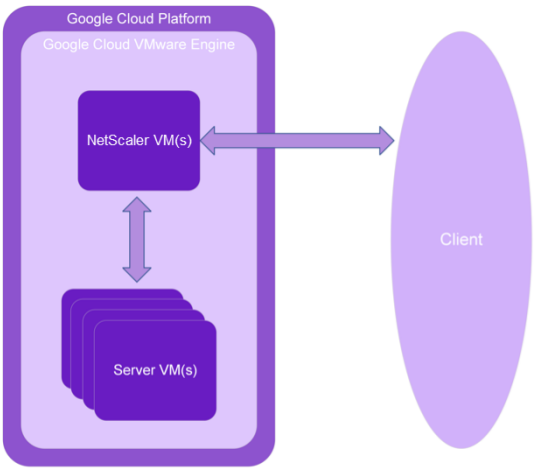
Google Cloud VMware Engine に **NetScaler VPX** インスタンスをインストールする

October 17, 2024

Google Cloud VMware Engine（GCVE）は、専用のベアメタルの Google Cloud Platform インフラストラクチャから構築された vSphere クラスタを含むプライベートクラウドを提供します。最小初期デプロイメントは 3 ホストですが、追加ホストは一度に 1 つずつ追加できます。プロビジョニングされたすべてのプライベートクラウドには、vCenter Server、vSAN、vSphere、NSX-T があります。

GCVE を使用すると、必要な数の ESX ホストを使用して Google Cloud Platform 上にクラウドソフトウェア定義データセンター（SDDC）を作成できます。GCVE は NetScaler VPX の導入をサポートします。GCVE はオンプレミス vCenter と同じユーザーインターフェイスを提供します。ESX ベースの Citrix ADC VPX デプロイメントと同じように機能します。

次の図は、管理者またはクライアントがインターネット経由でアクセスできる Google Cloud Platform 上の GCVE を示しています。管理者は GCVE を使用してワークロードまたはサーバー VM を作成、管理、構成できます。管理者は OpenVPN 接続を使用して GCVE のウェブベースの vCenter と NSX-T Manager にアクセスできます。vCenter を使用して GCVE 内に NetScaler VPX インスタンス（スタンドアロンまたは HA ペア）とサーバ仮想マシンを作成し、NSX-T Manager を使用して対応するネットワークを管理できます。GCVE 上の NetScaler VPX インスタンスは、オンプレミスの VMware ホストクラスタと同様に機能します。GCVE は、管理インフラストラクチャへの OpenVPN 接続を使用して管理できます。



前提条件

仮想アプライアンスのインストールを開始する前に、次の操作を行います。

- Google Cloud VMware エンジンとその前提条件の詳細については、Google Cloud VMware エンジンのドキュメントを参照してください。
- Google Cloud VMware Engine のデプロイに関する詳細については、「Google Cloud VMware Engine プライベートクラウドのデプロイ」を参照してください。
- ポイントツーサイト VPN ゲートウェイを使用してプライベートクラウドに接続し、Google Cloud VMware Engine にアクセスして管理する方法の詳細については、「Google Cloud VMware Engine プライベートクラウドへのアクセス」を参照してください。
- VPN クライアントマシンで、NetScaler VPX アプライアンスのセットアップファイルをダウンロードします。
- 仮想マシンが接続する VMware SDDC 上に、適切な NSX-T ネットワークセグメントを作成します。詳細については、「Google Cloud VMware Engine でのネットワークセグメントの追加」を参照してください。
- VPX ライセンスファイルを入手します。NetScaler VPX インスタンスライセンスの詳細については、「ライセンスの概要」を参照してください。
- GCVE プライベートクラウドに作成または移行された仮想マシン (VM) は、ネットワークセグメントに接続する必要があります。

VMware クラウドのハードウェア要件

次の表に、VMware SDDC が各 VPX nCore 仮想アプライアンスに対して提供する必要がある仮想コンピューティングリソースを示します。

表 1. NetScaler VPX インスタンスの実行に必要な最小限の仮想コンピューティングリソース

コンポーネント	条件
メモリ	2 GB
仮想 CPU（VCPU）	2
仮想ネットワークインターフェイス	VMware SDDC では、VPX ハードウェアをバージョン 7 以上にアップグレードすると、最大 10 個の仮想ネットワーク インターフェイスをインストールできます。
ディスク領域	20GB

> 注
>
> これは、ハイパーバイザーのディスク要件に加えて必要になります。

VPX 仮想アプライアンスを実稼働で使用するには、フルメモリ割り当てを予約する必要があります。

OVF ツール 1.0 のシステム要件

Open Virtualization Format Tool（OVF Tool）は、Windows および Linux システムで実行できるクライアントアプリケーションです。次の表は、OVF ツールをインストールするための最小システム要件を示しています。

表 2. OVF ツールのインストールに必要な最小システム要件

コンポーネント	条件
オペレーティングシステム	VMware からの詳細な要件については、http://kb.vmware.com/で『OVF ツールユーザーガイド』の PDF ファイルを検索してください。
CPU	最低 750MHz、1GHz 以上推奨
RAM	最小 1 GB、推奨 2 GB
NIC	100Mbps 以上の NIC。

OVF のインストールについては、http://kb.vmware.com/で『OVF ツールユーザーガイド』の PDF ファイルを検索してください。

NetScaler VPX セットアップファイルのダウンロード

VMware ESX 用の NetScaler VPX インスタンスセットアップパッケージは、オープン仮想マシン（OVF）フォーマット標準に準拠しています。これらのファイルは、Citrix Web サイトからダウンロードできます。ログオンするには、Citrix アカウントが必要です。Citrix アカウントをお持ちでない場合は、http://www.citrix.comのホームページにアクセスしてください。［新しいユーザー］リンクをクリックし、指示に従って新しい Citrix アカウントを作成します。

ログオンしたら、Citrix のホームページから次のパスをナビゲートします。

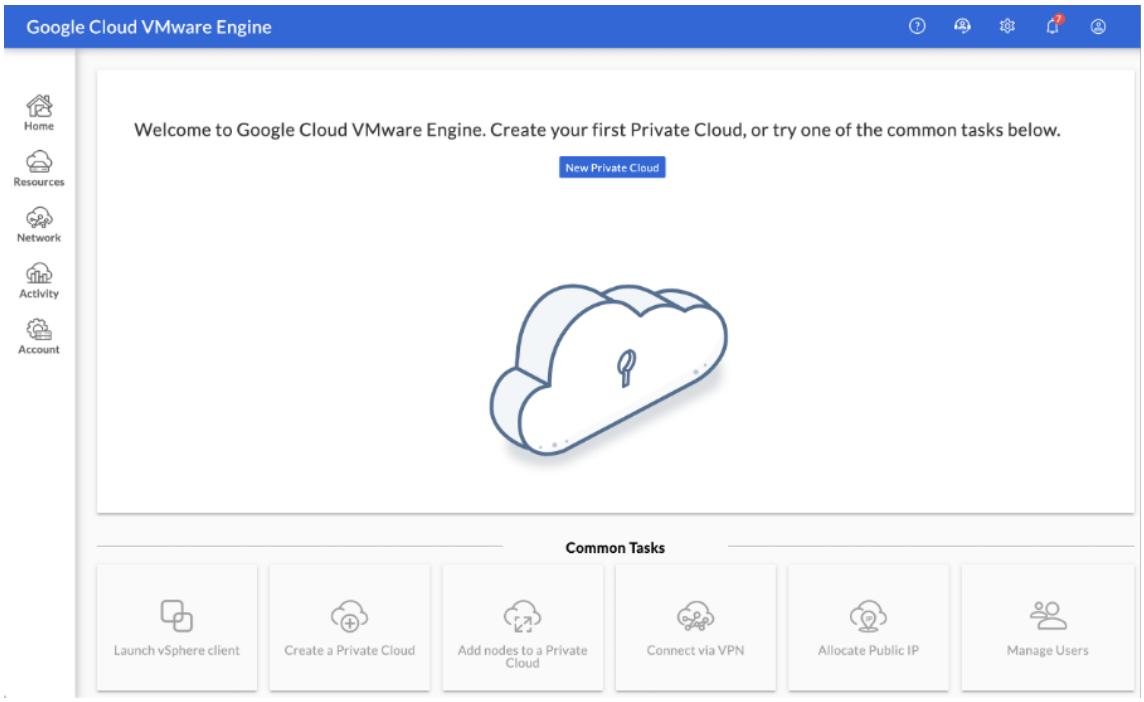
Citrix.com > ダウンロード **> NetScaler >** 仮想アプライアンス。

次のファイルを、ESX サーバーと同じネットワーク上のワークステーションにコピーします。3 つのファイルをすべて同じフォルダーにコピーします。

- NSVPX-ESX-<release number>-<build number>-disk1.vmdk（例えば、NSVPX-ESX-13.0-79.64-disk1.vmdk）
- NSVPX-ESX-<release number>-<build number>.ovf（例えば、NSVPX-ESX-13.0-79.64.ovf）
- NSVPX-ESX-<release number>-<build number>.mf（例えば、NSVPX-ESX-13.0-79.64.mf）

Google Cloud VMware エンジンをデプロイする

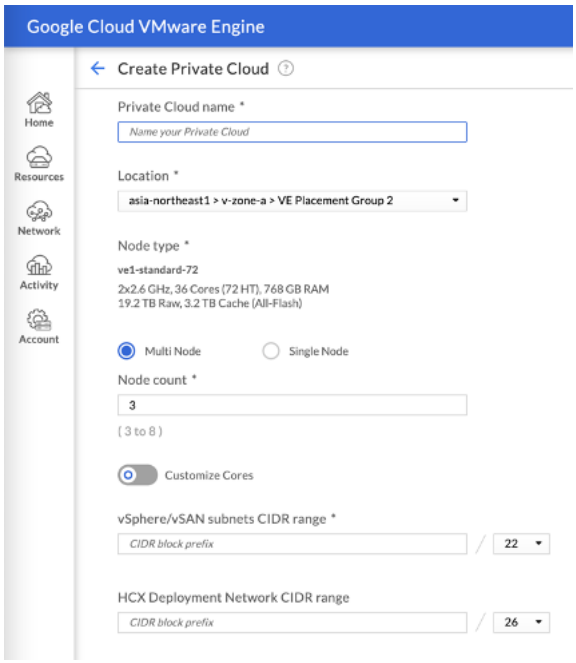
1. GCVE ポータルにログインし、ホームに移動します。



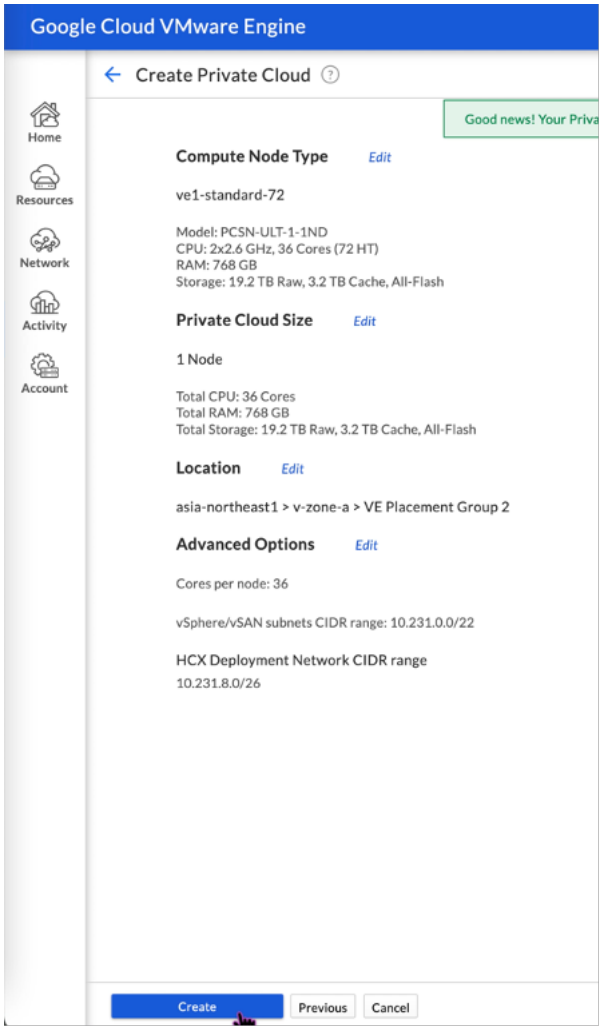
2. 「新規プライベートクラウド」ページで、次の詳細を入力します。

    - プライベートクラウドのデフォルトクラスタを作成するには、最低 3 つの ESXi ホストを選択します。

- **vSphere/vSAN** サブネットの **CIDR** 範囲フィールドには、/22 アドレススペースを使用します。
- **HCX** デプロイメントネットワーク **CIDR** 範囲フィールドには、/26 アドレススペースを使用します。
- 仮想ネットワークの場合、CIDR 範囲がオンプレミスまたは他の GCP サブネット (仮想ネットワーク) と重複していないことを確認します。

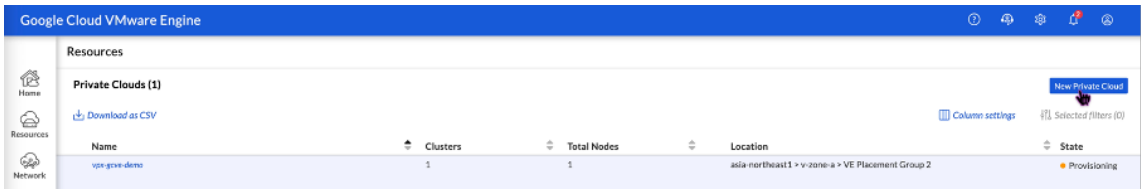


3. [確認して作成] をクリックします。
4. 設定を確認します。設定を変更する必要がある場合は、[前へ] をクリックします。



5. [**Create**] をクリックします。プライベートクラウドのプロビジョニングプロセスが開始されます。プライベートクラウドのプロビジョニングには最大 2 時間かかることがあります。

6. 「リソース」に移動して、作成されたプライベートクラウドを確認します。

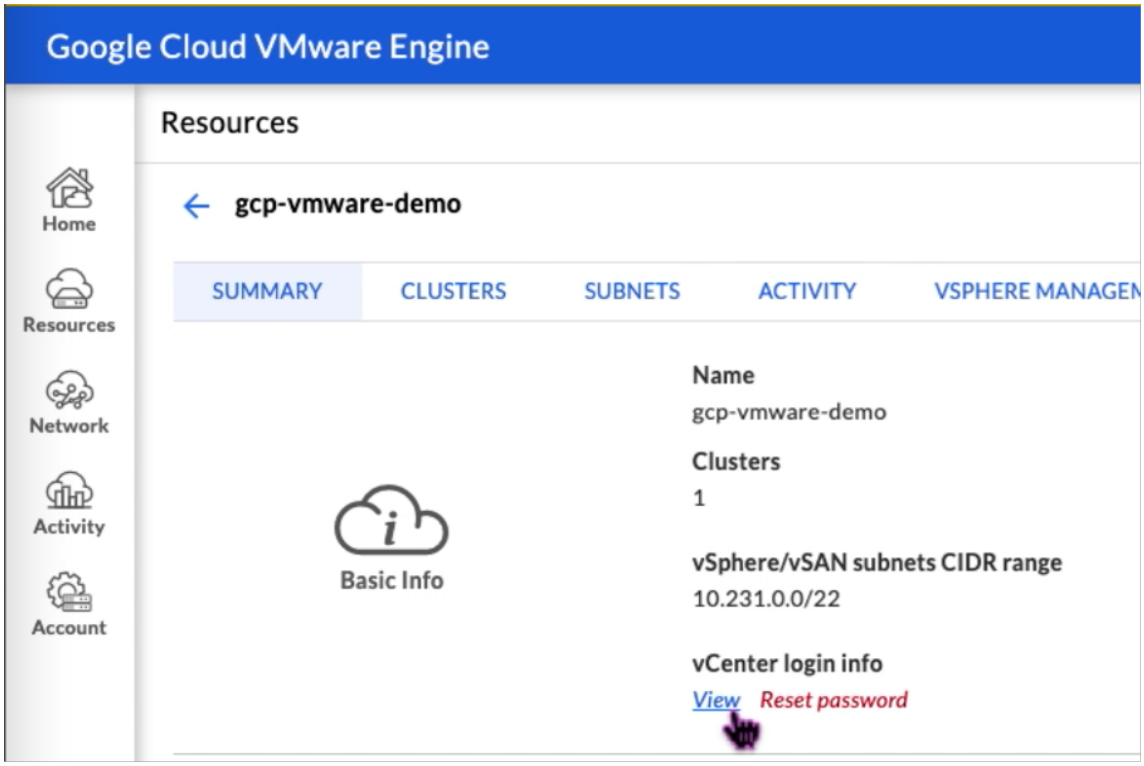


7. このリソースにアクセスするには、ポイントツーサイト VPN を使用して GCVE に接続する必要があります。詳細については、次のドキュメントを参照してください。

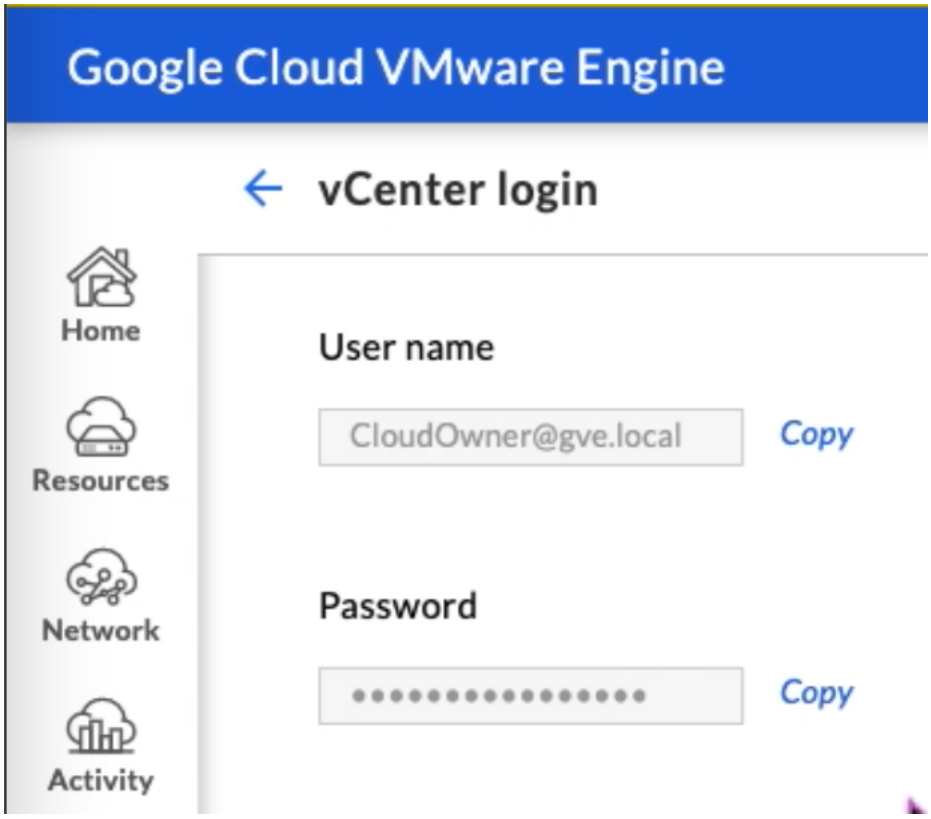
   - VPN ゲートウェイ
   - VPN を使用して接続する

プライベートクラウド **vCenter** ポータルにアクセスする

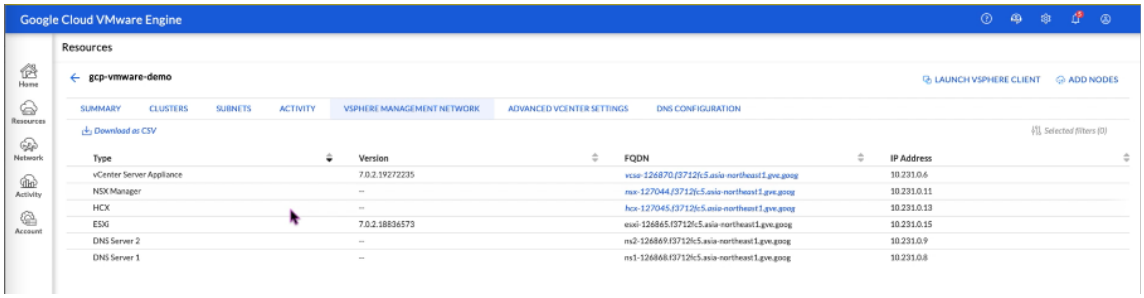
1. Google Cloud VMware Engine プライベートクラウドに移動します。[概要] タブの [**vCenter** ログイン情報] で、[表示] をクリックします。



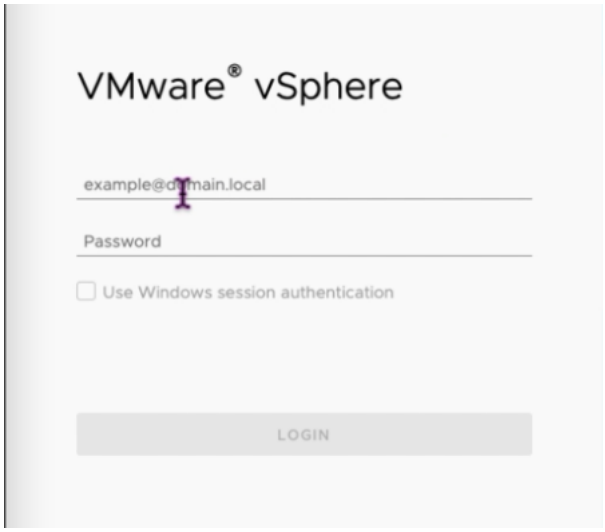
2. vCenter の認証情報を書き留めます。



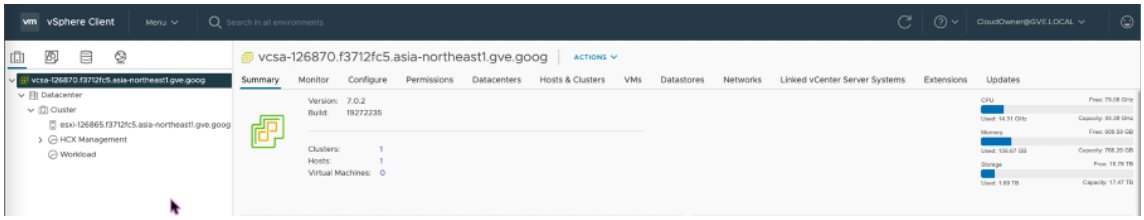
3. vSPHERE CLIENT の起動をクリックして **vSphere** クライアントを起動するか、**VSPHERE** 管理ネットワークに移動して **vCenter Server** アプライアンスの FQDN をクリックします。



4. この手順のステップ 2 でメモした vCenter 認証情報を使用して VMware vSphere にログインします。



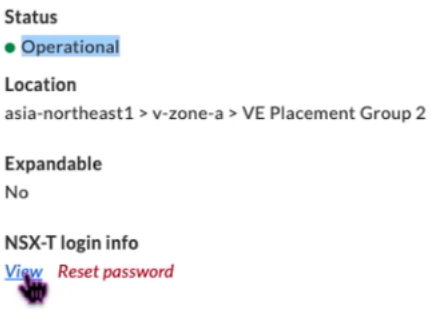
5. vSphere クライアントでは、GCVE ポータルで作成した ESXi ホストを確認できます。



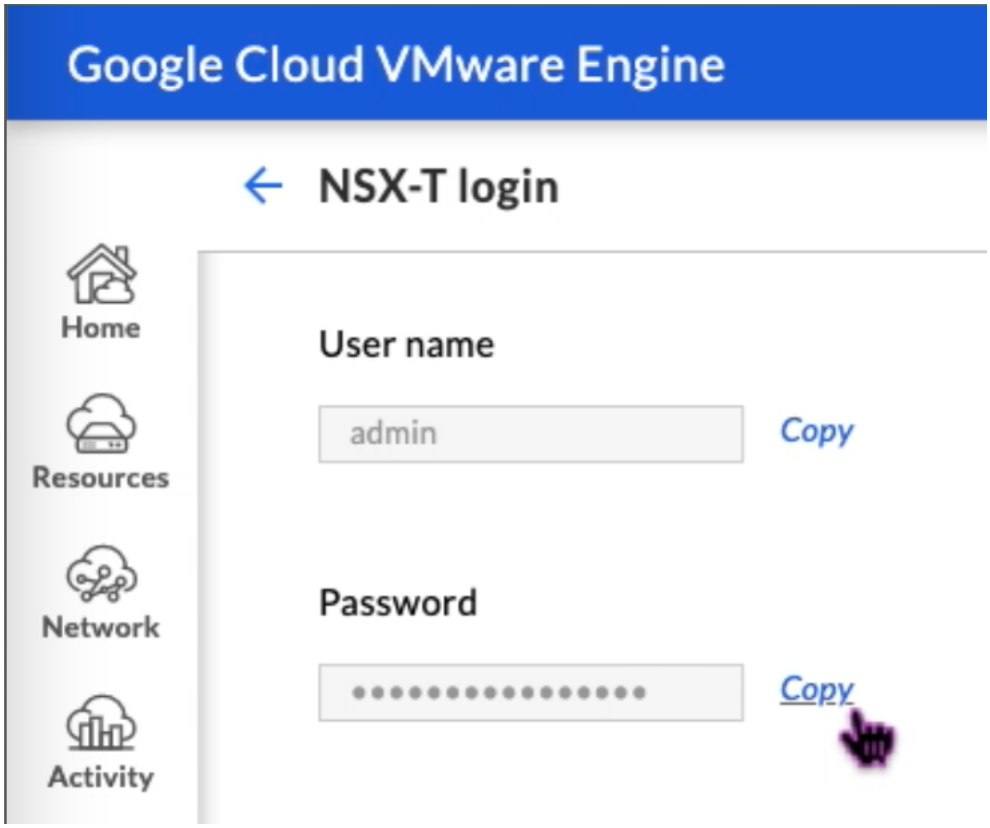
GCVE NSX-T ポータルで NSX-T セグメントを作成します

NSX-T セグメントは、Google Cloud VMware エンジンコンソールの NSX マネージャから作成および設定できます。これらのセグメントはデフォルトの Tier-1 ゲートウェイに接続され、これらのセグメントのワークロードは East-West および North-South 接続を取得します。セグメントを作成すると、vCenter に表示されます。

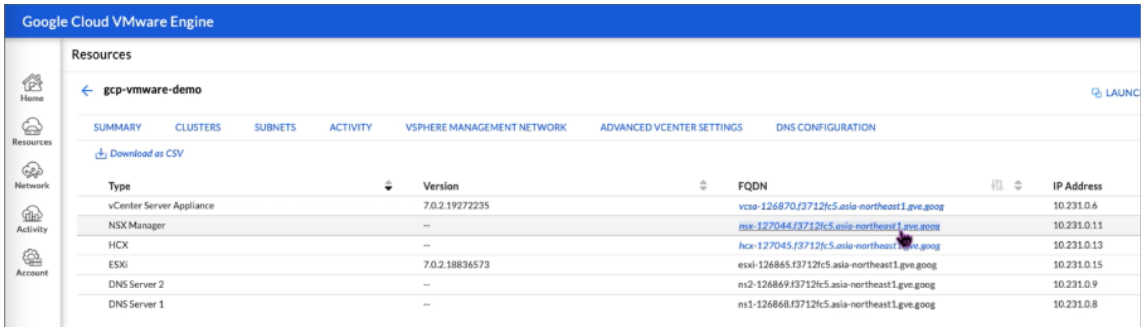
1. GCVE プライベートクラウドの **[概要]-> [NSX-T** ログイン情報] で、[表示] を選択します。



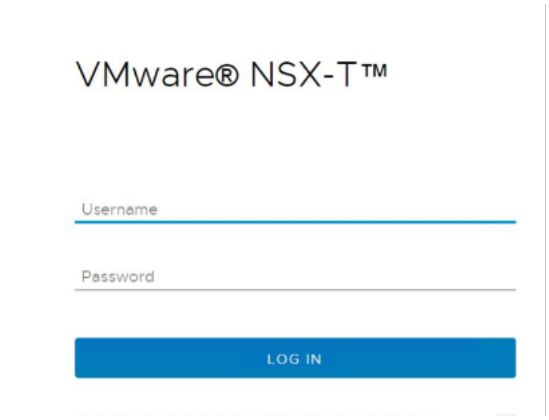
2. NSX-T の認証情報を書き留めておきます。



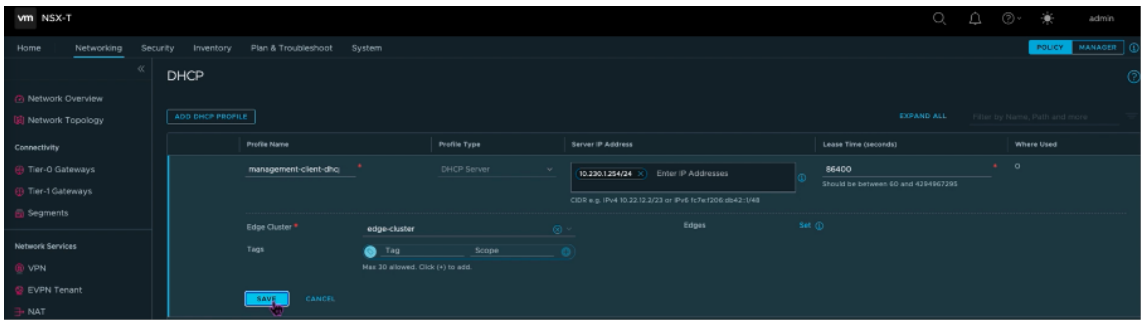
3. [**VSPHERE** 管理ネットワーク] に移動して NSX Manager を起動し、**NSX Manager** の FQDN をクリックします。



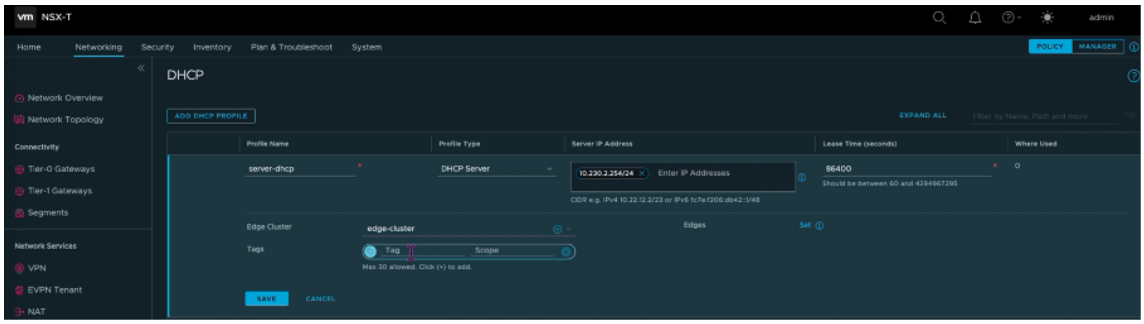
4. この手順のステップ 2 でメモした認証情報を使用して NSX Manager にログインします。



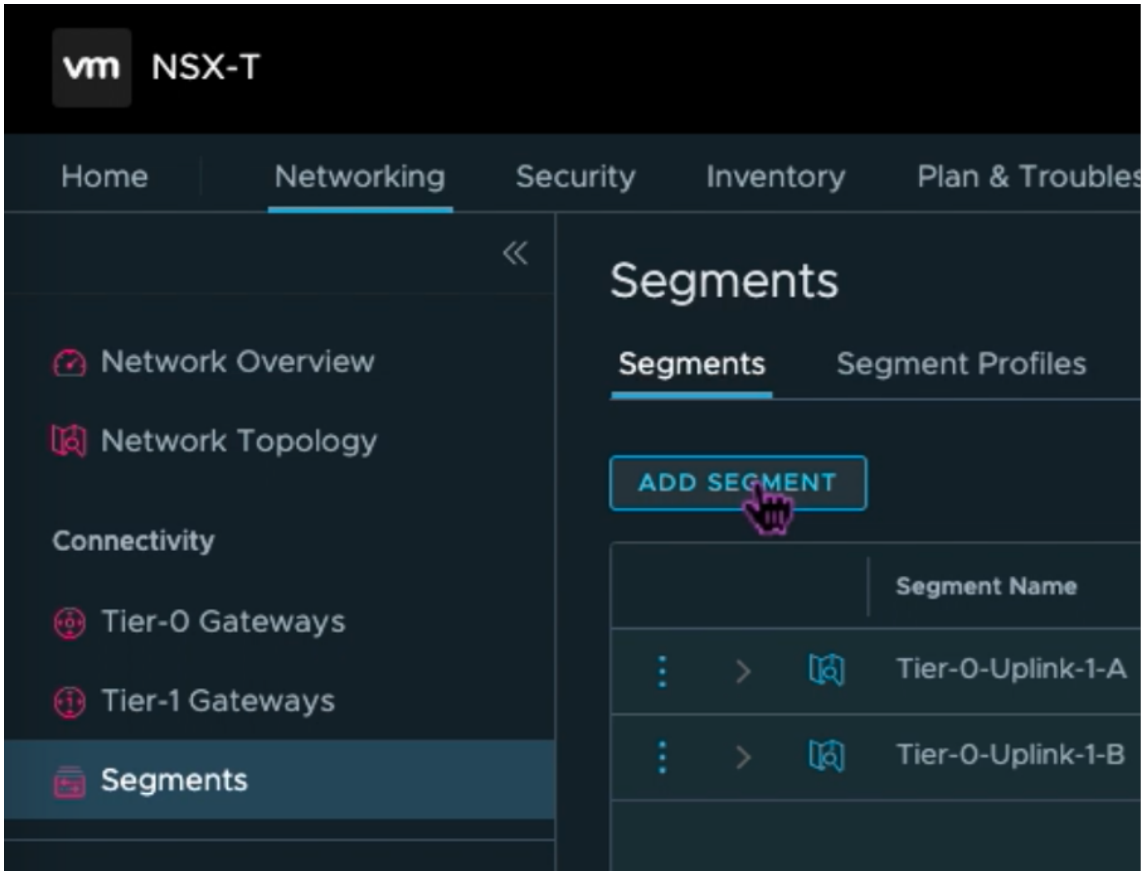
5. 新しいセグメントまたはサブネットの DHCP サービスを設定します。
6. サブネットを作成する前に、DHCP サービスを設定します。
7. NSX-T で、[ネットワーク] **> [DHCP]** に移動します。ネットワークダッシュボードには、サービスが Tier-0 ゲートウェイを 1 つと Tier-1 ゲートウェイを 1 つ作成していることがわかります。
8. DHCP サーバーのプロビジョニングを開始するには、「**DHCP** プロファイルの追加」をクリックします。
9. DHCP 名フィールドに、クライアント管理プロファイルの名前を入力します。
10. プロファイルタイプとして **DHCP** サーバーを選択します。
11. 「サーバー **IP** アドレス」列に、DHCP サービスの IP アドレス範囲を指定します。
12. **Edge** クラスタを選択します。
13. [**Save**] をクリックして、DHCP サービスを作成します。



14. サーバの DHCP 範囲について、手順 6～13 を繰り返します。



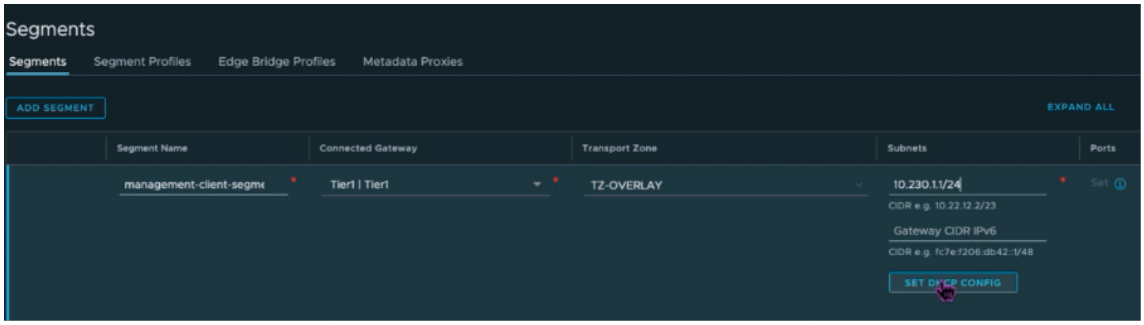
15. 2 つのセグメントを作成します。1 つはクライアントと管理インターフェイス用、もう 1 つはサーバーインターフェイス用です。
16. NSX-T で、[ネットワーク**] > [**セグメント**]** に移動します。
17. [**Add Segment**] をクリックします。



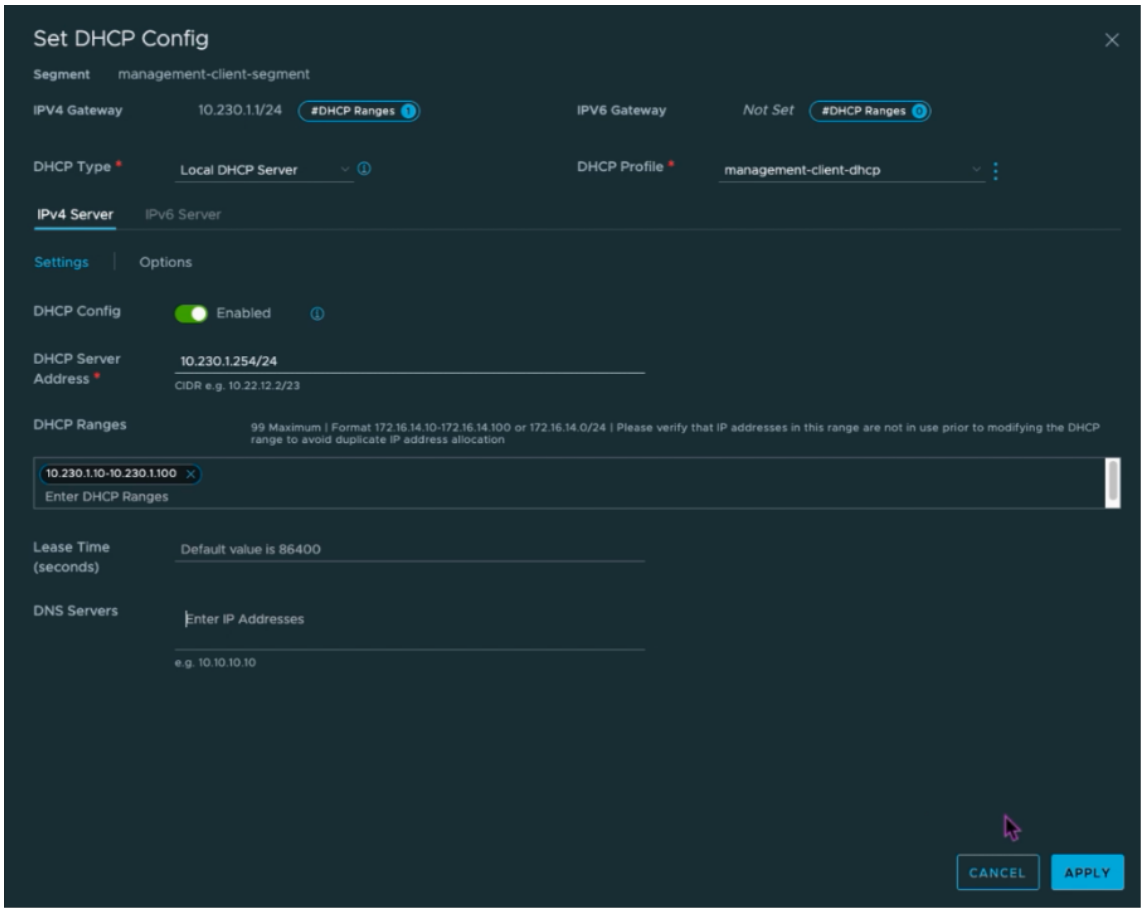
18. セグメント名フィールドに、クライアント管理セグメントの名前を入力します。
19. 接続されたゲートウェイリストで、**Tier1** を選択して Tier-1 ゲートウェイに接続します。

「トランスポートゾーン」リストで「**TZ-OVERLAY」を　オーバーレイ **。
選択します

20.
21. [サブネット] 列に、サブネット範囲を入力します。サブネット範囲で **.** 1を最後のオクテットとして指定します。例：10.12.2.1/24。

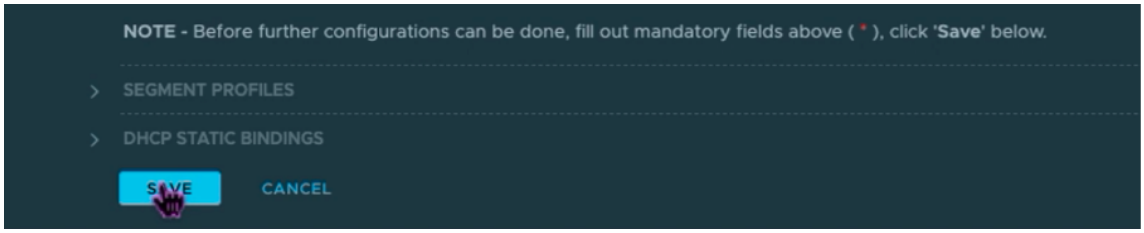


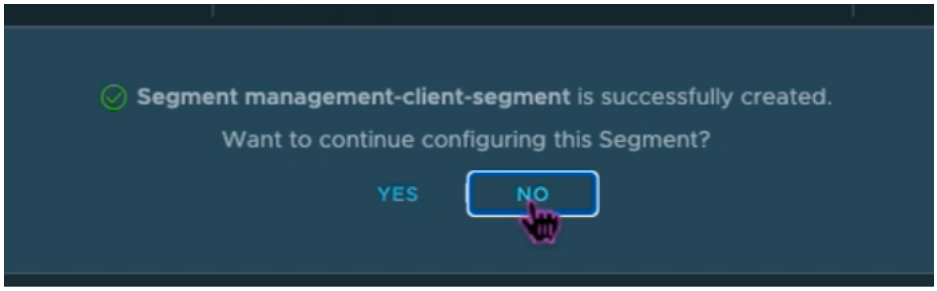
22.「**DHCP** 構成を設定」をクリックし、「**DHCP** 範囲」フィールドに値を入力します。



23. [**Apply**] をクリックして DHCP 設定を保存します。

24. [保存] をクリックします。





25. サーバーセグメントについても手順 17 ～24 を繰り返します。
26. 仮想マシンの作成時に vCenter でこれらのネットワークセグメントを選択できるようになりました。

詳細については、「最初のサブネットの作成」を参照してください。

VMware クラウドへの Citrix ADC VPX インスタンスのインストール

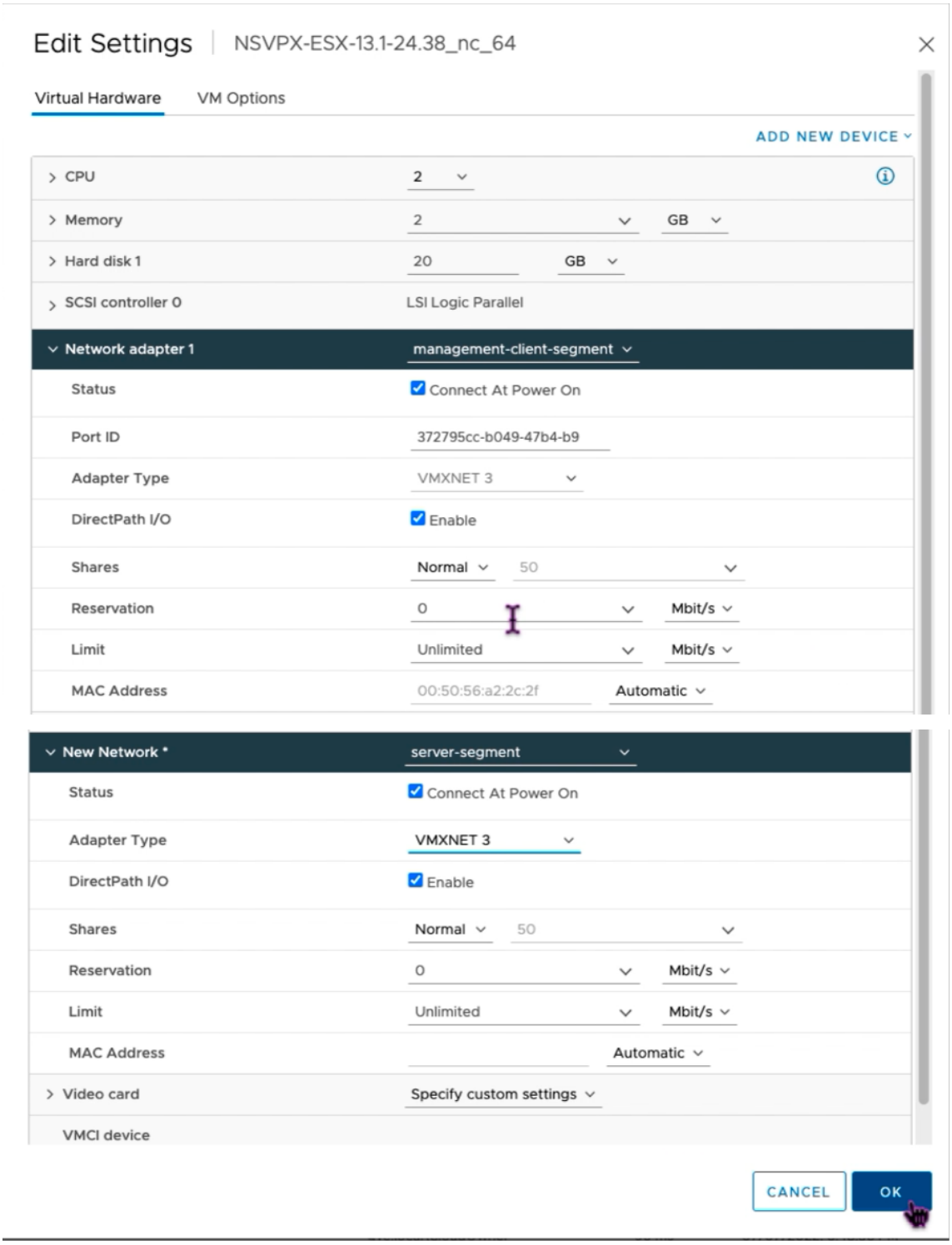
GCVE にプライベートクラウドをインストールして設定したら、vCenter を使用して VMware Engine に仮想アプライアンスをインストールできます。インストールできる仮想アプライアンスの数は、プライベートクラウドで使用可能なリソースの量によって異なります。

NetScaler VPX インスタンスをプライベートクラウドにインストールするには、プライベートクラウドのポイントツーサイト VPN に接続されたデスクトップで以下の手順を実行します。

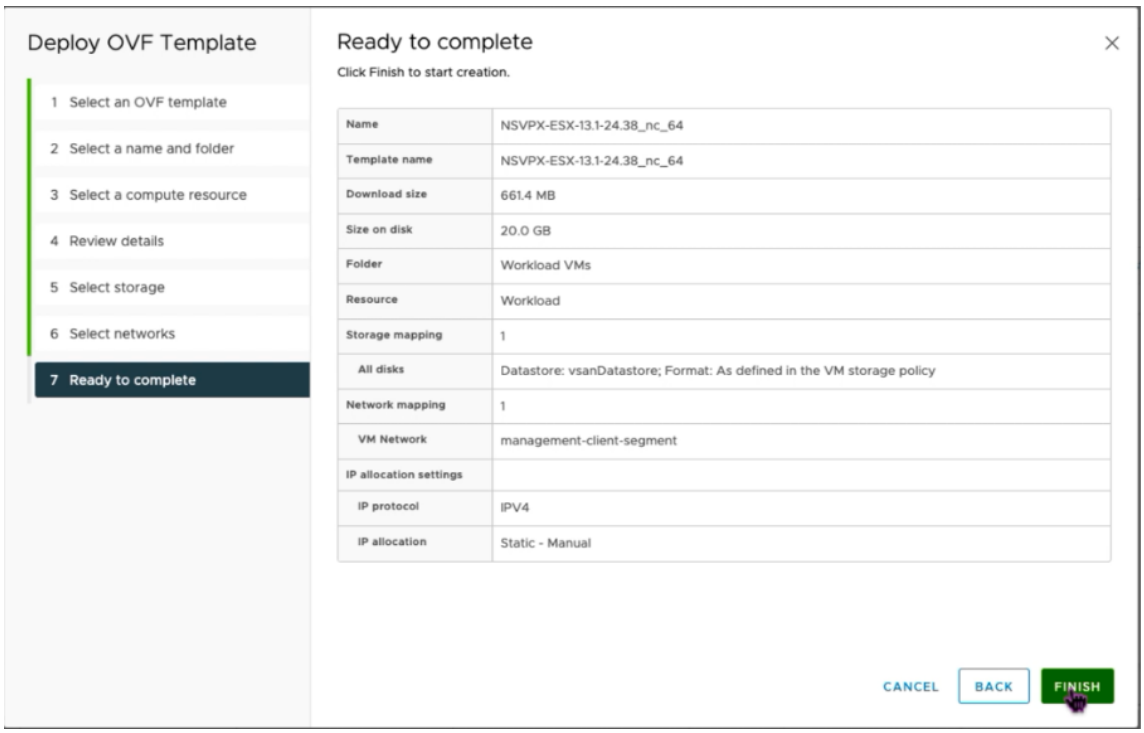
1. ESXi ホスト用の NetScaler VPX インスタンスセットアップファイルを、NetScaler ダウンロードサイトからダウンロードします。
2. プライベートクラウドのポイントツーサイト VPN に接続されたブラウザで VMware vCenter を開きます。
3. [ユーザー名] フィールドと [パスワード] フィールドに管理者の資格情報を入力し、[ログイン] をクリックします。
4. [**File**] メニューの [**Deploy OVF Template**] を選択します。
5. [**OVF** テンプレートのデプロイ] ダイアログボックスの [ファイルからの展開] フィールドで、NetScaler VPX インスタンスセットアップファイルを保存した場所を参照し、.ovf ファイルを選択し、[次へ] をクリックします。

   > 注
   >
   > デフォルトでは、NetScaler VPX インスタンスは E1000 ネットワークインターフェイスを使用します。VMXNET3 インターフェイスで ADC を展開するには、E1000 ではなく VMXNET3 インターフェイスを使用するように OVF を変更します。VMXNET3 インターフェイスの可用性は GCP インフラストラクチャによって制限され、Google Cloud VMware Engine では利用できない場合があります。

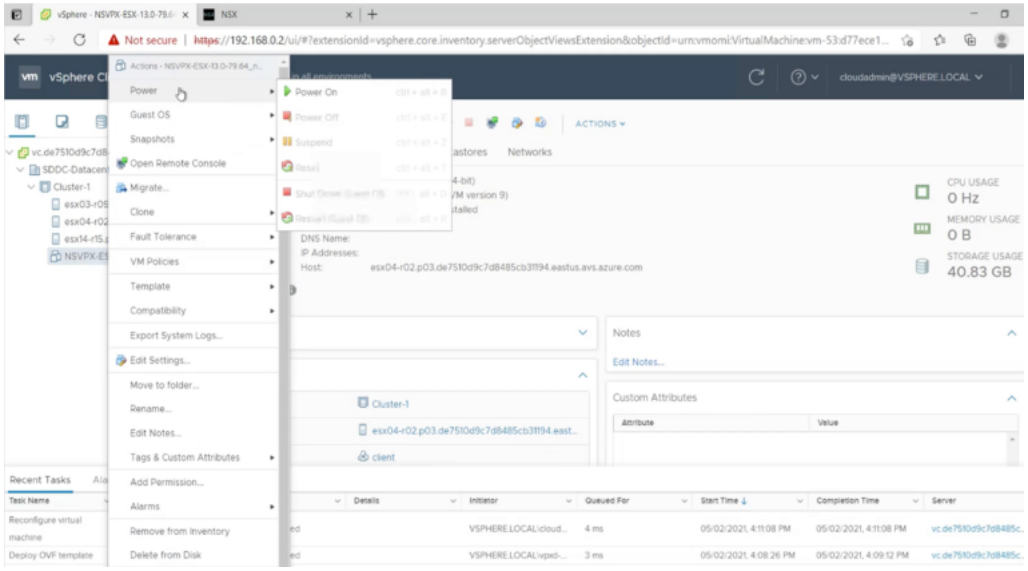
6. 仮想アプライアンスの OVF テンプレートに表示されるネットワークを、NSX-T Manager で設定したネットワークにマッピングします。[作成] または [**OK]** をクリックします。



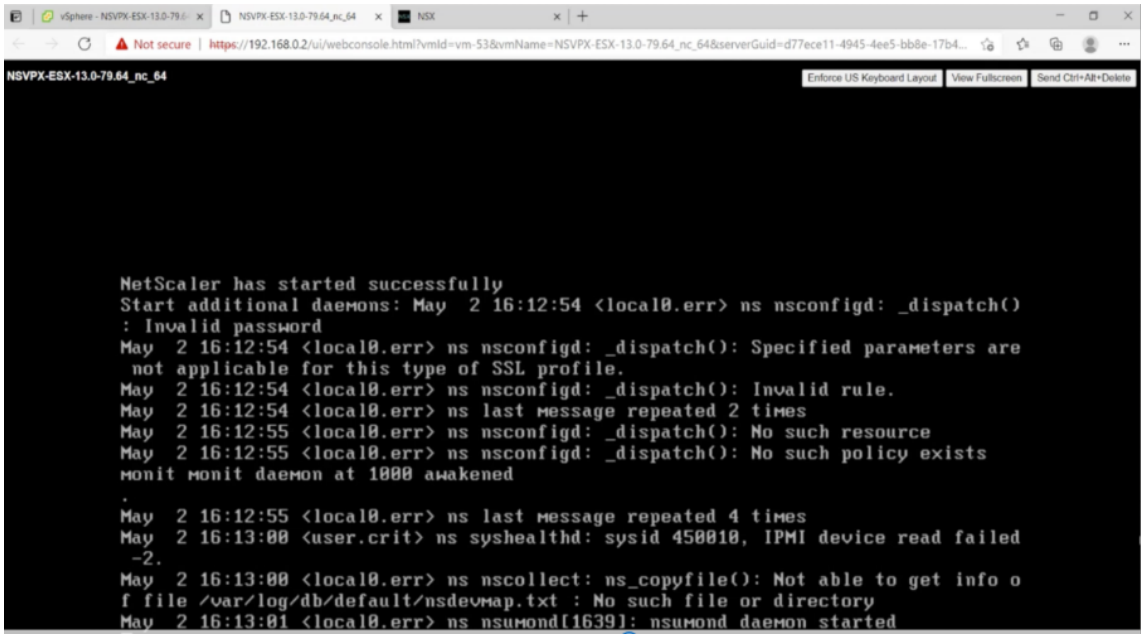
7. [完了] をクリックして VMware クラウドへの仮想アプライアンスのインストールを開始します。



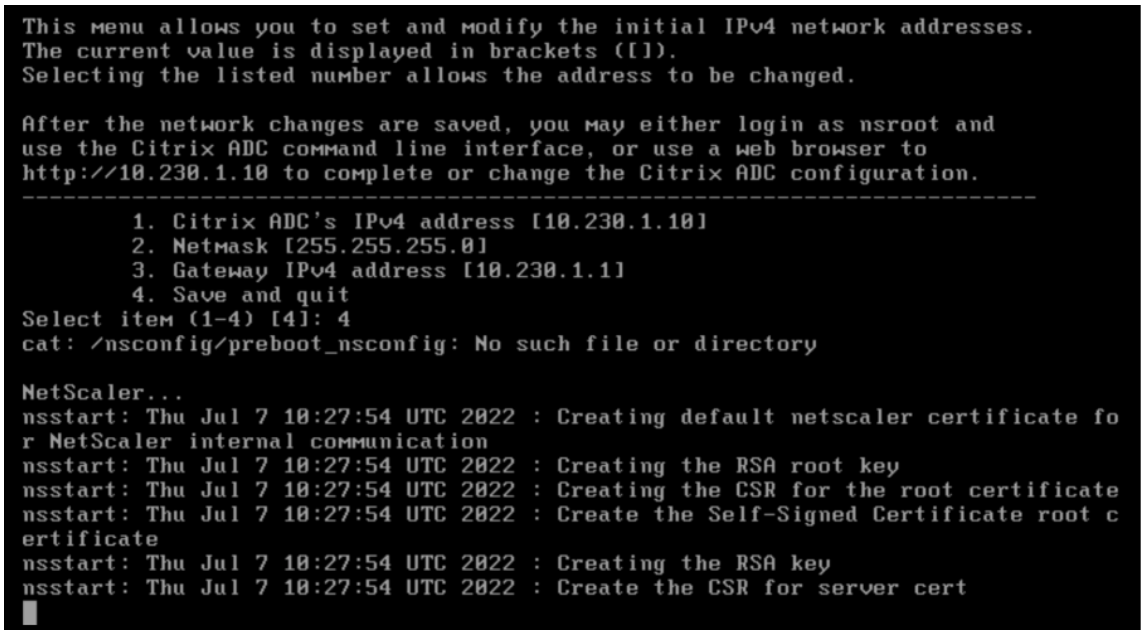
8. これで、NetScaler VPX インスタンスを起動する準備ができました。ナビゲーションペインで、インストールした Citrix ADC VPX インスタンスを選択し、右クリックメニューから［パワーオン］を選択します。コンソールポートをエミュレートするには、「**Web** コンソールの起動」タブをクリックします。



9. これで、vSphere クライアントから NetScaler 仮想マシンに接続されています。



10. 初回起動時に、ADC インスタンスの管理 IP とゲートウェイを設定します。



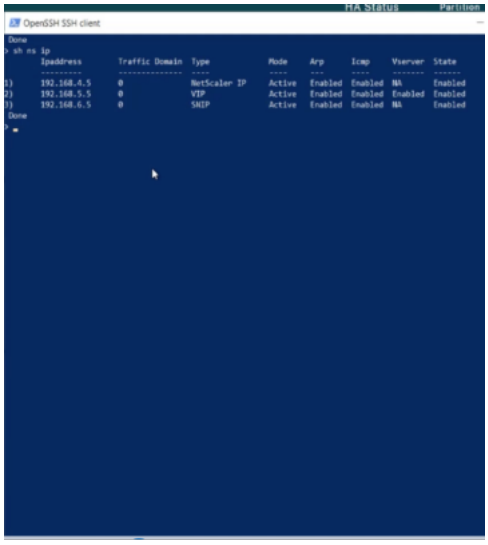
11. SSH キーを使用して NetScaler アプライアンスにアクセスするには、CLI で次のコマンドを入力します。

```
ssh nsroot@<management IP address>
```

例

```
ssh nsroot@10.230.1.10
```

12. ADC の設定は、`show ns ip`コマンドを使用して確認できます。

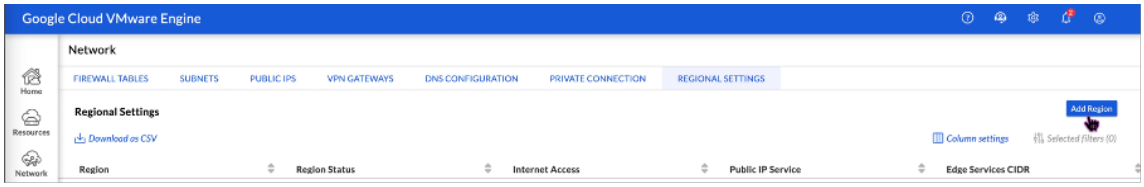


VMware クラウド上の NetScaler VPX インスタンスにパブリック IP アドレスを割り当てる

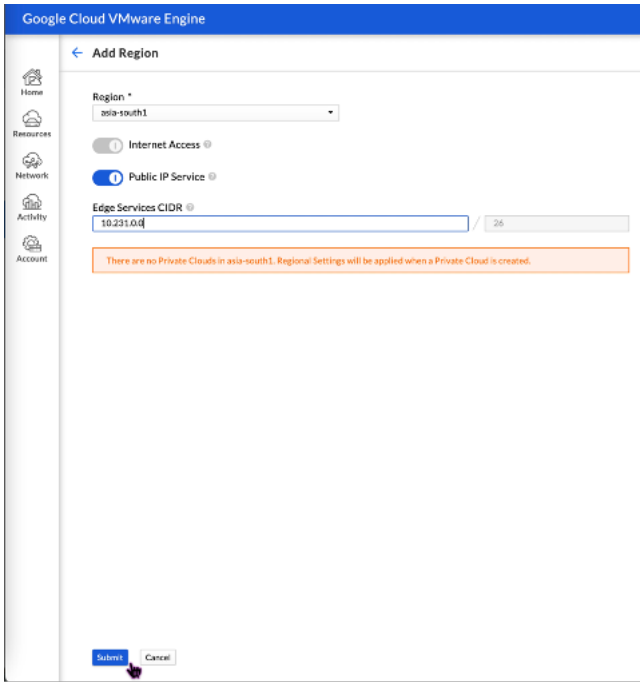
GCVE に NetScaler VPX インスタンスをインストールして構成したら、クライアントインターフェイスにパブリック IP アドレスを割り当てる必要があります。VM にパブリック IP アドレスを割り当てる前に、Google Cloud リージョンでパブリック IP サービスが有効になっていることを確認してください。

新しいリージョンのパブリック IP サービスを有効にするには、次の手順に従います。

1. GCVE コンソールで、[ネットワーク] **>** [地域設定] **>** [地域の追加] に移動します。



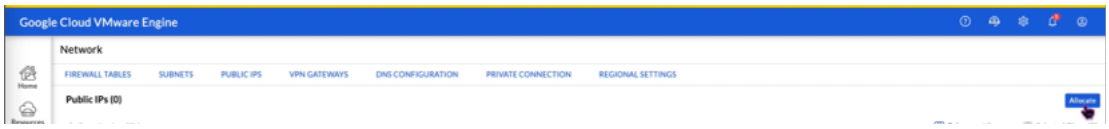
2. 地域を選択し、インターネット アクセス と パブリック **IP** サービスを有効にします。
3. エッジサービス CIDR を割り当てて、CIDR 範囲がオンプレミスまたは他の GCP/GCVE サブネット (仮想ネットワーク) と重複しないようにします。



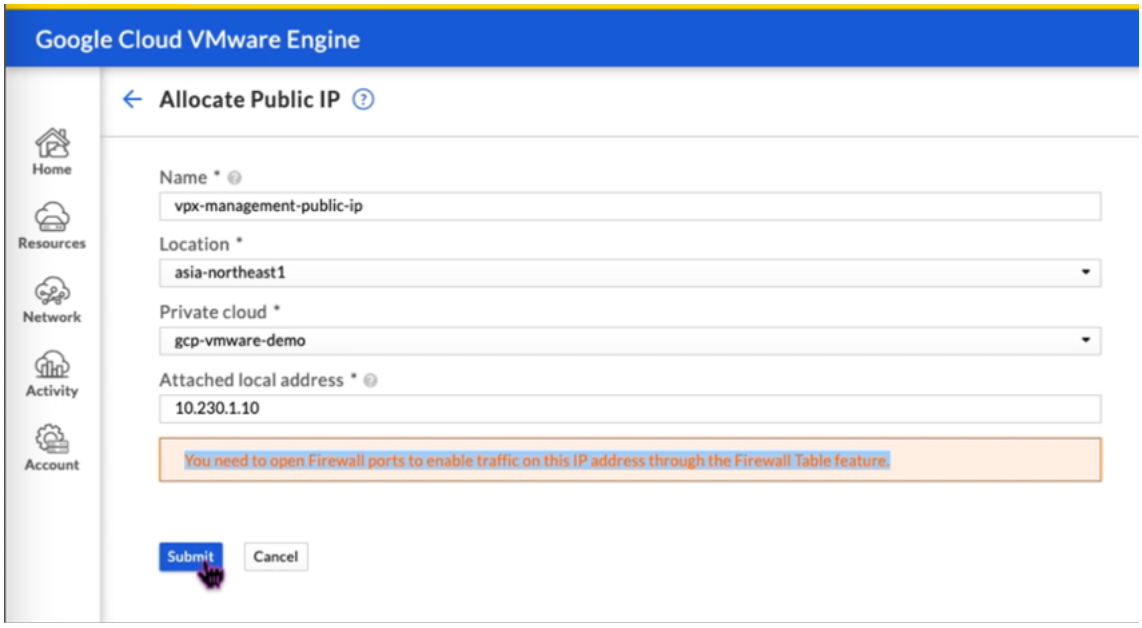
4. 数分後に、選択したリージョンのパブリック IP サービスが有効になります。

GCVE 上の NetScaler VPX インスタンスのクライアントインターフェイスにパブリック IP を割り当てるには、GCVE ポータルで以下の手順を実行します。

1. GCVE コンソールで、[ネットワーク**] > [**パブリック **IP] > [**割り当て] に移動します。



2. パブリック IP の名前を入力します。地域を選択し、IP を使用するプライベートクラウドを選択します。
3. パブリック IP をマッピングするインターフェイスのプライベート IP を指定します。** これはクライアントインターフェースのプライベート **IP** になります ** 。
4. [**Submit**] をクリックします。



5. パブリック IP は数分で使用可能になります。
6. パブリック IP を使用する前に、パブリック IP へのアクセスを許可するファイアウォール ルールを追加する必要があります。詳細については、「ファイアウォールルール」を参照してください。

バックエンドの **GCP Auto Scaling** サービスを追加する

October 17, 2024

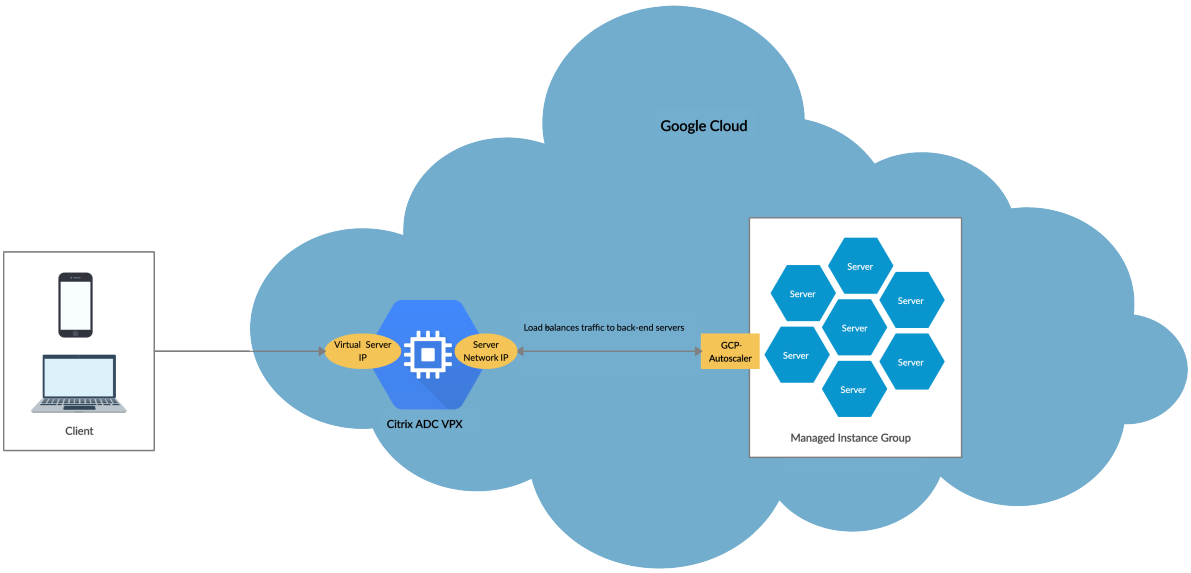
クラウドでアプリケーションを効率的にホストするには、アプリケーションの需要に応じて、簡単で費用対効果の高いリソース管理が必要です。増加する需要を満たすには、ネットワークリソースを拡大する必要があります。需要が収まったら、十分に活用されていないリソースによる不必要なコストを避けるために規模を縮小する必要があります。アプリケーションの実行コストを最小限に抑えるには、トラフィック、メモリ、CPU の使用などを常に監視する必要があります。しかし、トラフィックを手動で監視するのは煩雑です。アプリケーション環境を動的にスケールアップまたはスケールダウンするには、トラフィックの監視プロセスを自動化し、必要に応じてリソースを増減する必要があります。

GCP 自動スケーリングサービスと統合された NetScaler VPX インスタンスには、次の利点があります。

- 負荷分散と管理: 需要に応じてサーバーをスケールアップおよびスケールダウンするように自動構成します。VPX インスタンスはバックエンドサブネット内のマネージドインスタンスグループを自動検出し、負荷を分散するマネージドインスタンスグループを選択できます。仮想 IP アドレスとサブネット IP アドレスは、VPX インスタンスで自動構成されます。
- 高可用性: 複数のゾーンにまたがるマネージドインスタンスグループを検出し、サーバーの負荷を分散します。
- ネットワークの可用性の向上:VPX インスタンスは以下をサポートします。

- 同じ配置グループのバックエンドサーバー
- 異なるゾーンのバックエンドサーバー

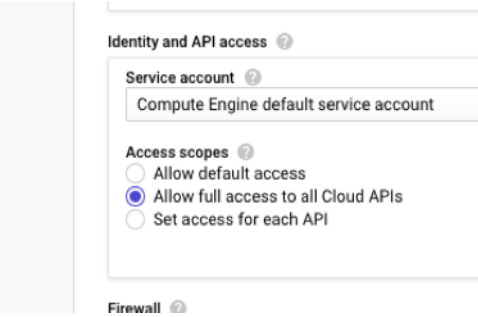
この図は、負荷分散仮想サーバーとして機能する NetScaler ADC VPX インスタンスで GCP 自動スケーリングサービスがどのように機能するかを示しています。



はじめに

NetScaler VPX インスタンスで自動スケーリングの使用を開始する前に、次のタスクを完了する必要があります。

- 要件に応じて、GCP 上に NetScaler ADC VPX インスタンスを作成します。
  - NetScaler VPX インスタンスを作成する方法の詳細については、「Google Cloud Platform に NetScaler VPX インスタンスをデプロイする」を参照してください。
  - VPX インスタンスを HA モードでデプロイする方法の詳細については、「Google Cloud Platform に VPX 高可用性ペアをデプロイする」をご覧ください。
- GCP プロジェクトで クラウドリソースマネージャー **API** を有効にします。



- インスタンスの作成中に、すべての Cloud API へのフルアクセスを許可します。
- GCP サービスアカウントに次の IAM 権限があることを確認してください。

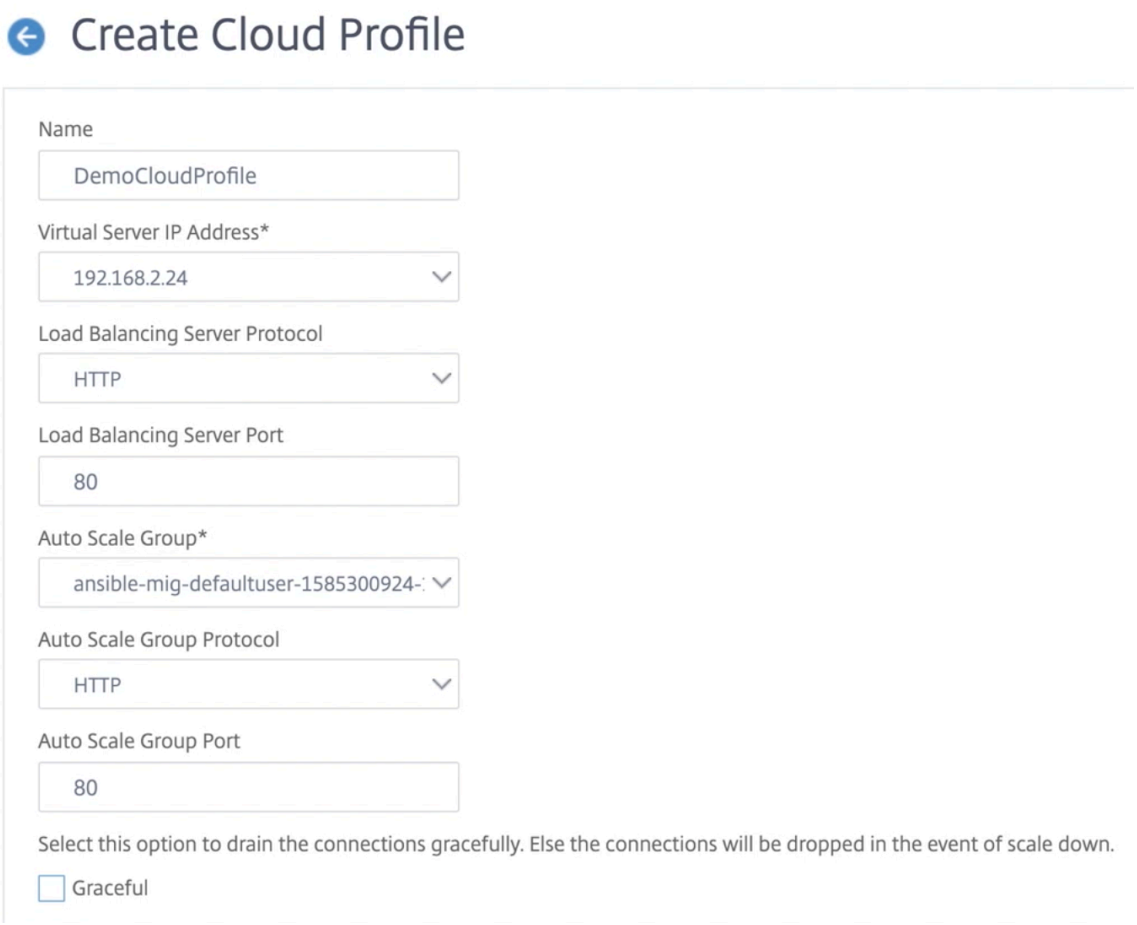

```
REQUIRED_INSTANCE_IAM_PERMS = [
"compute.instances.get",
"compute.instanceGroupManagers.get",
"compute.instanceGroupManagers.list",
"compute.zones.list",
"logging.sinks.create",
"logging.sinks.delete",
"logging.sinks.get",
"logging.sinks.list",
"logging.sinks.update",
"pubsub.subscriptions.consume",
"pubsub.subscriptions.create",
"pubsub.subscriptions.delete",
"pubsub.subscriptions.get",
"pubsub.topics.attachSubscription",
"pubsub.topics.create",
"pubsub.topics.delete",
"pubsub.topics.get",
"pubsub.topics.getIamPolicy",
"pubsub.topics.setIamPolicy",
]
```

- 自動スケーリングを設定するには、以下が設定されていることを確認してください。
  - インスタンステンプレート
  - マネージドインスタンスグループ
  - 自動スケーリングポリシー

GCP 自動スケーリングサービスを NetScaler VPX インスタンスに追加する

GUI を使用して、ワンクリックで VPX インスタンスに自動スケーリングサービスを追加できます。次の手順を実行して、VPX インスタンスに自動スケーリングサービスを追加します。

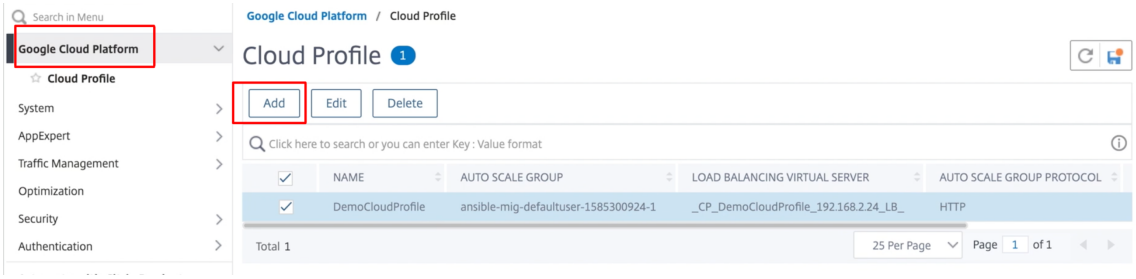
1. nsrootの認証情報を使用して VPX インスタンスにログオンします。
2. NetScaler VPX インスタンスに初めてログオンすると、デフォルトの Cloud Profile ページが表示されます。ドロップダウンメニューから GCP マネージドインスタンスグループを選択し、[作成] をクリックしてクラウドプロファイルを作成します。



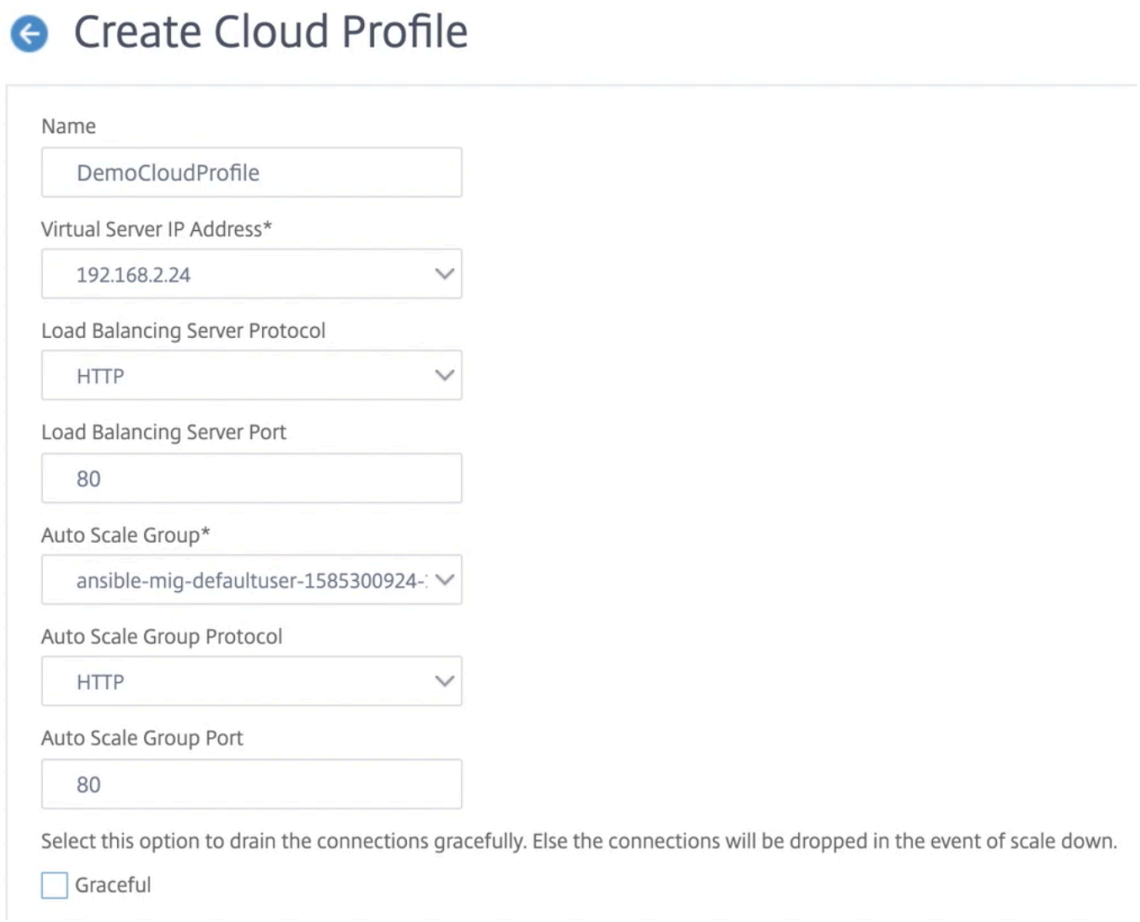
- [**Virtual Server IP Address**] フィールドは、インスタンスに関連付けられたすべての IP アドレスから自動的に入力されます。
- **Autoscale** グループは、GCP アカウントで設定されたマネージドインスタンスグループから事前設定されています。
- [自動スケールグループプロトコル] と [自動スケールグループポート] を選択するときは、サーバが構成済みのプロトコルとポートでリッスンしていることを確認します。サービスグループに適切なモニターをバインドします。デフォルトでは、TCP モニターが使用されます。
- サポートされていないため、「**Graceful**」チェックボックスはオフにします。

> 注
>
> SSL プロトコルタイプ Auto Scaling の場合、クラウドプロファイルを作成すると、証明書がないために負荷分散仮想サーバーまたはサービスグループがダウンします。証明書は、仮想サーバまたはサービスグループに手動でバインドできます。

3. 初めてログオンした後、クラウドプロファイルを作成する場合は、GUI で [システム] **> [Google Cloud Platform] > [**クラウドプロファイル] に移動し、[追加] をクリックします。

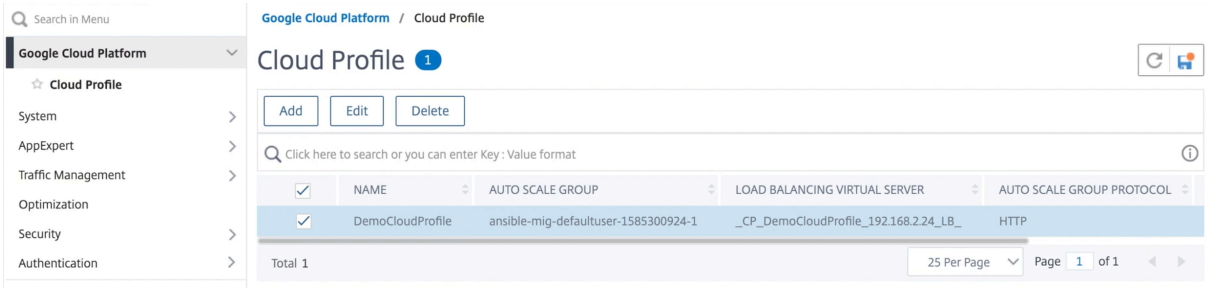


クラウドプロファイルの作成設定ページが表示されます。



Cloud Profile は、NetScaler 負荷分散仮想サーバーと、マネージドインスタンスグループのサーバーとしてメンバーを含むサービスグループを作成します。バックエンドサーバーは、VPX インスタンスで構成された SNIP を介して到達可能である必要があります。

> 注
>
> NetScaler リリース 13.1-42.x 以降では、GCP の同じマネージドインスタンスグループを使用して、（異なるポートを使用して）サービスごとに異なるクラウドプロファイルを作成できます。したがって、NetScaler VPX インスタンスは、パブリッククラウド内の同じ自動スケーリンググループを持つ複数のサービスをサポートします。



GCP 上の NetScaler VPX インスタンスの VIP スケーリングサポート

October 17, 2024

NetScaler アプライアンスはクライアントとサーバーの間に設置され、クライアント要求とサーバー応答は NetScaler アプライアンスを経由します。一般的な設置では、アプライアンス上で構成された仮想サーバーによって接続ポイントが提供され、クライアントはこれを使用してアプライアンスの背後にあるアプリケーションにアクセスします。展開に必要なパブリック仮想 IP（VIP）アドレスの数は、ケースバイケースで異なります。

GCP アーキテクチャでは、インスタンスの各インターフェイスが異なる VPC に接続されるように制限します。GCP 上の VPC はサブネットの集合であり、各サブネットはリージョンのゾーンにまたがることができます。さらに、GCP には次の制限があります。

- パブリック IP アドレス数と NIC の数が 1:1 でマッピングされています。NIC に割り当てることができるパブリック IP アドレスは 1 つだけです。
- 大容量のインスタンスタイプには最大 8 つの NIC しか接続できません。

たとえば、n1-standard-2 インスタンスは 2 つの NIC しか持つことができず、追加できるパブリック VIP は 2 つに制限されています。詳細については、「VPC リソースクォータ」を参照してください。

NetScaler VPX インスタンスでより大規模なパブリック仮想 IP アドレスを実現するために、インスタンスのメタデータの一部として VIP アドレスを構成できます。NetScaler VPX インスタンスは、GCP が提供する転送ルールを内部で使用して VIP スケーリングを実現します。NetScaler VPX インスタンスは、構成された VIP に高可用性も提供します。ADC VPX インスタンスは、構成された VIP に高可用性も提供します。メタデータの一部として VIP アドレスを設定した後、転送ルールの作成に使用するのと同じ IP を使用して LB 仮想サーバを設定できます。そのため、転送ルールを使用することで、GCP 上の NetScaler VPX インスタンスでパブリック VIP アドレスを使用する際のスケーリング上の制限を緩和できます。

転送ルールの詳細については、「転送ルールの概要」を参照してください。

HA の詳細については、「高可用性」を参照してください。

注意事項

- Google は、各仮想 IP 転送ルールに対して追加費用を請求します。実際のコストは、作成されるエントリの数によって異なります。関連するコストは、Google の価格設定ドキュメントから確認できます。
- 転送ルールは、パブリック VIP にのみ適用されます。展開でプライベート IP アドレスが VIP として必要な場合は、エイリアス IP アドレスを使用できます。
- 転送ルールは、LB 仮想サーバーを必要とするプロトコルに対してのみ作成できます。VIP は、その場で作成、更新、または削除できます。同じ VIP アドレスを持つが、プロトコルが異なる新しい負荷分散仮想サーバーを追加することもできます。

はじめに

- NetScaler VPX インスタンスは、GCP にデプロイする必要があります。
- 外部 IP アドレスは予約する必要があります。詳細については、「静的外部 IP アドレスの予約」を参照してください。
- GCP サービスアカウントに次の IAM 権限があることを確認します。

```
REQUIRED_IAM_PERMS = [
"compute.addresses.list",
"compute.addresses.get",
"compute.addresses.use",
"compute.forwardingRules.create",
"compute.forwardingRules.delete",
"compute.forwardingRules.get",
"compute.forwardingRules.list",
"compute.instances.use",
"compute.subnetworks.use",
"compute.targetInstances.create"
"compute.targetInstances.get"
"compute.targetInstances.use",
]
```

- GCP プロジェクトで クラウドリソースマネージャー **API** を有効にします。
- スタンドアロン VPX インスタンスで VIP スケーリングを使用する場合は、GCP サービスアカウントに次の IAM 権限があることを確認してください。

```
REQUIRED_IAM_PERMS = [
"compute.addresses.list",
"compute.addresses.get",
"compute.addresses.use",
"compute.forwardingRules.create",
"compute.forwardingRules.delete",
"compute.forwardingRules.get",
"compute.forwardingRules.list",
"compute.instances.use",
"compute.subnetworks.use",
```



```
    "compute.targetInstances.create",
    "compute.targetInstances.list",
    "compute.targetInstances.use",
    ]
```

- 高可用性モードで VIP スケーリングを使用する場合は、GCP サービスアカウントに次の IAM 権限があることを確認してください。

```
    REQUIRED_IAM_PERMS = [
    "compute.addresses.get",
    "compute.addresses.list",
    "compute.addresses.use",
    "compute.forwardingRules.create",
    "compute.forwardingRules.delete",
    "compute.forwardingRules.get",
    "compute.forwardingRules.list",
    "compute.forwardingRules.setTarget",
    "compute.instances.use",
    "compute.instances.get",
    "compute.instances.list",
    "compute.instances.setMetadata",
    "compute.subnetworks.use",
    "compute.targetInstances.create",
    "compute.targetInstances.list",
    "compute.targetInstances.use",
    "compute.zones.list",
    ]
```

> 注
>
> 高可用性モードでは、サービスアカウントに所有者または編集者の役割がない場合は、サービスアカウントにサービスアカウントユーザーの役割を追加する必要があります。

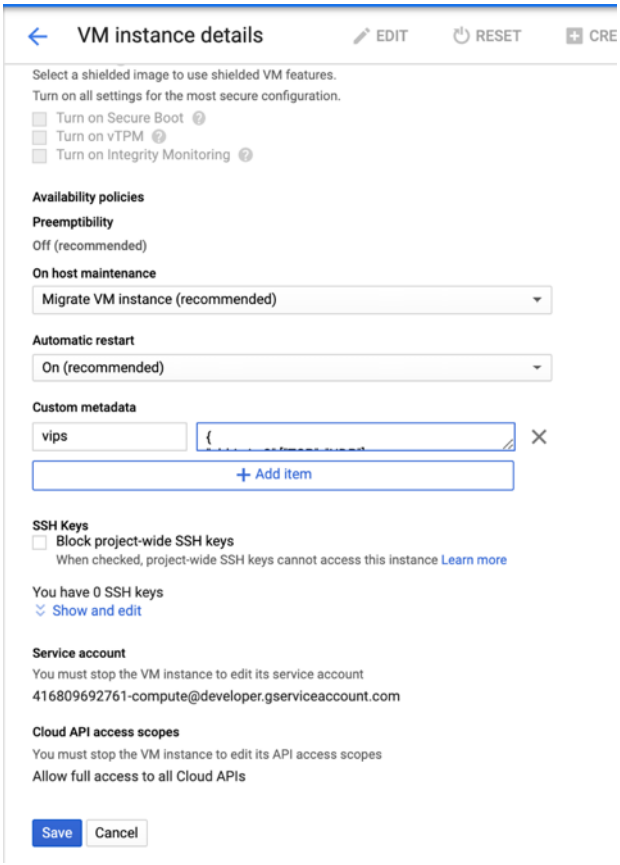
NetScaler VPX インスタンスでの **VIP** スケーリング用の外部 **IP** アドレスを構成する

1. Google Cloud コンソールで、[**VM** インスタンス] ページに移動します。
2. 新しい VM インスタンスを作成するか、既存のインスタンスを使用します。
3. インスタンス名をクリックします。**VM** インスタンスの詳細ページで、[編集] をクリックします。
4. 次のように入力して、カスタムメタデータを更新します。
   - キー = VIP
   - 値 = 次の JSON 形式で値を指定します。

     { 「外部予約 IP の名前」: [プロトコルのリスト], }

   GCP は次のプロトコルをサポートしています。
   - AH

- ESP
- ICMP
- SCT
- TCP
- UDP



詳細については、「カスタムメタデータ」を参照してください。

カスタムメタデータの例:

{ “外部 IP1 名”:[“TCP”, “UDP”], “外部 IP2 名”:[“ICMP”, “AH”] }

この例では、NetScaler VPX インスタンスは、IP、プロトコルのペアごとに 1 つの転送ルールを内部で作成します。メタデータエントリは、転送ルールにマッピングされます。この例では、メタデータエントリに対して作成される転送ルールの数を把握するのに役立ちます。

次の 4 つの転送ルールが作成されます。

a) external-ip1-name と TCP
b) external-ip1-name と UDP
c) external-ip2-name と ICMP
d) external-ip2-name と AH

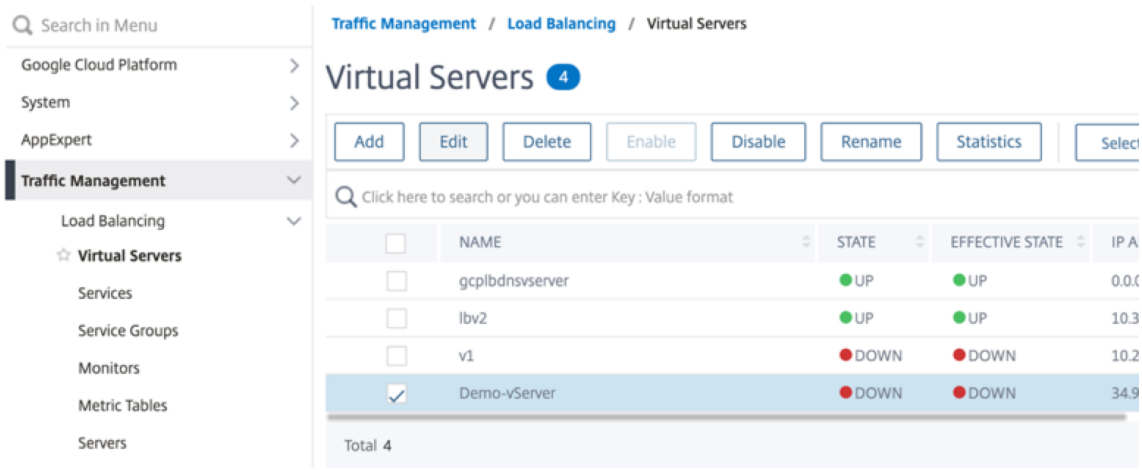
> 注
>
> HA モードでは、プライマリインスタンスにのみカスタムメタデータを追加する必要があります。フェールオーバー時に、カスタムメタデータが新しいプライマリに同期されます。

5. ［保存］をクリックします。

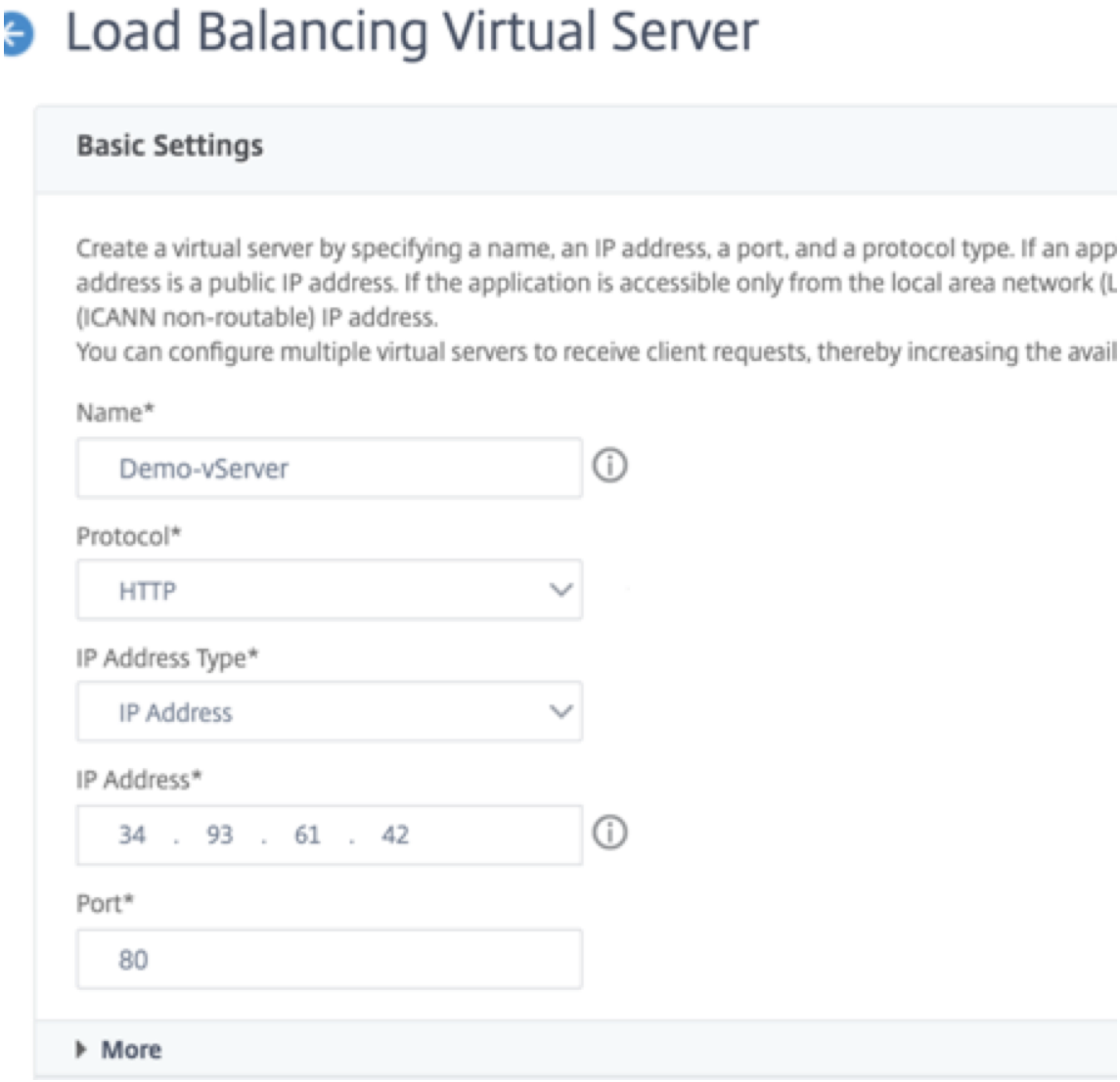
NetScaler VPX インスタンスで外部 **IP** アドレスを使用した負荷分散仮想サーバーのセットアップ

ステップ **1.** 負荷分散仮想サーバーを追加します。

1. ［設定**] > [**トラフィック管理**] > [**負荷分散**] > [**仮想サーバー**] > [**追加**]** に移動します。

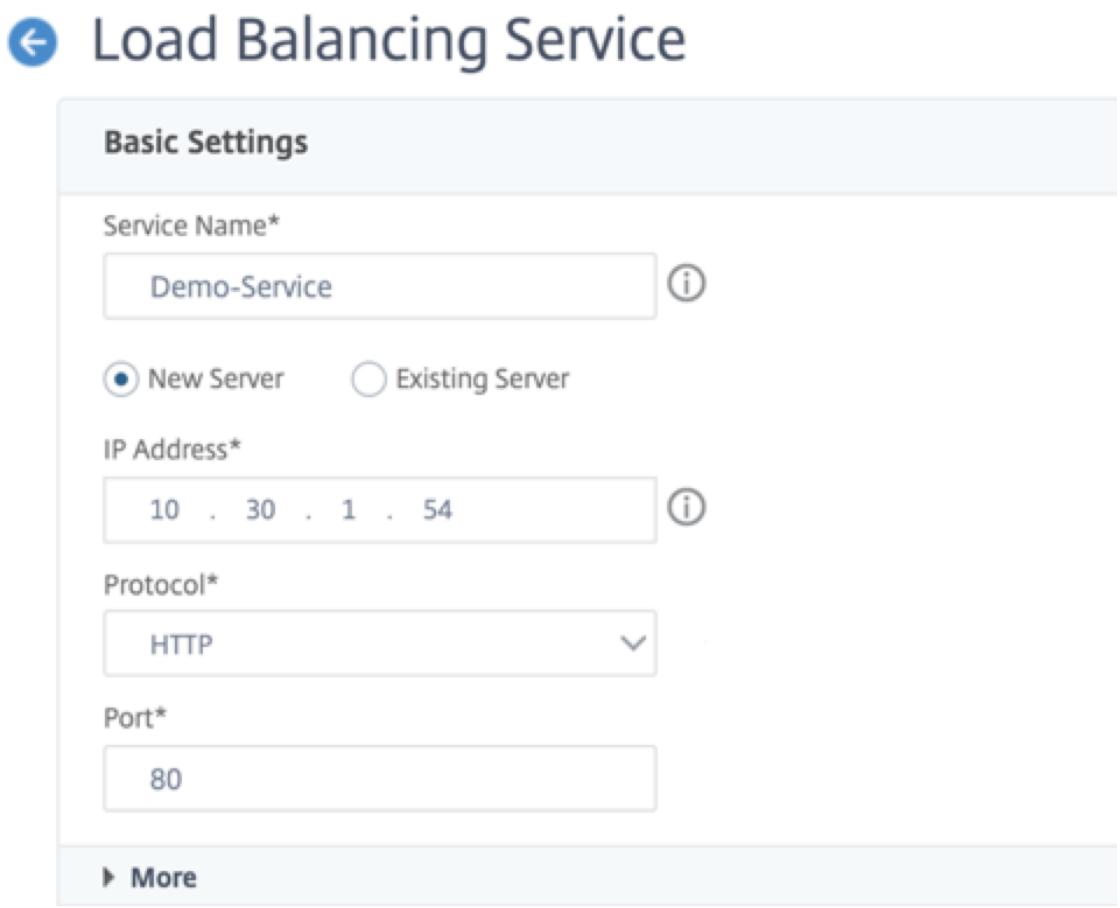


2. 名前、プロトコル、IP アドレスタイプ（IP アドレス）、IP アドレス（ADC で VIP として追加される転送ルールの外部 IP アドレス）、およびポートに必要な値を追加し、「**OK**」をクリックします。



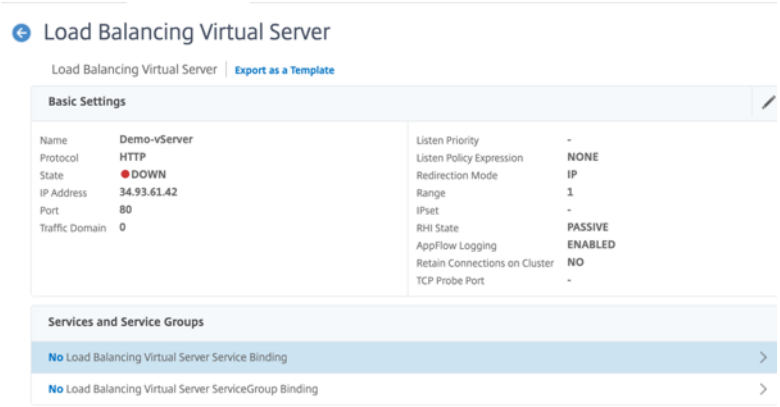
ステップ **2**. サービスまたはサービス グループを追加します。

1. **[**設定**] > [**トラフィック管理**] > [**負荷分散**] > [**サービス**] >[**追加**]** に移動します。
2. サービス名、IP アドレス、プロトコル、およびポートに必要な値を追加し、[**OK**] をクリックします。

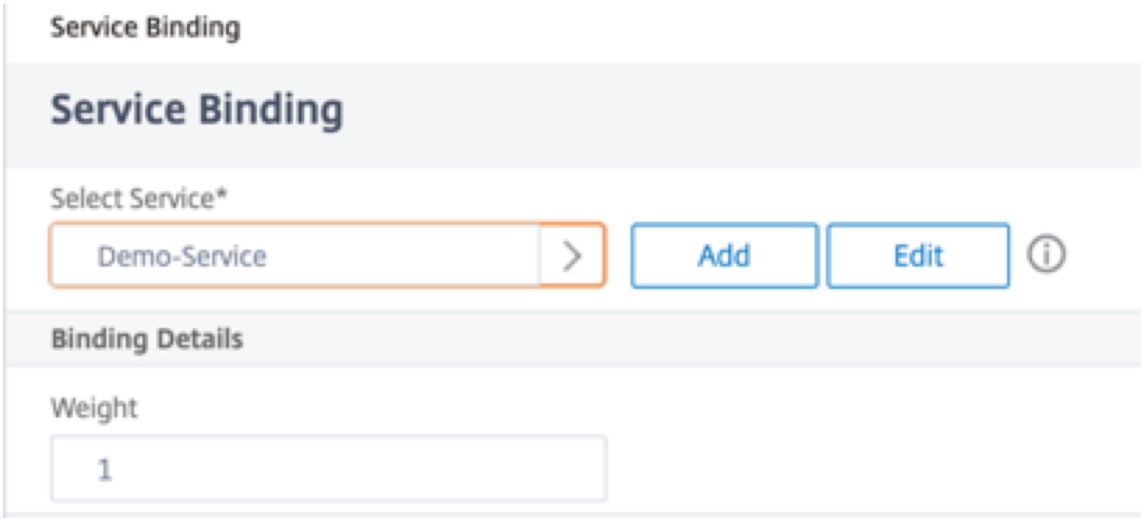


ステップ **3:** プライマリ・インスタンスに仮想サーバを追加します。サービスまたはサービス グループを負荷分散仮想サーバーにバインドします。

1. **[**設定**] > [**トラフィック管理**] > [**負荷分散**] > [**仮想サーバー**]** に移動します。
2. 手順 **1** で構成した負荷分散仮想サーバーを選択し、[編集] をクリックします。
3. [サービスとサービスグループ] ページで、[負荷分散仮想サーバーサービスのバインドなし] をクリックします。



4. 手順 **3** で構成したサービスを選択し、[バインド] をクリックします。



5. 構成を保存します。

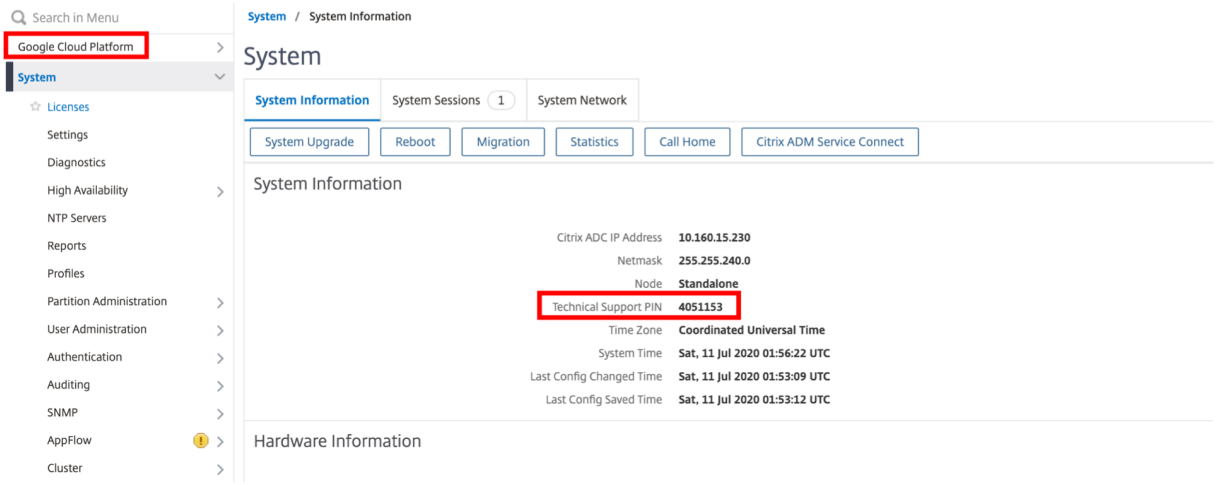
GCP での VPX インスタンスのトラブルシューティング

October 17, 2024

Google Cloud Platform （GCP）は、NetScaler VPX インスタンスへのコンソールアクセスを提供します。デバッグできるのは、ネットワークが接続されている場合だけです。インスタンスのシステムログを表示するには、コンソールにアクセスし、システムログファイルを確認します。

NetScaler は、GCP 上で有料の NetScaler VPX インスタンス（時間単位のユーティリティライセンス）をサポートします。サポートケースを提出するには、GCP アカウント番号とサポート PIN コードを見つけて、NetScaler サポートに連絡してください。名前と E メールアドレスの入力を求められます。サポート PIN を検索するには、VPX GUI にログオンし、［システム］ページに移動します。

サポート PIN を示すシステムページの例を次に示します。



NetScaler VPX インスタンスのジャンボフレーム

October 17, 2024

NetScaler VPX アプライアンスは、最大 9216 バイトの IP データを含むジャンボフレームの送受信をサポートしています。ジャンボフレームでは、標準の IP MTU サイズ（1500 バイト）を使用するよりも効率的に大きなファイルを送信することができます。

NetScaler アプライアンスは、以下の展開シナリオでジャンボフレームを使用することができます。

- ジャンボからジャンボへ。アプライアンスがデータをジャンボフレームで受信し、それを通常のフレームで送信します。
- 非ジャンボからジャンボへ。アプライアンスはデータを通常のフレームとして受信し、ジャンボ フレームとして送信します。
- ジャンボから非ジャンボへ。アプライアンスがデータを通常のフレームで受信し、それをジャンボフレームで送信します。

詳細については、「Citrix ADC アプライアンスでのジャンボフレームサポートの構成」を参照してください。

ジャンボフレームのサポートは、次の仮想化プラットフォームで実行されている NetScaler ADC VPX アプライアンスで利用できます。

- VMware ESX
- Linux-KVM プラットフォーム
- Citrix XenServer
- Amazon Web Services（AWS）

VPX アプライアンスのジャンボフレームは、MPX アプライアンスのジャンボフレームと同様に機能します。ジャンボフレームおよびそのユースケースについて詳しくは、「MPX アプライアンスでのジャンボフレームの構成」を参照してください。MPX アプライアンスのジャンボフレームの使用例は、VPX アプライアンスにも当てはまります。

VMware ESX で実行中の **VPX** インスタンスのジャンボフレームを構成する

VMware ESX サーバーで実行されている NetScaler ADC VPX アプライアンスでジャンボフレームを構成するには、次のタスクを実行します。

1. VPX アプライアンスのインターフェイスまたはチャネルの MTU を 1501～9000 の範囲の値に設定します。CLI または GUI を使用して MTU サイズを設定します。VMware ESX 上で動作する NetScaler VPX アプライアンスは、最大 9000 バイトの IP データのみを含むジャンボフレームの送受信をサポートします。
2. 管理アプリケーションを使用して、VMware ESX サーバーの対応する物理インターフェイスで同じ MTU サイズを設定します。VMware ESX の物理インターフェイスでの MTU サイズの設定の詳細については、http://vmware.com/を参照してください。

Linux-KVM サーバーで実行されている **VPX** インスタンスのジャンボフレームを構成する

Linux-KVM サーバーで実行されている NetScaler VPX アプライアンスでジャンボフレームを構成するには、次のタスクを実行します。

1. VPX アプライアンスのインターフェイスまたはチャネルの MTU を 1501～9216 の範囲の値に設定します。NetScaler VPX CLI または GUI を使用して MTU サイズを設定します。
2. 管理アプリケーションを使用して、Linux-KVM サーバーの対応する物理インターフェイスで同じ MTU サイズを設定します。Linux-KVM の物理インターフェイスの MTU サイズの設定の詳細については、http://www.linux-kvm.org/を参照してください。

Citrix XenServer 上で実行されている **VPX** インスタンスのジャンボフレームを構成する

Citrix XenServer で実行されている NetScaler VPX アプライアンスでジャンボフレームを構成するには、次のタスクを実行します。

1. XenCenter を使用して XenServer に接続します。
2. MTU を変更する必要があるネットワークを使用するすべての VPX インスタンスをシャットダウンします。
3. [ネットワーク] タブで、ネットワーク-[ネットワーク 0/1/2] を選択します。
4. [プロパティ] を選択し、MTU を編集します。

XenServer でジャンボフレームを構成した後、ADC アプライアンスでジャンボフレームを構成できます。詳細については、「Citrix ADC アプライアンスでのジャンボフレームサポートの構成」を参照してください。

AWS で実行中の **VPX** インスタンスのジャンボフレームを設定する

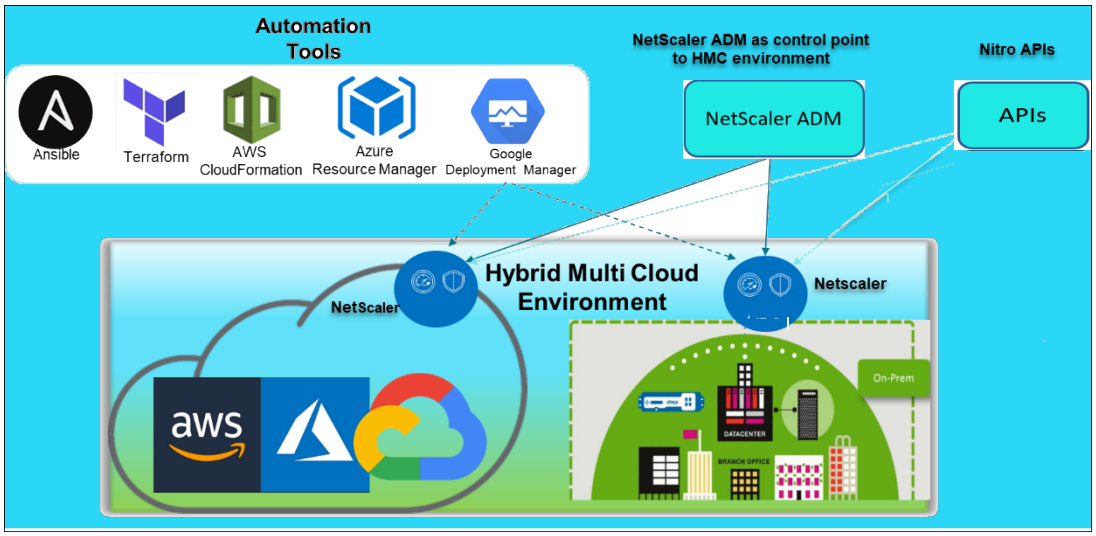
Azure 上の VPX では、ホストレベルの構成は不要です。VPX でジャンボフレームを構成するには、Citrix ADC アプライアンスでのジャンボフレームサポートの構成に記載されている手順に従います。

NetScaler の導入と構成を自動化する

October 17, 2024

NetScaler には、ADC の展開と構成を自動化するための複数のツールが用意されています。このドキュメントでは、さまざまな自動化ツールの概要と、ADC 構成の管理に使用できるさまざまな自動化リソースの参照について説明します。

次の図は、ハイブリッドマルチクラウド（HMC）環境における NetScaler 自動化の概要を示しています。



NetScaler ADM を使用して NetScaler を自動化する

NetScaler ADM は、分散 ADC インフラストラクチャへの自動化制御ポイントとして機能します。NetScaler ADM は、ADC アプライアンスのプロビジョニングからアップグレードまで、包括的な自動化機能セットを提供します。ADM の主な自動化機能は次のとおりです：

- AWS での NetScaler VPX インスタンスのプロビジョニング
- Azure での NetScaler VPX インスタンスのプロビジョニング
- StyleBooks
- 構成ジョブ
- 構成監査
- ADC アップグレード
- SSL 証明書の管理
- 統合- GitHub、ServiceNow、イベント通知の統合

NetScaler ADM の自動化に関するブログとビデオ

- StyleBooks を使用したアプリケーションの移行

- ADM スタイルブックを使用して ADC 構成を CI/CD と統合する
- ADM によるパブリッククラウドの NetScaler 導入の簡素化
- NetScaler ADM サービスが NetScaler のアップグレードを容易にする 10 の方法

NetScaler ADM は、全体的な IT 自動化の一環として NetScaler ADM と NetScaler を統合するさまざまな機能用の API も提供します。詳細については、「NetScaler ADM サービス API」を参照してください。

Terraform を使用して **NetScaler** を自動化する

Terraform は、クラウド、インフラストラクチャ、またはサービスのプロビジョニングと管理に、インフラストラクチャをコードアプローチとして採用するツールです。NetScaler テラフォームリソースは、GitHub で使用できます。詳細なドキュメントと使用法については、GitHub を参照してください。

- NetScaler Terraform モジュールにより、負荷分散や GSLB などのさまざまなユースケースに合わせて ADC を構成できます
- AWS に ADC をデプロイするための Terraform クラウドスクリプト
- Azure に ADC をデプロイするための Terraform クラウドスクリプト
- Terraform クラウドスクリプトで ADC を GCP にデプロイ
- NetScaler VPX および Azure パイプラインを使用したブルーグリーン展開

Terraform の **ADC** 自動化に関するブログとビデオ

- Terraform で NetScaler の展開を自動化
- Terraform を使用した AWS の HA セットアップで ADC をプロビジョニングおよび設定する

領事**-Terraform-Sync** を使用して **NetScaler** を自動化する

NetScaler Consul-Terraform-Sync（CTS）モジュールにより、アプリケーションチームはサービスの新しいインスタンスを NetScaler に自動的に追加または削除できます。必要な ADC 構成の変更を行うために、IT 管理者やネットワーキングチームに手動でチケットを提出する必要はありません。

- ネットワークインフラストラクチャ自動化のための NetScaler 領事 Terraform-Sync モジュール
- Citrix-HashiCorp 共同ウェビナー: Terraform Enterprise および NetScaler 向けの領事 Terraform-Sync を使用した動的ネットワーキング

Ansible を使用して **NetScaler** を自動化する

Ansible は、インフラストラクチャをコードとして実現する、オープンソースのソフトウェアプロビジョニング、構成管理、およびアプリケーションデプロイメントツールです。NetScaler Ansible モジュールとサンプルプレイブックは、GitHub にあります。詳細なドキュメントと使用法については、GitHub を参照してください。

- ADC を構成するための Ansible モジュール
- ADC Ansible モジュールドキュメント/リファレンスガイド
- ADM 用の Ansible モジュール

Citrix は認定された AnsibleAutomation パートナーです。Red Hat Ansible オートメーションプラットフォームのサブスクリプションをお持ちのユーザーは、Red Hat オートメーションハブから NetScalerコレクションにアクセスできます。

Terraform と **Ansible** の自動化ブログ

- Citrix、HashiCorp 統合パートナー・オブ・ザ・イヤーに選出
- Citrix は Red Hat Ansible オートメーションプラットフォーム認定パートナーになりました
- アプリケーションの配信とセキュリティのための Terraform と Ansible Automation

ADC 展開用のパブリッククラウドテンプレート

パブリッククラウドテンプレートは、パブリッククラウドでのデプロイメントのプロビジョニングを簡素化します。さまざまな環境で、さまざまな NetScaler テンプレートを使用できます。使用方法の詳細については、それぞれの GitHub リポジトリを参照してください。

AWS CFT:

- AWS で NetScaler VPX をプロビジョニングするための CFT

Azure Resource Manager (ARM) テンプレート**:**

- Azure で NetScaler VPX をプロビジョニングするための ARM テンプレート

Google Cloud デプロイメントマネージャー **(GDM)** テンプレート**:**

- Google で NetScaler VPX をプロビジョニングするための GDM テンプレート

テンプレートに関する動画

- クラウドフォーメーションテンプレートを使用して NetScaler HA を AWS にデプロイ
- AWS クイックスタートを使用してアベイラビリティーゾーン全体に NetScaler HA
- GDM テンプレートを使用した GCP での NetScaler HA の展開

NITRO API

NetScaler NITRO プロトコルを使用すると、表現状態転送（REST）インターフェイスを使用して、NetScaler アプライアンスをプログラムで構成および監視できます。そのため、NITRO アプリケーションはあらゆるプログラミング言語で開発することができます。Java、.NET、または Python で開発する必要があるアプリケーションの場合、NITRO API は、個別のソフトウェア開発キット (SDK) としてパッケージ化された関連ライブラリを通じて公開されます。

- NITRO API ドキュメント
- NITRO API を使用した ADC ユースケースの設定例

よくある質問

October 17, 2024

次のセクションでは、Citrix アプリケーション Delivery Controller（ADC）VPX に基づいて FAQ を分類するのに役立ちます。

- 機能と機能
- 暗号化
- 価格設定と梱包
- NetScaler VPX Express と 90 日間の無料トライアル
- ハイパーバイザー
- キャパシティプランニングまたはサイジング
- システム要件
- その他の技術的なよくある質問

機能と機能

NetScaler VPX とは何ですか**?**

NetScaler VPX は、業界標準のサーバーにインストールされた Hypervisor でホストできる仮想 ADC アプライアンスです。

NetScaler VPX には、すべての **Web** アプリケーション最適化機能が **ADC** アプライアンスとして含まれていますか

はい。NetScaler VPX には、すべての負荷分散、トラフィック管理、アプリケーションアクセラレーション、アプリケーションセキュリティ（NetScaler Gateway および Citrix アプリケーションファイアウォールを含む）、およびオフロード機能が含まれています。NetScaler の機能と機能の完全な概要については、「アプリケーションの配信方法」を参照してください。

Citrix アプリケーションファイアウォールを **NetScaler VPX** で使用する場合、制限はありますか**?**

NetScaler VPX 上の Citrix アプリケーションファイアウォールは、NetScaler アプライアンスと同じセキュリティ保護を提供します。Citrix アプリケーションファイアウォールのパフォーマンスまたはスループットは、プラットフォームによって異なります。

NetScaler VPX 上の **NetScaler Gateway** と **NetScaler** アプライアンスの **NetScaler Gateway** の間に違いはありますか**?**

機能的には同じです。NetScaler VPX 上の NetScaler Gateway は、NetScaler ソフトウェアリリース 14.1 で利用可能なすべての NetScaler Gateway 機能をサポートします。ただし、NetScaler アプライアンスは専用の SSL アクセラレーションハードウェアを提供するため、NetScaler VPX インスタンスよりも優れた SSL VPN スケーラビリティを提供します。

NetScaler VPX はハイパーバイザー上で実行できるという明らかな違い以外に、**NetScaler** の物理アプライアンスとどう違うのですか

顧客に行動の違いが見られる主な領域は 2 つあります。1 つ目は、NetScaler VPX は多くの NetScaler アプライアンスと同じパフォーマンスを提供できないことです。2 つ目は、NetScaler アプライアンスは独自の L2 ネットワーク機能を組み込んでいるが、NetScaler VPX は L2 ネットワークサービスのために Hypervisor に依存しているということです。一般的に、NetScaler VPX の展開方法は制限されません。物理 NetScaler アプライアンスに構成されている特定の L2 機能は、基盤となる Hypervisor で構成する必要があります。

NetScaler VPX は、アプリケーションデリバリー市場でどのように役割を果たしていますか**?**

NetScaler VPX は、アプリケーション配信市場のゲームを次のように変えます。

- NetScaler アプライアンスをさらに手頃な価格にすることで、NetScaler VPX は、あらゆる IT 組織が NetScaler アプライアンスを展開できるようにします。これは、最もミッションクリティカルな Web アプリケーションだけでなく、すべての Web アプリケーション用です。

- NetScaler VPX を使用すると、データセンター内でネットワーキングと仮想化をさらに統合できます。NetScaler VPX は、仮想サーバーでホストされている Web アプリケーションを最適化するためだけに使用することはできません。また、Web アプリケーションの配信自体を、どこでも簡単かつ迅速に展開できる仮想化サービスにすることができます。IT 組織は、Web アプリケーション配信インフラストラクチャのプロビジョニング、自動化、チャージバックなどのタスクに標準的なデータセンタープロセスを使用します。
- NetScaler VPX は、物理アプライアンスだけを使用する場合は実用的ではない新しい展開アーキテクチャを開きます。NetScaler VPX および NetScaler MPX アプライアンスは、圧縮やアプリケーションファイアウォール検査などのプロセッサ負荷の高いアクションを処理するために、各アプリケーションの個々のニーズに合わせてベースで使用できます。データセンターエッジでは、NetScaler MPX アプライアンスは、初期トラフィック分散、SSL 暗号化または復号化、サービス拒否（DoS）攻撃防止、グローバル負荷分散など、大量のネットワーク全体のタスクを処理します。高性能の NetScaler MPX アプライアンスと展開しやすい NetScaler VPX 仮想アプライアンスを組み合わせることで、データセンター全体のコストを削減しながら、最新の大規模データセンター環境に比類のない柔軟性とカスタマイズ機能を提供します。

NetScaler VPX はシトリック **Citrix** デリバリーセンター戦略にどのように適合していますか

NetScaler VPX を利用することで、Citrix デリバリーセンターの全サービスを仮想化されたサービスとして利用できます。Citrix XenCenter で利用可能な強力な管理、プロビジョニング、監視、およびレポート機能によって、Citrix デリバリーセンター全体がメリットを得られます。これは、ほぼすべての環境に迅速に導入でき、どこからでも一元的に管理できます。1 つの統合された仮想化アプリケーション配信インフラストラクチャにより、組織はデスクトップ、クライアント/サーバーアプリケーション、Web アプリケーションを配信できます。

暗号化

NetScaler VPX は **SSL** オフロードをサポートしていますか**?**

はい。ただし、NetScaler VPX はすべての SSL 処理をソフトウェアで行うため、NetScaler VPX は NetScaler アプライアンスと同じ SSL パフォーマンスを提供しません。NetScaler VPX は、毎秒最大 750 の新しい SSL トランザクションをサポートできます。

NetScaler VPX をホストするサーバーにインストールされているサードパーティの **SSL** カードは、**SSL** 暗号化または復号化を高速化しますか**?**

いいえ。NetScaler VPX ライセンスは、基盤となる Hypervisor から独立しています。NetScaler VPX 仮想マシンをある Hypervisor から別の Hypervisor に移動する場合は、新しいライセンスを取得する必要はありません。ただし、既存の NetScaler VPX ライセンスを再ホストする必要がある場合があります。

NetScaler VPX は、物理的な **NetScaler** アプライアンスと同じ暗号化暗号をサポートしていますか**?**

VPX は、ECDSA を除くすべての暗号化暗号を物理 NetScaler アプライアンスとしてサポートします。

NetScaler VPX SSL トランザクションスループットとは何ですか**?**

SSL トランザクションのスループットについては、NetScaler VPX のデータシートを参照してください 。

価格設定と梱包

NetScaler VPX はどのようにパッケージ化されていますか

NetScaler VPX の選択は、NetScaler アプライアンスの選択に似ています。まず、お客様は、機能要件に基づいて NetScaler エディションを選択します。次に、スループット要件に基づいて、特定の NetScaler VPX 帯域幅層を選択します。NetScaler VPX は、スタンダード、アドバンスエディション、およびプレミアムエディションで利用できます。NetScaler VPX は、10Mbps（VPX 10）から 100Gbps（VPX 100G）まで対応している。詳細については、NetScaler VPX のデータシートを参照してください。

NetScaler VPX の価格はすべての **Hypervisor** で同じですか**?**

はい。

すべての **Hypervisor** で **VPX** に同じ **NetScaler SKU** が使用されていますか**?**

はい。

NetScaler VPX ライセンスをある **Hypervisor** から別の **Hypervisor** に移動できますか（たとえば、**VMware** から **Hyper-V** へ）**?**

はい。NetScaler VPX ライセンスは、基盤となるハイパーバイザーとは独立しています。NetScaler VPX 仮想マシンをあるハイパーバイザーから別のハイパーバイザーに移動する場合、新しいライセンスを取得する必要はありません。ただし、既存の NetScaler VPX ライセンスを再ホストする必要がある場合があります。

NetScaler VPX インスタンスはアップグレードできますか**?**

はい。スループット制限と NetScaler ファミリエディションの両方をアップグレードできます。両方のタイプのアップグレードのアップグレード SKU が利用可能です。

NetScaler VPX を高可用性ペアに展開する場合、必要なライセンスはいくつですか

NetScaler 物理アプライアンスと同様に、NetScaler 高可用性構成には 2 つのアクティブなインスタンスが必要です。したがって、お客様は 2 つのライセンスを購入する必要があります。

NetScaler VPX Express と 90 日間の無料トライアル

NetScaler VPX Express には、**NetScaler** 標準機能がすべて含まれていますか? **NetScaler Gateway** と、**Citrix Virtual Apps Web** インターフェイスと **XML** ブローカーの負荷分散が含まれていますか

はい。NetScaler VPX Express には、NetScaler Premium の完全な機能が含まれています。NetScaler リリース 14.1～29.65 以降、NetScaler は VPX Express の動作を変更しました。

NetScaler VPX Express にはライセンスが必要ですか

最新の NetScaler VPX Express リリース (14.1～29.65 以降) では、VPX Express は無料で使用でき、インストールや使用にライセンス ファイルは必要ありません。いかなる約束も必要ありません。すでに VPX Express ライセンスをお持ちの場合は、以前のライセンス動作が引き続き有効になります。ただし、既存の VPX Express ライセンス ファイルを削除し、バージョン 14.1～29.65 以降を使用すると、更新された VPX Express の動作が適用されます。

NetScaler VPX Express ライセンスは期限切れになりますか

新しい VPX Express にはライセンスも有効期限もありません。すでに VPX エクスプレス ライセンスをお持ちの場合、ライセンスはダウンロード後 1 年で期限切れになります。

NetScaler VPX Express は、**NetScaler MPX** アプライアンスと同じ暗号化暗号をサポートしていますか

一般的な可用性のために、NetScaler アプライアンスでサポートされている同じ強力な暗号化暗号はすべて、NetScaler VPX および NetScaler VPX Express で利用できます。これは、同じ輸出入規制の対象となります。

NetScaler VPX Express のテクニカルサポートケースを報告できますか

いいえ。NetScaler VPX Express ユーザーは、NetScaler VPX ナレッジ センターを自由に使用でき、ディスカッション フォーラムを使用してコミュニティにサポートをリクエストすることもできます。

NetScaler VPX Express を製品版にアップグレードできますか**?**

はい。必要な小売用 NetScaler VPX ライセンスを購入し、対応するライセンスを NetScaler VPX Express インスタンスに適用するだけです。

ハイパーバイザー

NetScaler VPX はどのバージョンの **VMware** をサポートしていますか

NetScaler VPX は、バージョン 3.5 以降では、VMware ESX と ESXi の両方をサポートしています。詳細については、サポート マトリックスと使用ガイドラインを参照してください。

VMware の場合、**VPX** に割り当てることができる仮想ネットワーク・インタフェースはいくつですか**?**

最大 10 個の仮想ネットワークインターフェイスを NetScaler VPX に割り当てることができます。

vSphere から、**NetScaler VPX** コマンドラインにどのようにアクセスできますか

VMware vSphere クライアントは、コンソールタブから NetScaler VPX コマンドラインへの組み込みアクセスを提供します。また、任意の SSH または Telnet クライアントを使用してコマンドラインにアクセスすることもできます。NetScaler VPX の NSIP アドレスは、SSH または Telnet クライアントで使用できます。

NetScaler VPX GUI にはどのようにアクセスできますか

NetScaler VPX GUI にアクセスするには、任意のブラウザーのアドレスフィールドに、NetScaler VPX の NSIP（たとえば、`http://NSIP address`）を入力します。

同じ **VMware ESX** にインストールされている **2** つの **NetScaler VPX** インスタンスを高可用性セットアップで構成できますか**?**

はい、でもお勧めできません。ハードウェア障害は、両方の NetScaler VPX インスタンスに影響します。

2 つの異なる **VMware ESX** システム上で実行されている **2** つの **NetScaler VPX** インスタンスを、高可用性セットアップで構成できますか**?**

はい。これは、高可用性セットアップで推奨されます。

VMware の場合、インターフェイス関連のイベントは **NetScaler VPX** でサポートされていますか

いいえ。インターフェイス関連のイベントはサポートされていません。

VMware の場合、タグ付き **VLAN** は **NetScaler VPX** でサポートされていますか**?**

はい。NetScaler タグ付き VLAN は、リリース 11.0 以降の NetScaler VPX でサポートされています。詳しくは、NetScaler のドキュメントを参照してください。

VMware の場合、リンクアグリゲーションと **LACP** は **NetScaler VPX** でサポートされていますか**?**

いいえ。なしリンクアグリゲーションと LACP は、NetScaler VPX ではサポートされていません。リンクアグリゲーションは VMware レベルで設定する必要があります。

NetScaler VPX ドキュメントにはどのようにアクセスするのですか

このドキュメントは、NetScaler VPX GUI から入手できます。ログイン後、[ドキュメント] タブを選択します。

キャパシティプランニングまたはサイジング

NetScaler VPX で期待できるパフォーマンスは何ですか

NetScaler VPX は、優れたパフォーマンスを提供します。NetScaler VPX を使用して達成可能な特定のパフォーマンスレベルについては、NetScaler VPX のデータシートを参照してください 。

サーバーの **CPU** パワーが変化することを考えると、**NetScaler** インスタンスの最大パフォーマンスをどのように見積もることができますか**?**

より高速な CPU を使用すると（ライセンスで許可されている最大値まで）パフォーマンスが向上しますが、低速の CPU を使用すると、パフォーマンスが確実に制限されます。

NetScaler VPX の帯域幅またはスループットの制限は、インバウンドのみのトラフィック、またはインバウンドとアウトバウンドの両方のトラフィックですか

NetScaler VPX 帯域幅制限は、要求トラフィックか応答トラフィックかにかかわらず、NetScaler への着信トラフィックにのみ適用されます。これは、NetScaler VPX-1000（たとえば）が 1 Gbps のインバウンドトラフィックと 1 Gbps のアウトバウンドトラフィックの両方を同時に処理できることを示します。インバウンドおよびアウトバウンドトラフィックは、要求および応答トラフィックと同じではありません。NetScaler では、エンドポイントからのトラフィック（リクエストトラフィック）とオリジンサーバーからのトラフィック（レスポンストラフィック）の両方が「インバウンド」（つまり、NetScaler に着信）です。

同じサーバー上で **NetScaler VPX** 複数のインスタンスを実行できますか**?**

はい。Hypervisor 上に物理 NIC が 1 つしかない NetScaler VPX 構成ユーティリティを使用して、最大 7 つのインターフェイス（VMware では 10）を追加できます。

NetScaler VPX の複数のインスタンスが物理サーバーで実行されている場合、**NetScaler VPX** インスタンスごとの最小ハードウェア要件は何ですか**?**

各 NetScaler VPX インスタンスには、2 GB の物理 RAM、20 GB のハードディスク容量、および 2 つの vCPU を割り当てる必要があります。2 ギガバイトの VPX 重要な展開では、システムがメモリに制約のある環境で動作するため、VPX に 2 GB の RAM を使用することはお勧めしません。これにより、スケール、パフォーマンス、または安定性に関連する問題が発生する可能性があります。4 GB または 8 GB のメモリが推奨されます。

> 注
>
> NetScaler VPX は、レイテンシーに敏感で高性能な仮想アプライアンスです。期待されるパフォーマンスを実現するには、アプライアンスに vCPU 予約、メモリ予約、ホストでの vCPU ピン接続が必要です。また、ホスト上でハイパースレッディングを無効にする必要があります。ホストがこれらの要件を満たさない場合、高可用性フェイルオーバー、VPX インスタンス内の CPU スパイク、VPX CLI へのアクセスにおける低速化、ピットボスデーモンのクラッシュ、パケットドロップ、低スループットなどの問題が発生します。
>
> すべての VPX インスタンスが事前定義された条件を満たしていることを確認してください。

NetScaler VPX と他のアプリケーションを同じサーバーでホストできますか**?**

はい。たとえば、NetScaler VPX、Citrix Virtual Apps Web インターフェイス、および Citrix Virtual Apps XML ブローカーはすべて仮想化でき、同じサーバー上で実行できます。最高のパフォーマンスを得るには、実行中のすべてのワークロードをサポートするのに十分な CPU および I/O 容量が物理ホストにあることを確認します。

単一の **NetScaler VPX** インスタンスに **CPU** コアを追加すると、そのインスタンスのパフォーマンスが向上しますか**?**

ライセンスに応じて、NetScaler VPX インスタンスは現在、最大 4 つの vCPU を使用できます。より多くの CPU を使用できる NetScaler VPX インスタンスに CPU を追加すると、パフォーマンスが向上します。

NetScaler VPX がアイドル状態のにもかかわらず、**CPU** の **90**%以上を消費しているように見えるのはなぜですか**?**

これは正常な動作であり、NetScaler アプライアンスは同じ動作を示します。NetScaler VPX CPU 使用率の実程度を確認するには、NetScaler CLI で stat CPU コマンドを使用するか、NetScaler GUI から NetScaler VPX CPU 使用率を表示します。NetScaler パケット処理エンジンは、やるべき作業がない場合でも、常に「仕事を探している」

ことです。したがって、CPU を制御し、それを解放しないためにすべてを行います。NetScaler VPX がインストールされたサーバーでは、NetScaler VPX が CPU 全体を消費しているような外観になります（Hypervisor の観点から）。CLI または GUI を使用した「NetScaler 内部」からの CPU 使用率を見ると、NetScaler VPX CPU 容量が使用されている様子が表示されます。

システム要件

NetScaler VPX 最小ハードウェア要件を教えてください

次の表では、NetScaler VPX 最小ハードウェア要件について説明しています。

種類	要件
——	————————————————————————--
プロセッサ	インテル Xeon または AMD EPYC を搭載したデュアルコアサーバー。
メモリ	最低 2 GB。ただし、4 GB が推奨されます。
ディスク	最低 20 GB のハードドライブ。
ハイパーバイザー	Citrix Hypervisor 5.6 以降、VMware ESX/ESXi 3.5 以降、または Hyper-V を搭載した Windows Server 2008 R2
ネットワーク接続性	最小値は 100 Mbps ですが、1 Gbps が推奨されます。
NIC	使用しているハイパーバイザーと互換性のある NIC。

> 注
>
> 重要な導入環境では、NetScaler VPX には 4 GB のメモリが推奨されます。2 GB のメモリを搭載した NetScaler VPX は、メモリに制約のある環境で動作します。これにより、スケール、パフォーマンス、または安定性に関連する問題が発生する可能性があります。

システム要件の詳細については、NetScaler VPX のデータシートを参照してください。

> 注
>
> NetScaler 13.1 リリース以降、VMware ESXi ハイパーバイザー上の NetScaler VPX インスタンスは AMD EPYC プロセッサをサポートしています。

インテル **VT-x** とは何ですか**?**

これらの機能は「ハードウェアアシスト」または「仮想化アシスト」と呼ばれることもあり、ゲスト OS が実行する機密性の高い CPU 命令や特権が必要な CPU 命令をハイパーバイザーにトラップアウトします。これにより、Hypervisor でのゲスト OS（NetScaler VPX 用の BSD）のホスティングが簡単になります。

VT-X はどれくらい一般的ですか

事実上、過去 2 年以内に出荷されたすべてのサーバが VT-X をサポートしている可能性があります。多くのサーバには、BIOS で仮想化支援が無効になっている状態で出荷されます。NetScaler VPX を実行できないと仮定する前に、サーバーでこの設定を変更する必要があるかどうかを確認してください。

NetScaler VPX ハードウェア互換性リスト（**HCL**）はありますか**?**

サーバーが Intel VT-X をサポートしている限り、NetScaler VPX は、基盤となる Hypervisor と互換性のあるサーバーで実行する必要があります。サポートされるプラットフォームの包括的なリストについては、Hypervisor HCL を参照してください。

NetScaler VPX はどのバージョンの **NetScaler OS** をベースにしていますか

NetScaler VPX は、NetScaler 9.1 以降のリリースをベースにしています。

NetScaler VPX は **BSD** 上で動作するので、**BSD Unix** がインストールされているサーバーでネイティブに実行できますか

いいえ。NetScaler MPX アプライアンスは、NetScaler VPX が提供するよりも高い SSL スループットが必要な場合に使用する必要があります。ハイパーバイザーのサポートの詳細については、NetScaler VPX データ シートを参照してください。

その他の技術的なよくある質問

複数の **NIC** を持つ物理サーバでのリンクアグリゲーションは機能しますか**?**

LACP はサポートされていません。Citrix Hypervisor では、静的リンクアグリゲーションがサポートされ、4 つのチャネルと 7 つの仮想インターフェイスの制限があります。VMware の場合、静的リンクアグリゲーションは NetScaler VPX 内ではサポートされていませんが、VMware レベルで構成できます。

MAC ベースの転送 **(MBF)** は **VPX** でサポートされていますか**? NetScaler** アプライアンスの実装から変更はありますか**?**

MBF はサポートされており、NetScaler アプライアンスと同じように動作します。Hypervisor は基本的に、NetScaler VPX から受信したすべてのパケットを外部に切り替え、逆に切り替えます。

NetScaler VPX のアップグレードプロセスはどのように実行されますか**?**

アップグレードは、NetScaler アプライアンスの場合と同じ方法で実行されます。カーネルファイルをダウンロードし、GUI で install ns またはアップグレードユーティリティを使用します。

フラッシュとディスク容量はどのように割り当てられますか。それを変えることはできますか

/フラッシュ = 965M /var = 14G 各 NetScaler VPX インスタンスには、最低 2 GB のメモリを割り当てる必要があります。NetScaler VPX ディスクイメージは、最大 4 GB のコアダンプ、ログファイル、トレースファイルを取得して格納するためのスペースなど、保守性を考慮して 20 GB のサイズにしました。これより小さいディスクイメージを生成することは可能ですが、現時点ではこれを実行する予定はありません。/flashおよび/varは両方とも同じディスクイメージ内にあります。互換性を保つために、これらは別々のファイル システムとして保持されます。互換性のために別々のファイルシステムとして保管されています。メモリ割り当ての推奨事項の詳細については、NetScaler VPX データシートを参照してください。

新しいハードドライブを追加して、**NetScaler VPX** インスタンスのスペースを増やすことはできますか

はい。NetScaler リリース 13.1 ビルド 21.x 以降では、2 台目のディスクを追加して NetScaler VPX インスタンスのディスク容量を増やすことができます。2 番目のディスクを接続すると、「/var/crash」ディレクトリが自動的にこのディスクにマウントされます。2 つ目のディスクは、コアファイルの保存とロギングに使用されます。コアファイルとログファイルの保存に使用される既存のディレクトリは、以前と同様に機能します。

> 注
>
> データの損失を防ぐために、NetScaler アプライアンスのダウングレード時に外部バックアップを作成します。

クラウド上の NetScaler VPX インスタンスに新しいハードディスクドライブ（HDD）を接続する方法については、以下を参照してください。

- Azure ドキュメンテーション

  > 注
  >
  > Azure にデプロイされた NetScaler VPX インスタンスにセカンダリ ディスクを接続するには、Azure VM サイズにローカル一時ディスクがあることを確認します。詳細については、「ローカル一時ディスクなしの Azure VM サイズ」を参照してください。

- AWS ドキュメント
- GCP ドキュメント

> 警告:
>
> NetScaler VPX に新しい HDD を追加した後、新しい HDD に移動されたファイルに対して機能するスクリプトの一部が、次の条件下で失敗する可能性があります。

> 「link」シェルコマンドを使用して、新しい HDD に移動されたファイルへのハードリンクを作成した場合。
>
> シンボリックリンクを使用するには、このようなコマンドはすべて「ln-s」に置き換えてください。また、失敗したスクリプトを適宜修正してください。

NetScaler VPX のプライマリ ディスクのサイズを増やすことはできますか**?**

NetScaler リリース 14.1 ビルド 21.x 以降、管理者は NetScaler VPX プライマリディスクサイズを一度に 20 GB から 1TB に動的に増やすことができます。その後、再び最大 1 TB まで増やすことができます。ディスク容量を増やすには、それぞれのクラウドまたはハイパーバイザー UI でプライマリディスクサイズを少なくとも 1 GB に拡張します。

> 注
>
> ディスクのサイズを増やすことしかできません。新しいサイズを割り当てると、後でサイズを減らすことはできません。そのため、必要な場合にのみディスクサイズを増やしてください。

NetScaler VPX プライマリディスクサイズを手動で増やすにはどうすればよいですか**?**

次の手順に従って、ハイパーバイザーまたはクラウドから VPX プライマリディスクサイズを手動で増やします:

1. 仮想マシンをシャットダウンします。
2. デフォルトのディスクサイズである 20 GB をより高い値に拡張します。たとえば、20 ギガバイトから 30 ギガバイトまたは 40 ギガバイトです。Azure の場合は、デフォルトのディスクサイズである 32 GB を 64 GB に拡張します。
3. VM の電源を入れ、ブートプロンプトを入力します。
4. 「boot-s」コマンドを使用してシングルユーザーモードにログインします。
5. ディスク容量を確認してください。新しく割り当てられたディスク容量は、「gpart show」コマンドを使用して確認できます。
6. パーティション名を書き留めておきます。たとえば、VM パーティションは da0 です。
7. 「gpart resize」コマンドを使用してディスクパーティションのサイズを変更します。

   例**:** 次のコマンドを実行して、da0 MBR パーティションのサイズを変更し、10 GB の空き領域を含めます。

   ```
   gpart resize -i 1 da0
   ```

8. 空き領域を最後のパーティションにマージします。

   例

   ```
   gpart resize -i 5 da0s1
   ```


9. 「growfs」コマンドを使用して、新しく割り当てられた空き領域を含むようにファイルシステムを拡張します。

   例

   ```
   growfs /dev/ada0s1e
   ```

10. VM を再起動し、シェルプロンプトで「df-h」コマンドを使用して増加したディスク容量を確認します。

NetScaler VPX ビルドの番号付けと、他のビルドとの相互運用性に関して、どのようなことが期待できますか**?**

NetScaler VPX には、9.1 と同様のビルド番号が付けられています。Cl (クラシック) と 9.1. Nc (nCore) リリース。たとえば、9.1_97.3.vpx、9.1_97.3.nc、9.1_97.3.cl。

NetScaler VPX を **NetScaler** アプライアンスを使用した高可用性セットアップの一部にすることはできますか**?**

サポートされていない構成です。

NetScaler VPX に表示されるすべてのインターフェイスは、**Hypervisor** 上のインターフェイスの数に直接関係していますか

いいえ。ハイパーバイザー上の物理 NIC が 1 つだけの場合、NetScaler VPX 構成ユーティリティを使用して最大 7 つのインターフェイス (VMware の場合は 10) を追加できます。

Citrix Hypervisor XenMotion、**VMware VMotion**、または **Hyper-V** ライブマイグレーションを使用して **NetScaler VPX** のアクティブなインスタンスを移動できますか**?**

NetScaler VPX は Hyper-V ライブマイグレーションをサポートしていません。vMotion は NetScaler リリース 13.0 以降でサポートされています。ライブマイグレーション（以前の XenMotion）は、NetScaler リリース 14.1 ビルド 17.38 以降でサポートされています。