

NetScaler VPX 13.1

Contents

サポートマトリックスと使用ガイドライン	28
XenServer上のVPXインスタンス	28
VMware ESXiハイパーバイザー上のVPXインスタンス	28
Microsoft Hyper-V上のVPXインスタンス	31
Azure Local上のVPXインスタンス	31
Nutanix AHV 上の VPX インスタンス	31
汎用 KVM 上の VPX インスタンス	32
パブリッククラウド上のVPXインスタンス	33
ハイパーバイザーでサポートされるVPX機能	33
パブリッククラウドでサポートされるVPX機能	35
サポートされているブラウザ	37
NetScaler VPX™でサポートされているプロセッサ	37
NetScaler VPXでサポートされているNIC	38
使用ガイドライン	39
パケットエンジンCPU使用率を制御するコマンド	40
その他の参照情報	41
VMware ESX、Linux KVM、およびCitrix Hypervisor上でのNetScaler VPXパフォーマンスを最適化する	43
VMware ESXハイパーバイザー上のNetScaler VPXインスタンス	43
ESXホストでの推奨構成	43
E1000ネットワークインターフェイスを備えたNetScaler VPX	44
pNICあたり2つのvNIC展開	45
VMXNET3 ネットワークインターフェイスを備えた NetScaler VPX	46
VMXNET3デバイスのVMware ESXでマルチキューおよびRSSサポートを構成する	47
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 VPX	48
Linux-KVM プラットフォーム上の NetScaler VPX インスタンス	49

KVM のパフォーマンス設定	49
PV ネットワークインターフェイスを備えた NetScaler VPX	53
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を備えたNetScaler VPX	55
Citrix Hypervisor上でのNetScaler VPXインスタンス	56
Citrix Hypervisorのパフォーマンス設定	56
SR-IOVネットワークインターフェイスを備えたNetScaler VPX	57
パラ仮想化インターフェースを備えたNetScaler VPX	57
NetScaler VPXディスク容量の増加のサポート	59
NetScaler VPXでプライマリディスクサイズを動的に増やす	59
セカンダリディスクを追加する	60
NetScaler VPXのプライマリディスクサイズを手動で増やす	61
クラウドでのNetScaler VPXアプライアンスの初回起動時に構成を適用する	62
ユーザーデータとは	62
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	62
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	62
AWS CLI を使用してプリブートユーザーデータを提供する	63
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	63
Azure CLI を使用してプリブートユーザーデータを提供する	64
GCP 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	64
gcloud CLI を使用してプリブートユーザーデータを提供する	64
プリブートユーザーデータ形式	65
NetScaler構成	66
ユーザースクリプト	68
ライセンス	71
ブートストラップ	72
デフォルトのブートストラップ構成	72
ブートストラップ構成のカスタマイズ	73

方法1：インターフェイスの詳細のみを指定するカスタムブートストラップ	73
AWS のカスタムブートストラップの例	74
Azure のカスタムブートストラップの例	74
GCPのカスタムブートストラップの例	75
方式2：インターフェース、IPアドレス、およびサブネットマスクを指定することによるカスタムブートストラップ	75
AWSのカスタムブートストラップの例	76
Azureのカスタムブートストラップの例	76
GCPのカスタムブートストラップの例	77
方法 3: <code><NS-CONFIG></code> セクションでブートストラップ関連コマンドを指定することによるカスタムブートストラップ	77
AWS のカスタムブートストラップの例	78
Azure のカスタムブートストラップの例	79
GCP のカスタムブートストラップの例	82
AWSおよびAzureにおけるNICの接続と切断の影響	83
パブリッククラウドプラットフォームでのSSL-TPSパフォーマンスの向上	85
NetScaler CLIを使用してPEモードを構成する	85
クラウド内のNetScaler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	86
パブリッククラウド上のNetScaler VPXで同時マルチスレッディングを構成する	87
SMTサポートマトリックス	88
制限事項	88
SMTの構成	88
トラブルシューティング	90
ベアメタルサーバーにNetScaler VPXインスタンスをインストールする	92
ベアメタルサーバーにNetScaler VPXインスタンスをインストールするための前提条件	92
ベアメタルサーバーにNetScaler VPXインスタンスをインストールする	92
Citrix Hypervisor/XenServerにNetScaler VPXインスタンスをインストールする	94
HypervisorにNetScaler VPXインスタンスをインストールするための前提条件	94

Hypervisorのハードウェア要件	94
XenCenterのシステム要件	96
XenCenterを使用してHypervisorにNetScaler VPXインスタンスをインストールする	96
VPXインスタンスをシングルルートI/O仮想化 (SR-IOV) ネットワークインターフェースを使用するように構成する	97
制限事項	98
Intel 82599 NICの制限事項	98
Intel X710 10GおよびIntel XL710 40G NICの制限事項	98
Intel 82599 NIC の前提条件	99
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る	99
Citrix Hypervisorホストを使用して、Intel 82599 SR-IOV VFをNetScaler VPXインスタンスから割り当て解除する	100
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る	100
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を削除する	103
SR-IOVインターフェイスでリンクアグリゲーションを構成する	103
SR-IOVインターフェイスでVLANを構成する	104
その他の参照	104
NetScaler VPXインスタンスをVMware ESXにインストールする	105
前提条件	105
VMware ESXのハードウェア要件	106
VMware vSphereクライアントのシステム要件	107
OVF Tool 1.0のシステム要件	108
NetScaler VPX セットアップファイルのダウンロード	108
VMware ESX に NetScaler VPX インスタンスをインストールする	109
VMware vMotionを使用してNetScaler VPXインスタンスを移行する	110

NetScaler VPX インスタンスをVMXNET3ネットワークインターフェースを使用するように構成する	111
NetScaler VPXインスタンスをSR-IOVネットワークインターフェースを使用するように構成する	115
制限事項	115
前提条件	116
E1000からSR-IOVまたはVMXNET3ネットワークインターフェースへのNetScaler VPXの移行	121
NetScaler VPXインスタンスをPCIパススルーネットワークインターフェースを使用するように構成する	121
概要	121
前提条件	121
ホストでパススルーデバイスを構成する	122
NetScaler VPXインスタンスでパススルーデバイスを構成する	122
VMware ESXハイパーバイザー上のNetScalerアプライアンスの初回起動時にNetScaler VPX構成を適用する	123
ESXハイパーバイザーでプレブートユーザーデータを提供する方法	124
CD/DVD ISOを使用してユーザーデータを提供する	124
ESX Web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	127
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	130
AWS 上の VMware Cloud に NetScaler VPX インスタンスをインストールする	134
前提条件	134
VMware Cloud のハードウェア要件	134
OVF Tool 1.0 のシステム要件	135
NetScaler VPXセットアップファイルのダウンロード	135
VMware Cloud に NetScaler VPX インスタンスをインストールする	136
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	138
MicrosoftサーバーにNetScaler VPXインスタンスをインストールするための前提条件	138
Microsoftサーバーのハードウェア要件	139
NetScaler VPXセットアップファイルをダウンロードする	139

MicrosoftサーバーにNetScaler VPXインスタンスをインストールする	140
Hyper-V 上で NetScaler VPX インスタンスを自動プロビジョニングする	141
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	146
制限事項と使用ガイドライン	146
一般的な推奨事項	146
制限事項	146
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ための前提条件	147
ソフトウェア要件	147
ゲストVMのハードウェア要件	147
ネットワーク要件	148
ソースインターフェースとモード	148
ソースインターフェースのプロパティ	149
プロミスキュースモード	152
必要なモジュール	153
OpenStack を使用して NetScaler VPX インスタンスをプロビジョニングする	153
ユーザーデータファイル	154
ネットワーク構成	158
カスタマースクリプト	158
SSHキーペア認証	158
開始する前に	159
VPXインスタンスのプロビジョニング	159
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	160
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	161
Virtual Machine Managerを使用してNetScaler VPXインスタンスをプロビジョニングする	162
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	163
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する	166
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する	167
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	167

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ースを追加する	168
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	169
制限事項	170
前提条件	171
NetScaler VPX インスタンスを SR-IOV ネットワークインターフェースを使用するように構成する	173
SR-IOVインターフェースで静的LA/LACPを構成する	174
SR-IOVインターフェースでのVLANの設定	174
NetScaler VPXインスタンスをPCIパススルーネットワークインターフェースを使用するように構成する	174
前提条件	175
virsh プログラムを使用してNetScaler VPXインスタンスをプロビジョニングする	176
NetScaler VPXインスタンスに virsh プログラムを使用してインターフェースを追加する	179
NetScaler VPXゲストVMを管理する	181
Virtual Machine Managerを使用してVPXゲストVMを管理する	181
virsh プログラムを使用してNetScaler VPXゲストVMを管理する	182
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する	184
前提条件	184
ホスト上でSR-IOV VFを有効にする	184
VFを構成し、OpenStackで利用できるようにする	185
OpenStackにNetScaler VPXインスタンスをプロビジョニングする	185
CLI を介して OpenStack に NetScaler VPX インスタンスをプロビジョニングする	186
追加でサポートされる構成: ホストからSR-IOV VF上のVLANを作成および削除する	189
KVM 上で NetScaler VPX インスタンスを OVS DPDK ベースのホストインターフェースを使用するように構成する	189
前提条件	190
DPDKをインストールする	190

OVNをビルドしてインストールする	190
OVNブリッジを作成する	190
物理インターフェースをOVNブリッジに接続する	191
<code>vhost-user</code> ポートをOVNデータパスに接続する	192
OVN-DPDKベースの <code>vhost-user</code> ポートを持つKVM-VPXをプロビジョニングする	192
注意点	209
KVMハイパーバイザー上のNetScalerアプライアンスの初回起動時にNetScaler VPX構成を適用する	210
KVMハイパーバイザーでプレブートユーザーデータを提供する方法	211
CDROM ISOファイルを使用してユーザーデータを提供する方法	211
AWS 上の NetScaler VPX	214
はじめに	214
AWS で NetScaler VPX インスタンスを展開する	214
ハイブリッド展開	215
ライセンス	215
自動化	215
ブログ	215
ビデオ	216
顧客事例	216
ソリューション	216
サポート	216
その他の参考資料	216
AWS 用語	217
AWS-VPX サポートマトリックス	219
制限事項と使用ガイドライン	223
前提条件	225
NetScaler VPXインスタンスでAWS IAMロールを設定する	229
IAMロールを作成する	233

IAMポリシーシミュレーターでIAMポリシーをテストする	236
関連情報	237
AWS 上での NetScaler VPX インスタンスの動作方法	237
AWS に NetScaler VPX スタンドアロンインスタンスを展開する	238
AWS ウェブコンソールを使用して AWS に NetScaler VPX インスタンスを展開する	238
Citrix 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	243
AWS CLI を使用して NetScaler VPX インスタンスを構成する	244
シナリオ: スタンドアロンインスタンス	244
NetScaler VPXライセンスをダウンロードする	249
AWSサブスクリプションをBYOLに移行する方法	250
制限事項	253
異なるアベイラビリティゾーンのサーバーの負荷分散	253
AWSでの高可用性の仕組み	254
同一ゾーン内での高可用性	254
異なるゾーン間での高可用性	256
展開を開始する前に	256
トラブルシューティング	257
同じAWSアベイラビリティゾーンにVPX HAペアをデプロイする	257
同じゾーンにVPX HAペアをデプロイする方法	257
ステップ1. 同じVPCを使用して、それぞれ3つのNIC（Ethernet 0、Ethernet 1、Ethernet 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	258
ステップ2. プライマリノードで、Ethernet 1（クライアントIPまたはVIP）とEthernet 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	258
ステップ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を構成する	259
ステップ4: 両方のインスタンスでHAを構成する	260
VPX HAペアをデプロイするためのレガシーメソッド	260
注意事項	261

Citrix CloudFormationテンプレートを使用して高可用性ペアを展開する	262
Amazon CloudWatch を使用してインスタンスを監視する	265
注意点	265
CloudWatch メトリックの表示方法	265
高可用性セットアップでのSR-IOVの構成	266
関連リソース	266
異なるAWSアベイラビリティゾーン間での高可用性	266
注意事項	266
追加の参照情報	267
異なるAWSゾーン間でエラスティックIPアドレスを持つVPX高可用性ペアをデプロイする	267
異なるAWSゾーン間でEIPアドレスを使用するHAの仕組み	267
異なるAWSゾーン間でエラスティックIPアドレスを持つVPX高可用性ペアをデプロイする方 法	268
CLIを使用して高可用性を構成する	268
GUIを使用して高可用性を構成する	269
シナリオ	270
異なるAWSゾーンにプライベートIPアドレスを持つVPX高可用性ペアを展開する	271
AWS非共有VPCを使用したプライベートIPアドレスによる高可用性ペア	272
前提条件	272
AWS非共有VPCを使用してプライベートIPアドレスでVPX HAペアを展開する	273
CLIを使用して高可用性を構成する	275
GUIを使用して高可用性を構成する	275
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	276
前提条件	276
クロスアカウントIAMロールを作成する	278
NetScaler CLIを使用してクロスアカウントIAMロールを設定する	278
NetScaler GUIを使用してクロスアカウントIAMロールを設定する	279
シナリオ	279

HAプライベートIPソリューション向けのAWS Transit Gateway の設定	280
トラブルシューティング	280
AWS Outposts に NetScaler VPX インスタンスを展開する	281
AWS Outposts の仕組み	281
前提条件	282
AWS ウェブコンソールを使用して AWS Outposts に NetScaler VPX インスタンスを展開する	282
AWS Outposts を使用したハイブリッドクラウドへの NetScaler VPX インスタンスの展開	283
参照	283
NetScaler Web App Firewall を使用した AWS API Gateway の保護	284
AWS API Gateway の前に NetScaler アプライアンスを展開する	284
WAFが有効なNetScalerアプライアンスを構成する	285
NetScalerの機能と性能	287
バックエンドAWSオートスケーリングサービスを追加	288
開始する前に	289
AWS AutoscalingサービスをNetScaler VPXインスタンスに追加する	290
AWSにNetScaler GSLBを展開する	293
DBSの概要	293
ELBを使用したドメイン名ベースサービス (DBS)	294
AWSコンポーネントの構成	294
セキュリティグループ	294
グローバルサーバーロードバランシングのドメイン名ベースサービスを設定する	296
展開タイプ	296
CFT展開	296
展開手順	296
ハイブリッドおよびマルチクラウド展開のためのNetScalerグローバル負荷分散	298
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	298
セキュリティグループの設定	299

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	300
他のロードバランシングソリューションとの互換性	300
GSLBメソッド	300
GSLBトポロジ	301
IPv6のサポート	301
監視	301
パーシステンス	302
NetScaler ADM StyleBookを使用した構成	302
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	302
NetScaler LBノードでGSLBを構成するためのStyleBookの使用	304
LBノード用のマルチクラウドGSLB StyleBookを使用して親子トポロジで子サイトを構成する	305
CloudFormationテンプレートの展開	307
CFT 3 NIC展開	307
前提条件	309
GSLB の前提条件	310
その他のリソース	311
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	311
インターフェイスタイプをSR-IOVに変更する	311
インターフェースタイプをSR-IOVに変更するには	312
高可用性セットアップでの SR-IOV の構成	312
AWS ENAで拡張ネットワーキングを使用するようにNetScaler VPXインスタンスを構成する	313
AWS上のNetScaler VPXインスタンスをアップグレードする	314
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	314
AWS上の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をアップグレードする	315
カスタマーライセンスを使用する (Bring-Your-Own-License)	315

ユーティリティライセンスの使用（時間単位料金のユーティリティライセンス）	315
AWS上のNetScaler VPXインスタンスのシステムソフトウェアをアップグレードする	315
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	316
前提条件と考慮事項	316
AWS上のVPXインスタンスのトラブルシューティング	320
AWS よくある質問	321
Microsoft AzureへのNetScaler VPXインスタンスの展開	325
前提条件	325
Azure での NetScaler VPX インスタンスの動作	326
マルチNICマルチIPアーキテクチャ	327
シングルNICマルチIPアーキテクチャ	327
シングルNICシングルIP	327
NetScaler VPX ライセンス	328
VPX のパフォーマンスと推奨される Azure インスタンスタイプ	328
留意事項	329
AzureにおけるNetScaler VPXインスタンスのIPv6サポート	330
制限事項	330
Azure 用語	331
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	333
ネットワークアドレス変換によるトラフィックフロー	335
ポート使用ガイドライン	335
NetScaler VPXスタンドアロンインスタンスを構成する	336
開始する前に	336
構成手順の概要	337
リソースグループを構成する	337
ネットワークセキュリティグループを構成する	338
仮想ネットワークとサブネットを構成する	338
2番目のサブネットを構成する	339

ストレージアカウントを構成する	339
可用性セットを構成する	340
NetScaler VPXインスタンスを構成する	340
NetScaler VPXスタンドアロンインスタンスの複数のIPアドレスを構成する	343
ユースケース	343
始める前に	344
ステップ2: NetScalerが所有するIPアドレスを構成する	345
複数のIPアドレスとNICを持つ高可用性セットアップを構成する	345
アベイラビリティセットを使用した高可用性	346
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	347
可用性ゾーンを使用した高可用性	349
Azure Monitorでメトリックを使用してインスタンスを監視する	350
注意事項	350
Azure Monitor でメトリックを表示する方法	350
PowerShellコマンドを使用して、複数のIPアドレスとNICを持つ高可用性セットアップを構成する	351
PowerShellコマンドを使用してHA-INCノードを構成する	352
シナリオ: HA-INC PowerShell展開	352
パラメータ設定	353
ALBのフローティングIP無効モードでAzure上にNetScaler高可用性ペアを展開する	371
フローティングIP無効モードでのALBを使用したHA展開アーキテクチャ	371
前提条件	371
ALB フローティング IP 無効モードで Azure に VPX HA ペアを展開する方法	372
Azureアクセラレーテッド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	379
前提条件	380
制限事項	380
高速ネットワークでサポートされるNIC	380

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	381
Azure PowerShellを使用してアクセラレーテッドネットワークを有効にする	382
NetScalerのFreeBSDシェルを使用してインターフェース上のアクセラレーテッドネットワークを確認するには	383
ADC CLIを使用してインターフェースで高速ネットワークを検証するには	384
NetScalerで注意すべき点	384
PVインターフェースへのVLANの構成	385
Azure ILBでNetScaler®高可用性テンプレートを使用してHA-INCノードを構成する構成例	389
インターネットに公開されるアプリケーション向けNetScaler®高可用性テンプレートを使用してHA-INCノードを構成する設定例	393
Azureの外部および内部ロードバランサーを同時に使用して高可用性セットアップを構成する	394
Azure VMware Solution に NetScaler VPX インスタンスをインストールする前提条件	398
VMware クラウドのハードウェア要件	399
OVF Tool 1.0 のシステム要件	399
NetScaler VPX セットアップファイルのダウンロード	400
Azure VMware Solution をデプロイする	400
Windowsを実行しているAzure仮想マシンに接続する	401
プライベートクラウド vCenter ポータルにアクセスする	402
AzureポータルでNSX-Tセグメントを作成する	403
VMware Cloud に NetScaler VPX インスタンスをインストールする	403
Azure VMware Solution で NetScaler VPX スタンドアロンインスタンスを構成する前提条件	405
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	405

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	407
Azure VMware Solution 上で NetScaler VPX 高可用性セットアップを構成する	407
前提条件	408
構成手順	408
NetScaler VPX HAペアでAzureルートサーバーを構成する	409
前提条件	410
NetScaler VPX HAペアでAzureルートサーバーを構成する方法	410
Azure Autoscale™ 設定を追加する	413
始める前に	414
NetScaler VPXインスタンスにVMSSを追加する	415
クラウドプロファイル作成時の留意事項	415
Azureでアンマネージドディスクからマネージドディスクに移行する	416
参照	416
NetScaler VPX展開のためのAzureタグ	416
VPX展開におけるAzureタグの仕組み	417
開始する前に	417
VPX GUIを使用してAzureタグ設定を追加する方法	417
VPX CLIを使用してAzureタグ設定を追加する方法	418
トラブルシューティング	420
NetScaler VPXインスタンスでGSLBを構成する	421
シナリオ	421
パラメーター	422
VMの作成	424
リソースグループの作成	425
ストレージアカウントの作成	425
可用性セットの作成	425
仮想ネットワークの作成	425

パブリック IP アドレスの作成	426
NIC の作成	426
VM構成オブジェクトを作成	427
資格情報を取得し、OSプロパティを設定する	428
NICを追加	428
OSディスクを指定し、VMを作成する	428
IP構成の詳細	429
GSLBサイトとその他の設定を構成する	431
ディザスタリカバリ	431
ディザスタリカバリのためのマルチNICマルチIP（3NIC）展開	432
ディザスタリカバリのためのシングルNICマルチIP（1NIC）展開	432
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	432
シナリオ	433
パラメータ設定	434
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ルールを使用して ALBを構成する	435
各高可用性ペアでGSLBを有効にする	436
Azure で NetScaler GSLB を展開する	438
DBS の概要	438
Azure ロードバランサー (ALB) を使用したドメイン名ベースのサービス	438
Azure GSLB の前提条件	439
Azureコンポーネントの構成	439
NetScaler GSLBドメインベースサービスの構成	440
トラフィック管理設定	440
GSLB設定	440
その他のリソース	442
NetScaler VPX インスタンスを Azure Local に展開する	443
NetScaler VPX インスタンスを Azure Local に展開するための前提条件	443

Windows Admin Center から VHD を使用して NetScaler VPX インスタンスを Azure Local に展開する方法	443
Azure に NetScaler® Web App Firewall を展開する	445
NetScaler Web App Firewall の展開戦略	446
Webアプリケーションファイアウォールの設定	447
SQLインジェクション	447
クロスサイトスクリプティング	448
バッファオーバーフローチェック	448
仮想パッチ/シグネチャ	448
アプリケーションセキュリティ分析	448
ADMでの集中学習	449
StyleBooks	450
セキュリティインサイト分析	451
セキュリティ侵害に関する詳細情報の取得	451
ログ式の詳細を表示	451
セキュリティ違反	452
アプリケーションセキュリティ違反の詳細を表示	453
違反カテゴリ	453
設定	453
高度なセキュリティ分析を有効にする	453
Webトランザクション設定を有効にする	454
セキュリティ違反ダッシュボード	454
ボットインサイト	454
ボットの表示	455
CAPTCHAボットの表示	456
ボットトラップの表示	456
ボットの詳細を表示	456
高度な検索	457

異常に高いリクエストレート	458
ボット検出	458
IP レピュテーション	459
NetScaler Web App Firewall と OWASP トップ 10 – 2021	459
A1:2021 不適切なアクセス制御	460
NetScaler Web App Firewall の保護機能	460
A2:2021 - 暗号化の失敗	461
NetScaler Web App Firewall の保護機能	461
A3:2021 - インジェクション	461
NetScaler Web App Firewallの保護機能	461
A5:2021 - セキュリティ設定ミス	462
NetScaler Web App Firewallの保護機能	462
A6:2021 - 脆弱で古いコンポーネント	463
NetScaler Web App Firewallの保護	463
A7:2021 – 認証の不備	464
NetScaler Web App Firewallの保護	464
A8:2021 - ソフトウェアとデータの整合性の不備	464
NetScaler Web App Firewallの保護	464
A9:2021 - セキュリティログと監視の不備	464
NetScaler Web App Firewallの保護	465
参照	465
NetScaler GatewayアプライアンスのIPアドレスプールイントラネットIPを構成する	465
AzureポータルでプライベートIPアドレスを登録する	466
NetScaler Gatewayアプライアンスでアドレスプールを構成する	466
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	467
ユースケース	467
スクリプト	468

パラメータ	468
1. リソースグループを作成する	471
2. ストレージアカウントを作成する	471
3. 可用性セットを作成する	472
4. ネットワークセキュリティグループを作成する	472
5. 仮想ネットワークを作成する	472
6. パブリックIPアドレスを作成する	473
7. IP構成の割り当て	473
8. NICの作成	474
9. NetScaler VPXインスタンスの作成	474
10. NIC構成を確認する	475
11. VPX側の構成を確認する	475
Azure展開用の追加PowerShellスクリプト	475
NetScaler VPXスタンドアロンインスタンスのプロビジョニング	476
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 ビジョニングする	481
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のNetScaler VPXペアのプ ロビジョニング	492
Azure よくある質問	500
Google Cloud Platform に NetScaler VPX インスタンスを展開する	502
サポートされる機能	502
制限事項	502
ハードウェア要件	502
注意事項	502
シナリオ: 複数のNIC、複数のIPを持つスタンドアロンVPXインスタンスを展開する	503
展開手順の概要	503
シナリオ: シングルNIC、スタンドアロンVPXインスタンスの展開	505
展開手順の概要	505

GCPにVPXインスタンスをデプロイした後の注意点	508
NetScaler VPXライセンス	509
GCPでサポートされているNetScaler VPX製品	509
サポートされているGCPマシンタイプファミリー	510
NetScaler VPXインスタンスをデプロイするためのGDMテンプレート	510
NetScaler マーケットプレイスイメージ	510
リソース	511
関連情報	511
Google Cloud Platform で VPX 高可用性ペアを展開する	511
GCP で VPX 高可用性ペアを展開するための GDM テンプレート	512
GCP上のVPX高可用性ペアに対する転送ルールサポート	512
前提条件	512
制限事項	512
Google Cloud Platformで外部静的IPアドレスを持つVPX高可用性ペアを展開する	512
開始する前に	513
Google Cloud PlatformでVPX HAペアを展開する方法	514
ステップ 1. VPC ネットワークを作成する	514
ステップ 2. 2 つの VPX インスタンスを作成する	515
ステップ 3. 高可用性を構成する	515
Google Cloud PlatformでプライベートIPアドレスを持つシングルNIC VPX高可用性ペアを展開する	522
開始する前に	522
Google Cloud PlatformでVPX高可用性ペアを展開する方法	523
ステップ 1. 1つのVPCネットワークを作成する	523
ステップ 2. 2つのVPXインスタンスを作成する	523
ステップ3. 高可用性を構成する	524
GUIを使用して高可用性を構成する	524
CLI を使用して高可用性を構成する	527

Google Cloud PlatformでプライベートIPアドレスを使用してVPX高可用性ペアを展開する	529
開始する前に	529
Google Cloud PlatformでVPX高可用性ペアを展開する方法	531
ステップ 1. VPC ネットワークを作成する	531
ステップ 2. 2 つの VPX インスタンスを作成する	531
手順3. 高可用性を構成する	532
GUIを使用して高可用性を構成する	532
CLI を使用して高可用性を構成する	535
Google Cloud VMware Engine に NetScaler VPX インスタンスをインストールする	537
前提条件	538
VMware クラウドのハードウェア要件	538
OVF Tool 1.0 のシステム要件	539
NetScaler VPX セットアップファイルのダウンロード	540
Google Cloud VMware Engineのデプロイ	540
プライベートクラウドのvCenterポータルにアクセスする	541
GCVE NSX-TポータルでNSX-Tセグメントを作成する	541
VMware クラウドに NetScaler VPX インスタンスをインストールする	543
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	545
バックエンドGCPオートスケーリングサービスを追加	546
開始する前に	546
NetScaler VPXインスタンスにGCPオートスケーリングサービスを追加する	547
GCP上のNetScaler VPXインスタンスにおけるVIPスケーリングのサポート	549
注意点	549
開始する前に	550
NetScaler VPXインスタンスでのVIPスケーリングのための外部IPアドレスの構成	552
NetScaler VPXインスタンスで外部IPアドレスを持つロードバランシング仮想サーバーを設定する	554
GCP上のVPXインスタンスのトラブルシューティング	555

NetScaler VPXインスタンスでのジャンボフレーム	556
VMware ESX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	557
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	557
Citrix XenServer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	557
AWS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	558
よくある質問	559
機能	559
NetScaler VPXとは何か?	559
NetScaler VPXには、ADCアプライアンスと同様に、すべてのWebアプリケーション最適化機能が含まれるか?	559
NetScaler VPXでCitrix Application Firewallを使用する場合、何か制限はありますか?	559
NetScaler VPX上のNetScaler GatewayとNetScalerアプライアンス上のNetScaler Gatewayに違いはありますか?	560
ハイパーバイザー上で実行できるという明白な違いを除いて、NetScaler VPXはNetScaler物理アプライアンスとどのように異なります?	560
NetScaler VPXはアプリケーションデリバリー市場でどのような役割を果たします?	560
NetScaler VPXは、当社のCitrixデリバリーセンター戦略にどのように適合しますか?	561
暗号化	561
NetScaler VPXはSSLオフロードをサポートしていますか?	561
NetScaler VPXをホストするサーバーにインストールされたサードパーティ製SSLカードは、SSL暗号化または復号化を高速化しますか?	561
NetScaler VPXは、物理NetScalerアプライアンスと同じ暗号化方式をサポートしていますか?	561
NetScaler VPXのSSLトランザクションスループットはどのくらいですか?	562
価格とパッケージ	562
NetScaler VPXはどのようにパッケージ化されていますか?	562
NetScaler VPXはすべてのハイパーバイザーで同じ価格ですか?	562
すべてのハイパーバイザーでVPXに同じNetScaler SKUが使用されますか?	562

NetScaler VPXライセンスをあるハイパーバイザーから別のハイパーバイザー（たとえば、VMwareからHyper-Vへ）に移動できますか？	562
NetScaler VPXインスタンスはアップグレードできますか？	562
NetScaler VPXを高可用性ペアで展開する場合、いくつかのライセンスが必要ですか？	563
NetScaler VPX Expressと90日間無料トライアル	563
NetScaler VPX Expressには、すべてのNetScaler標準機能が含まれていますか NetScaler Gatewayと、Citrix Virtual Apps（旧XenApp） Web InterfaceおよびXMLブローカーのロードバランシングが含まれていますか？	563
NetScaler VPX Expressの機能：NetScaler GatewayとCitrix Virtual Apps Web InterfaceおよびXMLブローカーのロードバランシング？	563
NetScaler VPX Expressのライセンス要件？	563
NetScaler VPX Expressライセンスは期限切れになりますか？	563
NetScaler VPX Expressには、5つの無料のNetScaler Gateway同時ライセンスが含まれていますか？	563
お客様がダウンロードできるNetScaler VPX Expressの数に制限はありますか？	564
NetScaler VPX Expressは、NetScaler MPXアプライアンスと同じ暗号化方式をサポートしていますか？	564
NetScaler VPX Expressのテクニカルサポートケースを申請できますか？	564
NetScaler VPX Expressをリテール版にアップグレードできますか？	564
ハイパーバイザー	564
NetScaler VPXはどのVMwareバージョンをサポートしていますか？	564
VMwareの場合、VPXにいくつかの仮想ネットワークインターフェースを割り当てることができますか？	564
vSphereからNetScaler VPXコマンドラインにアクセスするにはどうすればよいですか？	565
NetScaler VPX GUIにはどのようにアクセスできますか？	565
同じVMware ESXにインストールされた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？	565
2つの異なるVMware ESXシステムで実行されている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？	565
VMwareの場合、NetScaler VPXでインターフェース関連イベントはサポートされていますか？	565

VMwareの場合、NetScaler VPXでタグ付きVLANはサポートされていますか？	565
VMwareの場合、NetScaler VPXでリンクアグリゲーションとLACPはサポートされていますか？	565
NetScaler VPXのドキュメントにはどのようにアクセスしますか？	566
キャパシティプランニングまたはサイジング	566
NetScaler VPXでどのようなパフォーマンスが期待できますか？	566
サーバーのCPUパワーが異なることを考慮すると、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？	566
NetScaler VPXの帯域幅またはスループットの制限は、インバウンドトラフィックのみか、それともインバウンドとアウトバウンドの両方のトラフィックか？	566
同じサーバーで複数のNetScaler VPXインスタンスを実行できるか？	566
物理サーバーで複数のNetScaler VPXインスタンスが実行されている場合、NetScaler VPXインスタンスあたりの最小ハードウェア要件は何ですか？	567
同じサーバーでNetScaler VPXと他のアプリケーションをホストできますか？	567
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は向上しますか？	567
NetScaler VPXはアイドル状態なのに、なぜCPUを90%以上消費しているように見えるのですか？	568
システム要件	568
NetScaler VPXの最小ハードウェア要件は何ですか？	568
Intel VT-xとは何ですか？	569
VT-xはどのくらい一般的ですか？	569
NetScaler VPX のハードウェア互換性リスト (HCL) はありますか？	569
NetScaler VPX はどのバージョンの NetScaler OS に基づいていますか？	569
NetScaler VPX は BSD 上で動作しますが、BSD Unix がインストールされたサーバー上でネイティブに実行できますか？	569
その他の技術的な FAQ	570
複数の NIC を持つ物理サーバーでのリンクアグリゲーションは機能しますか？	570
VPX で MAC ベースの転送 (MBF) は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点はありますか？	570

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？	570
フラッシュとディスクスペースはどのように割り当てられますか？変更できますか？	570
NetScaler VPXインスタンスの容量を増やすために新しいハードドライブを追加できますか？	571
NetScaler VPXのビルド番号付けと他のビルドとの相互運用性に関して、何を期待できますか？	572
NetScaler VPXは、NetScalerアプライアンスとの高可用性セットアップの一部になり得ますか？	572
NetScaler VPXで表示されるすべてのインターフェースは、ハイパーバイザー上のインターフェース数に直接関係していますか？	572
NetScaler VPX上の準仮想化（PV）インターフェースが、基盤となる物理インターフェースが10Gであるにもかかわらず、なぜ1Gインターフェース（1/x）として表示されるのか 最大スループットは1 Gbpsに制限されるのか？	572
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のアクティブインスタンスを移動できるか？	572
fix	573

サポートマトリックスと使用ガイドライン

このドキュメントでは、特定のハイパーバイザーがVPX FIPSでサポートされていないと明示的に記載されている場合を除き、NetScaler VPXおよびVPX FIPS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、その使用ガイドラインと既知の制限についても説明します。

XenServer上のVPXインスタンス

XENSERVENバージョン	SYSID	パフォーマンス範囲
8.2 (13.0 64.x以降でサポート), 8.0, 7.6, 7.1	450000	10 Mbpsから40 Gbps

VMware ESXiハイパーバイザー上のVPXインスタンス

ESXiバージョン	ESXiリリース日 (YYYY/MM/DD)	ESXiビルド番号	NETSCALER VPXバージョン	パフォーマンス 範囲
ESXi 9.0.2	2026/01/20	25148076	13.1-61.x およ びそれ以降のビ ルド	10 Mbps から 100 Gbps まで
ESXi 8.0 アッ プデート 3k	2026/07/29	25595708	13.1-64.x およ びそれ以降のビ ルド	
ESXi 8.0 アッ プデート 3j	2026/05/27	25429389	13.1-63.x およ びそれ以降のビ ルド	
ESXi 8.0 アッ プデート 3g	2025/07/29	24859861	13.1-58.x 以降 のビルド	同上
ESXi 8.0 アッ プデート 3f	2025/07/15	24784735	13.1-58.x 以降 のビルド	同上

ESXiバージョン	ESXiリリース日 (YYYY/MM/DD)	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 3e	2025/04/10	24674464	13.1-58.x 以降 のビルド	同上
ESXi 8.0 アップデート 3d	2025/03/04	24585383	13.1-56.x 以降 のビルド	
ESXi 8.0 アップデート 3c	2025/01/23	24414501	13.1-55.x 以降 のビルド	
ESXi 8.0 アップデート 3b	2024/09/17	24280767	13.1-53.x 以降 のビルド	
ESXi 8.0 アップデート 3	2024/06/25	24022510	13.1-53.x 以降 のビルド	
ESXi 8.0 アップデート 2c	2024/05/21	23825572	13.1-53.x 以降 のビルド	
ESXi 8.0 アップデート 2b	2024/02/29	23305546	13.1-49.15、および 13.1-52.x 以降のビルド	
ESXi 8.0 アップデート 2	2023/09/21	22380479	13.1-52.x 以降 のビルド	
ESXi 8.0 アップデート 1	2023/04/18	21495797	13.1-45.x 以降 のビルド	
イーエスエックス アイ 8.0c	2023/03/30	21493926	13.1-45.x 以降 のビルド	
イーエスエックス アイ 8.0	2022/10/11	20513097	13.1-42.x 以降 のビルド	
ESXi 7.0 アップデート 3w	2025/07/15	24784741	13.1-58.x 以降 のビルド	
ESXi 7.0 アップデート 3s	2025/03/04	24585291	13.1-55.x 以降 のビルド	

ESXiバージョン	ESXiリリース日 (YYYY/MM/DD)	ESXiビルド番号	NETSCALER VPXバージョン	パフォーマンス 範囲
ESXi 7.0 アップ デート 3r	2024/12/12	24411414	13.1-55.x 以降 のビルド	
ESXi 7.0 アップ デート 3q	2024/05/21	23794027	13.1-53.x 以降 のビルド	
ESXi 7.0 アップ デート 3p	2024/03/05	23307199	13.1-52.x 以降 のビルド	
ESXi 7.0 アップ デート 3o	2023/09/28	22348816	13.1-51.x 以降 のビルド	
ESXi 7.0 アップ デート 3n	2023/07/06	21930508	13.1-49.x 以降 のビルド	
ESXi 7.0 アップ デート 3m	2023/05/03	21686933	13.1-48.x 以降 のビルド	
ESXi 7.0 アップ デート 3i	2022/12/08	20842708	13.1-37.x 以降 のビルド	
ESXi 7.0 アップ デート 3f	2022/07/12	20036589	13.1-33.x 以降 のビルド	
ESXi 7.0 アップ デート 3d	2022/03/29	19482537	13.1-27.x 以降 のビルド	
ESXi 7.0 アップ デート 3c	2022/01/27	19193900	13.1-21.x 以降 のビルド	
ESX 7.0 アップ デート 2d	2021/09/14	18538813	13.1-9.x 以降の ビルド	
ESX 7.0 アップ デート 2a	2021/04/29	17867351	13.1-4.x 以降の ビルド	

注記:

各ESXiパッチのサポートは、前の表で指定されたNetScaler VPXバージョンで検証済みであり、NetScaler VPX 13.1バージョンのそれ以降のすべてのビルドに適用されます。

Microsoft Hyper-V上のVPXインスタンス

HYPER-Vバージョン	SYSID	NETSCALER VPX バージョン	パフォーマンス範囲
2016、2019	450020	13.1-4.x 以降	10 Mbps から 3 Gbps まで
2022	同上	13.1-58.x 以降	同上
2025		13.1-60.x 以降	

Azure Local上のVPXインスタンス

コンポーネント	サポートされているバージョン / ビルド	SYSID
NetScaler VPX	13.1-61.x 以降	450020
Azure Local OSビルド	25398.1965、26100.7171、 および20349.3692	

Azure Localのリリースバージョンの詳細については、[Microsoftドキュメント](#)を参照してください。

Nutanix AHV 上の VPX インスタンス

NetScaler VPX は、[Citrix Ready パートナースhip](#) 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ベンダーがデジタルワークスペース、ネットワーク、および分析のための NetScaler テクノロジーと製品を開発および統合するのに役立つテクノロジーパートナープログラムです。

Nutanix AHV に NetScaler VPX インスタンスを展開するステップバイステップの方法の詳細については、「[Deploying a NetScaler VPX on Nutanix AHV](#)」を参照してください。

サードパーティサポート:

NetScaler® 環境で特定のサードパーティ (Nutanix AHV) 統合に関する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なる支援を求めることができます。パートナーの専任技術担当者が、問題が解決されるまで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	450070	10 Mbps から 100 Gbps
Ubuntu 16.04、Ubuntu 18.04、Ubuntu 22.04		

注意:

KVMハイパーバイザーを使用する際は、以下の点に注意してください。

- VPXインスタンスは、表1-4に記載されているハイパーバイザーの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対しては認定されていません。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でもシームレスに動作することが期待されます。動作しない場合は、トラブルシューティングとデバッグのためにサポートケースを記録してください。
- RHEL 7.6を使用する前に、KVMホストで以下の手順を完了してください。
 - /etc/default/grub を編集し、`GRUB_CMDLINE_LINUX` 変数に `"kvm_intel.preemption_timer=0"` を追加します。
 - `"# grub2-mkconfig -o /boot/grub2/grub.cfg"` コマンドを使用して grub.cfg を再生成します。
 - ホストマシンを再起動します。
- Ubuntu 18.04を使用する前に、KVMホストで以下の手順を完了してください。
 - /etc/default/grub を編集し、`GRUB_CMDLINE_LINUX` 変数に `"kvm_intel.preemption_timer=0"` を追加します。

- b. `"# grub-mkconfig -o /boot/grub/grub.cfg "` コマンドを使用して grub.cfg を再生成します。
- c. 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	10 Mbpsから30 Gbpsまで
アジュール	450020	10 Mbps から 10 Gbps まで
GCP	450070	10 Mbps から 10 Gbps まで

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 → 機能 ↓	XEN SER VER 上で の VPX	VM WA RE ESX 上で の VPX	マイ クロ ソフト トハ イ パー V上 での VPX	汎用 KVM 上の VPX						
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
Mult i-PE 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 リン グ	はい	はい 1	はい	はい 1	はい	はい	はい	はい	はい 1	はい

ハイパーバイザー → 機能 ↓	XEN SER VER 上で の VPX	VM WA RE ESX 上で の VPX	マイ クロ ソフト ハイ パー V上 での VPX	汎用 KVM 上の VPX						
サ ポー ト										
VLA Nタ ギン グ	はい	はい	はい	はい	はい	はい	はい (201 2R2 のみ)	はい	はい	はい
リン クイ ベン トの 検 出/ HA Mon	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 タ設 定	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
LAC P	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

ハイパーバイザー → 機能 ↓	XEN SER VER 上で の VPX	VM WA RE ESX 上で の VPX	マイ クロ ソフト ハイ パー V上 での VPX	汎用 KVM 上の VPX						
静的 CLA G	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
LAC P CLA G	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 ブ ラグ	いいえ	いいえ	いいえ	不可	不可	不可	不可	不可	不可	不可

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 → 機能 ↓	AWS上のVPX	AZURE上のVPX	GCP上のVPX
Multi-PE サポート	はい	はい	はい
クラスタリングサポート	はい	いいえ	はい
VLAN タグ付け	いいえ	いいえ	いいえ
リンクイベント/HAMon の検出	いいえ ²	不可 ²	不可 ²

パブリッククラウド → 機能 ↓	AWS上のVPX	AZURE上のVPX	GCP上のVPX
インターフェースパラメータ設定	不可	不可	不可
静的LA	不可	不可	不可
LACP	不可	不可	不可
静的CLAG	いいえ	いいえ	いいえ
LACP CLAG	いいえ	いいえ	いいえ
ホットプラグ	はい	いいえ	いいえ

前の2つの表で使用されている上付き数字 (1, 2, 3) は、それぞれの番号に対応する以下の点を指します。

1. クラスタリングサポートは、クライアント側およびサーバー側のインターフェースでSRIOVで利用可能ですが、バックプレーンでは利用できません。
2. インターフェースのDOWNイベントは、NetScaler VPXインスタンスには記録されません。
3. 静的LAの場合、物理ステータスがDOWNのインターフェースでもトラフィックが送信される可能性があります。

以下の点は、前の2つの表に記載されている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ースの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の場合、VM間でインターフェースを共有しないでください。
- ダイナミックルーティングの場合、リンクイベントが検出されないため、コンバージェンス時間はルーティングプロトコルに依存します。

- モニター対象のスタティックルート機能は、ルートの状態がVLANステータスに依存し、VLANステータスがリンクステータスに依存するため、モニターをスタティックルートにバインドしないと失敗します。
- リンク障害が発生した場合、高可用性環境では部分的な障害検出は行われません。リンク障害が発生した場合、高可用性スプリットブレイン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（無効化、有効化、リセット）が生成されても、リンクの物理ステータスは変更されません。スタティックLAの場合、ピアによって開始されたトラフィックはインスタンスで破棄されます。
 - VMware ESXでVLANタギング機能を使用するには、VMware ESXサーバーのvSwitchでポートグループのVLAN IDを1～4095に設定します。
- ENAインターフェースを持つVPXインスタンスではホットプラグはサポートされておらず、ホットプラグを試みるとインスタンスの動作が予測不能になる可能性があります。ホットアドは、AWS上のNetScalerでPVおよびSRIOVインターフェースの場合にのみサポートされます。
- AWS WebコンソールまたはAWS CLIインターフェースを介したホットリムーブは、NetScalerのPV、SRIOV、およびENAインターフェースではサポートされていません。ホットリムーブを試みると、インスタンスの動作が予測不能になる可能性があります。

サポートされているブラウザ

NetScaler GUIバージョン14.1および13.1へのアクセスでサポートされているブラウザについては、「[互換性のあるブラウザ](#)」を参照してください。

NetScaler VPX™でサポートされているプロセッサ

プラットフォーム	INTEL プロセッサ	AMD プロセッサ
シトリックス ハイパーバイザー	はい	いいえ
ESXi ハイパーバイザー	はい	はい
ハイパーV	はい	いいえ
KVM	はい	いいえ
AWS	はい	はい
アジュール	はい	はい

プラットフォーム	INTEL プロセッサ	AMD プロセッサ
GCP	はい	はい

NetScaler VPXでサポートされているNIC

次の表に、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を示します。

NIC → プラットフォーム ↓	メラノックス CX-3	メラノックス CX-4	メラノックス CX-5	インテル 82599 SRIOV VF	インテル X710/ X722/ XL710 SRIOV VF	INTEL X710/ XL710/ XXV710 PCIパス スルー モード
Citrix ハイパーバイザー	該当なし	該当なし	該当なし	はい	はい	いいえ
ESXi ハイパーバイザー	いいえ	はい	いいえ	はい	いいえ	はい
ハイパーV	該当なし	該当なし	該当なし	いいえ	いいえ	不可
KVM	不可	可	可	可	可	不可
AWS	N/A	N/A	N/A	可	N/A	N/A
アジュール	はい	はい	はい	該当なし	該当なし	該当なし
GCP	該当なし	該当なし	該当なし	該当なし	該当なし	該当なし

使用ガイドライン

以下の使用ガイドラインに従ってください。

- VPXインスタンスは、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。

[Performance Best Practices for VMware vSphere 6.5](#) 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。以下はその抜粋です。

- CPU/メモリ要求の高い仮想マシンを、オーバーコミットされたホストまたはクラスターに配置することはお勧めしません。
- 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オーバーコミットを許可します。ホスト上では、そのホスト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、ホスト上の仮想マシンやその他の負荷がホストが持つすべてのCPUリソースを要求する場合、レイテンシに敏感なワークロードはうまく機能しない可能性があります。この場合、CPU負荷を軽減するために、例えば一部の仮想マシンをシャットダウンしたり、別のホストに移行したり（またはDRSに自動的に移行させたり）することを検討してください。
- Citrixは、仮想マシン向けESXiハイパーバイザーの最新機能セットを利用するために、最新のハードウェア互換性バージョンを推奨しています。ハードウェアとESXiバージョンの互換性に関する詳細については、[VMwareドキュメント](#)を参照してください。
- NetScaler VPXは、レイテンシに敏感な高性能仮想アプライアンスです。期待されるパフォーマンスを提供するために、アプライアンスはvCPU予約、メモリ予約、ホスト上でのvCPUピンニングを必要とします。また、ホストでは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へのアクセス遅延、ピットボスデーモンクラッシュ、パケットドロップ、低スループットなどの問題が発生します。

ハイパーバイザーは、以下の2つの条件のいずれかが満たされた場合にオーバープロビジョニングされていると見なされます。

- 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) の総数よりも多い場合。
- プロビジョニングされたVMの総数が、pCPUの総数よりも多くのvCPUを消費する場合。

インスタンスがオーバー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、インスタンスに予約され

たリソース（CPU、メモリなど）が保証されない可能性があります。この動作は、NetScalerのCPUリソース不足を引き起こし、「**使用ガイドライン**」の最初の項目で述べた問題につながる可能性があります。管理者としては、ホストにプロビジョニングされるvCPUの総数がpCPUの総数以下になるように、ホスト上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、`esxstop` コマンド出力でVPX vCPUの `%RDY%` パラメータが0より大きい場合、ESXホストはスケジューリングオーバーヘッドを抱えているとされ、VPXインスタンスのレイテンシ関連の問題を引き起こす可能性があります。

このような状況では、`%RDY%` が常に0に戻るようにホスト上のテナンシーを減らしてください。または、リソース予約が尊重されない理由を特定するためにハイパーバイザーベンダーに連絡してください。

- ホットアドは、AWS上のNetScalerでPVおよびSRIOVインターフェースでのみサポートされています。ENAインターフェースを持つVPXインスタンスはホットプラグをサポートしておらず、ホットプラグを試みるとインスタンスの動作が予測不能になる可能性があります。
- AWS WebコンソールまたはAWS CLIインターフェースを介したホットリムーブは、NetScalerのPV、SRIOV、およびENAインターフェースではサポートされていません。ホットリムーブを試みると、インスタンスの動作が予測不能になる可能性があります。

パケットエンジン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ために、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、他の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: 割り当て済みだが未使用のCPUリソースを解放するか、解放しないか。

- **YES:** 割り当て済みだが未使用のCPUリソースを別のVMが使用できるようにします。

- **NO:** すべてのCPUリソースを、それらが割り当てられているVMのために予約します。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のVPX CPU使用率において、より高いパーセンテージを示します。
- **DEFAULT:** いいえ。

注:

すべてのNetScaler VPXプラットフォームでは、ホストシステム上のvCPU使用率は100パーセントです。この使用率を上書きするには、`set ns vpxparam -cpuyield YES` コマンドを入力します。

クラスターノードを「yield」に設定したい場合は、CCOで以下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、すべてのノードは「yield=DEFAULT」で起動します。
- 「yield=YES」に設定済みのノードを使用してクラスターが形成された場合、そのノードは「DEFAULT」のyieldを使用してクラスターに追加されます。

注記:

クラスターノードを「yield=YES」に設定したい場合、クラスター形成後にのみ設定でき、形成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以下のオプションがあります。

- **YES:** VMがメインクロックソースをCPU0からCPU1に移動することを許可します。
- **NO:** VMはメインクロックソースとして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`

現在の `vpxparam` 設定を表示します。

その他の参照情報

- Citrix Ready製品については、[Citrix Ready Marketplace](#) を参照してください。
- Citrix Ready製品のサポートについては、[Citrix Ready partners page](#) 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、[Upgrading VMware Tools](#) を参照してください。

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

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。目的のパフォーマンスを達成するには、まずVPXデータシートの推奨事項に従い、その後、このドキュメントで提供されているベストプラクティスを使用してさらに最適化してください。

VMware ESXハイパーバイザー上のNetScaler 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の詳細、およびVMware ESXハイパーバイザー上のNetScaler 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提案について説明します。

- ESXホストでの推奨構成([vpx-performance-on-esx-kvm-xen.html#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 VPX([vpx-performance-on-esx-kvm-xen.html#citrix-adc-vpx-with-e1000-network-interfaces](#))
- VMXNET3ネットワークインターフェイスを備えたNetScaler VPX([vpx-performance-on-esx-kvm-xen.html#citrix-adc-vpx-with-vmxnet3-network-interfaces](#))
- 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 VPX([vpx-performance-on-esx-kvm-xen.html#citrix-adc-vpx-with-sr-iov-and-pci-passthrough-network-interfaces](#))

ESXホストでの推奨構成

E1000、VMXNET3、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VPXで高いパフォーマンスを達成するには、次の推奨事項に従ってください。

- ESXホストにプロビジョニングされる仮想CPU (vCPU) の総数は、ESXホスト上の物理CPU (pCPU) の総数以下である必要があります。

- 良好な結果を達成するには、Non-uniform Memory Access (NUMA) アフィニティとCPUアフィニティをESXホストに設定する必要があります。

VmnicのNUMAアフィニティを見つけるには、ホストにローカルまたはリモートでログインし、次のように入力します。

```
#vsish -e get /net/pNics/vmnic7/properties | grep NUMA
```

```
Device NUMA Node: 0
```

- VMのNUMAおよびvCPUアフィニティを設定するには、[VMwareドキュメント](#)を参照してください。

E1000ネットワークインターフェイスを備えたNetScaler VPX

VMware ESXホストで以下の設定を実行します。

- VMware ESXホストで、1つのpNIC vSwitchから2つのvNICを作成します。複数のvNICはESXホストで複数のRxスレッドを作成します。これにより、pNICインターフェースのRxスループットが向上します。
- 作成した各vNICについて、vSwitchポートグループレベルでVLANを有効にします。
- vNICの送信（Tx）スループットを向上させるには、vNICごとにESXホストで個別のTxスレッドを使用します。以下のESXコマンドを使用します。
 - ESXバージョン5.5の場合：

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

- ESXバージョン6.0以降の場合：

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

- vNICのTxスループットをさらに向上させるには、デバイス（NIC）キューごとに個別のTx完了スレッドとRxスレッドを使用します。以下のESXコマンドを使用します。

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

注：

更新された設定を適用するには、VMware ESXホストを再起動してください。

pNICあたり2つのvNIC展開

以下は、より優れたネットワークパフォーマンスを提供する**pNICあたり2つのvNIC**展開モデルのサンプル構成と設定コマンドです。

pNICあたり2つのvNIC展開(/en-us/vpx/media/two-vnics-per-pnic-deployment.png)

NetScaler VPXの構成例：

前述のサンプル構成図に示されている展開を実現するには、NetScaler VPXインスタンスで以下の構成を実行します。

- クライアント側で、SNIP (1.1.1.2) をネットワークインターフェース1/1にバインドし、VLAN タグモードを有効にします。

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

bind vlan 2 -IPAddress 1.1.1.2 255.255.255.0
```

- サーバー側で、SNIP (2.2.2.2) をネットワークインターフェース 1/1 にバインドし、VLAN タグモードを有効にします。

```
bind vlan 3 -ifnum 1/2 -tagged

bind vlan 3 -IPAddress 2.2.2.2 255.255.255.0
```

- HTTP 仮想サーバー (1.1.1.100) を追加し、サービス (2.2.2.100) にバインドします。

```
add lb vserver v1 HTTP 1.1.1.100 80 -persistenceType NONE -Listenpolicy
None -cltTimeout 180

add service s1 2.2.2.100 HTTP 80 -gs1b NONE -maxClient 0 -maxReq 0 -cip
DISABLED -usip NO -useproxyport YES -sp ON -cltTimeout 180 -svrTimeout
360 -CKA NO -TCPB NO -CMP NO
```

```
bind lb vserver v1 s1
```

注:

ルートテーブルに次の2つのエントリを含めるようにしてください。

- 1.1.1.0/24 サブネット（ゲートウェイはSNIP 1.1.1.2 を指す）
- 2.2.2.0/24 サブネット（ゲートウェイはSNIP 2.2.2.2 を指す）

VMXNET3 ネットワークインターフェースを備えた NetScaler VPX

VMXNET3 ネットワークインターフェースを備えた VPX で高いパフォーマンスを実現するには、VMware ESX ホストで次の設定を行います。

- 1つの pNIC vSwitch から2つの vNIC を作成します。複数の vNIC は ESX ホスト内に複数の Rx スレッドを作成します。これにより、pNIC インターフェースの Rx スループットが向上します。
- 作成した各 vNIC の vSwitch ポートグループレベルで VLAN を有効にします。
- vNIC の送信 (Tx) スループットを向上させるには、vNIC ごとに ESX ホストで個別の Tx スレッドを使用します。次の ESX コマンドを使用します。
 - ESX バージョン 5.5 の場合:

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

- ESX バージョン 6.0 以降の場合:

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

VMware ESX ホストで、次の構成を実行します。

- VMware ESX ホストで、1つの pNIC vSwitch から2つの vNIC を作成します。複数の vNIC は ESX ホスト内に複数の Tx および Rx スレッドを作成します。これにより、pNIC インターフェースの Tx および Rx スループットが向上します。
- 作成した各vNICについて、vSwitchポートグループレベルでVLANを有効にします。
- vNICのTxスループットを向上させるには、デバイス（NIC）キューごとに個別のTx完了スレッドとRxスレッドを使用します。次のコマンドを使用します。

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

- VMの構成に次の設定を追加して、VMがvNICごとに1つの送信スレッドを使用するように構成します。

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

- VMの構成に次の設定を追加して、VMがvNICごとに最大8つの送信スレッドを使用するように構成します。

```
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。上記の構成を行う前に、十分なCPUリソースが利用可能であることを確認してください。

注:

更新された設定を適用するには、VMware ESXホストを再起動してください。

VMXNET3を**pNICあたり2つのvNIC展開**として構成できます。詳細については、「[pNICあたり2つのvNIC展開](#)」を参照してください。

VMXNET3デバイスのVMware ESXでマルチキューおよびRSSサポートを構成する

デフォルトでは、VMXNET3デバイスは8つのRxおよびTxキューのみをサポートします。VPX上のvCPUの数が8を超えると、VMXNET3インターフェース用に構成されたRxおよびTxキューの数はデフォルトで1に切り替わります。ESXで特定の構成を変更することにより、VMXNET3デバイス用に最大19のRxおよびTxキューを構成できます。このオプションにより、VPXインスタンスのvCPU全体でのパフォーマンスとパケットの均一な分散が向上します。

注:

NetScalerリリース13.1ビルド48.x以降、NetScaler VPXはESX上のVMXNET3デバイスで最大19のRxおよびTxキューをサポートします。

前提条件:

VMXNET3デバイスのESXで最大19のRxおよび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 のバージョンは 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ています。

8 を超える Rx および Tx キューをサポートするように VMXNET3 インターフェイスを構成する:

1. 仮想マシン構成ファイル (.vmx) を開きます。
2. `ethernetX.maxTxQueues` と `ethernetX.maxRxQueues` の値を構成して、Rx および TX キューの数を指定します (X は構成する仮想 NIC の数です)。構成されるキューの最大数は、仮想マシン内の vCPU の数を超えてはなりません。

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として特定されるシナリオでは、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。このような状況では、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、次に 20、というように増やします。最大制限まで直接増やすのではなく、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 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 VPX で高い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。

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

このセクションでは、Linux-KVM プラットフォーム上で NetScaler 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、構成可能なオプションと設定、およびその他の提案の詳細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 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を備えた NetScaler VPX](#)

KVM のパフォーマンス設定

KVM ホストで以下の設定を実行します。

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

VPX のメモリと CPU が同じ場所にピン留めされていることを確認してください。以下の出力では、10G NIC 「ens2」が 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、メモリが割り当てられている NUMA ドメインを示します。以下の出力では、約 10 GB の RAM が NUMA ノード #0 から割り当てられています。



NUMA ノードの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 ノードのマッピングで 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、ホストで `numactl -hardware` コマンドの出力を確認し、VPX の更新されたメモリ割り当てを確認します。



VPXのvCPUを物理コアに固定します。

- VPXのvCPUとpCPU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>
```



vCPU 0～4は物理コア 8～11にマッピングされています。

- 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
```



この出力では、8は管理CPUで、9～11はパケットエンジンです。

- vCPUとpCPUのピンニングを変更するには、2つのオプションがあります。
 - VPXの起動後、実行時に次のコマンドを使用して変更します。

```
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、以前と同様に次のタグを使用して `.xml` ファイルを編集します。

- a. ホスト上でVPXの.xmlファイルを編集します

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

- b. 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>
```

- c. VPXをシャットダウンします。
- d. 次のコマンドを使用して、NUMAノードマッピングでVMの構成情報を更新します。

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

- e. VPXの電源をオンにします。次に、ホスト上で `virsh vcpupin <VPX name>` コマンドの出力を確認して、更新されたCPUピンニングを確認します。

ホスト割り込みのオーバーヘッドを排除する。

- `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 vCPU 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 が定期的に強制終了される可能性があります。

ホストを実行している VM によって行われた VM 終了を見つけるには、`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 のオーダーで値が高い場合は、問題があることを示します。

単一の VM が存在する場合、期待される値は 30~100 K です。それ以上の場合、1つ以上のホスト割り込みベクトルが同じ pCPU にマッピングされてい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」ファイルに対して `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 にマッピングされている場合、対応するカウンタが増加します。

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リングが行わ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
```

カウンタが多数ないことを確認するために、`kvm_stat` コマンドを実行してください。

PV ネットワークインターフェイスを備えた NetScaler VPX

パラ仮想化 (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を、**pNIC あたり 2 つの vNIC 展開**として構成できます。詳細については、「[pNIC あたり 2 つの vNIC 展開](#)」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が関連付けられているNUMAドメインを特定します。
- VPXのメモリとvCPUは、同じNUMAドメインにピン留めする必要があります。
- Vhost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にバインドする:

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）のアフィニティを特定します。
3. 次のコマンドを使用して、以前に特定したNUMAドメイン内の物理コアに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、次のコマンドで特定できます。

```
[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する:

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、上記の出力をご覧ください。
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Fortville 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 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が関連付けられているNUMAドメインを特定します。
- VPXのメモリとvCPUは、同じNUMAドメインにピン留めする必要があります。

Linux KVMでのvCPUとメモリのピン留め用VPX XMLファイルの例:

```
<domain type='kvm'>

  <name>NetScaler-VPX</name>

  <uuid>138f7782-1cd3-484b-8b6d-7604f35b14f4</uuid>

  <memory unit='KiB'>8097152</memory>

  <currentMemory unit='KiB'>8097152</currentMemory>

  <vcpu placement='static'>4</vcpu>

  <cpuset>

    <vcpupin vcpu='0' cpuset='8' />

    <vcpupin vcpu='1' cpuset='9' />
```

```

        <vcupin vcpu='2' cpuset='10' />

        <vcupin vcpu='3' cpuset='11' />

    </cputune>

    <numatune>

    <memory mode='strict' nodeset='1' />

    </numatune>

</domain>

```

Citrix Hypervisor上でのNetScaler VPXインスタンス

このセクションでは、設定可能なオプションと設定の詳細、およびCitrix Hypervisor上でのNetScaler 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提案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 VPX](#)
- [準仮想化インターフェイスを備えたNetScaler 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ドメインを検索する：

```
xl info -n
```

VPXのvCPUを物理コアにピン留めします。

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

vCPUのバインディングを確認します。


```
x1 vcpu-list
```

NetScaler VMに8つ以上のvCPUを割り当てます。

8つ以上のvCPUを設定するには、Citrix Hypervisor™ コンソールから次のコマンドを実行します。

```
xe vm-param-set uuid=your_vms_uuid VCPUs-max=16  
  
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 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、次の手順に従います。

- PCIeスロットまたはNICが関連付けられている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。
- Domain-0 vCPUを残りのCPUにバインドします。

パラ仮想化インターフェイスを備えたNetScaler 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あたり2つのvNIC、およびpNICあたり1つのvNIC構成が推奨されます。

パラ仮想化（netfront）インターフェイスの最適なパフォーマンスを達成するには、次の手順に従います。

- PCIeスロットまたはNICが関連付けられている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。
- Domain-0 vCPUを同じNUMAドメインの残りのCPUにバインドします。
- vNICのホストRx/TxスレッドをDomain-0 vCPUに固定します。

ホストスレッドをDomain-0 vCPUに固定する:

1. Citrix Hypervisorホストシェルで `x1 list` コマンドを使用して、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特定します。

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

次の例では、これらの値は以下を示します。

- **vif5.0** - XenCenterでVPXに割り当てられた最初のインターフェースのスレッド（管理インターフェース）。
- **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Domain-0 vCPUにピン留めします:

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

例:

```
taskset -pc 1 29189
```

NetScaler VPXディスク容量の増加のサポート

NetScaler VPXはデフォルトで20 GBのディスク容量をサポートしています。さまざまな理由でディスク容量の制約に直面した場合、VPXディスク容量を増やすために次のオプションが利用可能です。

- プライマリディスクサイズを動的に増やす
- セカンダリディスクを追加する
- プライマリディスクサイズを手動で増やす

注:

NetScaler VPXのディスク容量を増やす機能は、オンプレミスVPXとクラウドVPXの両方の展開で利用できます。

NetScaler VPXでプライマリディスクサイズを動的に増やす

NetScalerリリース14.1ビルド21.x以降、管理者はNetScaler VPXのプライマリディスクサイズを一度に20 GBから最大1 TBまで動的に増やすことができます。その後の各増加では、再び最大1 TBまで拡張できます。プライマリディスクサイズを増やすたびにVMをシャットダウンしてください。これにより、システムは新しいディスクサイズを適切に認識し、パーティションテーブルを更新し、システムの安定性を維持できます。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のUIでプライマリディスクサイズを少なくとも1 GB拡張します。

注:

ディスクのサイズは増やすことしかできません。新しいサイズが割り当てられると、後で減らすことはできません。したがって、ディスクサイズは必要な場合にのみ増やしてください。

セカンダリディスクを追加する

NetScalerリリース13.1ビルド21.x以降、セカンダリディスクを追加することで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。セカンダリディスクを接続すると、`/var/crash` ディレクトリがこのディスクに自動的にマウントされます。セカンダリディスクはコアファイルとログの保存に使用されます。コアファイルとログファイルの既存のディレクトリは、以前と同様に機能し続けます。

注:

データ損失を避けるため、NetScalerアプライアンスをダウングレードする前に外部バックアップを取得してください。

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ドキュメント](#)

注:

Azureにデプロイされた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てください。詳細については、[ローカル一時ディスクのない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:

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を操作する一部のスクリプトは、以下の条件下で失敗する可能性があります。

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するために、「link」シェルコマンドを使用する場合。

そのようなすべてのコマンドを「ln -s」に置き換えてシンボリックリンクを使用してください。また、失敗するスクリプトもそれに応じて修正してください。

NetScaler VPXのプライマリディスクサイズを手動で増やす

ハイパーバイザーまたはクラウドプラットフォームを使用してVPXプライマリディスクサイズを手動で増やすには、以下の手順に従ってください。

1. VM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を20 GBから30 GBまたは40 GBなどのより大きな値に拡張します。Azureの場合、デフォルトのディスクサイズを32 GBから64 GBに拡張します。
3. VMの電源をオンにし、ブートプロンプトに入ります。
4. `boot -s` コマンドを使用してシングルユーザーモードでログインします。
5. ディスク容量を確認します。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メモします。以下の例では、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: 以下のコマンドを実行して、da0 MBRパーティションのサイズを10 GBの空き領域を含むように変更しましょう。

```
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/da0s1e
```

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。

クラウドでのNetScaler VPXアプライアンスの初回起動時に構成を適用する

クラウド環境でNetScaler VPXアプライアンスを初回起動する際に、NetScaler VPX構成を適用できます。この段階は、このドキュメントではプレブート段階として扱われます。そのため、ADCプールライセンスなどの特定のケースでは、VPXインスタンスの起動時間が大幅に短縮されます。この機能は、Microsoft Azure、Google Cloud Platform、およびAWSクラウドで利用できます。

ユーザーデータとは

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際、インスタンスにユーザーデータを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、一般的な自動構成タスクを実行したり、インスタンスの起動動作をカスタマイズしたり、インスタンスの起動後にスクリプトを実行したりできます。初回起動時に、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た構成を解釈します。
- 起動時に新しく追加された構成を適用します。

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クラウドインスタンスにXML形式で提供できます。ユーザーデータを提供するインターフェイスは、クラウドによって異なります。

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

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際は、> に移動し、フィールドにプレブートユーザーデータ構成を入力します。

各手順の詳細については、「[AWSウェブコンソールを使用してAWSにNetScaler VPXインスタンスを展開する](#)」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。



注記：

プレ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 を使用してプリ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、AWS ドキュメントの [インスタンスの実行](#) を参照してください。

詳細については、AWS ドキュメントの [インスタンスユーザーデータの使用](#) 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際は、**仮想マシンの作成 > 詳細** 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構成を入力します。

Azure コンソール (/en-us/vpx/media/preboot-azure-console.png)

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構成をファイルとして「--custom-data」パラメーター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 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リ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際は、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**、**ネットワーク**、**単一テナンシー**を展開します。**管理** タブに移動します。**自動化** セクションの **起動スクリプト** フィールドに、プリブートユーザーデータ構成を入力します。

GCP を使用した VPX インスタンスの作成に関する詳細については、([/ja-jp/vpx/13-1/deploy-vpx-google-cloud.html#scenario-deploy-a-multi-nic-multi-ip-standalone-vpx-instance](#)) を参照してください。



gcloud CLI を使用してプリ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 - <LOCAL_FILE_PATH> に保存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。

プリブートユーザーデータ形式

プリ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する NetScaler プリブートユーザーデータは、次の4つのセクションで構成できます。

- **<NS-CONFIG>** タグで表される NetScaler 構成。
- **<NS-BOOTSTRAP>** タグで表される NetScaler のカスタムブートストラップ。
- **<NS-SCRIPTS>** タグで表される NetScaler へのユーザースクリプトの保存。
- **<NS-LICENSE-CONFIG>** タグで表されるプールライセンス構成。

前述の4つのセクションは、ADCプリ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する際は、以下のセクションに示されている書式設定を厳密に守ってください。

注:

プリブートユーザーデータ構成全体は、以下の例に示すように **<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、プリブート段階でVPXインスタンスに適用する必要がある特定の NetScaler VPX構成を提供します。

注:

<NS-CONFIG> セクションには、有効なADC CLIコマンドが含まれている必要があります。CLIは、構文エラーや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、プリブート段階でVPXインスタンスに適用する必要がある特定の NetScaler VPX構成を提供します。

注:

<NS-CONFIG> セクションには、有効なADC CLIコマンドが含まれている必要があります。CLIは、構文エラーや形式について検証されません。

例:

次の例では、`<NS-CONFIG>` セクションに構成の詳細が記載されています。ID '5' のVLANが構成され、SNIP (5.0.0.1) にバインドされています。ロードバランシング仮想サーバー (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されている構成は、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

    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

    add lb vserver preboot_v4_101 HTTP 4.0.0.101 80 -persistenceType
    NONE -cltTimeout 180

  </NS-CONFIG>

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

NetScaler VPXインスタンスは、`<NS-CONFIG>` セクションで適用された構成で、次の図に示すように起動します。



ユーザースクリプト

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提供するには、`<NS-SCRIPTS>` タグを使用します。

`<NS-SCRIPTS>` タグ内に複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、以下のサブタグを使用してスクリプトのすべての詳細を含みます。

- `<SCRIPT-NAME>`: 保存する必要があるスクリプトファイルの名前を示します。
- `<SCRIPT-CONTENT>`: 保存する必要が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>`: スクリプトを実行するため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と、そのセクションで提供されるコマンドは「/nsconfig/nsafter.sh」に保存され、パケットエンジンが起動した後、「nsafter.sh」の実行の一部としてコマンドが実行されます。
 - `<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1つのスクリプト「script-1.sh」の詳細のみが含まれています。「script-1.sh」スクリプトは「/var」ディレクトリに保存されます。スクリプトには指定された内容が入力され、パケットエンジンが起動した後、「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されている構成は、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。



例 2:

以下の例では、`<NS-SCRIPTS>` タグには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「/var」ディレクトリに「script-1.sh」として保存されます。スクリプトには指定された内容が入力され、パケットエンジンが起動した後、「sh /var/script-1.sh」コマンドで実行されます。
- 2番目のスクリプトは「/var」ディレクトリに「file-2.txt」として保存されます。このファイルには指定された内容が入力されます。しかし、起動実行コマンド `<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されている構成は、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>

```

以下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時にNetScalerの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。プールライセンスコマンドを提供するには、`<NS-LICENSE-CONFIG>` セクション内の `<LICENSE-COMMANDS>` タグを使用します。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>` を適用すると、VPXは起動時に要求されたエディションで起動し、ライセンスサーバー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した場合、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が失敗した場合、ライセンスは、約10～12分以内にライセンスサーバーから取得されません。その結果、システムは再起動し、ライセンスのない状態になります。

例:

以下の例では、`<NS-LICENSE-CONFIG>` を適用した後、VPXは起動時にPremiumエディションで起動し、VPXはライセンスサーバー (10.102.38.214) から設定されたライセンスをチェックアウトしようとします。



上記のスクリーンショットに示されている構成は、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>

```

以下の図に示すように、「show license server」コマンドを実行して、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できます。



ブートストラップ

カスタムブートストラップ情報を提供するには、`<NS-BOOTSTRAP>` タグを使用します。`<NS-BOOTSTRAP>` セクション内で `<SKIP-DEFAULT-BOOTSTRAP>` および `<NEW-BOOTSTRAP-SEQUENCE>` タグを使用できます。このセクションは、NetScalerアプライアンスがデフォルトのブートストラップを回避するかどうかを通知します。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、以下のインターフェイス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向けインターフェイス。

ブートストラップ構成のカスタマイズ

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、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提供できます。カスタムブートストラップ情報を提供するには、`<NS-BOOTSTRAP>` タグを使用します。たとえば、管理インターフェイス (NSIP)、クライアント向けインターフェイス (VIP)、およびサーバー向けインターフェイス (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。

以下の表は、`<SKIP-DEFAULT-BOOTSTRAP>` および `<NEW-BOOTSTRAP-SEQUENCE>` タグに許可される異なる値でのブートストラップの動作を示しています。

SKIP-DEFAULT-BOOTSTRAP	NEW-BOOTSTRAP-SEQUENCE	ブートストラップの動作
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、 <code><NS-BOOTSTRAP></code> 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。 <code><NS-CONFIG></code> セクションで提供され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指定する
- インターフェイスの詳細をIPアドレスとサブネットマスクとともに指定する
- `<NS-CONFIG>` セクションでブートストラップ関連コマンドを指定する

方法1：インターフェイスの詳細のみを指定する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、およびサーバー向けインターフェイスを指定しますが、それらの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を照会することによって入力されます。

AWS のカスタムブートストラップの例

カスタムブートストラップシーケンスは、次の例に示すように指定します。詳細については、「[クラウドインスタンスでプリ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) として、Eth0 インターフェイスはクライアントインターフェイス (VIP) として、Eth2 インターフェイスはサーバーインターフェイス (SNIP) として割り当てられます。 `<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ており、IP アドレスとサブネットマスクの詳細は含まれていません。



VM インスタンスが作成されたら、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 instances** に移動し、カスタムブートストラップ情報を提供して作成したインスタンスを選択します。
2. **Description** タブで、次の図に示すように、各ネットワークインターフェイ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 NetScaler VPX インスタンスに適用されたネットワークインターフェイスを確認できます。



Azure のカスタムブートストラップの例

カスタムブートストラップシーケンスは、次の例に示すように指定します。詳細については、「[クラウドインスタンスでプリ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 インターフェイスはクライアントインターフェイス (VIP) として、Eth0 インターフェイスはサーバーインターフェイス (SNIP) として割り当てられます。 `<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ており、IP アドレスとサブネットマスクの詳細は含まれていません。



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 instance > Networking** 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。

「Azureサーバー方式1」 (/en-us/vpx/media/azure-server-method1.png)

「Azureクライアント方式1」 (/en-us/vpx/media/azure-client-method1.png)

「Azure管理方式1」 (/en-us/vpx/media/azure-mgmt-method1.png)

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。

「Azure show nsipコマンド」 (/en-us/vpx/media/azure-show-nsip-method1.png)

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リ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インターフェースは管理インターフェース (NSIP) として、Eth0インターフェースはクライアントインターフェース (VIP) として、Eth2インターフェースはサーバーインターフェース (SNIP) として割り当てられます。`<NS-BOOTSTRAP>` セクションには、インターフェースの詳細のみが含まれており、IPアドレスとサブネットマスクの詳細は含まれていません。

「GCP方式1」 (/en-us/vpx/media/gcp-custom-bootstrap-method1.png)

GCPポータルでVMインスタンスが作成されたら、次のようにネットワークインターフェースのプロパティを確認できます。

1. カスタムブートストラップ情報を提供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ースのプロパティに移動し、次のようにNICの詳細を確認します。

「GCP方式1」 (/en-us/vpx/media/gcp-method1.png)

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NetScaler VPXインスタンスに適用されたネットワークインターフェースを確認できます。

GCPでのNSIP表示方法1(/en-us/vpx/media/gcp-show-nsip-method1.png)

方式2：インターフェース、IPアドレス、およびサブネットマスクを指定することによるカスタムブートストラップ

管理インターフェース、クライアント向けインターフェース、サーバー向けインターフェースを、それぞれのIPアドレスとサブネットマスクとともに指定します。

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、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には、以下の詳細を指定します。

- **管理インターフェース:** インターフェース - Eth1、NSIP - 172.31.52.88、サブネットマスク - 255.255.240.0
- **クライアント向けインターフェース:** インターフェース - Eth0、VIP - 172.31.5.155、サブネットマスク - 255.255.240.0。
- **サーバー向けインターフェース:** インターフェース - Eth2、SNIP - 172.31.76.177、サブネットマスク - 255.255.240.0。

AWSカスタムブートストラップ方法2(/en-us/vpx/media/aws-custom-bootstrap-method2.png)

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。サブネットマスクを確認するには、「`show route`」コマンドを実行します。

AWSでのnsip表示（方法2）(/en-us/vpx/media/aws-show-nsip-method2.png)

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示されており、デフォルトのブートストラップはスキップされます。インターフェースの詳細とIPアドレスおよびサブネットマスクを次のように指定します。

- 管理インターフェース (eth2)、NSIP (172.27.2.53)、およびサブネットマスク (255.255.255.0)
- クライアント向けインターフェース (eth1)、VIP (172.27.1.53)、およびサブネットマスク (255.255.255.0)
- サーバー向けインターフェース (eth0)、SNIP (172.27.0.53)、およびサブネットマスク (255.255.255.0)

Azureカスタムブートストラップ方法2(/en-us/vpx/media/azure-custom-bootstrap-method2-new.png)

NetScaler VPXインスタンスが3つのネットワークインターフェー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、以下の図に示すように、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

次の例では、ADCの新しいブートストラップシーケンスが示されており、デフォルトのブートストラップはスキップされています。インターフェースの詳細とIPアドレスおよびサブネットマスクを次のように指定します。

- 管理インターフェース (eth2)、NSIP (10.128.4.31)、およびサブネットマスク (255.255.255.0)
- クライアント向けインターフェース (eth1)、VIP (10.128.0.43)、およびサブネットマスク (255.255.255.0)
- サーバー向けインターフェース (eth0)、SNIP (10.160.0.75)、およびサブネットマスク (255.255.255.0)



カスタムブートストラップを使用してGCPポータルでVMインスタンスが作成された後、ネットワークインターフェー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提供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ー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。



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>` セクションでブートストラップ関連コマンドを指定することによるカスタムブートストラップ

`<NS-CONFIG>` セクションでブートストラップ関連コマンドを指定できます。`<NS-BOOTSTRAP>` セクションでは、`<NS-CONFIG>` セクションでブートストラップコマンドを実行するために、`<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を指定してください。

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、`<NS-CONFIG>` セクションでブートストラップ関連コマンドが提供されています。

`<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た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ています。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されている構成は、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 インスタンスが作成された後、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 ポータル > 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提供して作成したインスタンスを選択します。
2. **Description** タブで、次の図に示すように、各ネットワークインターフェイ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NetScaler VPXインスタンスに適用されたネットワークインターフェースを確認できます。



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コマンドが `<NS-CONFIG>` セクションに提供されています。

`<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た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ています。

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) および DNS サーバーは、プライマリインターフェース (Eth0) を介してのみアクセス可能です。したがって、Eth0 インターフェースが管理インターフェース (NSIP) として使用されていない場合でも、IMDS または DNS アクセスが機能するためには、Eth0 インターフェースを少なくとも SNIP として構成する必要があります。Eth0 のゲートウェイを介した 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
```

```
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>
```

NetScaler VPX インスタンスが3つのネットワークインターフェースで作成されていることがわかります。**Azure portal > VM instance > Networking** に移動し、以下の図に示すように、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コマンドは `<NS-CONFIG>` セクションで提供されています。

`<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た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ています。

GCP メソッド3(/en-us/vpx/media/gcp-method3.png)

前のスクリーンショットに示されている設定は、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インスタンスが作成された後、ネットワークインターフェー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提供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ースのプロパティに移動し、図に示されているNICの詳細を確認します。

GCPコンソールに表示されているNICの詳細(/en-us/vpx/media/gcp-nic-method3.png)

ADC CLIで `show nsip` コマンドを実行し、前の `<NS-CONFIG>` セクションで提供された設定がADCアプライアンスの初回起動時に適用されていることを確認できます。

NSIP出力の表示(/en-us/vpx/media/gcp-show-nsip-method3.png)

AWSおよびAzureにおけるNICの接続と切断の影響

AWSとAzureは、ネットワークインターフェースをインスタンスに接続したり、インスタンスからネットワークインターフェースを切断したりするオプションを提供しています。インターフェースの接続または切断は、インターフェースの位置を変更する可能性があります。そのため、CitrixはNetScaler VPXインスタンスからインターフェースを切断しないことを推奨します。カスタムブートストラップが設定されているときにインターフェースを切断または接続すると、NetScaler VPXインスタンスは、管理インターフェースの位置にある新しく利用可能なインターフェースのプライマリIPをNSIPとして再割り当てします。切断したインターフェースの後に利用可能なインターフェースが他にない場合、最初のインターフェースがNetScaler VPXインスタンスの管理インターフェースになります。

例えば、NetScaler VPXインスタンスが3つのインターフェース（Eth0 (SNIP)、Eth1 (NSIP)、Eth2 (VIP)）で起動されたとします。管理インターフェースであるEth1インターフェースをインスタン

スから切断すると、ADCは次に利用可能なインターフェース（Eth2）を管理インターフェースとして設定します。これにより、NetScaler VPXインスタンスはEth2インターフェースのプライマリIPを介して引き続きアクセスされます。Eth2も利用できない場合、残りのインターフェース（Eth0）が管理インターフェースになります。したがって、NetScaler VPXインスタンスへのアクセスは継続されます。

インターフェースの割り当てを次のように変更して考えてみましょう。Eth0 (SNIP)、Eth1 (VIP)、Eth2 (NSIP)。Eth2 (NSIP)を切断した場合、Eth2の後に新しいインターフェースが利用できないため、最初のインターフェース（Eth0）が管理インターフェースになります。

パブリッククラウドプラットフォームでのSSL-TPSパフォーマンスの向上

パケットエンジン (PE) の重みを均等に分散することで、AWSおよびGCPクラウドでより優れたSSL-TPSパフォーマンスを得ることができます。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を搭載したNetScaler VPXインスタンスは、PEの重みがデフォルトで均等に分散されているため、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ています。この機能は、Azureインスタンスのパフォーマンスを向上させません。

NetScaler CLIを使用してPEモードを構成する

PEモードを設定した後、設定変更を有効にするにはシステムを再起動する必要があります。

コマンドプロンプトで、次のように入力します。

```
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。

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設定では、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。

```
Operation not permitted on CLIP
```

構成されている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で同時マルチスレッディングを構成する

NetScaler®は、管理機能とデータプレーン機能にそれぞれ異なる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、残りの利用可能なコアはデータプレーン機能に割り当てられます。

次の図は、4コアのNetScaler VPXの簡略化された図を示しています。

図1. 4コアシステムにおけるNetScalerの管理およびデータプレーンのワークロード

SMT機能なしのNetScaler(/en-us/vpx/media/ns-4core-without-smt.png)

前の図は、利用可能なコア全体でのNetScaler機能の分散を示していますが、必ずしも基盤となるハードウェアを正確に描写しているわけではありません。ほとんどの最新のx86 CPUは、Intel Hyperthreading (HT) または AMD simultaneous multithreading (SMT) として商業的に知られている機能を通じて、物理コアあたり2つの論理コアを提供します。

次の図は、SMTが無効化された最新のCPU上で動作するNetScaler VPXを示しています。各CPU コアは、一般的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。各スレッドは、独自のレプリケートされたリソースセットと分割されたリソースの一部を持ち、兄弟スレッドと共有リソースを競合します。

図2. SMTが無効化された4コア/8スレッドシステムにおけるNetScalerの管理およびデータプレーンのワークロード

SMT機能が無効化されたNetScaler(/en-us/vpx/media/ns-4core-with-smt-disabled.png)

次の図は、SMTが有効化された最新のCPU上で動作するNetScaler VPXを示しています。

図3. SMTが有効化された4コアシステムにおけるNetScalerの管理およびデータプレーンのワークロード

SMT機能が有効化されたNetScaler(/en-us/vpx/media/ns-4core-with-smt-enabled.png)

SMTを有効にすると、NetScalerのパフォーマンスは次の方法で向上します。

- すべての物理コアでデータプレーン機能を実行する。
- 管理プレーン機能を兄弟スレッドに移動する。

- 柔軟なリソース制限メカニズムを導入し、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のを防ぎます。

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、および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	エムファイブ、エムファイブ エヌ、シーファイブ、シー ファイブエヌ	13.1-48.x以降

注：

SMT機能を有効にすることで、サポートされているタイプでNetScaler VPXのパフォーマンスが向上します。

制限事項

SMT機能は、NetScalerアプライアンスで利用可能なvCPUを実質的に2倍にします。NetScalerアプライアンスがそれら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。

例えば、図3に示すNetScaler VPXを考えてみましょう。スループットベースのライセンスを使用する場合、SMT機能で8つのvCPUを有効にするには、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ライセンスで十分でした。vCPUライセンスを使用する場合、NetScaler VPXは、適切な動作のためにvCPU数の2倍のライセンスをチェックアウトするように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の構成

SMT 機能を有効にする前に、お使いの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、`.smt_handling` という名前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 はその機能が利用可能であり、かつ有効になっていることを示します。

```
smt_handling is set to "1"

> shell sysctl -a | grep smt_handling

netscaler.smt_handling_platform: 1

netscaler.smt_handling: 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 はその機能が利用可能であるが、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
```

トラブルシューティング

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

```
...
```

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

```
>
```

```
...
```

このコマンドは、次のいずれかの出力を返します。

- SMT 機能が見つかりません。

`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、以下のいずれかの理由によりサポートされていません。

- NetScaler VPXが13.1-48.xまたは14.1-12.xより古い。
- 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（例：vCPU数が8より多い）。

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

- 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)  
  
netscaler.smt_handling: 0 (not enabled)
```

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

ベアメタルは、物理的な分離を提供し、クラウド環境に完全に統合された、完全に専用の物理サーバーです。これはシングルテナントサーバーとも呼ばれます。シングルテナンシーにより、ノージーネイバー効果を回避できます。ベアメタルでは、唯一のユーザーであるため、ノージーネイバー効果を経験することはありません。

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、仮想化の柔軟性と信頼性を得るために、ワークロードを個別の仮想マシンに仮想化することです。

ベアメタルサーバーにNetScaler VPXインスタンスを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取得する必要があります。

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

ベアメタルサーバーにNetScaler VPXインスタンスを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ベアメタルサーバーをクラウドベンダーから取得する必要があります。そのベアメタルサーバーには、NetScaler VPXインスタンスをデプロイする前に、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へのNetScaler VPXインスタンスのインストールに関する詳細については、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/XenServerにNetScaler VPXインスタンスをインストールする

Citrix Hypervisor/XenServerにVPXインスタンスをインストールするには、まず十分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、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されているCitrix XenCenterを使用します。

Hypervisorの詳細については、[Citrix Hypervisorドキュメント](#)を参照してください。

次の図は、Hypervisor上のNetScaler 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/XenServer上のNetScaler VPXインスタンス



HypervisorにNetScaler 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、以下を実行してください。

- 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入手します。仮想アプライアンスの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Hypervisorプラットフォームの最小ハードウェア要件について説明しています。

表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	要件
CPU	仮想化支援 (Intel-VT) が有効な 2 つ以上の 64 ビット x86

コンポーネント	要件
	CPU。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が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てください。詳細については、BIOS ドキュメントを参照してください。
RAM	3 GB
ディスク容量	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 ホスト制御ドメイン用に 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 インスタンスおよびその他の仮想マシンで利用できます。
NIC	1 Gbps NIC 1 つ。推奨: 1 Gbps NIC 2 つ。

詳細については、[XenServer 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。

表 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	要件
メモリ	2 GB

コンポーネント	要件
仮想CPU (vCPU)	2
仮想ネットワークインターフェース	2

注:

NetScaler VPXインスタンスを本番環境で使用する場合、Citrixは、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、CPU優先度（仮想マシンプロパティ内）を最高レベルに設定することを推奨しています。

XenCenterのシステム要件

XenCenterはWindows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のインストールに関する詳細については、以下のHypervisor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HypervisorにNetScaler VPXインスタンスをインストールする

HypervisorとXenCenterをインストールして構成した後、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利用可能なメモリ量によって異なります。

XenCenter を使用して Hypervisor に NetScaler 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. サーバーメニューで、**追加**をクリックします。
3. **新しいサーバーの追加**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Hypervisor の IP アドレスまたは DNS 名を入力します。
4. **ユーザー名とパスワード**のテキストボックスに管理者資格情報を入力し、**接続**をクリックします。ナビゲーションペインにHypervisor 名が緑色の丸で表示され、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**メニューで、**インポート**をクリックします。
7. **Import** 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に移動します。Exported VM オプションが選択されていることを確認し、**Next** 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 Hypervisor 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、Import をクリックしてインポートプロセスを開始します。
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、**次へ**をクリックします。
11. インポートプロセスを完了するには、**完了**をクリックします。

注:

インポートプロセスのステータスを表示するには、**ログ**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 から 11 を繰り返します。

注記:

VPXインスタンスの初期設定後、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。

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

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

以下のNICがサポートされています。

- インテル 82599 10G
- インテル X710 10G
- インテル XL710 40G

制限事項

Citrix Hypervisor™は、SR-IOVインターフェー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に関する制限事項は、以下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は以下の機能をサポートしていません。

- L2モード切り替え
- クラスターリング
- 管理者パーティショニング \[共有VLANモード\]
- 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
- クラスター環境でのIPv6プロトコル

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

Intel X710 10GおよびIntel XL710 40G NICには、以下の制限事項があります。

- L2モード切り替えはサポートされていません。
- 管理者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。
- クラスター環境では、XL710 NICがデータインター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。
- インターフェースが切断および再接続されると、インターフェースリストが再順序付けされます。
- 速度、デュプレックス、自動ネゴシエーションなどのインターフェースパラメータ設定はサポートされていません。
- Intel X710 10GおよびIntel XL710 40G NICの両方で、インターフェースは40/xインターフェースとして起動します。
- VPXインスタンスでは、最大16個のIntel X710/XL710 SR-IOVインターフェースのみがサポートされます。

注：

Intel X710 10GおよびIntel XL710 40G NICで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）で信頼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、以下を確実に行ってください。

- ホストに Intel 82599 NIC (NIC) を追加します。
- `/etc/modprobe.d/blacklist.conf` ファイルに次のエントリを追加して、`ixgbevf` ドライバーをブロックリストに登録します。

ブロックリスト ixgbevf

- `/etc/modprobe.d/ixgbe` ファイルに次のエントリを追加して、SR-IOV 仮想機能 (VF) を有効にします。

```
options ixgbe max\_vfs=\&lt;number\_of\_VF\>
```

ここで、`\<number\_VF\>` は作成する SR-IOV VF の数です。

- BIOS で SR-IOV が有効になっていることを確認します。

注:

IXGBE ドライバーバージョン 3.22.3 が推奨されます。

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る

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 NetScaler VPX インスタンスに割り当てます。

```
xe host-call-plugin plugin=iovirt host-uuid=\&lt;Xen host UUID\> fn=assign\_free\_vf  
args:uuid=\&lt;NetScaler VM UUID\> args:ethdev=\&lt;interface name\> args:mac=\&lt;Mac  
addr\>
```

説明:

- `\<Xen host UUID\>` は Citrix Hypervisor ホストの 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=` パラメータで使用したいMACアドレスを指定します。指定しない場合、`iovirt` スクリプトがMACアドレスをランダムに生成して割り当てます。また、SR-IOV VFをリンクアグリゲーションモードで使用したい場合は、MACアドレスを `00:00:00:00:00:00` として指定してください。

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インスタンスから割り当て解除する

誤ったSR-IOV VFを割り当てた場合、または割り当て済みのSR-IOV VFを変更したい場合は、SR-IOV VFをNetScaler VPXインスタンスから割り当て解除し、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NetScaler 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_UUID>
```

説明:

- `<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る

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

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

```
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ネットワークが構成されるNICのPIFユニバーサル一意識別子 (UUID) を特定します。

```
xe pif-list

          uuid ( R0)  : e2874343-f1de-1fa7-8fef-98547c348783

          device ( R0): eth18

          currently-attached ( R0): true

          VLAN ( R0): -1

          network-uuid ( R0): f865bd85-44dd-b865-ab65-dcd6ae28c16e
```

3. ネットワークをSR-IOVネットワークとして構成します。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-98547c3487831629b44f-832a-084e-d67d-5d6d314d5e0f
```

SR-IOVネットワークパラメータに関する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-TOP0 ~]# xe network-sriov-param-list
uuid=1629b44f-832a-084e-d67d-5d6d314d5e0f

      uuid ( R0): 1629b44f-832a-084e-d67d-5d6d314d5e0f

      physical-PIF ( R0): e2874343-f1de-1fa7-8fef-98547c348783

      logical-PIF ( R0): 85d52771-5814-c62d-45fa-f37b536144ff

      requires-reboot ( R0): false

      remaining-capacity ( R0): 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=8ee59b73-7319-6998-cd69-b9fa3e8d7503 vm-uuid=b507e8a6-
f5ca-18eb-561d-308218a9dd68

3e1e2e58-b2ad-6dc0-61d4-1d149c9c6466
```

注:

VMのNICインデックス番号は0から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-TOP0 ~]# xe vm-list

      uuid ( R0): b507e8a6-f5ca-18eb-561d-308218a9dd68

      name-label ( RW): sai-vpx-1

      power-state ( R0): 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を削除する

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. Citrix Hypervisorホストで次のコマンドを実行して、VIFを破棄します。

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

例:

```
[root@citrix-XS82-T0P0 ~]# xe vif-destroy uuid=3e1e2e58-b2ad-6dc0-61d4-1d149c9c6466
```

SR-IOVインターフェイスでリンクアグリゲーションを構成する

リンクアグリゲーションモードでSR-IOV仮想機能（VF）を使用するには、作成した仮想機能のなりすまし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なりすましチェックを無効にします。

```
ip link set \&lt;interface\_name\> vf \&lt;VF\_id\> spoofchk off
```

ここで、：

- \<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なりすまし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構成します。手順については、「[リンクアグリゲーションの構成](#)」を参照してください。

重要:

SR-IOV VFをNetScaler VPXインスタンスに割り当てる際は、VFにMACアドレス 00:00:00:00:00:00を指定してください。

SR-IOVインターフェイスでVLANを構成する

SR-IOV仮想機能でVLANを構成できます。手順については、「[VLANの構成](#)」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。

その他の参照

[SR-IOV対応NIC](#)

[SR-IOVネットワークの追加](#)

NetScaler VPXインスタンスをVMware ESXにインストールする

VMware ESXにNetScaler VPXインスタンスをインストールする前に、十分なシステムリソースを持つマシンにVMware ESXサーバーがインストールされていることを確認してください。VMware ESXiにNetScaler VPXインスタンスをインストールするには、VMware vSphereクライアントを使用します。このクライアントまたはツールは、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。

このセクションには、次のトピックが含まれています。

- 前提条件
- VMware ESXへのNetScaler VPXインスタンスのインストール

重要:

標準のVMware Toolsをインストールしたり、NetScaler VPXインスタンスで利用可能なVMware Toolsの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 Toolsは、NetScaler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アプライアンスの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」([/ja-jp/citrix-adc/13-1/licensing.html](http://ja-jp/citrix-adc/13-1/licensing.html))を参照してください。

VMware ESXのハードウェア要件

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

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	要件
CPU	仮想化支援 (Intel-VT) が有効な64ビットx86 CPUを2つ以上。NetScaler VPXインスタンスを実行するには、VMware ESX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。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ます。
RAM	2 GB VPX。重要な展開の場合、システムがメモリ制約のある環境で動作するため、VPXに2 GBのRAMは推奨しません。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。推奨は4 GB RAMまたは8 GB RAMです。
ディスク容量	ESXiセットアップのためのVMwareの最小サーバー要件より20 GB多く。最小サーバー要件については、VMwareのドキュメントを参照してください。
ネットワーク	1 Gbps NIC (NIC) 1つ。1 Gbps NIC 2つを推奨。

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ースまたはPCIパススルーサポートの場合、以下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
- AMD-VをサポートするAMDプロセッサ

- I/Oメモリ管理ユニット (IOMMU) またはSR-IOVがBIOSで有効になっている

以下のNICはSR-IOVモードでサポートされています。

- Mellanox ConnectX-4 NIC (NetScalerリリース13.1-42.x以降)
- インテル 82599 ネットワークインターフェースカード

以下の表に、VMware ESXサーバーが各VPX nCore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。

表2. NetScaler 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	要件
メモリ	4 GB
仮想CPU (vCPU)	2
仮想ネットワークインターフェース	ESXでは、VPXハードウェアがバージョン7以降にアップグレードされている場合、最大10個の仮想ネットワークインターフェースをインストールできます。
ディスク容量	20 GB

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です。

VPX仮想アプライアンスを本番環境で使用する場合、メモリはすべて割り当てて予約する必要があります。ESXの1つのCPUコアの速度と同等以上のCPUサイクル（MHz単位）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は、WindowsおよびLinux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サーバーと同じマシンでは実行できません。次の表に、最小システム要件を示します。

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	要件
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 Compatibility Matrixes」 PDFファイルを検索してください。
CPU	750 MHz; 1ギガヘルツ (GHz) 以上を推奨
RAM	1 GB。2 GBを推奨
NIC (NIC)	100 Mbps以上のNIC

OVF Tool 1.0のシステム要件

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サーバーと同じマシンでは実行できません。次の表に、最小システム要件を示します。

表4. OVFツールインストールの最小システム要件

コンポーネント	要件
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 Tool User Guide」 PDFファイルを検索してください。
CPU	最小750 MHz、1 GHz以上を推奨
RAM	1 GB 最小、2 GB 推奨
NIC (NIC)	100 Mbps 以上の NIC

OVF のインストールに関する情報については、<<http://kb.vmware.com/>> で「OVF Tool User Guide」 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、Open Virtual Machine (OVF) 形式標準に準拠しています。ファイルは Citrix ウェブ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、**New Users link** をクリックして、指示に従って Citrix アカウントを作成してください。

ログオンしたら、Citrix® ホームページから次のパスに移動します。

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てください。

- NSVPX-ESX-`<release number>-<build number>-disk1.vmdk` (例: NSVPX-ESX-13.0-71.44_nc_64-disk1.vmdk)
- NSVPX-ESX-`<release number>-<build number>.ovf` (例: NSVPX-ESX-13.0-71.44_nc_64.ovf)
- NSVPX-ESX-`<release number>-<build number>.mf` (例: NSVPX-ESX-13.0-71.44_nc_64.mf)

VMware ESX に NetScaler 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 クライアントを使用して 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、VMware ESX を実行しているハードウェアで利用可能なメモリ量によって異なります。

VMware vSphere Client を使用して VMware ESX に NetScaler 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 VMware vSphere クライアントを起動します。
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. **User Name** および **Password** テキストボックスに管理者資格情報を入力し、「Login」をクリックします。
4. **File** メニューで、**Deploy OVF Template** をクリックします。
5. **Deploy OVF Template** ダイアログボックスの **Deploy from file** で、NetScaler VPX インスタンスのセットアップファイルを保存した場所に移動し、.ovf ファイルを選択して **Next** をクリックします。

6. 仮想アプライアンスのOVFテンプレートに表示されているネットワークを、ESXホストで構成したネットワークにマッピングします。**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、インストールが成功したことを知らせるポップアップウィンドウが表示されます。
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NetScaler VPXインスタンスを選択し、右クリックメニューから **Power On** を選択します。
8. VMが起動したら、コンソールからNetScaler IP、ネットマスク、およびゲートウェイアドレスを構成します。構成が完了したら、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、手順6から手順8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 Client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タギングを有効にするには、vSwitchでポートグループのVLAN IDをすべて（4095）に構成します。vSwitchでVLAN IDを設定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。

VMware vMotionを使用してNetScaler 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。

以下の使用ガイドラインに従ってください。

- VMware は、PCI Passthrough 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の 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、インスタンスがサポートされているインターフェイスで構成されていることを確認してください。
- 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。

NetScaler VPX インスタンスをVMXNET3ネットワークインターフェースを使用するように構成する

VMware ESX に NetScaler VPX インスタンスをインストールして構成した後、VMware vSphere Web Client を使用して、仮想アプライアンスが VMXNET3 ネットワークインターフェースを使用するように構成できます。

VMware vSphere Web Client を使用して、NetScaler VPX インスタンスが VMXNET3 ネットワークインターフェースを使用するように構成するには、次の手順を実行します。

1. vSphere Web Client で、 を選択します。
2. NetScaler VPX インスタンスの互換性設定を次のように 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、 > を選択します。
- c. ダイアログボックスで、 ドロップダウンリストから を選択し、 をクリックします。

3. NetScaler VPX インスタンスを右クリックし、 をクリックします。

設定の編集メイン(/en-us/vpx/media/edit_settings_main.png)

4. ダイアログボックスで、 セクションをクリックします。

CPU設定の編集メイン(/en-us/vpx/media/edit_settings_cpu_main.png)

5. セクションで、以下を更新します。

- CPU数
- ソケット数
- 予約
- 制限
- 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. ソケットあたりのコア数ドロップダウンリストで、ソケットの数を選択します。
- c. (オプション) CPUホットプラグフィールドで、チェックボックスを選択または選択解除します。

注:

Citrix®は、デフォルト（無効）を受け入れることを推奨します。

d. 予約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、と最大値として表示されている数値を選択します。



6. 「メモリ」セクションで、以下を更新します。

- RAMのサイズ
- 予約
- 制限
- 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である必要があります。たとえば、vCPUの数が4の場合、RAM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、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は4 x 4 GB = 16 GBになります。

メモリ設定の編集(/en-us/vpx/media/edit_settings_mem_set1.png)

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 (すべてロック済み)」チェックボックスを選択します。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、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は4 x 4 GB = 16 GBになります。

メモリ設定1の編集(/en-us/vpx/media/edit_settings_mem_set2.png)

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。

「メモリ設定」 (/en-us/vpx/media/edit_settings_mem_set3.png)

d. 「共有」ドロップダウンリストで、「カスタム」と最大値として表示されている数値を選択します。

「メモリ編集設定」 (/en-us/vpx/media/edit_settings_mem_set4.png)

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。

「VMネットワークインターフェイス」 (/en-us/vpx/media/vmxnet3_1.png)

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. 「アダプタータイプ」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は共存できません。E1000ネットワークインターフェイスを削除し、VMXNET3 (0/1) を管理インターフェイスとして使用してください。

「VMネットワークの選択」 (/en-us/vpx/media/vmxnet3_select.png)

9. 「OK」をクリックします。

10. NetScaler VPXインスタンスの電源をオンにします。

11. NetScaler VPXインスタンスの電源がオンになったら、次のコマンドを使用して構成を確認できます。

インターフェースの概要を表示

出力には、構成したすべてのインターフェイスが表示されているはずです。

```
> 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
4	L0/1	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した後、VMware ESXハイパーバイザーが、NICがVPXアプライアンスに提示される順序を変更する可能性があります。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アプライアンスへの管理接続が失われる可能性があります。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それに応じて変更してください。

これはVMware ESXハイパーバイザーの制限事項です。

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

VMware ESXにNetScaler 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、仮想アプライアンスがシングルルートI/O仮想化（SR-IOV）ネットワークインターフェースを使用するように構成できます。

制限事項

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

- ESX VPXでIntel 82599 10G NICを使用するSR-IOVインターフェースでは、以下の機能はサポートされていません。
 - L2モードスイッチング
 - 静的リンクアグリゲーションとLACP
 - クラスタリング
 - 管理パーティショニング \[共有VLANモード\]
 - 高可用性
 - ジャンボフレーム
 - IPv6
- KVM VPXでIntel 82599 10G NICを使用するSR-IOVインターフェースでは、以下の機能はサポートされていません。
 - 静的リンクアグリゲーションとLACP
 - L2モードスイッチング
 - クラスタリング
 - 管理パーティショニング \[共有VLANモード\]
 - 高可用性
 - ジャンボフレーム
 - IPv6
 - ハイパーバイザーでのSR-IOV VFインターフェースに対するVLAN設定は、`ip link` コマンドを介してはサポートされていません

前提条件

- 以下のいずれかのNICをESXホストに追加していることを確認してください。
 - Intel 82599 NIC、IXGBEドライババージョン3.7.13.7.14iov以降を推奨します。
 - Mellanox ConnectX-4 ニック

- ホストの物理アダプターでSR-IOVを有効にします。

ホストの物理アダプターでSR-IOVを有効にするには、次の手順に従います。

- vSphere Web Clientで、ホストに移動します。
- 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ステータスフィールドには、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。



- 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。

SR-IOVの編集(/en-us/vpx/media/sr-iov_img1.jpg)

- SR-IOVの下で、**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。

SR-IOVを有効にする(/en-us/vpx/media/sr-iov_img3.jpg)

- 仮想機能の数**フィールドに、アダプタ用に構成する仮想機能の数を入力します。

SR-IOV仮想機能(/en-us/vpx/media/sr-iov_img4.jpg)

- OK**をクリックします。
- ホストを再起動します。

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

注：

Citrixは、DVSと **Portgroups** でのみSR-IOV構成を認定しています。

VMware vSphere Web Client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 VPXインスタンスを構成するには：

- vSphere Web Clientで、**ホストとクラスタ**を選択します。
- 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、「**Compatibility > Upgrade VM Compatibility**」を選択します。

c. 「**Configure VM Compatibility**」ダイアログボックスで、「**Compatible with**」ドロップダウンリストから「**ESXi 5.5 and later**」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、「**Edit Settings**」をクリックします。



4. 「**<virtual_appliance> - Edit Settings**」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数
- ソケット数
- 予約
- 制限
- 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. 「**ソケットあたりのコア数**」ドロップダウンリストで、ソケットの数を選択します。

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®は、デフォルト（無効）を受け入れることを推奨します。

d. 「**予約**」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」と最大値として表示されている数値を選択します。



6. 「メモリ」セクションで、次の設定を更新します。

- RAMサイズ
- 予約
- 制限
- 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。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、RAM = 4 x 2 GB = 8 GBです。

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、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 x 4 GB = 16 GBです。



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 (すべてロック済み)** チェックボックスを選択します。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、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 x 4 GB = 16 GBです。



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ている数値を選択します。



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. 「**New Network**」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、以下を実行します。

a. **Adapter Type**ドロップダウンリストで、**SR-IOV passthrough**を選択します。



b. **Physical function**ドロップダウンリストで、**Portgroup** にマッピングされている物理アダプターを選択します。



c. **Guest OS MTU Change**ドロップダウンリストで、**Disallow**を選択します。

9. 「**<virtual_appliance> - Edit Settings**」ダイアログボックスで、**VM Options**タブをクリックします。

10. **VM Options**タブで、**Advanced**セクションを選択します。**Latency Sensitivity**ドロップダウンリストから、**High**を選択します。



11. **OK**をクリックします。

12. NetScaler VPXインスタンスの電源をオンにします。

13. NetScaler VPXインスタンスの電源がオンになったら、次のコマンドを使用して構成を確認できます。

インターフェースの概要を表示

出力には、構成したすべてのインターフェイスが表示されるはずです。

```
> 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      L0/1     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 0h21m53s

      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
```

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

E1000ネットワークインターフェイスを使用している既存の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。

既存のNetScaler VPXインスタンスを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。

NetScaler VPXインスタンスを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構成する

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後、vSphere Web Clientを使用して、仮想アプライアンスが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構成できます。

PCIパススルー機能により、ゲスト仮想マシンはホストに接続されている物理PCIおよびPCIeデバイスに直接アクセスできます。

前提条件

- ホスト上の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 ネットワークインターフェースカード
- Intel XL710デュアルポート40G NIC
- Intel XL710シングルポート40G NIC

ホストでパススルーデバイスを構成する

仮想マシンでパススルーPCIデバイスを構成する前に、ホストマシンで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、次の手順に従います。

1. vSphere Web Client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。使用可能なすべてのパススルーデバイスが表示されます。
3. 設定したいデバイスを右クリックし、**編集**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。

PCIパススルーを構成(/en-us/vpx/media/config_pcipassthrough_host.png)

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

NetScaler VPXインスタンスでパススルーPCIデバイスを構成するには、以下の手順に従います。

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。

PCIパススルーを構成(/en-us/vpx/media/config_pcipassthrough_vm.png)

4. **新しいPCIデバイス**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て、**OK**をクリックします。

注:

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

PCIパススルー(/en-us/vpx/media/config_pcipassthrough_vm_2.png)

5. ゲスト仮想マシンをパワーオンします。

NetScaler VPX で 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ための構成手順が完了しました。

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

VMware ESXハイパーバイザー上のNetScalerアプライアンスの初回起動時にNetScaler VPX構成を適用できます。そのため、特定のセットアップやVPXインスタンスをより短時間で起動できる場合があります。

プレ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アプライアンスの初回起動時にNetScaler VPX構成を適用する](#)」を参照してください。

注:

ESX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、デフォルトゲートウェイ構成を `<NS-CONFIG>` セクションで渡す必要があります。 `<NS-CONFIG>` タグの内容の詳細については、「[サンプル-`<NS-CONFIG>`-セクション](#)」を参照してください。

サンプル `<NS-CONFIG>` セクション:

```
<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>
```

ESXハイパーバイザーでプレブートユーザーデータを提供する方法

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

- CD/DVD ISOを使用する
- OVFプロパティを使用する

CD/DVD ISOを使用してユーザーデータを提供する

VMware vSphere クライアントを使用して、CD/DVD ドライブを介して ISO イメージとしてユーザーデータを仮想マシンに挿入することができます。

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の内容を含む、ファイル名 `userdata` のファイルを作成します。
`<NS-CONFIG>` タグの内容の詳細については、「サンプル `<NS-CONFIG>` セクション」を参照してください。

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. **userdata** ファイルをフォルダーに保存し、そのフォルダーを使用してISOイメージを構築します。

userdata ファイルを含むISOイメージは、次の2つの方法で構築できます。

- PowerISOなどの画像処理ツールを使用する。
- Linuxで **mkisofs** コマンドを使用する。

以下のサンプル構成は、Linuxで **mkisofs** コマンドを使用してISOイメージを生成する方法を示しています。

```
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 VPXインスタンスをプロビジョニングし、VMを作成します。ただし、VMを自動的にパワーオンしないでください。



4. VMが正常に作成されたら、ISOファイルをCD/DVDドライブとしてVMにアタッチします。



5. **新規CD/DVDドライブ** に移動し、ドロップダウンメニューから **データストアISOファイル** を選択します。



6. vSphere Clientでデータストアを選択します。



7. VMの電源を入れます。

ESX Web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

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. ユーザーデータコンテンツを含むファイルを作成します。

ESXのプリブートユーザーデータ(/en-us/vpx/media/ns-preboot-config-esx-userdata.png)

2. ユーザーデータコンテンツをBase64エンコーディングでエンコードします。Base64エンコーディングは、次の2つの方法で実行できます。

- Linuxでは、次のコマンドを使用します。

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

例:

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

Base64エンコードされたユーザーデータ(/en-us/vpx/media/base64-encoded-userdata.png)

- オンラインツールを使用してユーザーデータコンテンツをエンコードします。たとえば、Base64 Encode and Decodeなどです。
3. ESXハイパーバイザー上のNetScaler VPXインスタンスのOVFテンプレートに**Product**セクションを含めます。

Productセクションの例:

```
<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. Productセクションの `guestinfo.userdata` プロパティの `ovf:value` として、Base64エンコードされたユーザーデータを提供します。

```
<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="PE5TLVBRSRS1CT09ULUNPTkZJRz4KICAgIDx0Uy1DT05GSUc+Cg1hZGQgcm91
dGUgMC4wLjAuMCAw"
```


LjAuMC4wIDEwLjEwMi4zOC4xCiAgICA8L05TLUNPTkZJRz4KCjAgICA8TlMtQk9PVFNuUkFQPgog

ICAgICAgICAgICA8U0tJUC1ERUZBVUxULUJPT1RTVFJBUD5ZRVM8L1NLSVAtREVGQVVMVC1CT09U

U1RSQVA+CjAgICAgICAgICAgIDx0RVctQk9PVFNuUkFQLVNFUVVFTkNFP1lFUzwwTkVXLUJPT1RT

VFJBUC1TRVFRU5DRT4KCjAgICAgICAgPE1HTVQtSU5URVJGQUNFLUNPTkZJRz4KICAgICAgICAg

ICAgICAgIDxJTlRFUkZBQ0UtTlVNPiBlbGgwIDwvSU5URVJGQUNFLU5VTT4KICAgICAgICAgICAg

ICAgIDxJUD4gICAgMTAuMTAyLjM4LjIxOSA8L0lQPgogICAgICAgICAgICAgICAgPFNVQk5FVC1N

QVNLPiAyNTUuMjU1LjI1NS4wIDwvU1VCTkVULU1BU0s+CjAgICAgICAgPC9NR01ULU10VEVSrkFD

RS1DT05GSUc+CjAgICA8L05TLUJPT1RTVFJBUD4KPC90Uy1QUkUtQk9PVC1DT05GSUc+Cg==>

<Label>Userdata</Label>

<Description> Userdata for ESX VPX </Description>

</Property>

</ProductSection>

5. Productセクションを含む変更されたOVFテンプレートをVMデプロイメントに使用します。

Productセクションを含むOVFテンプレート (/en-us/vpx/media/ovf-template-with-product-section.png)

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

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. ユーザーデータコンテンツを含むファイルを作成します。

ESXのプリブートユーザーデータ (/en-us/vpx/media/ns-preboot-config-esx-userdata.png)

2. ユーザーデータコンテンツをBase64エンコーディングでエンコードします。Base64エンコーディングは、次の2つの方法で実行できます。

- Linuxでは、次のコマンドを使用します。

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

例:

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

Base64でエンコードされたユーザーデータ (/en-us/vpx/media/base64-encoded-userdata.png)

- オンラインツールを使用してユーザーデータコンテンツをエンコードします。たとえば、Base64 Encode and Decodeなどです。

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

Productセクションの例:

```
<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. Productセクションの `guestinfo.userdata` プロパティの `ovf:value` として、Base64でエンコードされたユーザーデータを提供します。

```

<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="PE5TLVBSRS1CT09ULUNPTkZJRz4KICAgIDx0Uy1DT05GSUc+Cg1hZGQgcm91
    dGUgMC4wLjAuMCAw

    LjAuMCAwIDEwLjEwMi4zOC4xCiAgICA8L05TLUNPTkZJRz4KCiAgICA8TlMtQk9PVFNuUkF
    QPgog

    ICAGICAgICAgICA8U0tJUC1ERUZBVUxULUJPT1RTVFJBUD5ZRVM8L1NLSVAtREVGVVMVC1
    CT09U

    U1RSQVA+CiAgICAgICAgICAgIDx0RVctQk9PVFNuUkFQLVNFUVVFTkNFP1lFUzwwTkVXLUJ
    PT1RT

    VFJBUC1TRVFRU5DRT4KCiAgICAgICAgICAgPE1HTVQtSU5URVJGQUFLUNPTkZJRz4KICAgICA
    gICAg

    ICAGICAgIDxJTlRFUkZBQ0UtTlVNPiBlbGgwIDwvSU5URVJGQUFLU5VTT4KICAgICAgICA
    gICAg

    ICAGIDxJUD4gICAgMTAuMTAyLjM4LjIx0SA8L0lQPgogICAgICAgICAgICAgICAgPFNVQk5
    FVC1N

    QVNLPiAyNTUuMjU1LjI1NS4wIDwvU1VCTkVULU1BU0s+CiAgICAgICAgPC9NR01ULU10VEV
    SRkFD

    RS1DT05GSUc+CiAgICA8L05TLUJPT1RTVFJBUD4KPC90Uy1QUkUtQk9PVC1DT05GSUc+Cg=
    =">

    <Label>Userdata</Label>

```

```
<Description> Userdata for ESX VPX </Description>

</Property>

</ProductSection>
```

5. VirtualHardwareSectionにプロパティ `ovf:transport="com.vmware.guestInfo"` を次のように追加します。

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

6. Productセクションを含む変更されたOVFテンプレートをVMデプロイメントに使用します。

Productセクションを含むOVFテンプレート (/en-us/vpx/media/ovf-template-with-product-section.png)

AWS 上の VMware Cloud に NetScaler VPX インスタンスをインストールする

AWS 上の VMware Cloud (VMC) を使用すると、必要な数の ESX ホストを持つ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 の展開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ーフェイスを提供します。これは ESX ベースの NetScaler VPX 展開と同一に機能します。

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、以下を実行してください。

- 少なくとも1つのホストを持つVMware SDDCが1つ存在している必要があります。
- 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上に適切なネットワークセグメントを作成します。
- VPX ライセンスファイルを入手します。NetScaler VPX インスタンスのライセンスの詳細については、<https://www.citrix.com/licenses/netScaler/13.1/licensing/licensing-guide-for-netscaler.html> の『*NetScaler VPX Licensing Guide*』を参照してください。

VMware Cloud 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仮想コンピューティングリソース

コンポーネント	要件
メモリ	2 GB
仮想CPU (vCPU)	2
仮想ネットワークインターフェース	VMware SDDCでは、VPXハードウェアがバージョン7以降にアップグレードされている場合、最大10個の仮想ネットワークインターフェースをインストールできます。

コンポーネント	要件
ディスク容量	20 GB

注記:

これは、ハイパーバイザーのディスク要件に加えて必要です。

VPX仮想アプライアンスを本番環境で使用する場合、メモリ割り当て全体を予約する必要があります。

OVF Tool 1.0 のシステム要件

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、最小システム要件を示します。

表 2. OVF Tool 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	要件
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 Tool User Guide」 PDFファイルを検索してください。
CPU	750 MHz以上、1 GHz以上を推奨
RAM	1 GB以上、2 GBを推奨
NIC	100 Mbps以上のNIC

OVFのインストールに関する情報については、<http://kb.vmware.com/> で「OVF Tool User Guide」 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、Open Virtual Machine (OVF) 形式標準に準拠しています。ファイルはCitrixの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。「**New Users link**」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てください。

ログオンしたら、Citrix®のホームページから次のパスに移動します。

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てください。

- NSVPX-ESX-\\<release number>-\\<build number>-disk1.vmdk (例: NSVPX-ESX-13.0-79.64-disk1.vmdk)
- NSVPX-ESX-\\<release number>-\\<build number>.ovf (例: NSVPX-ESX-13.0-79.64.ovf)
- NSVPX-ESX-\\<リリース番号>-\\<ビルド番号>.mf (例: NSVPX-ESX-13.0-79.64.mf)

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

VMware SDDC をインストールして構成したら、その SDDC を使用して VMware Cloud 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 で利用可能なメモリ量によって異なります。

VMware Cloud に NetScaler 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 VMware SDDC を開きます。
2. **ユーザー名とパスワード**のテキストボックスに管理者資格情報を入力し、ログインをクリックします。
3. **ファイルメニュー**で、**OVF テンプレートの展開**をクリックします。
4. **OVF テンプレートの展開**ダイアログボックスの**ファイルから展開**で、NetScaler VPX インスタンスのセットアップファイルを保存した場所に移動し、.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、OVF を変更して E1000 の代わりに VMXNET3 インターフェイスを使用するようにします。

5. 仮想アプライアンスの OVF テンプレートに表示されているネットワークを、VMware SDDC で構成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 NetScaler 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。**コンソールタブ**をクリックして、コンソールポートをエミュレートします。
7. 別の仮想アプライアンスをインストールする場合は、手順 6 から繰り返します。
8. 管理ネットワークとして選択されているのと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 および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する

Microsoft Windows ServerにNetScaler VPXインスタンスをインストールするには、まず十分なシステムリソースを持つマシンに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ロールをインストールする際は、Hyper-Vが仮想ネットワークを作成するために使用するサーバー上のNICを指定してください。一部のNICはホスト用に予約できます。NetScaler VPXインスタンスのインストールにはHyper-Vマネージャーを使用します。

Hyper-V用のNetScaler VPXインスタンスは、仮想ハードディスク（VHD）形式で提供されます。これには、CPU、ネットワークインターフェース、ハードディスクのサイズとフォーマットなどの要素のデフォルト設定が含まれています。NetScaler VPXインスタンスをインストールした後、仮想アプライアンスのネットワークアダプターを構成し、仮想NICを追加し、NetScalerのIPアドレス、サブネットマスク、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設定後、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したい場合は、「[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:

Intermediate System-to-Intermediate System (ISIS) プロトコルは、HyperV-2012プラットフォームでホストされているNetScaler VPX仮想アプライアンスではサポートされていません。

MicrosoftサーバーにNetScaler 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、以下を実行してください。

- Windows ServerでHyper-Vロールを有効にします。詳細については、[<http://technet.microsoft.com/ja-jp/library/ee344837\(ws.10\).aspx>](http://technet.microsoft.com/ja-jp/library/ee344837(ws.10).aspx) を参照してください。
- 仮想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。

- NetScaler VPXインスタンスのライセンスファイル入手します。NetScaler VPXインスタンスの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にある「NetScaler VPX Licensing Guide」を参照してください。

Microsoftサーバーのハードウェア要件

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	要件
CPU	1.4 GHz 64ビットプロセッサ
RAM	8 GB
ディスク容量	32 GB以上

次の表に、各 NetScaler VPXインスタンスの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仮想コンピューティングリソース

コンポーネント	要件
RAM	4 GB
仮想CPU	2
ディスク容量	20 GB
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 VPX インスタンスは、仮想ハードディスク (VHD) 形式で提供されます。ファイルはCitrixの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」をクリックして、指示に従ってCitrixアカウントを作成してください。

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ブラウザで `<http://www.citrix.com/>` にアクセスします。
2. ユーザー名とパスワードでサインインします。
3. 「ダウンロード」 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 Release X.X > 仮想アプライアンス」 で、「NetScaler VPX Release 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。

MicrosoftサーバーにNetScaler VPXインスタンスをインストールする

Microsoft Server で Hyper-V の役割を有効にし、仮想アプライアンスファイルを抽出したら、Hyper-V マネージャーを使用して NetScaler 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た後、Hyper-V によって作成された仮想ネットワークに仮想 NIC を関連付けて構成する必要があります。

仮想NICは最大8つまで構成できます。物理NICがDOWNしていても、仮想アプライアンスは同じホスト（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NICはUPしていると見なします。

注:

仮想アプライアンスの実行中は、設定を変更できません。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ってください。

Hyper-V マネージャーを使用して Microsoft Server に NetScaler VPX インスタンスを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**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 マネージャー**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. **操作メニュー**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が含まれるフォルダーのパスを指定し、**仮想マシンをコピーする (新しい一意の ID を作成する)** を選択します。このフォルダーは、スナップショット、仮想ハードディスク、および仮想マシンフォルダーを含む親フォルダーです。

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、ファイルをフォルダーに抽出してください。

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から6を繰り返します。

重要:

手順4で、ファイルを別のフォルダーに抽出してください。

Hyper-V 上で NetScaler 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が行われない場合、仮想アプライアンスはIPアドレスなどを設定するオプションを提供します。

Hyper-V 上で NetScaler 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. 例に示されているように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、以下を使用できます。

- 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"
```

```
oe:id=""
```

```
xmlns="http://schemas.dmtf.org/ovf/environment/1">
```

```
<PlatformSection>
```

```
<Kind>HYPER-V</Kind>
```

```
<Version>2013.1</Version>
```

```
<Vendor>CITRIX</Vendor>
```

```
<Locale>en</Locale>
```

```
</PlatformSection>
```

```
<PropertySection>
```

```
<Property oe:key="com.citrix.netscaler.ovf.version" oe:value="1.0"/>

<Property oe:key="com.citrix.netscaler.platform" oe:value="NS1000V"/>

<Property oe:key="com.citrix.netscaler.orch\_env" oe:value="cisco-orch-
env"/>

<Property oe:key="com.citrix.netscaler.mgmt.ip"
oe:value="10.102.100.122"/>

<Property oe:key="com.citrix.netscaler.mgmt.netmask"
oe:value="255.255.255.128"/>

<Property oe:key="com.citrix.netscaler.mgmt.gateway"
oe:value="10.102.100.67"/></PropertySection>

</Environment>
```

2. ISOイメージをHyper-Vサーバーにコピーします。
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て右クリックし、**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コントローラー** をクリックします。
5. 右側のウィンドウペインで、**DVDドライブ** を選択し、**追加** をクリックします。DVDドライブが左側のウィンドウペインの **IDEコントローラー** セクションの下に追加されます。
6. ステップ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、ステップ2でHyper-VサーバーにコピーしたISOイメージを選択します。

7. **適用** をクリックします。

注:

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

- DVDドライブが接続されており、ISOファイルが提供されていません。
- ISOファイルにユーザーデータファイルが含まれていません。
- ユーザーデータファイルの名前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. 「**<仮想アプライアンス名> の設定**」ダイアログボックスで、左ペインの「**ハードウェアの追加**」をクリックします。
3. 右ペインのデバイスリストから、**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**ネットワークアダプター (未接続)**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ー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、**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. 以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設定が完了しました。Webブラウザに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:

SCVMMを使用して、仮想マシン (VM) テンプレートでNetScaler 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チームングソリューションを使用している場合は、詳細については記事[CTX224494](#)を参照してください。

Linux-KVMプラットフォームにNetScaler VPXインスタンスをインストールする

Linux-KVMプラットフォーム用にNetScaler VPXをセットアップするには、グラフィカルなVirtual Machine Manager (Virtual Manager) アプリケーションを使用できます。Linux-KVMコマンドラインを好む場合は、`virsh` プログラムを使用できます。

ホストLinuxオペレーティングシステムは、KVMモジュールやQEMUなどの仮想化ツールを使用して、適切なハードウェアにインストールする必要があります。ハイパーバイザーに展開できる仮想マシン (VM) の数は、アプリケーションの要件と選択されたハードウェアによって異なります。

NetScaler VPXインスタンスをプロビジョニングした後、さらに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

予期しない動作を避けるため、以下の推奨事項を適用してください。

- VPX VM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てください。
- VPX VMの強制シャットダウンは行わないでください。つまり、**強制オフ**コマンドを使用しないでください。
- ホストLinuxで行われた設定は、Linuxディストリビューションの設定によっては永続的である場合とそうでない場合があります。ホストLinuxオペレーティングシステムの再起動後も一貫した動作を保証するために、これらの設定を永続的にすることを選択できます。
- プロビジョニングされたNetScaler VPXインスタンスごとに、NetScalerパッケージは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NetScaler VPXインスタンスをインストールするための前提条件

NetScaler VPXインスタンスを実行するLinux-KVMサーバーの最小システム要件を確認してください。

CPU要件:

- Intel VT-Xプロセッサ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。

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。

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

上記の拡張機能の**BIOS**設定が無効になっている場合は、BIOSで有効にする必要があります。

- ホストLinuxに少なくとも2つのCPUコアを提供します。
- プロセッサ速度に関する特定の推奨事項はありませんが、速度が速いほど、VMアプリケーションのパフォーマンスが向上します。

メモリ (RAM) 要件:

ホストLinuxカーネルには最低4GB。VMが必要とする場合は、さらにメモリを追加してください。

ハードディスク要件:

ホストLinuxカーネルとVMの要件に必要な容量を計算します。単一のNetScaler VPX VMには20GBのディスク容量が必要です。

ソフトウェア要件

使用するホストカーネルは、すべての仮想化ツールを備えた64ビットLinuxカーネル、リリース2.6.20以降である必要があります。Citrixは、3.6.11-4以降のような新しいカーネルを推奨しています。

Red Hat、CentOS、Fedoraなどの多くのLinuxディストリビューションには、テスト済みのカーネルバージョンと関連する仮想化ツールがあります。

ゲストVMのハードウェア要件

NetScaler VPXは、IDEおよびvirtIOハードディスクタイプをサポートしています。ハードディスクタイプは、NetScalerパッケージの一部であるXMLファイルで設定されています。

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ースをサポートしています。

サポートされてい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する([/ja-jp/citrix-adc/13-1/install-vpx-on-kvm/provision-on-kvm-using-virtual-machine-manager.html](http://ja-jp/citrix-adc/13-1/install-vpx-on-kvm/provision-on-kvm-using-virtual-machine-manager.html))
- 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([/ja-jp/vpx/13-1/install-vpx-on-kvm/configure-SR-IOV-KVM.html](http://ja-jp/vpx/13-1/install-vpx-on-kvm/configure-SR-IOV-KVM.html))
- PCIパススルー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([/ja-jp/vpx/13-1/install-vpx-on-kvm/configure-PCI-passthrough-KVM.html](http://ja-jp/vpx/13-1/install-vpx-on-kvm/configure-PCI-passthrough-KVM.html))

ソースインターフェースとモード

ソースデバイスタイプは、ブリッジまたはMacVTapのいずれかです。MacVTapでは、VEPA、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。以下に従って、使用できるインターフェー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てください。

ブリッジ:

- Linuxブリッジ。
- ホストLinux上の **Ebtables** および **iptables** 設定は、正しい設定を選択しないか、**IPtable** サービスを無効にしない場合、ブリッジ上のトラフィックをフィルタリングする可能性があります。

MacVTap (VEPAモード):

- ブリッジよりも優れたパフォーマンス。
- 同じ下位デバイスからのインターフェースは、VM間で共有できます。
- 同じものを使用したVM間通信
- 下位デバイスは、アップストリーム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):

- ブリッジよりも優れたパフォーマンス。
- 同じ下位デバイスからのインターフェースは、VM間で共有できます。
- 同じ下位デバイスを使用したVM間通信はできません。

MacVTap (ブリッジモード):

- ブリッジと比較して優れています。
- 同じ下位デバイスからのインターフェースは、VM間で共有できます。
- 同じ下位デバイスを使用したVM間通信は、下位デバイスリンクがUPの場合に可能です。

MacVTap (パススルーモード):

- ブリッジと比較して優れています。
- 同じ下位デバイスからのインターフェースは、VM間で共有できません。
- 1つのVMのみが下位デバイスを使用できます。

注記:

VPXインスタンスの最高のパフォーマンスのために、ソースインターフェースで **gro** および **lro** 機能が無効になっていることを確認してください。

ソースインターフェースのプロパティ

ソースインターフェースのgeneric-receive-offload (**gro**) および large-receive-offload (**lro**) 機能を無効にしてください。 **gro** および **lro** 機能を無効にするには、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 off ethool -K eth6 lro off
```

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ブリッジがソースデバイスとして使用されている場合、**lro** 機能は VNetインターフェースで無効にする必要があります。これらはホストをゲストVMに接続する仮想インターフェースです。

```
[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ースはeth6_brから派生し、vnet0およびvnet2として表されます。これらのインターフェースで **gro** および **lro** 機能を無効にするには、次のコマンドを実行します。

```
ethtool -K vnet0 gro off
```

```
ethtool -K vnet2 gro off
```

```
ethtool -K vnet0 lro off
```

```
ethtool -K vnet2 lro off
```

プロミスカスモード

次の機能が動作するためには、プロミスカス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

プロミスカスモードを有効にするには、以下のコマンドを使用します。

```
[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ホストで以下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、以下のコマンドを入力して実行します。

```
modprobe vhost\_net
```

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

NetScaler VPX インスタンスは、**Nova boot** コマンド (OpenStack CLI) または Horizon (OpenStack ダッシュボード) のいずれかを使用して OpenStack 環境にプロビジョニングできます。

VPX インスタンスのプロビジョニングには、オプションで設定ドライブのデータを使用することが含まれます。設定ドライブは、インスタンスの起動時に CD-ROM デバイスとしてインスタンスにアタッチされる特殊な設定ドライブです。この設定ドライブは、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のネットワーク設定を渡し、顧客スクリプトを挿入するために使用できます。

NetScaler アプライアンスでは、デフォルトの認証メカニズムはパスワードベースです。現在、OpenStack 環境の NetScaler 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 (公開鍵と秘密鍵) は、公開鍵暗号メカニズムを使用する前に生成されます。キーペアを生成するには、Horizon、Windows 用の Puttygen.exe、Linux 環境用の `ssh-keygen` など、さまざまなメカニズムを使用できます。キーペアの生成に関する詳細については、各メカニズム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、秘密鍵を、許可された人物がアクセスできる安全な場所にコピーします。OpenStack では、Horizon または Nova boot コマンドを使用して、公開鍵を VPX インスタンスにデプロイできます。OpenStack を使用して VPX インスタンスがプロビジョニングされると、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、/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 実装で利用可能な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。

設定ドライブが検出されると、インスタンスは、提供されていればネットワーク設定、カスタムスクリプト、および SSH キーペアの読み取りを試みます。

ユーザーデータファイル

NetScaler VPX インスタンスは、ネットワーク設定やカスタムスクリプトを挿入するために、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 ファイルを使用します。このファイルは設定ドライブの一部として提供されます。以下に、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>

    </Scripts>

  </ScriptSettingSection>

</cs:ScriptSection>
```

```
<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>

...

```

上記の OVF ファイルでは、「PropertySection」が NetScaler ネットワーク設定に使用され、`<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バンドルするために使用されます。各スクリプトは `<Script>` タグの間に定義されます。各スクリプトタグには、次のフィールド/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。可能な値: Shell/Perl/Python/NSLCI (NetScaler CLI スクリプト用)
- b) `<Parameter>`: スクリプトにパラメーターを提供します。各スクリプトには複数の `<Parameter>` タグを含めることができます。
- c) `<BootScript>`: スクリプトの実行ポイントを指定します。このタグの可能な値: before/after。「before」は、PE が起動する前にスクリプトが実行され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。

注記:

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。管理者として、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が存在する必要はありません。空の「PropertySection」を使用して初回起動時に実行するスクリプトのみを定義するか、または空の<cs:ScriptSection>を使用してネットワーク構成のみを定義します。

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取ります。

- 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

パラメータが正常に読み取られると、それらはNetScaler®構成に設定され、インスタンスをリモートで管理できるようになります。パラメータが正常に読み取られない場合、または構成ドライブが利用できない場合、インスタンスはデフォルトの動作に移行します。その動作は次のとおりです。

- インスタンスはDHCPからIPアドレス情報の取得を試みます。
- DHCPが失敗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構成（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、Shell、Perl、Python、およびNetScaler 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公開鍵を、その「authorized_keys」ファイルにコピーします。これにより、ユーザーは秘密鍵を使用してインスタンスにアクセスできます。

注記:

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、該当するSSH秘密鍵でログオンし、手動でパスワードを設定してください。

開始する前に

OpenStack環境に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、構築します

qcow2イメージからOpenStackイメージを。次の手順に従ってください。

1. 次のコマンドを入力して、 .tgz 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、ステップ1で抽出した .qcow2 ファイルを使用してOpenStackイメージを構築します。

```
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: 次の図は、glance image-createコマンドのサンプル出力を示しています。



VPXインスタンスのプロビジョニング

次のいずれかのオプションを使用して、VPXインスタンスを2つの方法でプロビジョニングできます。

- 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、次の手順に従ってください。

1. OpenStack ダッシュボードにログインします。
2. ダッシュボード左側の「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起動ウィザードを開きます。

インスタンス起動ウィザード(/en-us/vpx/media/instance-launching-wizard.png)

4. インスタンス起動ウィザードで、次のような詳細を入力します。
 - a. インスタンス名
 - b. インスタンスフレーバー
 - c. インスタンス数
 - d. インスタンスブートソース
 - e. イメージ名

起動ウィザードの詳細(/en-us/vpx/media/launch-wizard-details.png)

5. 以下の手順を完了して、Horizon を介して新しいキーペアまたは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、既存のメカニズムを使用してキーを作成します。既存のキーがある場合は、この手順をスキップします。
 - 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**に移動します。
 - d) **Access & Security** をクリックします。
 - e) **Key Pair** ドロップダウンメニューの横にある + 記号をクリックし、表示されたパラメータに値を入力します。
 - f) **Public key** ボックスに公開鍵の内容を貼り付け、鍵に名前を付け、**Import Key Pair** をクリックします。

キーペアのインポート OpenStack(/en-us/vpx/media/import-keypair-openstack.png)

6. ウィザードの **Post Creation** タブをクリックします。Customization Script で、ユーザーデータファイルの内容を追加します。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、ゲートウェイの詳細、および顧客スクリプトが含まれています。
7. キーペアが選択またはインポートされたら、config-drive オプションをチェックし、**Launch** をクリックします。

インスタンスの起動(/en-us/vpx/media/launch-instance.png)

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

以下の手順に従って、OpenStack CLI を使用して VPX インスタンスをプロビジョニングします。

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ための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. 特定のネットワークにNICをアタッチするには、ネットワークリストからネットワークIDを選択するために、次のコマンドを入力します。

```
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。

```
openstack server create --flavor FLAVOR_ID --image IMAGE_ID --key-name KEY_NAME

--user-data USER_DATA_FILE_PATH --config-drive True --nic net-id=net-uuid

INSTANCE_NAME

openstack server create --image VPX-ToT-Image --flavor m1.medium --user-data

ovf.xml --config-drive True --nic net-id=2734911b-ee2b-48d0-
```

```
a1b6-3efd44b761b9
```

```
VPX-ToT
```

図2: 次の図は出力例を示しています。



Virtual Machine Managerを使用してNetScaler VPXインスタンスをプロビジョニングする

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。これにより、新しいVMゲストや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。内蔵のVNCビューアを使用してVMゲストのグラフィカルコンソールにアクセスし、ローカルまたはリモートでパフォーマンス統計を表示できます。

お好みのLinuxディストリビューションをインストールし、KVM仮想化を有効にした後、仮想マシンのプロビジョニングに進むことができます。

Virtual Machine Managerを使用してNetScaler VPXインスタンスをプロビジョニングする際には、2つのオプションがあります。

- IPアドレス、ゲートウェイ、ネットマスクを手動で入力する
- 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インスタンスをプロビジョニングするには、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、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.qcow2
```

例:

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

KVMでの一般的なNetScaler VPX展開には、次の手順が含まれます。

- 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
- 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する
- 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する
- 仮想マシンマネージャーを使用してVPXインスタンスにインターフェースを追加する

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を使用します。この機能が有効になっている場合、初期設定中に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、デフォルトゲートウェイを入力する必要はありません。

VPXインスタンスを自動プロビジョニングする前に、以下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Open Virtualization Format (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 (例: 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、KVMホストにISOイメージをマウントします。

OVF XMLファイルのサンプル:

これはOVF XMLファイルの内容の例であり、ファイルを作成するためのサンプルとして使用できます。

```
<?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>
```

```
<Kind></Kind>
```

```
<Version>2016.1</Version>
```

```
<Vendor>VPX</Vendor>
```

```
<Locale>en</Locale>
```

```
</PlatformSection>
```

```
<PropertySection>
```

```
<Property oe:key="com.citrix.netscaler.ovf.version" oe:value="1.0"/>
```

```
<Property oe:key="com.citrix.netscaler.platform" oe:value="NSVPX"/>
```

```
<Property oe:key="com.citrix.netscaler.orch\_env" oe:value="KVM"/>

<Property oe:key="com.citrix.netscaler.mgmt.ip" oe:value="10.1.2.22"/>

<Property oe:key="com.citrix.netscaler.mgmt.netmask"
oe:value="255.255.255.0"/>

<Property oe:key="com.citrix.netscaler.mgmt.gateway" oe:value="10.1.2.1"/>

</PropertySection>

</Environment>
```

前述のOVF XMLファイルでは、「PropertySection」はNetScaler®のネットワーク構成に使用されます。ファイルを作成する際は、例の最後に強調表示されているパラメータの値を指定してください。

- 管理IPアドレス
- ネットマスク
- ゲートウェイ

重要:

OVFファイルが適切にXML形式でフォーマットされていない場合、VPXインスタンスには、ファイルで指定された値ではなく、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する

Virtual Machine Managerを使用すると、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。

Virtual Machine Manager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. Virtual Machine Manager (**アプリケーション > システムツール > Virtual Machine Manager**)を開き、**認証**ウィンドウにログイン資格情報を入力します。
2. (/en-us/vpx/media/icon.png)アイコンをクリックするか、**localhost (QEMU)**を右クリックして、新しいNetScaler VPXインスタンスを作成します。

(/en-us/vpx/media/localhost.png)

3. **名前**テキストボックスに、新しいVMの名前 (例: NetScaler-VPX) を入力します。
4. **新しいVM**ウィンドウで、「オペレーティングシステムのインストール方法を選択してください」の下にある**既存のディスクイメージをインポート**を選択し、**進む**をクリックします。

(/en-us/vpx/media/new-vm-window.png)

5. **既存のストレージパスを指定**フィールドで、イメージへのパスを参照します。OSタイプをUNIX、バージョンをFreeBSD 6.xとして選択します。次に、**進む**をクリックします。

(/en-us/vpx/media/provide-storage-path.png)

6. **メモリとCPUの設定を選択**の下で、次の設定を選択し、**進む**をクリックします。

- **メモリ (RAM)**— 2048 MB
- **CPU数**— 2



7. 「**インストール前に設定をカスタマイズ**」チェックボックスを選択します。オプションで、「**詳細オプション**」の下でMACアドレスをカスタマイズできます。選択されている「**仮想タイプ**」がKVMであり、選択されているアーキテクチャがx86_64であることを確認してください。「**完了**」をクリックします。



8. **NIC**を選択し、次の設定を指定します。

- **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—ブリッジ



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「**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**」セクションを参照してください。それ以外の場合は、「インストールを開始」をクリックします。KVM上にNetScaler VPXをプロビジョニングした後、さらにインターフェースを追加できます。

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する

Virtual Machine Managerを使用して、QCOW2イメージで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、次の手順を実行します。

1. [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するのステップ1](#)から[ステップ8](#)に従います。

注記:

ステップ 5でqcow2イメージを選択していることを確認してください。

2. **Disk 1**を選択し、**Advanced options**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にNetScaler VPXをプロビジョニングした後、さらにインターフェースを追加できます。

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. **Hardware > Storage > Device type > CDROM device**の順にクリックします。
2. **Manage**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ファイルを選択して、**Finish**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CD-ROM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、画面キャプチャの例に示すように、OVFファイルで提供されたネットワーク構成で自動プロビジョニングされます。



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、初期構成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ースを追加する

KVMにNetScaler VPX™インスタンスをプロビジョニングした後、さらにインターフェースを追加できます。

インターフェースを追加するには、次の手順に従います。

1. KVM上で実行されている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**開く**を選択します。

3. ヘッダーにある



アイコンをクリックして、仮想ハードウェアの詳細を表示します。

4. ハードウェアの追加をクリックします。新しい仮想ハードウェアの追加ウィンドウで、ナビゲーションメニューから**ネットワーク**を選択します。



5. **ホストデバイス**フィールドで、物理インターフェースタイプを選択します。ホストデバイスタイプは、ブリッジまたはMacVTapのいずれかです。MacVTapの場合、VEPA、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. ブリッジの場合

- i. ホストデバイス — 「共有デバイス名を指定」オプションを選択します。
- ii. KVMホストで設定されているブリッジ名を入力します。

注:

KVMホストでLinuxブリッジを設定し、物理インターフェー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てください。



iii. デバイスモデル— `virtio`。

iv. **完了**をクリックします。

b. MacVTapの場合

i. ホストデバイス—メニューから物理インターフェースを選択します。

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を表示できます。



iv. 新しく追加されたNICを選択し、このNICのソース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。インターフェースとモードの詳細については、「Source Interface and Modes」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インスタンスの電源をオンにして、初期設定を手動で完了します。

重要:

速度、デュプレックス、自動ネゴシエーションなどのインターフェースパラメータ設定はサポートされていません。

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

Linux-KVM プラットフォームで実行されている NetScaler VPX インスタンスを、シングルルート I/O 仮想化 (SR-IOV) を使用して、以下の NIC で構成できます。

- インテル 82599 10G
- インテル X710 10G
- インテル 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ースを使用するように NetScaler VPX インスタンスを構成する
- SR-IOV インターフェースで静的 LA/LACP を構成する
- SR-IOV インターフェースで VLAN を構成する

制限事項

Intel 82599、X710、XL710、および X722 NIC を使用する際は、制限事項に留意してください。以下の機能はサポートされていません。

Intel 82599 NIC の制限事項:

- L2 モード切り替え。
- 管理者パーティショニング (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インターフェースが少なくとも1つある場合、VPXインスタンスで最大30個の一意のIPv6アドレスのみを構成できます。
- SRIOV VFインターフェースのハイパーバイザーでのVLAN構成は、`ip link` コマンドを介してはサポートされていません。
- 速度、デュプレックス、自動ネゴシエーションなどのインターフェースパラメータ構成はサポートされていません。

Intel X710 10G、Intel XL710 40G、およびIntel X722 10G NICの制限事項:

- L2モード切り替え。
- 管理者パーティショニング (共有VLANモード)。
- クラスターでは、XL710 NICがデータインター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。
- インターフェースが切断および再接続された場合、インターフェースリストの並べ替えが発生します。
- 速度、デュプレックス、自動ネゴシエーションなどのインターフェースパラメータ構成はサポートされていません。
- インターフェース名は、Intel X710 10G、Intel XL710 40G、およびIntel X722 10G NICの場合、40/Xです。

- 最大16個のIntel XL710/X710/X722 SRIOVまたはPCIパススルーインターフェースがVPXインスタンスでサポートされます。

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) で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ースを使用するようにNetScaler VPXインスタンスを構成する前に、以下の前提条件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	インテル 82599 NIC	INTEL X710、XL710、および 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ホストでドライバーをブラックリストに登録します。	/etc/modprobe.d/blacklist.conf ファイルに次のエントリを追加します: <code>blacklist ixgbevf</code> 。IXGBE ドライバーバージョン 4.3.15 (推奨) を使用してください。	/etc/modprobe.d/blacklist.conf ファイルに次のエントリを追加します: <code>blacklist i40evf</code> 。i40e ドライバーバージョン 2.0.26 (推奨) を使用してください。
4.KVM ホストで SR-IOV 仮想機能 (VF) を有効にします。次の2つの列のコマンド	カーネル 3.8 以前のバージョンを使用している場合は、「/etc/modprobe.d/ixgbe」ファ	カーネル 3.8 以前のバージョンを使用している場合は、「/etc/modprobe.d/i40e.conf」

タスク	インテル 82599 NIC	INTEL X710、XL710、および X722 NIC
の両方で: <code>number_of_VFs</code> = 作成する仮想 VF の数。 <code>device_name</code> = インターフェイス名。	イルに次のエントリを追加し、KVM ホストを再起動します: <code>options ixgbe</code> <code>max_vfs=<number_of_VFs></code> 。カーネル 3.8 以降のバージョン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: <code>echo</code> <code><number_of_VFs> > /sys/class/net/<device_name>/device/sriov_numvfs</code>。図 1 の例を参照してください。	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: <code>options i40e</code> <code>max_vfs=<number_of_VFs></code> 。カーネル 3.8 以降のバージョン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: <code>echo<number_of_VFs> > /sys/class/net/<device_name>/device/sriov_numvfs</code>。図 2 の例を参照してください。
1. VF の作成に使用したコマンドを <code>rc.local</code> ファイルに追加して、VF を永続化します。	図 3 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 を作成する際は、VF に MAC アドレスを割り当てないようにしてください。

図 1: Intel 82599 10G NIC 用の KVM ホストで SR-IOV VF を有効にする。



図 2: Intel X710 10G および XL710 40G NIC 用の KVM ホストで SR-IOV VF を有効にする。



図 3: Intel X722 10G NIC 用の KVM ホストで SR-IOV VF を有効にする。

X722 用 SR-IOV VF(/en-us/vpx/media/sr-iov-x722-nic.png)

図 4: VF を永続化する。

SR-IOV VF の永続化(/en-us/vpx/media/enable_sriov_2.jpg)

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

Virtual Machine Manager を使用して 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、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、次に  を選択します。

SR-IOV VM1 を構成(/en-us/vpx/media/config_sriov_vmm1.jpg)

3.  KVM 上の仮想マシン > ウィンドウで、 アイコンを選択します。

SR-IOV VM2 を構成(/en-us/vpx/media/config_sriov_vmm2.jpg)

4. **ハードウェアの追加**を選択します。

SR-IOV仮想マシン(/en-us/vpx/media/config_sriov_vmm3.jpg)

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、作成した VF を選択し、 をクリックします。

 **図 4:** Intel 82599 10G NIC用VF



図 5: Intel XL710 40G NIC用VF



図 6: Intel X722 10G NIC用VF



6. 手順4と5を繰り返して、作成したVFを追加します。
7. NetScaler VPXインスタンスの電源をオンにします。
8. NetScaler VPXインスタンスの電源がオンになったら、次のコマンドを使用して構成を確認します。

```
show interface summary
```

出力には、構成したすべてのインターフェイスが表示されます。

 **図 6:** Intel 82599 NICの出力概要。



図 7. Intel X710およびXL710 NICの出力概要。



SR-IOVインターフェイスで静的LA/LACPを構成する

重要:

SR-IOV VFを作成する際は、VFにMACアドレスを割り当てないようにしてください。

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、次のコマンドを使用してなりすましチェックを無効にします。

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

説明:

- Interface_name – インターフェイス名です。
- VF_id – 仮想機能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。詳細な手順については、[リンクアグリゲーションの設定](#)を参照してください。

SR-IOVインターフェイスでのVLANの設定

SR-IOV VFでVLANを設定できます。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。

NetScaler VPXインスタンスを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構成する

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後、Virtual Machine Managerを使用して、仮想アプライアンスが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構成できます。

前提条件

- KVMホスト上のIntel XL710 NIC (NIC) のファームウェアバージョンは5.04です。
- KVMホストは入出力メモリ管理ユニット (IOMMU) とIntel VT-dをサポートしており、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: **intel_iommu=1**
- 次のコマンドを実行し、KVMホストを再起動します: **Grub2-mkconfig -o /boot/grub2/grub.cfg**

Virtual Machine Managerを使用して、**NetScaler VPX**インスタンスが**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、**Open** をクリックします。

SR-IOVを構成する(/en-us/vpx/media/config_sriov_vmm1.jpg)

3. **virtual_machine on KVM**> ウィンドウで、**i** アイコンをクリックします。

KVM上のVMを構成する(/en-us/vpx/media/config_sriov_vmm2.jpg)

4. **Add Hardware** をクリックします。
5. **Add New Virtual Hardware** ダイアログボックスで、次の操作を行います。
 - a. **PCI Host Device** を選択します。
 - b. 「**ホストデバイス**」セクションで、Intel XL710 物理機能を選択します。
 - c. 「**完了**」をクリックします。



6. 追加の Intel XL710 物理機能を追加するには、手順4と5を繰り返します。
7. NetScaler VPX インスタンスの電源を入れます。
8. NetScaler VPX インスタンスの電源がオンになったら、以下のコマンドを使用して構成を確認できます。



出力には、構成したすべてのインターフェイスが表示される必要があります。



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

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。その機能はVirtual Machine Managerの機能に似ています。これにより、VMゲストのステータス（開始、停止、一時停止など）を変更したり、新しいゲストやデバイスを設定したり、既存の構成を編集したりできます。**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 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属性を指定するドメインXMLファイル \[NSVPX-KVM-*_nc.xml\]
- NS-VMディスクイメージのチェックサム \[Checksum.txt\]
- NS-VMディスクイメージ \[NSVPX-KVM-*_nc.raw\]

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、次のパラメーターを指定します。

- 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**— 絶対ディスクイメージ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できます。

RAWイメージファイルを指定する場合は、以下の例に示すようにディスクイメージの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' />
```

以下の例に示すように、絶対QCOW2ディスクイメージのソースパスを指定し、ドライバタイプを**qcow2**として定義します。

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. ネットワークの詳細を設定するために、`\<DomainName\>-NSVPX-KVM-*\_nc.xml` ファイルを編集します。

- **source dev**— インターフェイスを指定します。
- **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 Bridge**です。

例: モード: MacVTap Bridge ターゲットインターフェイスを `ethx` として、モードをブリッジとして設定します。モデルタイプは `virtio` です。

```
<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VMに接続されている物理インターフェイスです。

5. 以下のコマンドを使用して、`\<DomainName\>-NSVPX-KVM-*\_nc.xml` ファイルで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. 以下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. コンソールを介してゲストVMに接続します。

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

例:

```
virsh console NetScaler-VPX
```

NetScaler VPXインスタンスに **virsh** プログラムを使用してインターフェースを追加する

KVM上にNetScaler VPXをプロビジョニングした後、追加のインターフェースを追加できます。

インターフェースを追加するには、次の手順に従います。

1. KVM上で実行されているNetScaler VPXインスタンスをシャットダウンします。
2. 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の場合

- インターフェースタイプ— インターフェースタイプを「direct」として指定します。
- 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ース全体で一意であることを確認します。
- source dev— インターフェース名を指定します。
- mode— モードを指定します。サポートされているモードは、Bridge、VEPA、Private、およびPass-throughです。
- モデルタイプ— モデルタイプを **virtio** として指定します。

例:

モード: MacVTap パススルー

ターゲットインターフェースを **ethx**、モードを bridge、モデルタイプを **virtio** に設定します。

```
<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VMに接続されている物理インターフェースです。

b. ブリッジモードの場合

注:

KVMホストでLinuxブリッジを設定し、物理インターフェー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てください。

- インターフェースタイプ— インターフェースタイプを「bridge」として指定します。
- 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ース全体で一意であることを確認してください。
- ソースブリッジ— ブリッジ名を指定します。
- モデルタイプ— モデルタイプを `virtio` として指定します。

例: ブリッジモード

```
<interface type='bridge'>

    <mac address='52:54:00:2d:43:a4' />

    <source bridge='br0' />

    <model type='virtio' />

</interface>
```

NetScaler VPXゲストVMを管理する

Virtual Machine Manager と `virsh` プログラムを使用して、VMゲストの起動または停止、新しいゲストとデバイスのセットアップ、既存の構成の編集、Virtual Network Computing (VNC) を介した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。

Virtual Machine Managerを使用してVPXゲストVM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、接続されている各VMホストサーバーのすべてのVMゲストのリストが表示されます。各VM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）が含まれています。

- グラフィカルコンソールを開く

VMゲストのグラフィカルコンソールを開くと、VNC接続を介して物理ホストを操作するのと同じように、マシンを操作できます。Virtual Machine Managerでグラフィカルコンソールを開くには、VMゲストエントリを右クリックし、ポップアップメニューから「開く」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ゲストを起動または停止できます。VMの状態を変更するには、VMゲストエントリを右クリックし、ポップアップメニューから「実行」または「シャットダウン」オプションのいずれかを選択します。



- ゲストを再起動する

Virtual Machine ManagerからVMゲストを再起動できます。VMを再起動するには、VMゲストエントリを右クリックし、ポップアップメニューから「シャットダウン」>「再起動」を選択します。

- ゲストを削除する

VMゲストを削除すると、デフォルトでそのXML構成が削除されます。ゲストのストレージファイルを削除することもできます。これにより、ゲストは完全に消去されます。

- a. Virtual Machine Managerで、VMゲストエントリを右クリックします。
- b. ポップアップメニューから「削除」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。

- c. 削除をクリックします。
- d. ゲストを完全に消去するには、「関連ストレージファイルの削除」チェックボックスを選択して、関連する.rawファイルを削除します。

virsh プログラムを使用してNetScaler VPXゲストVM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。

virsh を使用してゲストに関する情報を表示するには

```
virsh list --all
```

コマンド出力には、すべてのドメインとその状態が表示されます。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。

コンソールを介してゲストVMを接続します

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

例:

```
virsh console NetScaler-VPX
```

- ゲストの起動およびシャットダウン。

ゲストはDomainNameまたはDomain-UUIDを使用して起動できます。

```
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
```

- ゲストの再起動

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

例:

```
virsh reboot NetScaler-VPX
```

ゲストの削除

ゲストVM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、
<DomainName>-NSVPX-KVM-*_nc.xmlを未定義に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 :

削除コマンドはディスクイメージファイルを削除しません。これは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する

OpenStack上で、シングルルートI/O仮想化（SR-IOV）テクノロジーを使用する高性能なNetScaler VPXインスタンスを展開できます。

OpenStack上でSR-IOVテクノロジーを使用するNetScaler VPXインスタンスを、次の3つのステップで展開できます。

- ホスト上でSR-IOV仮想機能（VF）を有効にします。
- VFを構成し、OpenStackで利用できるようにします。
- OpenStack上でNetScaler VPXをプロビジョニングします。

前提条件

以下を確認してください。

- Intel 82599 NIC（NIC）をホストに追加します。
- Intelから最新のIXGBEドライバーをダウンロードしてインストールします。
- ホスト上でIXGBEVFドライバーをブロックリストに登録します。/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 `ixgbev`

注:

`ixgbe` ドライバーのバージョンは最低でも5.0.4である必要があります。

ホスト上で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。

- カーネルバージョン3.8より前のもの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、ホストを再起動してください: `options ixgbe max_vfs=<number> of_VFs<>`
- カーネルバージョン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
```

ここで、：

- number_of_VFs は、作成する仮想機能の数です。
- device_name はインターフェース名です。

重要:

SR-IOV VFを作成する際は、VFにMACアドレスを割り当てないようにしてください。

以下に、4つのVFが作成される例を示します。

SR-IOVを有効にする(/en-us/vpx/media/enable_sriov_1.jpg)

VFを永続化するには、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。以下に、rc.localファイルの内容の例を示します。

SR-IOV VFを有効にする(/en-us/vpx/media/enable_sriov_2.jpg)

詳細については、この[Intel SR-IOV Configuration Guide](#)を参照してください。

VFを構成し、OpenStackで利用できるようにする

OpenStackでSR-IOVを構成するには、以下のリンクの手順に従ってください: <https://wiki.openstack.org/wiki/SR-IOV-Passthrough-For-Networking>。

OpenStackにNetScaler 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 環境に NetScaler VPX インスタンスをプロビジョニングできます。

VPX インスタンスのプロビジョニングには、オプションで設定ドライブからのデータを使用します。設定ドライブは、インスタンスの起動時にインスタンスにアタッチされる特別な設定ドライブです。この設定ドライブは、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ために使用できます。

OpenStack が VPX インスタンスをプロビジョニングするとき、まず OpenStack を示す特定の BIOS 文字列 (OpenStack Foundation) を読み取ることにより、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。設定ドライブが検出されると、インスタンスは **nova** 起動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、このファイルは「userdata.txt」と呼ばれます。

- 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

パラメータが正常に読み取られると、それらは NetScaler® スタックに設定されます。これにより、インスタンスをリモートで管理できます。パラメータが正常に読み取られない場合、または設定ドライブが利用できない場合、インスタンスは次のデフォルトの動作に移行します。

- インスタンスは DHCP から IP アドレス情報を取得しようとしています。
- DHCP が失敗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構成 (192.168.100.1/16) で起動します。

CLI を介して OpenStack に NetScaler 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 環境に VPX インスタンスをプロビジョニングできます。OpenStack に NetScaler VPX インスタンスをプロビジョニングする手順の概要は次のとおりです。

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 イメージを構築する
3. VPX インスタンスのプロビジョニング

OpenStack 環境に 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. .tgz ファイルから qcow2 ファイルを次のコマンドを入力して抽出します。

```
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. ステップ1で抽出した .qcow2 ファイルを使用して、次のコマンドを入力して OpenStack イメージを構築します。

```
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>
```

```
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
```

次の図は、glance image-create コマンドの出力例を示しています。



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。

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

上記のコマンドでは、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、ユーザーが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。

次の図は、nova boot コマンドの出力例を示しています。



次の図は、userdata.txt ファイルの例を示しています。<PropertySection></PropertySection> タグ内の値は、ユーザーが設定可能で、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含んでいます。

```
<?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">

<PlatformSection>

<Kind>NOVA</Kind>

<Version>2013.1</Version>

<Vendor>Openstack</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="vpx"/>

citrix.com 4

<Property oe:key="com.citrix.netscaler.orch_env"

oe:value="openstack-orch-env"/>

<Property oe:key="com.citrix.netscaler.mgmt.ip"

oe:value="10.1.0.100"/>

<Property oe:key="com.citrix.netscaler.mgmt.netmask"

oe:value="255.255.0.0"/>

<Property oe:key="com.citrix.netscaler.mgmt.gateway"

oe:value="10.1.0.1"/>

</PropertySection>
```

</Environment>

追加でサポートされる構成: ホストからSR-IOV VF上のVLANを作成および削除する

SR-IOV VF上に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
```

上記のコマンドでは、「enp8s0f0」は物理機能の名前です。

例: vf 6上に作成されたVLAN 10



](/en-us/vpx/media/vlan-10-created-on-vf6.png)

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がvf 6から削除されました



これらの手順で、OpenStack上でSRIOVテクノロジーを使用するNetScaler VPXインスタンスを展開する手順が完了します。

KVM 上で NetScaler VPX インスタンスを OVS DPDK ベースのホストインターフェースを使用するように構成する

KVM (Fedora および RHOS) 上で実行されている NetScaler VPX インスタンスを、ネットワークパフォーマンス向上のために Open vSwitch (OVS) と Data Plane Development Kit (DPDK) を使用するように構成できます。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 NetScaler VPX インスタンスを動作させるための構成方法について説明します。

OVS は、オープンソースの Apache 2.0 ライセンスの下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ーのセットです。

NetScaler VPX インスタンスの構成には、以下の Fedora、RHOS、OVS、および DPDK のバージョンが認定されています。

フェドラ	RHOS
フェドラ 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 1 GB のヒュージ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。KVM 上で NetScaler VPX インスタンスを OVS DPDK ベースのホストインターフェースを使用するように構成するために必要な手順の概要を以下に示します。

- DPDKをインストールします。
- OVSをビルドしてインストールします。
- OVSブリッジを作成します。
- 物理インターフェースをOVSブリッジにアタッチします。
- `vhost-user` ポートをOVSデータパスにアタッチします。
- 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をインストールする

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手順に従ってください。

OVSをビルドしてインストールする

OVS [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をビルドしてインストールします。[Installing Open vSwitch](#)ドキュメントに記載されている手順に従ってください。

詳細については、[DPDK Getting Started Guide for Linux](#)を参照してください。

OVSブリッジを作成する

必要に応じて、Fedoraまたは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
```

物理インターフェースをOVSブリッジに接続する

ポートをDPDKにバインドし、以下のFedoraまたはRHOSコマンドを入力してOVSブリッジに接続します。

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ている `dpdk-devargs` は、それぞれの物理NICのPCI BDFを指定します。

vhost-user ポートをOVSデータパスに接続する

以下のFedoraまたはRHOSコマンドを入力して、vhost-user ポートをOVSデータパスに接続します。

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をプロビジョニングする

Fedora KVMにOVS-DPDKベースの vhost-user ポートを持つVPXインスタンスをプロビジョニングするには、以下のQEMUコマンドを使用してCLIからのみ可能です。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>  
\  

```

```
-netdev type=vhost-user,id=mynet2,chardev=char1,vhostforce -device virtio-  
net  
  
pci,mac=00:00:00:00:00:02,netdev=mynet2,mrg_rxbuf=on \  
  
--nographic
```

RHOSの場合、`virsh` 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、以下のサンプルXMLファイルを使用します。

```
<domain type='kvm'>  
  
  <name>dppdk-vpx1</name>  
  
  <uuid>aedb844b-f6bc-48e6-a4c6-36577f2d68d6</uuid>  
  
  <memory unit='KiB'>16777216</memory>  
  
  <currentMemory unit='KiB'>16777216</currentMemory>  
  
  <memoryBacking>  
  
    <hugepages>
```

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

</hugepages>

</memoryBacking>

<vcpu placement='static'>6</vcpu>

<cputune>

<shares>4096</shares>

<vcpupin vcpu='0' cpuset='0' />

<vcpupin vcpu='1' cpuset='2' />

<vcpupin vcpu='2' cpuset='4' />

<vcpupin vcpu='3' cpuset='6' />

<emulatorpin cpuset='0,2,4,6' />
```

```
</cputune>
```

```
<numatune>
```

```
  <memory mode='strict' nodeset='0' />
```

```
</numatune>
```

```
<resource>
```

```
  <partition>/machine</partition>
```

```
</resource>
```

```
<os>
```

```
  <type arch='x86\_64' machine='pc-i440fx-rhel7.0.0'>hvm</type>
```

```
  <boot dev='hd' />
```

```
</os>
```

```
<features>
```

```
<acpi/>
```

```
<apic/>
```

```
</features>
```

```
<cpu mode='custom' match='minimum' check='full'>
```

```
<model fallback='allow'>Haswell-noTSX</model>
```

```
<vendor>Intel</vendor>
```

```
<topology sockets='1' cores='6' threads='1'/>
```

```
<feature policy='require' name='ss'/>
```

```
<feature policy='require' name='pcid'/>
```

```
<feature policy='require' name='hypervisor'/>
```

```
<feature policy='require' name='arat' />

<domain type='kvm'>

  <name>dpdk-vpx1</name>

  <uuid>aedb844b-f6bc-48e6-a4c6-36577f2d68d6</uuid>

  <memory unit='KiB'>16777216</memory>

  <currentMemory unit='KiB'>16777216</currentMemory>

  <memoryBacking>

    <hugepages>

      <page size='1048576' unit='KiB' />

    </hugepages>

  </memoryBacking>
```

```
<vcpu placement='static'>6</vcpu>
```

```
<cputune>
```

```
<shares>4096</shares>
```

```
<vcpupin vcpu='0' cpuset='0' />
```

```
<vcpupin vcpu='1' cpuset='2' />
```

```
<vcpupin vcpu='2' cpuset='4' />
```

```
<vcpupin vcpu='3' cpuset='6' />
```

```
<emulatorpin cpuset='0,2,4,6' />
```

```
</cputune>
```

```
<numatune>
```

```
<memory mode='strict' nodeset='0' />
```

```
</numatune>
```

```
<resource>
```

```
<partition>/machine</partition>
```

```
</resource>
```

```
<os>
```

```
<type arch='x86\_64' machine='pc-i440fx-rhel17.0.0'>hvm</type>
```

```
<boot dev='hd' />
```

```
</os>
```

```
<features>
```

```
<acpi/>
```

```
<apic/>
```



```
</features>
```

```
<cpu mode='custom' match='minimum' check='full'>
```

```
<model fallback='allow'>Haswell-noTSX</model>
```

```
<vendor>Intel</vendor>
```

```
<topology sockets='1' cores='6' threads='1' />
```

```
<feature policy='require' name='ss' />
```

```
<feature policy='require' name='pcid' />
```

```
<feature policy='require' name='hypervisor' />
```

```
<feature policy='require' name='arat' />
```

```
<feature policy='require' name='tsc\_adjust' />
```

```
<feature policy='require' name='xsaveopt' />
```

```
<feature policy='require' name='pdpe1gb' />

<numa>

    <cell id='0' cpus='0-5' memory='16777216' unit='KiB'
memAccess='shared' />

</numa>

</cpu>

<clock offset='utc' />

<on\_poweroff>destroy</on\_poweroff>

<on\_reboot>restart</on\_reboot>

<on\_crash>destroy</on\_crash>

<devices>

    <emulator>/usr/libexec/qemu-kvm</emulator>
```

```
<disk type='file' device='disk'>

    <driver name='qemu' type='qcow2' cache='none' />

    <source file='/home/NSVPX-KVM-12.0-52.18\_nc.qcow2' />

    <target dev='vda' bus='virtio' />

    <address type='pci' domain='0x0000' bus='0x00' slot='0x07'
function='0x0' />

</disk>

<controller type='ide' index='0'>

    <address type='pci' domain='0x0000' bus='0x00' slot='0x01'
function='0x1' />

</controller>

<controller type='usb' index='0' model='piix3-uhci'>
```

```
<address type='pci' domain='0x0000' bus='0x00' slot='0x01'  
function='0x2' />
```

```
</controller>
```

```
<controller type='pci' index='0' model='pci-root' />
```

```
<interface type='direct'>
```

```
<mac address='52:54:00:bb:ac:05' />
```

```
<source dev='enp129s0f0' mode='bridge' />
```

```
<model type='virtio' />
```

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

```
</interface>
```

```
<interface type='vhostuser'>
```

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

```
<source type='unix' path='/var/run/openvswitch/vhost-user1'  
mode='client'/>
```

```
<model type='virtio'/>
```

```
<address type='pci' domain='0x0000' bus='0x00' slot='0x04'  
function='0x0'/>
```

```
</interface>
```

```
<interface type='vhostuser'>
```

```
<mac address='52:54:00:2a:32:64'/>
```

```
<source type='unix' path='/var/run/openvswitch/vhost-user2'  
mode='client'/>
```

```
<model type='virtio'/>
```

```
<address type='pci' domain='0x0000' bus='0x00' slot='0x05'  
function='0x0'/>
```

```
</interface>
```

```
<interface type='vhostuser'>

    <mac address='52:54:00:2a:32:74' />

    <source type='unix' path='/var/run/openvswitch/vhost-user3'
mode='client' />

    <model type='virtio' />

    <address type='pci' domain='0x0000' bus='0x00' slot='0x06'
function='0x0' />

</interface>

<interface type='vhostuser'>

    <mac address='52:54:00:2a:32:84' />

    <source type='unix' path='/var/run/openvswitch/vhost-user4'
mode='client' />

    <model type='virtio' />
```

```
<address type='pci' domain='0x0000' bus='0x00' slot='0x09'  
function='0x0' />
```

```
</interface>
```

```
<serial type='pty'>
```

```
<target port='0' />
```

```
</serial>
```

```
<console type='pty'>
```

```
<target type='serial' port='0' />
```

```
</console>
```

```
<input type='mouse' bus='ps2' />
```

```
<input type='keyboard' bus='ps2' />
```

```
<graphics type='vnc' port='-1' autoport='yes'>
```

```
<listen type='address' />

</graphics>

<video>

  <model type='cirrus' vram='16384' heads='1' primary='yes' />

  <address type='pci' domain='0x0000' bus='0x00' slot='0x02'
function='0x0' />

</video>

<memballoon model='virtio'>

  <address type='pci' domain='0x0000' bus='0x00' slot='0x08'
function='0x0' />

</memballoon>

</devices>
```



```
</domain
```

注意点

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` のサイズは1 GBである必要があります。

```
<memoryBacking>

    <hugepages>

        <page size='1048576' unit='KiB' />

    </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。

```
<interface type='vhostuser'>

    <mac address='52:54:00:55:55:56' />

    <source type='unix' path='/var/run/openvswitch/vhost-user1'
mode='client' />

    <model type='virtio' />
```

```

        <address type='pci' domain='0x0000' bus='0x00' slot='0x04'
function='0x0' />

</interface>

```

NetScaler VPXインスタンスを起動するには、`virsh` コマンドを使用して開始します。

KVMハイパーバイザー上のNetScalerアプライアンスの初回起動時にNetScaler VPX構成を適用する

KVMハイパーバイザー上のNetScalerアプライアンスの初回起動時にNetScaler VPX構成を適用できます。これにより、VPXインスタンスでの顧客設定をより短時間で構成できます。

プレ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アプライアンスの初回起動時にNetScaler VPX構成を適用する」(</ja-jp/vpx/13-1/apply-vpx-config-at-preboot.html>)を参照してください。

注:

KVMハイパーバイザー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、デフォルトゲートウェイ構成を `<NS-CONFIG>` セクションで渡す必要があります。`<NS-CONFIG>` タグの内容の詳細については、以下の「サンプル `<NS-CONFIG>` セクション」を参照してください。

サンプル `<NS-CONFIG>` セクション:

```

<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レブートユーザーデータを提供する方法

KVMハイパーバイザーでは、CDROMデバイスを使用してアタッチされるISOファイルを介してプレブートユーザーデータを提供できます。

CDROM ISOファイルを使用してユーザーデータを提供する

Virtual Machine Manager (VMM) を使用して、CDROMデバイスを介してISOイメージとしてユーザーデータを仮想マシン (VM) に挿入できます。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することによって、VMゲスト内のCD-ROMをサポートします。

CDROM ISOファイルを使用してユーザーデータを提供するには、次の手順を実行します。

1. プレブートユーザーデータの内容を含む、ファイル名 `userdata` の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. `userdata` ファイルをフォルダーに保存し、そのフォルダーを使用してISOイメージを構築します。

`userdata` ファイルを使用して、以下の2つの方法でISOイメージを構築できます。

- PowerISOなどの任意のイメージ処理ツールを使用する。
- Linuxで `mkisofs` コマンドを使用する。

以下のサンプル構成は、Linuxで `mkisofs` コマンドを使用してISOイメージを生成する方法を示しています。

```
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 VPXインスタンスをプロビジョニングし、VMを作成します。ただし、VMを自動的に起動しないでください。
4. 以下の手順でVirtual Machine Managerを使用してCD-ROMデバイスを追加します。
 - a. Virtual Machine Manager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、**View > Details**で詳細ビューに切り替えます。
 - b. **Add Hardware > Storage > Device type > CDROM device**をクリックします。
 - c. **Manage**をクリックして正しいISOファイルを選択し、**Finish**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。
5. VMを起動します。

AWS 上の NetScaler VPX

Amazon Web Services (AWS) で NetScaler VPX インスタンスを起動できます。NetScaler VPX アプライアンスは、AWS Marketplace で Amazon マシンイメージ (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利用し、ビジネスニーズに合わせて NetScaler の負荷分散およびトラフィック管理機能を使用できます。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ており、スタンドアロンインスタンスとして、または HA ペアで展開できます。VPX の機能の詳細については、[VPX データシート](#) を参照してください。

はじめに

VPX の展開を開始する前に、以下の情報に精通している必要があります。

- [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 VPX インスタンスを展開する

AWS では、VPX インスタンスに対して以下の展開タイプがサポートされています。

- [スタンドアロン](#)
- [高可用性 \(アクティブ-パッシブ\)](#)
 - [同一ゾーン内の高可用性](#)
 - [Elastic IP を使用した異なるゾーン間の高可用性](#)
 - [プライベートIPを使用した異なるゾーン間での高可用性\(/ja-jp/vpx/13-1/deploy-aws/vpx-ha-pip-different-aws-zones.html\)](#)
- [アクティブ-アクティブ GSLB\(/ja-jp/citrix-adc/13-1/global-server-load-balancing/deployment-types/active-active-site-deployment.html\)](#)

ハイブリッド展開

- AWS OutpostへのNetScalerの展開(</ja-jp/vpx/13-1/deploy-aws/deploy-vpx-on-aws-outposts.html>)
- AWSのVMCへのNetScalerの展開(</ja-jp/vpx/13-1/install-vpx-on-vmc-on-aws.html>)

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インスタンスで利用可能なライセンスオプションは、Bring Your Own License (BYOL) です。

自動化

- NetScaler ADM: スマート展開(</ja-jp/citrix-application-delivery-management-service/getting-started.html>)
- [GitHub CFTs: AWS展開用のNetScalerテンプレートとスクリプト](#)
- [GitHub Ansible: AWS展開用のNetScalerテンプレートとスクリプト](#)
- [GitHub Terraform: AWS展開用の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

- AWS上のNetScalerが顧客のアプリケーションを安全に配信するのにどのように役立つか (<https://aws.amazon.com/blogs/apn/how-citrix-adc-on-aws-helps-customers-deliver-applications-securely/>)
- [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: パブリッククラウドに常に準備万端](#)
- [NetScalerを介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](#)
- [CitrixはAWS OutpostsでADC展開の選択肢を拡大](#)
- [Amazon VPCイングルーティングでNetScalerを使用する](#)
- [CitrixはAWSで選択肢、パフォーマンス、簡素化された展開を提供](#)
- [NetScaler Web App Firewallのセキュリティ – AWS Marketplaceで利用可能に](#)
- [Aria SystemsがAWSでNetScaler Web App Firewallを使用する方法](#)

ビデオ

- [ADMを介したパブリッククラウドNetScaler展開の簡素化](#)
- [すぐに使えるTerraformスクリプトを使用してAWSでNetScaler VPXをプロビジョニングおよび構成する](#)
- [CloudFormationテンプレートを使用してAWSにNetScaler HAを展開する](#)
- [AWS QuickStartを使用してアベイラビリティゾーン全体にNetScaler HAを展開する](#)
- [ADMを使用したNetScaler Autoscale](#)

顧客事例

- [テクノロジーソリューション - Xenit AB](#)
- [NetScalerとAWSの利点を発見する](#)

ソリューション

- [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を使用してAWSでクリックストリーム分析を強化する](#)

サポート

- [サポートケースを開く](#)
- NetScalerサブスクリプション製品については、[AWS上のVPXインスタスのトラブルシューティング](#)を参照してください。サポートケースを申請するには、AWSアカウント番号とサポートPINコードを見つけて、NetScalerサポートに電話してください。
- NetScalerの顧客ライセンス製品またはBYOLの場合、有効なサポートおよびメンテナンス契約があることを確認してください。契約がない場合は、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 マーケットプレイスのネットスケイラー(https://aws.amazon.com/marketplace/seller-profile?id=fb9c6078-b60f-47f6-8622-49d5e1d5aca7&ref=_ptnr_proddocs)

- [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 よくある質問\(/ja-jp/vpx/13-1/deploy-aws/aws-faqs.html\)](#)

AWS 用語

このセクションでは、一般的に使用されるAWSの用語とフレーズのリストについて説明します。詳細については、[AWS Glossary](#)を参照してください。

用語	定義
アマゾン マシン イメージ (AMI)	インスタンス (クラウド内の仮想サーバー) を起動するために必要な情報を提供するマシンイメージです。
エラスティック ブロック ストア	AWS クラウドの Amazon EC2 インスタンスで使用するための永続的なブロックストレージボリュームを提供します。
シンプル ストレージ サービス (S3)	インターネット用のストレージです。開発者がウェブスケールのコンピューティングをより簡単に行えるように設計されています。
エラスティック コンピュート クラウド (EC2)	クラウドで安全かつサイズ変更可能なコンピューティング容量を提供するウェブサービスです。開発者がウェブスケールのクラウドコンピューティングをより簡単に行えるように設計されています。
エラスティック ロード バランシング (ELB)	複数のアベイラビリティゾーンにある複数の EC2 インスタンスに、受信アプリケーショントラフィックを分散します。これにより、アプリケーションの耐障害性が向上します。
エラスティック ネットワーク インターフェイス (ENI)	Virtual Private Cloud (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ース。
Elastic IP (EIP) アドレス	Amazon EC2 または Amazon VPC で割り当て、インスタンスにアタッチした静的なパブリック IPv4 アドレス。Elastic IP アドレスは、特定のインスタンスではなく、アカウントに関連

用語	定義
	付けられています。必要に応じて簡単に割り当て、アタッチ、デタッチ、解放できるため、柔軟性があります。
インスタンスタイプ	Amazon EC2は、さまざまなユースケースに合わせて最適化された幅広いインスタンスタイプを提供しています。インスタンスタイプは、CPU、メモリ、ストレージ、ネットワーク容量のさまざまな組み合わせで構成されており、アプリケーションに適したリソースの組み合わせを柔軟に選択できます。
ID およびアクセス管理 (IAM)	AWS内でそのIDができることとできないことを決定するアクセス許可ポリシーを持つAWS ID。IAMロールを使用すると、EC2インスタンスで実行されてい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インスタンスをデプロイするには、IAMロールが必要です。
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	電子的に身元を証明するために使用するセキュリティ認証情報のセット。キーペアは、秘密鍵と公開鍵で構成されます。
ルートテーブル	ルートテーブルに関連付けられているサブネットから出るトラフィックを制御するルーティングルールのセット。複数のサブネットを1つのルートテーブルに関連付けることができますが、サブネットは一度に1つのルートテーブルにのみ関連付けることができます。
セキュリティグループ	インスタンスに対する許可されたインバウンドネットワーク接続の名前付きセット。
サブネット	EC2インスタンスをアタッチできるVPCのIPアドレス範囲のセグメントです。セキュリティおよび運用上のニーズに応じてインスタンスを

用語	定義
	グループ化するためにサブネットを作成できます。
バーチャルプライベートクラウド (VPC)	定義した仮想ネットワークでAWSリソースを起動できる、AWSクラウドの論理的に分離されたセクションをプロビジョニングするためのウェブサービスです。
オートスケーリング	ユーザー定義のポリシー、スケジュール、および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ためのウェブサービスです。
CloudFormation	関連するAWSリソースを単位としてまとめて作成および削除するテンプレートを作成または変更するためのサービスです。

AWS-VPX サポートマトリックス

以下の表に、サポートされているVPXモデル、AWSリージョン、インスタンスタイプ、およびサービスを記載します。

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
ネットスケーラー VPX アドバンスド - 200 Mbps
ネットスケーラー VPX プレミアム - 1 Gbps
ネットスケーラー VPX プレミアム - 5 Gbps
NetScaler VPX - お客様ライセンス
NetScaler VPX FIPS - お客様ライセンス
NetScaler VPX FIPS ENA - お客様ライセンス

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)
米国西部 (北カリフォルニア)
米国東部 (オハイオ)
米国東部 (バージニア北部)
アジアパシフィック (ムンバイ)
アジアパシフィック (メルボルン)
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)
欧州 (フランクフルト)
欧州 (アイルランド)
欧州 (ロンドン)
EU (パリ)
EU (ミラノ)
南米 (サンパウロ)

サポートされているAWSリージョン
AWS GovCloud (米国東部)
AWS GovCloud (米国西部)
AWS 極秘 (C2S)
中東 (バーレーン)
中東 (UAE)
アフリカ (ケープタウン)
C2S

注:

AWS香港リージョンでは、NetScaler VPXのサポートは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

サポートされている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
c6in.large, c6in.xlarge, c6in.2xlarge, c6in.4xlarge, c6in.8xlarge, c6in.12xlarge, c6in.16xlarge, c6in.24xlarge, c6in.32xlarge
d2.xlarge, d2.2xlarge, d2.4xlarge, d2.8xlarge
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

サポートされているAWSインスタンスタイプ

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にプロビジョニングされたNetScaler VPXは、ENA低遅延キュー (LLQ) 機能をサポートしていません。

表 4: サポートされている 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。

Lambda: 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イングレスルーティングは、ファイアウォール負荷分散ソリューションで使用されます。

Route53: NetScaler Autoscale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。

サポートされている AWS サービス

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を監視します。

AWS Autoscaling: バックエンドサーバーのオートスケーリングに使用されます。

Cloud formation: CloudFormationテンプレートは、NetScaler VPXインスタンスの展開に使用されます。

Simple Queue Service (SQS): バックエンドのオートスケーリングにおける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。

Simple Notification Service (SNS): バックエンドのオートスケーリングにおける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。

Identity and Access Management (IAM): AWS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にNetScaler VPXインスタンスをプロビジョニングします。

NetScalerは、以下のAWSインスタンスタイプを推奨します。

- M6iおよびC5nシリーズ（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）。

プール型またはフレキシブルライセンス（帯域幅ライセンス）のVPX	推奨AWSインスタンス
最大200 Mbps	m6i.xlarge
1から5 Gbps	m6i.2xlarge
5から8 Gbps	c5n.4xlarge
8～25 ギガビット/秒	c5n.9xlarge

パケット/秒やSSLトランザクションレートなどのメトリックに基づいて、特定のニーズに合った適切なインスタンスを決定するには、NetScaler担当者にお問い合わせください。

制限事項と使用ガイドライン

AWSにNetScaler VPXインスタンスを展開する際に、以下の制限事項と使用ガイドラインが適用されます。

- 開始する前に、[Deploy a NetScaler VPX instance on AWS](#)のAWS用語セクションをお読みください。
- クラスターリング機能は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、管理インターフェースに専用のNATデバイスを関連付けるか、NSIPにEIPを関連付けます。NATの詳細については、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と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アドレスのみが存在する必要があります。
- NSIPにEIPを割り当てる代わりにセキュリティのためにNATインスタンスを使用する場合、適切なVPCレベルのルーティング変更が必要です。VPCレベルのルーティング変更を行う手順については、AWSドキュメントの[Scenario 2: VPC with Public and Private Subnets](#)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タイプへ移動できます（例：m3.largeからm3.xlargeへ）。
- AWS上のVPXのストレージオプションとして、Citrix®はEBSを推奨しています。これは、耐久性があり、インスタンスからデタッチされた後もデータが利用可能であるためです。
- VPXへのENIの動的な追加はサポートされていません。更新を適用するには、VPXインスタンスを再起動してください。Citrixは、スタンドアロンまたはHAインスタンスを停止し、新しいENIをアタッチしてからインスタンスを再起動することを推奨します。
- 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あたりの最大IPアドレス数はEC2インスタンスタイプによって決定されます。詳細については、[Elastic Network Interfaces](#)の「IP Addresses Per Network Interface Per Instance Type」セクションを参照してください。ENIに割り当てる前に、AWSでIPアドレスを割り当てる必要があります。詳細については、[Elastic Network Interfaces](#)を参照してください。
- Citrixは、NetScaler VPXインターフェースでenableおよびdisable interfaceコマンドを使用しないことを推奨します。
- NetScaler `set ha node \<NODE\_ID\> -haStatus STAYPRIMARY` および `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、以下の機能はサポートされていません。
 - 無償ARP (GARP)
 - 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、[エラスティックネットワークインターフェース](#)の「送信元/送信先チェックの変更」を参照してください。
 - AWS上でのNetScaler VPX展開において、一部のAWSリージョンでは、AWSインフラストラクチャがAWS APIコールを解決できない場合があります。これは、NetScaler VPXインスタンスの非管理インターフェースを介してAPIコールが発行された場合に発生します。回避策として、APIコールを管理インターフェースのみに制限してください。そのためには、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ースをNSVLANにバインドします。例: `set ns config -nsvlan <vlan id> -ifnum 1/1 -tagged NO` `save config` 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の構成](#)を参照してください。
 - AWSコンソールで、**監視**タブの下に表示されるVPXインスタンスのvCPU使用率が、実際の使用率よりもはるかに低い場合でも、高くなる（最大100パーセント）ことがあります。実際のvCPU使用率を確認するには、**すべてのCloudWatchメトリクスを表示**に移動します。詳細については、[Amazon CloudWatchを使用したインスタンスの監視](#)を参照してください。

前提条件

AWS で VPX インスタンスを作成する前に、以下があることを確認してください。

- **AWS アカウント**: AWS Virtual Private Cloud (VPC) で NetScaler VPX AMI を起動するため。AWS アカウントは (<http://www.aws.amazon.com>) で無料で作成できます。
- **AWS Identity and Access Management (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制御するため。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ている IAM ロールには、さまざまなシナリオに対して以下の IAM 権限が必要です。

同じ AWS ゾーン内の IPv4 アドレスを持つ HA ペア:

```
"ec2:DescribeInstances",  
  
"ec2:AssignPrivateIpAddresses",  
  
"iam:SimulatePrincipalPolicy",  
  
"iam:GetRole",  
  
"ec2:CreateTags"
```

同じ AWS ゾーン内の IPv6 アドレスを持つ HA ペア:

```
"ec2:DescribeInstances",  
  
"ec2:AssignIpv6Addresses",  
  
"ec2:UnassignIpv6Addresses",  
  
"iam:SimulatePrincipalPolicy",  
  
"iam:GetRole",  
  
"ec2:CreateTags"
```

同じ AWS ゾーン内の IPv4 および IPv6 アドレスの両方を持つ HA ペア:

```
"ec2:DescribeInstances",  
  
"ec2:AssignPrivateIpAddresses",  
  
"ec2:AssignIpv6Addresses",  
  
"ec2:UnassignIpv6Addresses",  
  
"iam:SimulatePrincipalPolicy",  
  
"iam:GetRole",  
  
"ec2:CreateTags"
```

異なる AWS ゾーン間の Elastic IP アドレスを持つ HA ペア:

```
"ec2:DescribeInstances",  
  
"ec2:DescribeAddresses",  
  
"ec2:AssociateAddress",  
  
"ec2:DisassociateAddress",  
  
"iam:SimulatePrincipalPolicy",  
  
"iam:GetRole",  
  
"ec2:CreateTags"
```

異なる AWS ゾーン間のプライベート 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 ゾーン間のプライベート 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",  
  
"ec2:CreateTags"
```

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",  
  
"ec2:CreateTags"
```

注:

- 上記の機能のいずれかの組み合わせを使用する場合は、各機能の IAM 権限の組み合わせを使用してください。
- Citrix CloudFormation テンプレートを使用する場合、IAM ロールは自動的に作成されます。このテンプレートでは、すでに作成されている IAM ロールを選択することはできません。
- GUIを介して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ためのプロンプトが表示されます。権限をすでに設定している場合は、そ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 Management Consoleが提供するすべての機能を使用するため。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。ネットワークインターフェースタイプをSR-IOVに変更するには、AWS CLIも必要です。
- **Elastic Network Adapter (ENA):** ENAドライバーが有効なインスタンスタイプ（M5、C5インスタンスなど）の場合、ファームウェアバージョンは13.0以上である必要があります。
- NetScaler VPXの場合、EC2インスタンスでInstance Metadata Service (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に利用できる2つのモードです。IMDSv2はIMDSv1よりも安全です。インスタンスは、両方の方法を使用するように（デフォルトオプション）、またはIMDSv2モードのみを使用するように（IMDSv1を無効にすることで）構成できます。Citrix ADC VPXは、NetScaler VPXリリース13.1.48.x以降、IMDSv2のみのモードをサポートしています。

NetScaler VPXインスタンスでAWS IAMロールを設定する

Amazon EC2インスタンスで実行されるアプリケーションは、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。しかし、その場合、認証情報を管理し、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時期が来たら各Amazon EC2インスタンスを更新する必要があります。これは多くの追加作業となります。

その代わりに、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ために、Identity and Access Management (IAM) ロールを使用することができますし、使用しなければなりません。ロールを使用する場合、長期的な認証情報（ユーザー名とパスワード、またはアクセスキーなど）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際に使用できる一時的な権限を提供します。Amazon EC2インスタンスを起動する際、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、ロールによって提供される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ているIAMロールには、さまざまなシナリオに対して以下のIAM権限が必要です。

同じAWSゾーン内のIPv4アドレスを持つHAペア:

```
"ec2:DescribeInstances",  
  
"ec2:AssignPrivateIpAddresses",  
  
"iam:SimulatePrincipalPolicy",  
  
"iam:GetRole"
```

同じAWSゾーン内のIPv6アドレスを持つHAペア:

```
"ec2:DescribeInstances",  
  
"ec2:AssignIpv6Addresses",  
  
"ec2:UnassignIpv6Addresses",  
  
"iam:SimulatePrincipalPolicy",  
  
"iam:GetRole"
```

同じAWSゾーン内のIPv4およびIPv6アドレスの両方を持つ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ペア:

```
"ec2:DescribeInstances",  
  
"ec2:DescribeAddresses",  
  
"ec2:AssociateAddress",  
  
"ec2:DisassociateAddress",  
  
"iam:SimulatePrincipalPolicy",  
  
"iam:GetRole"
```

異なるAWSゾーンにまたがるプライベート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ゾーンにまたがるプライベートIPおよび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バックエンドのオートスケーリング:

```
"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"
```

注意点:

- 前述の機能のいずれかの組み合わせを使用する場合は、各機能のIAM権限の組み合わせを使用してください。
- Citrix CloudFormationテンプレートを使用する場合、IAMロールは自動的に作成されます。このテンプレートでは、すでに作成されているIAMロールを選択することはできません。
- GUIを介して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ためのプロンプトが表示されます。すでに権限を設定している場合は、このプロンプトを無視してください。
- IAMロールは、スタンドアロン展開と高可用性展開の両方で必須です。

IAMロールを作成する

この手順では、AWSバックエンドの自動スケーリング機能用のIAMロールを作成する方法について説明します。

注:

他の機能に対応するIAMロールを作成する場合も、同じ手順に従うことができます。

1. EC2用AWSマネジメントコンソールにログオンします。
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、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ロールの作成をクリックします。



- b. ロールページで、ロールの作成をクリックします。

ロールの作成(/en-us/vpx/media/iam-create-role.png)

- c. 「信頼されたエンティティタイプ」の下にある「AWSサービス」と、「一般的なユースケース」の下にある「EC2」を選択し、「次へ」をクリックします。

信頼されたエンティティ (/en-us/vpx/media/iam-trusted-entity.png)

- d. 「アクセス許可の追加」ページで、「ポリシーの作成」をクリックします。

ポリシーの作成(/en-us/vpx/media/iam-create-policy.png)

- e. 「JSON」タブをクリックして、JSONエディターを開きます。

JSON(/en-us/vpx/media/iam-json.png)

- f. JSONエディターで、すべて削除し、使用したい機能のIAMアクセス許可を貼り付けます。

たとえば、AWSバックエンドオートスケーリング機能の次のIAMアクセス許可を貼り付けます。

```
{
  "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"

    ],

    "Resource": "*"

}

]

}

```

指定する「Version」キーと値のペアが、AWSによって自動的に生成されたものと同じであることを確認してください。

- g. 「**次へ: レビュー**」をクリックします。

IAMのレビュー(/en-us/vpx/media/iam-next-review.png)

- h. 「**ポリシーのレビュー**」タブで、ポリシーに有効な名前を付け、「**ポリシーの作成**」をクリックします。

ポリシーのレビュー(/en-us/vpx/media/iam-review-policy.png)

- i. 「**IDおよびアクセス管理**」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、および作成**ページで、ロールに有効な名前を付けます。



k. **ロールの作成**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して、作成したロールを表示します。



7. **IAMロールの更新**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アクセス制御ポリシーを本番環境にコミットする前にその効果をテストできるツールです。これにより、権限の検証とトラブルシューティングが容易になります。

1. **IAM**ページで、テストするIAMロールを選択し、**シミュレート**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. 「**Users, Groups and Roles**」タブで、ドロップダウンメニューから「**Roles**」を選択し、既存のロールを選択します。

既存のロールを選択(/en-us/vpx/media/iam-users-groups-roles.png)

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。

既存のポリシーを選択(/en-us/vpx/media/iam-users-groups-policies.png)

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。「**Select actions**」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。

目的のポリシーを選択(/en-us/vpx/media/iam-policy-json.png)

6. 「**Run simulation**」をクリックします。

目的のポリシーを選択(/en-us/vpx/media/iam-run-simulation.png)

詳細については、[AWS IAM ドキュメント](#)を参照してください。

関連情報

[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権限を付与するための IAM ロールの使用](#)

AWS 上での NetScaler VPX インスタンスの動作方法

NetScaler VPX インスタンスは、AWS Marketplace で AMI として利用でき、AWS VPC 内で EC2 インスタンスとして起動できます。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。AWS VPC 内で起動された EC2 インスタンスは、VPX の構成に必要な複数のインターフェース、インターフェースごとの複数の IP アドレス、およびパブリック IP アドレスとプライベート 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
- クライアント向けサブネット (VIP)
- バックエンド向けサブネット (SNIP、MIP など)

Citrix® 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ースを推奨しています。

AWS は現在、マルチ IP 機能を AWS VPC 内で実行されているインスタンスにのみ提供しています。VPC 内の VPX インスタンスは、EC2 インスタンスで実行されているサーバーのロードバランシングに使用できます。Amazon VPC を使用すると、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を含む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、AWS アカウントごとに AWS リージョンあたり最大 5 つの VPC インスタンスを作成できます。より高い VPC 制限をリクエストするには、Amazon のリクエストフォーム <http://aws.amazon.com/contact-us/vpc-request> を提出してください。

図 1. AWS アーキテクチャにおける NetScaler VPX インスタンスの展開例



図 1 は、NetScaler VPX が展開された AWS VPC のシンプルなトポロジを示しています。AWS VPC には以下が含まれます。

1. VPC 内外の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. 管理用、クライアント用、サーバー用にそれぞれ1つずつ、計3つのサブネット。
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内に展開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3つのENIがあり、それぞれが各サブネットに接続されています。

AWS に NetScaler VPX スタンドアロンインスタンスを展開する

次のオプションを使用して、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、AWS に NetScaler VPX インスタンスを展開する手順について説明します。

展開を開始する前に、次のトピックをお読みください。

- [前提条件](#)
- [制限事項と使用ガイドライン](#)

AWS ウェブコンソールを使用して AWS に NetScaler VPX インスタンスを展開する

AWS ウェブコンソールを通じて AWS に NetScaler VPX インスタンスを展開できます。展開プロセスには、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. サブネットの追加
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. ルートテーブルを追加
6. インターネットゲートウェイを作成

7. NetScaler VPXインスタンスを作成
8. ネットワークインターフェイスを追加作成してアタッチ
9. 管理NICにElastic IPをアタッチ
10. VPXインスタンスに接続

ステップ1：キーペアを作成します。

Amazon EC2は、ログオン情報を暗号化および復号化するためにキーペアを使用します。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秘密鍵を提供する必要があります。

AWSインスタンス起動ウィザードを使用してインスタンスを確認して起動すると、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 Key Pairs](#)」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内に展開されます。VPCを使用すると、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、「[Getting Started With Amazon VPC](#)」を参照してください。

NetScaler VPXインスタンス用のVPCを作成する際は、次の点に注意してください。

- 「VPC with a Single Public Subnet Only」オプションを使用して、AWSアベイラビリティゾーンにAWS VPCを作成します。
- Citrixは、次の種類の**3つ以上のサブネット**を作成することをお勧めします。
 - 管理トラフィック用のサブネットを1つ。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、管理IPにはElastic Network Interface（ENI）eth0が使用されます。
 - クライアントアクセス（ユーザーからNetScaler VPXへ）トラフィック用のサブネットを1つ以上。クライアントは、このサブネットを介して、NetScalerロードバランシング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）トラフィック用のサブネットを1つ以上。サーバーは、このサブネットを介して、VPXが所有するサブネットIP（SNIP）アドレスに接続します。NetScalerのロードバランシングと仮想サーバー、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ゾーンにある必要があります。

ステップ3：サブネットを追加する。

VPCウィザードを使用した場合、作成されたサブネットは1つだけでした。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。サブネットの追加方法の詳細については、「[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する。

インバウンドおよびアウトバウンド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、グループにルールを追加します。グループの作成方法とルールの追加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。これらはAWS Marketplace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、要件に基づいてさらにセキュリティグループを作成することもできます。

注：

SSH、HTTP、HTTPSアクセス用に、セキュリティグループでポート22、80、443を開放する必要があります。

ステップ5：ルートテーブルを追加する。

ルートテーブルには、ネットワークトラフィックの転送先を決定するために使用される「ルート」と呼ばれる一連のルールが含まれています。VPC内の各サブネットは、ルートテーブルに関連付けられている必要があります。ルートテーブルの作成方法の詳細については、「[ルートテーブル](#)」を参照してください。

ステップ6：インターネットゲートウェイを作成する。

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。VPCルートテーブルでインターネットルーティング可能なトラフィックのターゲットを提供することと、パブリックIPv4アドレスが割り当てられたインスタンスの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用の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のタッチ](#)」セクション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 VPXインスタンスを作成する。」

AWS EC2サービスを使用してNetScaler VPXインスタンスを作成するには、次の手順を実行します。

1. AWSダッシュボードから、**Compute > EC2 > Launch Instance > AWS Marketplace** に移動します。

Launch Instance をクリックする前に、**Launch Instance** の下に表示されるメモを確認して、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」というキーワードで検索します。
3. デプロイしたいバージョンを選択し、**Select** をクリックします。NetScaler VPXバージョンには、次のオプションがあります。

- ライセンス版
- NetScaler VPX Expressアプライアンス（これは、NetScaler 12.0 56.20以降で利用可能な無料の仮想アプライアンスです。）
- 独自のデバイスを持ち込む

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。ウィザードは次のことを促します。

- インスタンスタイプの選択
- インスタンスの設定
- ストレージの追加
- タグの追加
- セキュリティグループを設定する
- 確認

EC2ウィザード(/en-us/vpx/media/ec2-wizard.png)

ステップ8：ネットワークインターフェースを追加作成してアタッチする。

VIPとSNIP用にさらに2つのネットワークインターフェースを作成します。ネットワークインターフェースの追加作成方法の詳細については、「[ネットワークインターフェースの作成](#)」セクションを参照してください。

ネットワークインターフェースを作成したら、それらをVPXインスタンスにアタッチする必要があります。インターフェースをアタッチする前に、VPXインスタンスをシャットダウンし、インターフェースをアタッチしてから、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ースのアタッチ方法の詳細については、「[インスタンス起動時のネットワークインターフェースのアタッチ](#)」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る。

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するまでのみ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、インスタンスから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用にElastic IPを割り当てて関連付けます。Elastic IPアドレスの割り当てと関連付けの方法の詳細については、次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](#)
- [実行中のインスタンスへのElastic IPアドレスの関連付け](#)

これらの手順で、AWS上にNetScaler VPXインスタンスを作成する手順は完了です。インスタンスが準備でき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ていることを確認してください。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する。

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に接続します。

- GUI

NetScaler VPXインスタンスにアクセスするためのデフォルトの管理者資格情報は次のとおりです。

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に設定されています。初回ログオン時には、セキュリティ上の理由からパスワードの変更を求められます。パスワードを変更した後、設定を保存する必要があります。設定が保存されずにインスタンスが再起動した場合、デフォルトパスワードでログオンする必要があります。プロンプトで再度パスワードを変更してください。

- 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、**Connect**をクリックします。**Connect to Your Instance**ページに表示される指示に従ってください。

AWSウェブ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 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。このテンプレートは、単一のNetScaler VPXインスタンスを起動する機能、またはNetScaler VPXインスタンスのペアで高可用性環境を作成する機能を提供します。

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環境が必要であり、3つのElastic Network Interface (ENI) を持つ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: 管理用、クライアントトラフィック用、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ための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

前提条件を完了する方法の詳細については、「[AWS Web コンソールを使用して AWS に NetScaler VPX インスタンスを展開する](#)」セクションまたはAWSドキュメントを参照してください。

さらに、GitHubで入手できるCitrix CloudFormationテンプレートを使用して、NetScaler VPX Expressのスタンドアロンインスタンスを構成して起動します。

```
<https://github.com/citrix/citrix-adc-aws-cloudformation/tree/master/templates/standalone/>
```

スタンドアロン展開ではIAMロールは必須ではありません。ただし、Citrix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、インスタンスにアタッチすることを推奨します。IAMロールにより、必要に応じてスタンドアロンインスタンスをSR-IOVを備えた高可用性ノードに簡単に変換できます。

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。

注:

AWS Web コンソールを使用して AWS に NetScaler VPX インスタンスを展開する場合、CloudWatch サービスはデフォルトで有効になっています。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 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

このシナリオでは、AWS GUIを使用してAWSにNetScaler VPXスタンドアロンEC2インスタンスを展開する方法を示します。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。ロードバランシング仮想サーバーとして構成されたインスタンスは、バックエンドサーバー（サーバーファーム）と通信します。この構成では、インスタンスとバックエンドサーバー間、および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、[NetScaler VPXスタンドアロンインスタンスをAWSに展開する](#)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）で構成できます。NICは以下の目的で使用されます。

NIC	目的	関連付けられているもの
eth0	管理トラフィックを処理します (NSIP)	パブリックIPアドレスとプライベートIPアドレス
eth1	クライアント側トラフィックを処理します (VIP)	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ーと通信します (SNIP)	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、**Networking & Content Delivery** \> **VPC** に移動します。「**Start VPC Wizard**」をクリックします。
2. 「**VPC with a Single Public Subnet**」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に名前を付けます。
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ます。
7. サブネットに名前を付けます。
8. 「**VPC**」を作成をクリックします。

シナリオ用のVPCを作成します。(en-us/vpx/media/createvpc-standalone-sceanario.png)

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、以下の詳細を入力した後、「サブネット」、「サブネットの作成」を選択します。
 - 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。
 - アベイラビリティゾーン: ステップ1でVPCを作成したアベイラビリティ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。

シナリオ用のサブネットを作成(en-us/vpx/media/createsubnet-standalone-scenario.png)

3. バックエンドサーバー用に、もう1つのサブネットを作成する手順を繰り返します。

シナリオ用の追加サブネット(en-us/vpx/media/subnet-server-scenario.png)

ステップ3: ルートテーブルを作成します。

1. Amazon VPCコンソールを `<https://console.aws.amazon.com/vpc/>` で開きます。
2. ナビゲーションペインで、**ルートテーブル** > **ルートテーブルの作成**を選択します。
3. ルートテーブルの作成ウィンドウで、名前を追加し、ステップ1で作成したVPCを選択します。

4. はい、作成をクリックします。

シナリオ用のルートテーブルを作成(/en-us/vpx/media/create-routetable-standalone-scenario.png)

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、あるサブネットのインスタンスからのトラフィック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. 管理およびクライアントサブネットをクリックし、「保存」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。

シナリオのルートテーブルを関連付ける(/en-us/vpx/media/routetable-associate-scenario.png)

7. 「Routes > Edit > Add another route」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、VPCウィザードが自動的に作成したインターネットゲートウェイである `igw-xxxx` を選択します。

9. 「保存」をクリックします。

シナリオのゲートウェイを追加(/en-us/vpx/media/add-gateway-scenario.png)

10. サーバー側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**」の下にある「**EC2**」をクリックします。
2. 「AWS Marketplace」をクリックします。「Search AWS Marketplace」バーに「NetScaler VPX」と入力し、Enterキーを押します。利用可能なNetScaler VPXエディションが表示されます。
3. 目的のNetScaler VPXエディションを選択するには、「**Select**」をクリックします。EC2インスタンスウィザードが起動します。
4. 「**Choose Instance Type**」ページで、「**m4. Xlarge**」（推奨）を選択し、「**Next: Configure Instance Details**」をクリックします。
5. 「Configure Instance Details」ページで、以下を選択し、その後「Next: Add Storage」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

シナリオのインスタンスを構成する(/en-us/vpx/media/configure-instance-scenario.png)

6. ページで、デフォルトオプションを選択し、 をクリックします。
7. ページで、インスタンスの名前を追加し、 をクリックします。
8. ページで、デフォルトオプション (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、 > をクリックします。
9. 既存のキーペアを選択するか、新しいキーペアを作成するように求められます。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. キーペアを承認するチェックボックスをオンにし、 をクリックします。

シナリオのキーペアを追加する(/en-us/vpx/media/keypair-scenario.png)

インスタンス起動ウィザードに起動ステータスが表示され、インスタンスが完全に起動す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、 > をクリックします。インスタンスを選択して名前を追加します。インスタンスの状態が「実行中」であり、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: ネットワークインターフェイスを追加で作成してアタッチする。

VPC を作成したとき、関連付けられているネットワークインターフェイスは 1 つだけでした。VIP と SNIP 用に、さらに 2 つのネットワークインターフェイスを VPC に追加します。

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、 を選択します。
3. 「ネットワークインターフェイスの作成」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には、グループを選択します。
8. 「はい、作成」をクリックします。

NICの作成(/en-us/vpx/media/create-nic-scenario.png)

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. サーバー側トラフィック用のネットワークインターフェースを作成する手順を繰り返します。

ネットワークインターフェースをアタッチする:

1. ナビゲーションペインで、「ネットワークインターフェース」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」を選択します。
3. 「ネットワークインターフェースのアタッチ」ダイアログボックスで、インスタンスを選択し、「アタッチ」を選択します。

NICのアタッチ(/en-us/vpx/media/attach-nic-scenario.png)

ステップ6: Elastic IPをNSIPにアタッチします。

1. AWS マネジメントコンソールから、**NETWORK & SECURITY > Elastic IPs** に移動します。
2. 利用可能な空き EIP があるか確認します。ない場合は、**Allocate new address** 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、**Actions > Associate address** を選択します。
4. **Network interface** ラジオ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. **Private IP** ドロップダウンメニューから、AWS が生成した IP アドレスを選択します。
7. **Reassociation** チェックボックスを選択します。
8. **Associate** をクリックします。

EIP をアタッチ(/en-us/vpx/media/attach-eip-scenario.png)

VPX インスタンスにアクセスする:

3つのNICを持つスタンドアロンのNetScaler VPXインスタンスを設定した後、VPXインスタンスにログオンしてNetScaler側の設定を完了します。以下のオプションを使用します。

- GUI: ブラウザで管理NICのパブリックIPを入力します。ユーザー名として **nsroot** を、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:

初回ログオン時には、セキュリティ上の理由からパスワードの変更を求められます。パスワードを変更した後、設定を保存する必要があります。設定が保存されずにインスタンスが再起動した場合、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。プロンプトで再度パスワードを変更し、設定を保存してください。

- SSH: 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\> ns root@\<public DNS of the instance\>
```

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、**Connect** をクリックします。

関連情報:

- NetScaler が所有する IP アドレス (NSIP、VIP、SNIP) を構成するには、「」を参照してください([/ja-jp/citrix-adc/13-1/networking/ip-addressing/configuring-citrix-adc-owned-ip-addresses.html](https://ja-jp.citrix-adc/13-1/networking/ip-addressing/configuring-citrix-adc-owned-ip-addresses.html))。
- 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、https://support.citrix.com/s/article/CTX255959-how-to-allocate-and-install-citrix-netscaler-vpx-licenses?language=en_US の VPX ライセンスガイドを参照してください。

NetScaler VPXライセンスをダウンロードする

AWS MarketplaceからNetScaler 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。

次の操作を行う必要があります。

1. Citrix®ウェブ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。

これが**有料**のMarketplaceインスタンスである場合、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は自動的に有効になります。

VPX 5000より高いモデル番号のNetScaler VPXインスタンスを使用する場合、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ものと同じではない可能性があります。ただし、SSLスループットや1秒あたりのSSLトランザクションなどの他の機能は向上する可能性があります。

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Bring your own license (BYOL)への移行手順、およびその逆の手順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、その他のすべてのステップはAWSポータルで実行されます。

1. 古いEC2インスタンスと同じセキュリティグループ、IAMロール、サブネットを持つ同じ Availability Zoneに、[NetScaler VPX - Customer Licensed](#)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つだけが必要です。
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. 「**System > Backup and Restore**」に移動します。
- b. 「**Welcome**」ページで、「**Backup/Import**」をクリックしてプロセスを開始します。

バックアップと復元ページ(/en-us/vpx/media/vpx-backup-and-restore-citrix.png)

- c. 「**Backup/Import**」ページで、以下の詳細を入力します。

- **名前** – バックアップファイルの名前。
- **レベル** – バックアップレベルを「**Full**」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。

バックアップとインポートページ(/en-us/vpx/media/vpx-backup-import-citrix.png)

- d. 「**Backup**」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。

バックアップファイルのダウンロード(/en-us/vpx/media/vpx-download-backup-file-citrix.png)

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. 「**System > Backup and Restore**」に移動します。
- b. 「**Backup/Import**」をクリックしてプロセスを開始します。
- c. 「**Import**」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。

バックアップファイルのインポート(/en-us/vpx/media/vpx-import-backup-file-citrix.png)

- d. ファイルを選択します。
- e. **Select Action**ドロップダウンメニューから、**Restore**を選択します。

ファイルを復元(/en-us/vpx/media/vpx-restore-file-citrix.png)

- f. **Restore**ページで、ファイルの詳細を確認し、**Restore**をクリックします。

復元するファイルを確認(/en-us/vpx/media/vpx-verify-file-for-restore-citrix.png)

- g. 復元後、EC2インスタンスを再起動します。

4. すべてのインターフェース（NSIPアドレスがバインドされている管理インターフェースを除く）を古いEC2インスタンスから新しいEC2インスタンスに移動します。ネットワークインターフェースをあるEC2インスタンスから別のEC2インスタンスに移動するには、次の手順に従います。

- a. **AWS Portal**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. **Network Interfaces**に移動し、古いEC2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ースを選択します。
- c. **Actions > Detach**をクリックして、EC2インスタンスをデタッチします。

AWSでEC2インスタンスをデタッチ(/en-us/vpx/media/aws-detach-ec2-instance.png)

- d. **Actions > Attach**をクリックして、ネットワークインターフェースを新しい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ー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。

AWSでネットワークインターフェースをアタッチ(/en-us/vpx/media/aws-attach-network-interface.png)

- e. アタッチされている他のすべてのインターフェースについて、**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、インターフェースの順序を維持するようにしてください。つまり、まずインターフェース2をデタッチしてアタッチし、次にインターフェース3をデタッチしてアタッチするというように続けます。

5. 管理インターフェースを古いEC2インスタンスからデタッチすることはできません。そのため、古いEC2インスタンスの管理インターフェース（プライマリネットワークインターフェース）

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ースから別のインターフェースに移動するには、次の手順に従います。

- a. 「**AWS Portal**」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「**停止**」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ている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**」をクリックし、割り当てられているすべてのセカンダリIPアドレス（もしあれば）をメモします。
- 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**」をクリックします。
- f. 「**IPv4アドレス**」の下で、「**新しいIPアドレスの割り当て**」をクリックします。
- g. 「**ステップ3**」でメモした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選択します。
- i. 「**保存**」をクリックします。

AWSでIPアドレスを移動する(/en-us/vpx/media/aws-move-ip-address.png)

6. 新しいEC2インスタンスを起動し、構成を確認します。すべての構成が移動された後、要件に応じて古い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したい場合は、古いインスタンスと新しいインスタンスの間で同じ手順を逆の順序で実行します。
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ウェブ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。

注記:

BYOLインスタンスをサブスクリプションインスタンス（有料のマーケットプレイスインスタンス）に移行する場合、ライセンスをインストールする必要はありません。適切な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。そのため、Citrixは管理インターフェイスを手動で構成することを推奨しています。詳細については、前述の手順の**ステップ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として作成されますが、NSIPアドレスのみが新しいIPアドレスを持ちます。

異なるアベイラビリティゾーンのサーバーの負荷分散

VPXインスタンスは、同じアベイラビリティゾーンで実行されているサーバー、または以下の場所で実行されているサーバーの負荷分散に使用できます。

- 同じAWS VPC内の異なるアベイラビリティゾーン (AZ)
- 異なるAWSリージョン
- VPC内のAWS EC2

VPXインスタンスが配置されているAWS VPCの外部で実行されているサーバーの負荷分散をVPXインスタンスで有効にするには、インスタンスが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構成します。手順は次のとおりです。

1. NetScaler CLIまたはGUIを使用して、NetScaler VPXインスタンスに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に面したサブネットを作成して、AZ外へのトラフィックのルーティングを有効にします。
3. AWS GUIコンソールを使用して、ルーティングテーブルにインターネットゲートウェイルートを追加します。
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アドレスにマッピング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での高可用性の仕組み

AWSで2つのNetScaler 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。一方のインスタンスをプライマリノードとして、もう一方をセカンダリノードとして構成すると、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。何らかの理由でプライマリノードが接続を受け入れられなくなった場合、セカンダリノードが引き継ぎます。

AWSでは、VPXインスタンスに対して以下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
- 異なるゾーン間での高可用性

注:

高可用性を機能させるには、両方のNetScaler VPXインスタンスにIAMロールがアタッチされ、NSIPにElastic IP (EIP) アドレスが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到達できる場合、NSIPにEIPを割り当てる必要はありません。

同一ゾーン内での高可用性

同一ゾーン内での高可用性デプロイでは、両方のVPXインスタンスが同様のネットワーク構成を持つ必要があります。

次の2つのルールに従ってください。

ルール1. あるVPXインスタンス上の任意のNICは、もう一方のVPXの対応するNICと同じサブネット内にある必要があります。両方のインスタンスは以下を持つ必要があります。

- 同じサブネット上の管理インターフェース (管理サブネットと呼ばれる)
- 同じサブネット上のクライアントインターフェース (クライアントサブネットと呼ばれる)
- 同じサブネット上のサーバーインターフェース (サーバーサブネットと呼ばれる)

ルール2. 両方のインスタンスにおける管理NIC、クライアントNIC、サーバーNICの順序は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。

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: サーバー

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が機能するためには、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のいずれかに存在する必要があります。

13.0 41.xx以降、プライマリHAノードのNIC（クライアント側およびサーバー側NIC）にアタッチされたセカンダリプライベートIPアドレスを、フェイルオーバー後にセカンダリHAノードに移行することで、高可用性を実現できます。このデプロイでは、次のようになります。

- 両方のVPXインスタンスは、NIC列挙に従って同じ数のNICとサブネットマッピングを持っています。
- 各VPX NICには、管理IPアドレスに対応する最初のNICを除いて、1つの追加のプライベートIPアドレスがあります。この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は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、この追加の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として構成されてはなりません。
- 必要に応じて、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として構成する必要があります。
- フェイルオーバー時、新しいプライマリノードは構成されたSNIPとVIPを検索し、以前のプライマリにアタッチされていたNICから、新しいプライマリの対応するNICにそれらを移動します。
- NetScalerインスタンスがHAを機能させるには、IAM権限が必要です。各インスタンスに追加されたIAMポリシーに、以下のIAM権限を追加してください。

```
"iam:GetRole" "ec2:DescribeInstances" "ec2:DescribeNetworkInterfaces"  
"ec2:AssignPrivateIpAddresses"
```

注記:

`unassignPrivateIpAddress` は不要です。

この方法は従来の方式よりも高速です。従来の方式では、HA はプライマリノードの AWS 弾性ネットワークインターフェースがセカンダリノードに移行することに依存していました。

従来の方式では、以下のポリシーが必要です。

```
"iam:GetRole" "ec2:DescribeInstances" "ec2:DescribeAddresses"
"ec2:AssociateAddress" "ec2:DisassociateAddress"
```

詳細については、[AWS に高可用性ペアを展開する](#) を参照してください。

異なるゾーン間での高可用性

2つの異なるサブネットまたは2つの異なるAWSアベイラビリティゾーンに、2つのNetScaler VPX インスタンスを独立ネットワーク構成（INC）モードの高可用性アクティブ/パッシブペアとして構成できます。フェイルオーバーが発生すると、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ぎます。フェイルオーバープロセスでは、AWS APIが以下を実行します。

- `IPSets` がアタッチ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（1つは仮想サーバーに直接アタッチされ、もう1つはIPセットを介してアタッチされている）の中から、関連付けられたパブリックIPを持つIPアドレスを見つけます。
- 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。

異なるゾーン間でのHAには、以下のポリシーが必要です。

```
"iam:GetRole" "ec2:DescribeInstances" "ec2:DescribeAddresses"
"ec2:AssociateAddress" "ec2:DisassociateAddress"
```

詳細については、[AWS アベイラビリティゾーン間での高可用性](#) を参照してください。

展開を開始する前に

AWSでHA展開を開始する前に、以下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にNetScaler 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 VPXインスタンスのHAフェイルオーバー中に発生した障害をトラブルシューティングするには、`/var/log/` 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アベイラビリティゾーンにVPX HAペアをデプロイする

注:

NetScalerリリース13.1ビルド27.x以降、同じAWSアベイラビリティゾーンのVPX HAペアはIPv6アドレスをサポートします。

AWS上に2つのNetScaler VPXインスタンスをHAペアとして構成できます。このHAペアは、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ゾーンに配置されます。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ー側NIC）にアタッチ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ことによるHAフェイルオーバーシナリオを示しています。

図1. プライベートIP移行を使用する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にNetScaler VPXインスタンスをデプロイする](#)
- [高可用性](#)

同じゾーンにVPX HAペアをデプロイする方法

同じゾーンにVPX HAペアをデプロイする手順の概要は次のとおりです。

1. AWS上に2つのVPXインスタンスを、それぞれ3つのNICで作成します。
2. プライマリノードのVIPとSNIPにAWSセカンダリプライベートIPアドレスを割り当てます。
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。

ステップ1. 同じVPCを使用して、それぞれ3つのNIC（Ethernet 0、Ethernet 1、Ethernet 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

[AWSウェブコンソールを使用してAWSにNetScaler VPXインスタンスを展開する](#)に記載されている手順に従ってください。

ステップ2. プライマリノードで、Ethernet 1（クライアントIPまたはVIP）とEthernet 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

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アドレスとして知られる追加のプライベートIPアドレスを割り当ててください。

ネットワークインターフェースにプライベートIPアドレスを割り当てるには、次の手順に従ってください。

1. `<https://console.aws.amazon.com/ec2/>` でAmazon EC2コンソールを開いてください。
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ースを選択してください。
3. **Actions > Manage IP Addresses** を選択してください。
4. 要件に基づいて、**IPv4 Addresses** または **IPv6 Addresses** を選択してください。
5. IPv4アドレスの場合:
 - a. **Assign new IP** を選択してください。
 - 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にIPアドレスを選択させます。
 - c. (オプション) **再割り当てを許可**を選択すると、セカンダリプライベートIPアドレスがすでに別のネットワークインターフェースに割り当てられている場合に、再割り当てできるようになります。
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にIPアドレスを選択させます。
- c. (オプション) **再割り当てを許可**を選択すると、プライマリまたはセカンダリプライベートIPアドレスがすでに別のネットワークインターフェースに割り当てられている場合に、再割り当てできるようになります。

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、インターフェース上のセカンダリIPv4アドレスのみを割り当て、それらをVIPおよびSNIPアドレスとして使用できます。しかし、IPv6 HAペア展開では、インターフェース上のプライマリIPv6またはセカンダリIPv6アドレスのいずれかを割り当て、それらをVIPおよびSNIPアドレスとして使用できます。

ステップ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を構成する

SSHを使用してプライマリノードにアクセスします。sshクライアントを開いて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を構成します。

VIPの場合、次のように入力します。

```
add ns ip <IPAddress> <netmask> -type <type>
```

SNIPの場合、次のように入力します。

```
add ns ip <IPAddress> <netmask> -type SNIP
```

save config と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
```

詳細については、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を構成する

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。

設定されたHAノードを表示するには、`show ha node` と入力します。

フェイルオーバーが発生すると、以前のプライマリノードでVIPおよびSNIPとして構成されてい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。

VPX HAペアをデプロイするためのレガシーメソッド

13.0 41.xリリース以前は、同一ゾーン内のHAはAWS Elastic Network Interface (ENI) の移行によって実現されていました。しかし、この方法は徐々に非推奨になっています。

次の図は、AWS上のNetScaler VPXインスタンスのHA展開アーキテクチャの例を示しています。

図 1. ENI移行を使用したAWS上のNetScaler VPX HAペア



以下のいずれかのオプションを使用して、AWS上に2つのVPXインスタンスをHAペアとして展開できます。

- AWS Management Consoleを使用してIAMロールを持つインスタンスを手動で作成し、それらのインスタンスでHAを構成します。
- または、Citrix CloudFormationテンプレートを使用して高可用性展開を自動化します。

CloudFormationテンプレートを使用すると、HAペアの作成に必要な手順が大幅に減少し、IAMロールが自動的に作成されます。このセクションでは、Citrix CloudFormationテンプレートを使用してNetScaler VPX HA（アクティブ-パッシブ）ペアを展開する方法について説明します。

2つのNetScaler VPXインスタンスをHAペアとして展開する際は、以下の点に留意してください。

注意事項

- AWS上のHAでは、プライマリノードに少なくとも2つのENI（1つは管理用、もう1つはデータトラフィック用）が必要であり、セカンダリノードには1つの管理用ENIが必要です。ただし、セキュリティ上の目的から、プライマリノードには3つのENIを作成してください。この設定により、プライベートネットワークとパブリックネットワークを分離できます（推奨）。
- セカンダリノードには常に1つのENIインターフェイス（管理用）があり、プライマリノードには最大4つのENIを持つことができます。
- 高可用性ペア内の各VPXインスタンスのNSIPアドレスは、インスタンスのデフォルトENIに構成する必要があります。
- AmazonはAWSでブロードキャスト/マルチキャストパケットを許可していません。そのため、HA設定では、プライマリVPXインスタンスが失敗した場合、データプレーンENIはプライマリからセカンダリVPXインスタンスに移行されます。
- デフォルト（管理用）ENIは別のVPXインスタンスに移動できないため、クライアントおよびサーバートラフィック（データプレーンのトラフィック）にはデフォルトENIを使用しないでください。
- /var/log/ns.log にあるメッセージ「AWSCONFIG IOCTL NSAPI_HOTPLUG_INTF success output 0」は、2つのデータENIがセカンダリインスタンス（新しいプライマリ）に正常にアタッチされたことを示しています。
- AWSのENIデタッチ/アタッチメカニズムにより、フェールオーバーに最大20秒かかる場合があります。
- フェイルオーバー時、失敗したインスタンスは常に再起動します。
- ハートビートパケットは管理インターフェースでのみ受信されます。
- プライマリおよびセカンダリVPXインスタンスの設定ファイルは、`nsroot` パスワードを含め同期されます。HA設定同期後、セカンダリノードの `nsroot` パスワードはプライマリノードのものに設定されます。

- AWS APIサーバーにアクセスするには、VPXインスタンスにパブリックIPアドレスが割り当てられているか、VPCサブネットレベルでVPCのインターネットゲートウェイを指すようにルーティングが正しく設定されている必要があります。
- ネームサーバー/DNSサーバーは、DHCPオプションを使用してVPCレベルで設定されます。
- Citrix CloudFormationテンプレートは、異なるアベイラビリティゾーン間でHAセットアップを作成しません。
- Citrix CloudFormationテンプレートはINCモードを作成しません。
- AWSデバッグメッセージは、VPXインスタンス上のログファイル `/var/log/ns.log` で利用可能です。

Citrix CloudFormationテンプレートを使用して高可用性ペアを展開する

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。

- 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
- クライアントおよび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します。

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表示されません。

Citrix CloudFormation テンプレートを起動するには：

1. AWS 認証情報を使用して、[AWS marketplace](#) にログインします。
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 オファリングをクリックします。
4. 「**Pricing**」タブをクリックして、「**Pricing Information**」に移動します。

5. リージョンと「**Fulfillment Option**」として「**NetScaler VPX – Customer Licensed**」を選択します。
6. 「**Continue to Subscribe**」をクリックします。
7. 「**Subscribe**」ページで詳細を確認し、「**Continue to Configuration**」をクリックします。
8. 「**Delivery Method**」として「**CloudFormation Template**」を選択します。
9. 必要な CloudFormation テンプレートを選択します。
10. 「**Software Version**」と「**Region**」を選択し、「**Continue to Launch**」をクリックします。



11. 「**アクションの選択**」で「**CloudFormation の起動**」を選択し、「**起動**」をクリックします。「**スタックの作成**」ページが表示されます。
12. 「**次へ**」をクリックします。



13. 「**スタックの詳細の指定**」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 内に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てください。
 - 「**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「**サーバー 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 アドレスを自動的に割り当てさせることもできます。
 - 「**VPCTenancy**」には「**default**」を選択します。
- 「**NetScaler 構成**」で、次の操作を実行します。
 - 「**インスタンスタイプ**」には「**m5.xlarge**」を選択します。
 - 「**キーペア**」のメニューから、すでに作成済みのキーペアを選択します。
 - デフォルトでは、「**カスタムメトリクスを CloudWatch に公開しますか？**」オプションは「**はい**」に設定されています。このオプションを無効にする場合は、「**いいえ**」を選択します。

CloudWatch メトリクスの詳細については、([#monitor-your-instances-using-amazon-cloudWatch](#)) を参照してください。

- 「オプション構成」で、次の操作を実行します。
 - デフォルトでは、管理インターフェースにpublicIP(EIP)を割り当てるべきですか？オプションはいいいえに設定されています。
 - デフォルトでは、クライアントインターフェースにpublicIP(EIP)を割り当てるべきですか？オプションはい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. レビューページが表示されます。設定を確認し、必要に応じて変更を加えてください。

19. **AWS CloudFormationがIAMリソースを作成する可能性があることを認識しています。**
チェックボックスを選択し、スタックの作成をクリックします。

20. **CREATE-IN-PROGRESS**ステータスが表示されます。ステータスが**CREATE-COMplete**になるまで待ちます。ステータスが**COMPLETE**に変わらない場合は、イベント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てください。



21. IAMリソースが作成された後、**EC2管理コンソール > 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インスタンスが見つかります。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ースで作成されます。

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。

CloudFormation VPX HAペアが構成されています(/en-us/vpx/media/cloudformation-adc-vpx-ha-pair.png)

Amazon CloudWatch を使用してインスタンスを監視する

Amazon CloudWatch サービスを使用して、CPU およびメモリ使用率、スループットなどの NetScaler VPX メトリックのセットを監視できます。CloudWatch は、AWS で実行されているリソースとアプリケーションをリアルタイムで監視します。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。

注意点

- AWS Web コンソールを使用して AWS に NetScaler VPX インスタンスを展開する場合、CloudWatch サービスはデフォルトで有効になっています。
- 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。
- メトリックは、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 (インバウンドおよびアウトバウンド) で利用できます。

CloudWatch メトリックの表示方法

インスタンスの CloudWatch メトリックを表示するには、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** にログインします。
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. **すべての CloudWatch メトリックを表示** をクリックします。

CloudWatch メトリック (/en-us/vpx/media/view-cloudwatch-metrics.png)

5. すべてのメトリックの下で、インスタンス ID をクリックします。

6. 表示したいメトリクスをクリックし、期間（分、時間、日、週、月単位）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状況の統計を表示します。**グラフオプション**を使用してグラフをカスタマイズします。

図：CPU使用率のグラフ化されたメトリクス



高可用性セットアップでのSR-IOVの構成

高可用性セットアップでのSR-IOVインターフェースのサポートは、NetScalerリリース12.0 57.19以降で利用可能です。SR-IOVの構成方法の詳細については、「[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](#)」を参照してください。

関連リソース

[AWSでの高可用性の仕組み](#)

異なるAWSアベイラビリティーゾーン間での高可用性

2つのNetScaler VPXインスタンスを、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、独立ネットワーク構成 (INC) モードの高可用性アクティブ/パッシブペアとして構成できます。何らかの理由でプライマリノードが接続を受け入れられなくなった場合、セカンダリ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構成](#) を参照してください。

注意事項

- 展開を開始する前に、以下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、同じアベイラビリティーゾーン内の異なるサブネットに配置すること、2つの異なるAWSアベイラビリティーゾーンに配置することもできます。
- Citrix®は、管理 (NSIP)、クライアントトラフィック (VIP)、およびバックエンドサーバー (SNIP) に異なるサブネットを使用することを推奨します。
- フェールオーバーが機能するには、高可用性を独立ネットワーク構成 (INC) モードで設定する必要があります。
- ハートビートに使用されるため、2つのインスタンスはUDPトラフィック用にポート3003を開いている必要があります。
- REST 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サーバーへのアクセス権を持っている必要があります。
- IAMロールは、パブリックIPまたはElastic IP (EIP) の移行のためのE2権限と、プライベートIPの移行のためのEC2ルートテーブル権限を持っている必要があります。

AWSアベイラビリティゾーン間で高可用性を展開するには、以下の方法があります。

- [Elastic IPアドレスの使用](#)
- [プライベートIPアドレスの使用](#)

追加の参照情報

AWS向けNetScaler Application Delivery Management (ADM) の詳細については、[AWSへのNetScaler ADMエージェントのインストール](#)を参照してください。

異なるAWSゾーン間でエラスティックIPアドレスを持つVPX高可用性ペアをデプロイする

INCモードでエラスティック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ゾーン間でEIPアドレスを使用するHAの仕組み

フェイルオーバーが発生すると、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ぎます。フェイルオーバープロセスでは、AWS APIは次のことを行います。

1. **IPSets** がアタッチ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（仮想サーバーに直接アタッチされているものと、IPセットを介してアタッチされているもの）の中から、関連付けられたパブリックIPを持つIPアドレスを見つけます。
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:

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、EIPを使用する際は、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。高可用性については、デプロイメントに応じて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アドレスを持つVPX高可用性ペアをデプロイする方法

以下は、2つの異なるサブネットまたは2つの異なるAWSアベイラビリティゾーンにVPXペアをデプロイする手順の概要です。

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ゾーン、または同じゾーン内の異なるサブネットにデプロイします。
3. 高可用性を設定します
 - a. 両方のインスタンスでINCモードで高可用性を設定します。
 - b. 両方のインスタンスに(/ja-jp/citrix-adc/13-1/load-balancing/load-balancing-customizing/multi-ip-virtual-servers.html)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ーを追加します。

ステップ1と2ではAWSコンソールを使用します。ステップ3ではNetScaler VPX GUIまたはCLIを使用します。

ステップ1. Amazon Virtual Private Cloud (VPC) を作成します。

ステップ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ゾーン、または同じゾーン内の異なるサブネットにデプロイします。プライマリVPXのVIPにEIPをアタッチします。

AWSでVPCを作成し、VPXインスタンスをデプロイする方法の詳細については、(/ja-jp/vpx/13-1/deploy-aws.html)および(/ja-jp/vpx/13-1/deploy-aws/scenarios.html)を参照してください。

ステップ3. 高可用性を構成します。NetScaler VPX CLIまたはGUIを使用して高可用性を設定できます。

CLIを使用して高可用性を構成する

1. 両方のインスタンスでINCモードで高可用性を設定します。

プライマリノードで:

```
add ha node 1 <sec_ip> -inc ENABLED
```

セカンダリノードで:

```
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次のコマンドを入力します。

```
add ipset <ipsetname>
```

3. 両方のインスタンスでIPセットを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セットをプライマリ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を仮想サーバーに追加し、その逆も同様です。

4. プライマリインスタンスに仮想サーバーを追加します。

次のコマンドを入力します。

```
add <server_type> vservers <vservers_name> <protocol> <primary_vip> <port>  
-ipset \<ipset_name>
```

GUIを使用して高可用性を構成する

1. 両方のインスタンスでINCモードの高可用性を設定します。
2. ユーザー名 `nsroot` とパスワードとしてインスタンスIDを使用して、プライマリノードにログオンします。
3. GUIから、**Configuration > System > High Availability** に移動します。**Add** をクリックします。
4. **Remote Node IP address** フィールドで、セカンダリノードの管理NICのプライベートIPアドレスを追加します。
5. 自己ノードで、**NIC (Independent Network Configuration)** を有効にする モードを選択します。

6. **Remote System Login Credential** の下で、セカンダリノードのユーザー名とパスワードを追加し、**Create** 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. 両方のインスタンスで、IPセットを追加し、IPセットをVIPセットにバインドします。
9. GUIから、**System > Network > IPs > Add** 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) の必要な値を追加し、**Create** をクリックします。
11. **System > Network > IP Sets > Add** に移動します。IPセット名を追加し、**Insert** をクリックします。
12. IPV4sページから、仮想IPを選択し、**Insert** をクリックします。IPセットを作成するには、**Create** をクリックします。
13. プライマリインスタンスで仮想サーバーを追加します

GUIから、**設定 > トラフィック管理 > 仮想サーバー > 追加** に移動します。

仮想サーバーを追加する(/en-us/vpx/media/vserver-aws-hainc.png)

シナリオ

このシナリオでは、単一の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にアタッチされます。

図：この図は、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。

NetScaler VPX高可用性セットアップ（INCモード） (/en-us/vpx/media/aws-hainc.png)

このシナリオでは、CLIを使用して高可用性を構成します。

1. 両方のインスタンスでINCモードの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。

プライマリで：

```
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。

セカンダリで：

```
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. 両方のインスタンスでIPセットを追加し、そのIPセットをVIPにバインドします。

プライマリで：

```
add ipset ipset123
```

```
add ns ip 192.168.7.68 255.255.255.0 -type VIP
```

```
bindipset ipset123 192.168.7.68
```

セカンダリで：

```
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ーを追加します。

次のコマンド：

```
add lbvserver vserver1 http 192.168.2.129 80 -ipset ipset123
```

4. 設定を保存します。



5. 強制フェイルオーバー後、セカンダリが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アドレスを持つVPX高可用性ペアを展開する

INCモードでプライベート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アベイラビリティゾーンに2つのNetScaler VPXインスタンスを設定できます。このソリューションは、既存のマルチゾーン(</ja-jp/vpx/13-1/deploy-aws/vpx-ha-eip-different-aws-zones.html>)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。

高可用性の詳細については、(</ja-jp/citrix-adc/13-1/system/high-availability-introduction.html>)を参照してください。INCの詳細については、(</ja-jp/citrix-adc/13-1/system/high-availability-introduction/configuring-high-availability-nodes-different-subnets.html>)を参照してください。

注記:

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アドレスによる高可用性ペア

前提条件

AWSアカウント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": [
    {
      "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アドレスでVPX HAペアを展開する

以下は、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アベイラビリティゾーンにVPXペアを展開する手順の概要です。

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ゾーンに2つのVPXインスタンスを展開します。
3. 高可用性を構成します。
 - a. 両方のインスタンスでINCモードの高可用性を設定します。
 - b. クライアントインターフェースを指すVPCにそれぞれのルートテーブルを追加します。
 - c. プライマリインスタンスに仮想サーバーを追加します。

ステップ1、2、3bについては、AWSコンソールを使用します。ステップ3aと3cについては、NetScaler VPX GUIまたはCLIを使用します。

ステップ1. Amazon Virtual Private Cloud (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ゾーンに、同じ数のENI (ネットワークインターフェース) を持つ2つのVPXインスタンスをデプロイします。

AWSでVPCを作成し、VPXインスタンスをデプロイ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。

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サブネットが選択され、このサブネットにVIPが作成されています。VPCサブネット (192.168.0.0/16) 以外の任意のサブネットを選択できます。

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ース (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、ルートを追加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、 を選択し、ルートテーブルを選択します。
3. を選択し、 をクリックします。
4. ルートを追加するには、**ルートの追加**を選択します。**送信先**には、送信先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ースの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**送信元/送信先チェック**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ースの送信元/送信先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ース**を選択します。
3. プライマリクライアントインターフェースのネットワークインターフェースを選択し、**アクション**を選択して、**送信元/送信先チェックの変更**をクリックします。
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. 高可用性を構成します。NetScaler VPX CLIまたはGUIを使用して高可用性を設定できます。

CLIを使用して高可用性を構成する

1. 両方のインスタンスでINCモードで高可用性を設定します。

プライマリノードで:

```
add ha node 1 \<sec\_ip\> -inc ENABLED
```

セカンダリノードで:

```
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. プライマリインスタンスに仮想サーバーを追加します。選択したサブネット (たとえば、10.10.10.0/24) 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を使用して高可用性を構成する

1. 両方のインスタンスでINCモードの高可用性を設定します
2. ユーザー名 **nsroot** とパスワードとしてインスタンスIDを使用して、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** フィールドで、セカンダリノードの管理NICのプライベートIPアドレスを追加します。
5. 自己ノードで **NIC (独立ネットワーク構成)** をオンにする モードを選択します。
6. **リモートシステムログイン資格情報** の下で、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ーを追加します

構成 > 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。したがって、VPC所有者アカウントと参加者アカウントが存在します。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で自身のアプリケーションリソースを表示、作成、変更、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アドレスを持つVPX HAペアをデプロイするための構成手順は、[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](#)と同じですが、以下の例外があります。

- クライアントインターフェースを指す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、以下の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により、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ロールを引き受けることができます。

- 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、以下のIAM権限を参加者アカウントに提供していることを確認してください。

```

{

    "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ロールを作成する

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。



4. 管理者アクセス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

次のコマンドにより、NetScaler VPXインスタンスは、VPC所有者アカウントに存在するクロスアカウントIAMロールを引き受けることができます。

```
set cloud awsParam -roleARN <string>
```

NetScaler GUIを使用してクロスアカウントIAMロールを設定する

1. NetScalerアプライアンスにサインインし、**Configuration > AWS > Change cloud parameters** に移動します。



2. **Configure AWS Cloud Parameters** ページで、**RoleARN** 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、単一のVPCが作成されます。そのVPC内に、2つのアベイラビリティゾーンに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モードにおける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ゾーンをまたいで使用できます。



このシナリオでは、CLIを使用して高可用性を構成します。

1. 両方のインスタンスでINCモードの高可用性を設定します。

プライマリノードとセカンダリノードで以下のコマンドを入力します。

プライマリノードで:

```
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。

セカンダリノードで:

```
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。

2. プライマリインスタンスに仮想サーバーを追加します。

以下のコマンドを入力します:

```
add lbvserver vserver1 http 10.10.10.10 80
```

3. 設定を保存します。
4. 強制フェイルオーバー後:
 - 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ソリューション向けのAWS Transit Gateway の設定

内部ネットワーク、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プライベートVIPサブネットをルーティング可能にするには、AWS Transit Gateway が必要です。VPCはAWS Transit Gatewayに接続する必要があります。AWS Transit Gateway ルートテーブル内に、VIPサブネットまたはIPプール用の静的ルートが作成され、VPCを指します。



AWS Transit Gateway を設定するには、次の手順に従います。

1. を開きます(<https://console.aws.amazon.com/vpc/>)。
2. ナビゲーションペインで、**Transit Gateway ルートテーブル**を選択します。
3. ルートタブを選択し、**静的ルートを作成**をクリックします。



4. CIDRがプライベートVIPサブネットを指し、アタッチメントがNetScaler VPXを持つVPCを指す静的ルートを作成します。



5. **静的ルートを作成**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主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が、同じIAM権限セットを持っていること。
- プライマリノードとセカンダリノードの両方でINCモードが有効になっていること。
- プライマリノードとセカンダリノードの両方が、同じ数のインターフェースを持っていること。

- インスタンスを作成する際、両方のノードでインターフェースをアタッチする順序を同じにしてください。プライマリノードで、クライアントインターフェースが最初にアタッチされ、サーバーインターフェースが2番目にアタッチされる場合、セカンダリノードでも同じ順序に従ってください。不一致がある場合は、インターフェースをデタッチして正しい順序で再アタッチしてください。
- トラフィックが流れない場合は、初回にプライマリノードのクライアントインターフェースで「Source/dest. Check」が無効になっていることを確認してください。
- cloudhadaemon コマンド (`ps -aux | grep cloudha`) が Shell で実行されていることを確認してください。
- 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、`/var/log/cloud-ha-daemon.log` にあるログファイルを確認してください。

AWS Outposts に NetScaler VPX インスタンスを展開する

AWS Outposts は、お客様のサイトにデプロイされる AWS 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 インフラストラクチャとサービスを提供します。AWS は、この容量を AWS リージョンの一部として運用、監視、管理します。オンプレミスと AWS クラウド全体で同じ 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、一貫性のあるハイブリッドエクスペリエンスを実現できます。

Outposts 上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際にそれらを指定できます。Outposts サブネット内のインスタンスは、同じ Amazon Virtual Private Cloud (VPC) 内で、プライベート IP アドレスを使用して AWS リージョン内の他のインスタンスと通信します。

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 Outposts の仕組み

AWS Outposts は、Outposts と AWS リージョン間の常時かつ一貫した接続で動作するように設計されています。リージョンへのこの接続と、オンプレミス環境のローカルワークロードへ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。インターネットはまた、オンプレミスのワークロードまたは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 アクセスも提供する必要があります。

前提条件

- お客様のサイトに 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、以下の 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 AWS Outposts に NetScaler VPX インスタンスを展開する

次の図は、Outposts 上での NetScaler VPX インスタンスのシンプルな展開を示しています。AWS Marketplace にある NetScaler AMI も Outposts に展開されます。



AWS ウェブコンソールにログインし、以下の手順を完了して、AWS Outposts に NetScaler VPX EC2 インスタンスを展開します。

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 ダッシュボードから、**Compute > EC2 > Launch Instance > AWS Marketplace** に移動します。
8. ネットワークインターフェイスを追加作成してアタッチします。
9. 管理 NIC に Elastic IP をアタッチします。
10. VPX インスタンスに接続します。

各手順の詳細については、「[AWS Web コンソールを使用して AWS に NetScaler VPX インスタンスを展開する](#)」を参照してください。

同じアベイラビリティーゾーン展開での高可用性については、「[AWS に高可用性ペアを展開する](#)」を参照してください。

AWS Outposts を使用したハイブリッドクラウドへの NetScaler VPX インスタンスの展開

AWS Outposts を含む AWS 環境のハイブリッドクラウドに NetScaler VPX インスタンスを展開できます。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の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。これらの柔軟な展開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 または AWS Outposts のどちらにネイティブに展開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ポートフォリオ全体を保護します。

次の図は、AWS を使用したハイブリッドクラウドでの 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、Metrics Exchange Protocol (MEP) を介して、リソースの可用性と健全性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でトラフィックを負荷分散し、GSLB構成で指定された定義済みの方法（ラウンドロビン、最小接続、静的近接）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展開は、次の目的で使用できます。

- すべてのノードをアクティブにして、リソースの使用率を最適化します。
- 個々のユーザーに最も近いサイトに要求を誘導することで、ユーザーエクスペリエンスを向上させます。
- ユーザー定義の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展開は、次の目的で使用できます。

- 災害復旧
- クラウドバースト

参照

- [AWSにNetScaler 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AWS OutpostsにNetScaler VPXインスタンスを展開する
- NetScaler VPXインスタンスでGSLBを構成する

NetScaler Web App Firewall を使用した AWS API Gateway の保護

AWS API Gateway の前に NetScaler アプライアンスを展開し、外部の脅威から API Gateway を保護できます。NetScaler Web App Firewall (WAF) は、OWASP Top 10 の脅威やゼロデイ攻撃から API を防御できます。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、あらゆる環境でセキュリティポリシーを一貫して適用および実施できます。NetScaler Web App Firewall は展開が簡単で、単一ライセンスとして利用できます。NetScaler Web App Firewall は、以下の機能を提供します。

- 簡素化された構成
- ボット管理
- 包括的な可視性
- 複数のソースからデータを照合し、統合された画面にデータを表示します

API Gateway の保護に加えて、他の NetScaler 機能も使用できます。詳細については、[NetScaler 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ために、ADC をアベイラビリティゾーン内またはアベイラビリティゾーン間で高可用性構成に配置できます。また、Autoscale 機能を使用してクラスタリングを使用または構成することもできます。

以前は、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 Application Firewall (WAF) の保護がないと、API はセキュリティ上の脅威にさらされやすくなりました。

AWS API Gateway の前に NetScaler アプライアンスを展開する

次の例では、NetScaler アプライアンスが AWS API Gateway の前に展開されています。



AWS Lambda サービスに対する正規の API リクエストがあると仮定します。このリクエストは、[Amazon API Gateway ドキュメント](#) に記載されている API サービスのいずれかに対するものである可能性があります。前の図に示すように、トラフィックフローは次のとおりです。

1. クライアントは AWS Lambda 関数 (XYZ) にリクエストを送信します。このクライアントリクエストは NetScaler 仮想サーバー (192.168.1.1) に送信されます。

2. 仮想サーバーはパケットを検査し、悪意のあるコンテンツがないか確認します。
3. NetScaler アプライアンスは、クライアントリクエストのホスト名と URL を変更するためにリライトポリシーをトリガーします。たとえば、`https://restapi.citrix.com/default/LambdaFunctionXYZ` を `https://citrix.execute-api.<region>.amazonaws.com/default/LambdaFunctionXYZ` に変更したいとします。
4. NetScalerアプライアンスはこのリクエストをAWS API Gatewayに転送します。
5. AWS API GatewayはさらにリクエストをLambdaサービスに送信し、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リクエストを送信した場合、その悪意のあるリクエストはNetScalerアプライアンスに到達します。
7. NetScalerアプライアンスはパケットを検査し、設定されたアクションに基づいてパケットを破棄します。

WAFが有効なNetScalerアプライアンスを構成する

NetScaler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ングまたはロードバランシング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. ドメイン名を変更するためにリライトポリシーを有効にします。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、バックエンドのAWS API Gatewayの「citrix.execute-api. <region> .amazonaws」ドメイン名に書き換えたいとします。
4. NetScaler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NetScalerアプライアンスに「restapi.citrix.com」ドメイン名を「citrix.execute-api. <region> .amazonaws.com」に、URLを「default/lambda/XYZ」に置き換えさせたいとします。

次の手順では、リライト機能を使用してクライアントリクエストのホスト名と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アプライアンスにログインします。
2. リライトアクションを追加します。

```
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. リライトアクションのリライト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. リライトポリシーを仮想サーバーにバインドします。

```
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
```

詳細については、「NetScalerアプライアンスでクライアント要求のホスト名とURLを変更するよう
にリライトを構成する」([/ja-jp/citrix-adc/13-1/appexpert/rewrite/rewrite-action-policy-examples/
example-12-rewrite-hostname-url-client.html](https://www.citrix.com/help/13.1/appexpert/rewrite/rewrite-action-policy-examples/example-12-rewrite-hostname-url-client.html))を参照してください。

NetScalerの機能と性能

NetScaler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アプライアンスは、次の主要な機能を提供します。

- **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、NetScalerアプライアンスを使用して複数のAPIゲートウェイの負荷を分散し、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法が利用可能です。たとえば、最小接続方式はAPI Gatewayの制限の過負荷を回避し、カスタム負荷方式は特定のAPIゲートウェイで特定の負荷を維持します。詳細については、「負荷分散アルゴリズム」([/ja-jp/citrix-adc/13-1/load-balancing/load-balancing-customizing-algorithms.html](https://www.citrix.com/help/netScaler/13-1/load-balancing/load-balancing-customizing-algorithms.html))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構成されます。
 - クライアントIPアドレスを保持するために、送信元IP (USIP) モードが有効になっています。
 - ユーザー定義のSSL設定: 独自の署名付き証明書とアルゴリズムを使用して、独自のSSL仮想サーバーを持つことができます。
 - バックアップ仮想サーバー: APIゲートウェイに到達できない場合、さらなるアクションのために要求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が利用可能です。詳細については、「NetScalerアプライアンスでのトラフィックの負荷分散」([/ja-jp/citrix-adc/13-1/getting-started-with-citrix-adc/load-balancing.html](https://www.citrix.com/help/netScaler/13-1/getting-started-with-citrix-adc/load-balancing.html))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リクエストを承認および監査できます。
- **レスポnder:** シャットダウン中にAPIリクエストを他のAPIゲートウェイにリダイレクトできます。
- **レート制限:** APIゲートウェイの過負荷を回避するために、レート制限機能を構成できます。
- **可用性の向上:** AWS APIトラフィックの可用性を向上させるために、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できます。
- **REST API:** REST API をサポートしており、クラウド本番環境での作業を自動化するために使用できます。
- **データの監視:** 参照用にデータを監視およびログに記録します。

NetScaler アプライアンスは、さらに多くの機能を提供し、これらはAWS API Gateway と統合できます。詳細については、[NetScaler ドキュメント](#) を参照してください。

バックエンドAWSオートスケーリングサービスを追加

クラウドでのアプリケーションの効率的なホスティングには、アプリケーションの需要に応じてリソースを簡単かつ費用対効果の高い方法で管理することが含まれます。需要の増加に対応するためには、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。需要が減少した場合は、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。アプリケーションの実行コストを最小限に抑えるには、特定の時間に必要とされる数のインスタンスのみをデプロイすることで、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面倒です。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と、必要に応じたりソースのスケールアップおよびスケールダウンの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たNetScaler VPXインスタンスは、以下の利点を提供します。

- **負荷分散と管理:** 需要に応じてサーバーを自動的にスケールアップおよびスケールダウンするように構成します。VPXインスタンスは、バックエンドサブネット内のAutoscale™グループを自動的に検出し、ユーザーが負荷を分散するAutoscaleグループを選択できるようにします。これらすべては、VPXインスタンス上の仮想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ゾーンにまたがるAutoscaleグループを検出し、サーバーの負荷分散を行います。
- **より優れたネットワーク可用性:** VPXインスタンスは以下をサポートします。
 - VPCピアリングを使用することで、異なるVPC上のバックエンドサーバー
 - 同じプレースメントグループ上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上のバックエンドサーバー
- **グレースフル接続終了:** グレースフルタイムアウト機能を使用することで、スケールダウンアクティビティが発生した際にクライアント接続の損失を回避し、Autoscaleサーバーをグレースフルに削除します。

 **図:** NetScaler VPXインスタンスを使用したAWS Autoscalingサービス



この図は、AWS Autoscaling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。詳細については、以下のAWS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](#)
- [シンプル・ノーティフィケーション・サービス \(SNS\)](#)
- [シンプル・キュー・サービス \(Amazon SQS\)](#)

開始する前に

NetScaler VPXインスタンスでAutoscaling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。

1. 次のトピックをお読みください。

- 「前提条件」 ([/ja-jp/vpx/13-1/deploy-aws/prerequisites.html](#))
- 「制限事項と使用ガイドライン」 ([/ja-jp/vpx/13-1/deploy-aws/vpx-aws-limitations-usage-guidelines.html](#))

2. 要件に応じて、AWS上に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展開する」 ([/ja-jp/vpx/13-1/deploy-aws/launch-vpx-for-aws-ami.html](#))および「シナリオ: スタンドアロンインスタンス」 ([/ja-jp/vpx/13-1/deploy-aws/scenarios.html](#))を参照してください。
- HAモードでVPXインスタンスを展開する方法の詳細については、「AWSに高可用性ペアを展開する」 ([/ja-jp/vpx/13-1/deploy-aws/vpx-aws-ha.html](#))を参照してください。

注:

Citrixは、AWS上にNetScaler VPXインスタンスを作成するためにCloudFormationテンプレートを推奨しています。

Citrix®は、管理用 (NSIP) に1つ、クライアント向けLB仮想サーバー (VIP) に1つ、サブネットIP (NSIP) に1つの、3つのインターフェイスを作成することを推奨しています。

3. AWS Autoscaleグループを作成します。既存のAutoscaling構成がない場合は、次の操作を行う必要があります。

- a) 起動構成を作成する
- b) オートスケーリンググループを作成する
- c) オートスケーリンググループを確認する

詳細については、<http://docs.aws.amazon.com/autoscaling/latest/userguide/GettingStartedTutorial.html>を参照してください。

4. AWS Autoscaleグループでは、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す。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。シンプル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 Autoscaling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AutoscalingサービスをVPXインスタンスに追加できます。

AutoscalingサービスをVPXインスタンスに追加するには、次の手順を実行します。

1. **nsroot** の資格情報を使用して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AWS Autoscalingグループを選択し、**Create**をクリックしてクラウドプロファイルを作成します。後でクラウドプロファイルを作成する場合は、**Skip**をクリックします。

クラウドプロファイルの作成時に留意すべき点: デフォルトでは、CloudFormationテンプレートは以下のIAMロールを作成し、アタッチします。

```
{

  "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"

}

\]
```

}

インスタンスのIAMロールに適切な権限があることを確認してください。

- 仮想サーバーのIPアドレスは、VPXインスタンスで利用可能な空きIPアドレスから自動入力されます。 <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/MultipleIP.html#ManageMultipleIP>
- オートスケールグループは、AWSアカウントで設定されたオートスケールグループから事前入力されます。 <http://docs.aws.amazon.com/autoscaling/latest/userguide/AutoScalingGroup.html>。
- 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、およびサービスグループに正しいモニターをバインドしていることを確認してください。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オートスケーリングの場合、クラウドプロファイルを作成した後、証明書がないためロードバランス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ーまたはサービスグループに手動でバインドできます。
- オートスケールサーバーを正常に削除するには、「Graceful Timeout」オプションを選択します。このオプションが選択されていない場合、負荷が低下するとすぐにオートスケールグループのサーバーが削除され、既存の接続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。「Graceful」を選択してタイムアウトを設定すると、スケールダウン時にVPXインスタンスはサーバーをすぐに削除せず、いずれかのサーバーを正常な削除のためにマークします。この期間中、インスタンスはこのサーバーへの新しい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、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。

図: デフォルトのクラウドプロファイルページ



3. 初回ログオン後、クラウドプロファイルを作成する場合は、GUIで **System > AWS > Cloud Profile** に移動し、**Add** をクリックします。



Create Cloud Profile 設定ページが表示されます。



クラウドプロファイルは、NetScalerロードバランシング仮想サーバーと、メンバーとしてオートスケーリンググループの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。

注:

NetScalerリリース13.1-42.x以降、AWSで同じオートスケーリンググループ（ASG）を使用して、異なるサービス（異なるポートを使用）に対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で同じオート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。

**注:**

AWSコンソールでオートスケール関連情報を表示するには、**EC2 > Dashboard > Auto Scaling > Auto Scaling Group** に移動します。

AWSにNetScaler GSLBを展開する

AWSでNetScalerのGSLBをセットアップすることは、基本的に、NetScalerが属するVPCの外部にあるサーバー（別の可用性リージョン内の別のVPCやオンプレミスデータセンター内など）にトラフィックをロードバランシングするようにNetScalerを設定することから構成されます。



DBSの概要

クラウドロードバランサー向けのDBS（ドメインベースサービス）を使用したNetScaler GSLBサポートは、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的に検出することを可能にします。この構成により、NetScalerはActive-Active環境でグローバルサーバーロードバランシングドメイン名ベースサービス（GSLB DBS）を実装できます。DBSは、DNS検出からAWS環境のバックエンドリソースのスケールリングを可能にします。

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。ドキュメントの最終セクションでは、AWSリージョンに固有の2つの異なるアベイラビリティゾーン（AZ）にまたがるNetScaler ADCのHAペアをセットアップする機能について詳しく説明します。

ELBを使用したドメイン名ベースサービス (DBS)

GSLB DBSは、ユーザーの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めるように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構成できます。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を手動で更新することなく、AWS内の異なるサーバーに動的にルーティングします。GSLBサービスグループ向けのNetScaler DBS機能は、DNS対応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。

NetScaler GSLB DBS AutoScaleコンポーネントとクラウドロードバランサー:



AWSコンポーネントの構成

セキュリティグループ

注:

ELB、NetScaler GSLBインスタンス、およびLinuxインスタンスには、それぞれ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。これらの各エンティティに必要なルールセットが異なるためです。この例では、簡潔にするために統合されたセキュリティグループ構成を使用しています。

仮想ファイアウォールの適切な構成を確実にするには、[Security Groups for Your VPC](#)を参照してください。

1. ユーザーのAWSリソースグループにログインし、**EC2 > NETWORK & SECURITY > Security Groups**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インバウンドポートルールを追加します。

注:

きめ細かな強化のためには、ソースIPアクセスを制限することをお勧めします。詳細については、[Webサーバーのルール](#)を参照してください。

4. Amazon LinuxバックエンドWebサービス

- a. ユーザーの**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. **インスタンスの起動**をクリックし、以下の詳細を使用して**Amazon Linux**インスタンスを構成します。

このインスタンスでWebサーバーまたはバックエンドサービスをセットアップするための詳細を入力します。

5. NetScalerの設定

- a. ユーザーの**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. **インスタンスの起動**をクリックし、以下の詳細を使用して**Amazon AMI**インスタンスを構成します。

6. Elastic IPの設定

注:

コスト削減のため、NSIPにパブリックIPを持たせないことで、必要に応じてNetScalerを単一のElastic IPで実行することもできます。代わりに、SNIPにElastic IPをアタッチすることで、GSLBサイトIPおよびADNS IPに加えて、ボックスへの管理アクセスをカバーできます。

- a. ユーザーの **AWS リソースグループ** にログインし、**EC2 > ネットワーク & セキュリティ > Elastic IP** に移動します。
- b. **新しいアドレスを割り当てる** をクリックして、Elastic IP アドレスを作成します。
- c. Elastic IP を設定して、AWS 内で実行中のユーザーの NetScaler インスタンスを指すようにします。
- d. 2つ目の Elastic IP を設定し、再度実行中のユーザーの NetScaler インスタンスを指すようにします。

7. エラスティックロードバランサー

- a. ユーザーの **AWS リソースグループ** にログインし、**EC2 > ロードバランシング > ロードバランサー** に移動します。



- a. **ロードバランサーの作成** 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。

ユーザーの Elastic Load Balancer を使用すると、バックエンドの Amazon Linux インスタンスをロードバランシングできるだけでなく、オンデマンドで起動される他のインスタンスもロードバランシングできます。

グローバルサーバーロードバランシングのドメイン名ベースサービスを設定する

トラフィック管理の設定については、[NetScaler GSLB ドメインベースサービスの設定](#) を参照してください。

展開タイプ

3-NIC 展開

- 一般的な展開
 - GSLB StyleBook
 - ADMを使用
 - GSLBを使用 (Route53、ドメイン登録あり)
 - ライセンス - プール型/マーケットプレイス
- ユースケース
 - 3-NIC展開は、データトラフィックと管理トラフィックの真の分離を実現するために使用されます。
 - 3-NIC展開は、ADCのスケールとパフォーマンスも向上させます。
 - 3-NIC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-NIC展開が推奨されます。

CFT展開

顧客は、展開をカスタマイズする場合、または展開を自動化する場合に、CloudFormationテンプレートを使用して展開します。

展開手順

展開手順は次のとおりです。

1. GSLB用3-NIC展開
2. ライセンス
3. 展開オプション

GSLB用3-NIC展開

NetScaler VPXインスタンスは、AWSマーケットプレイスでAmazon Machine Image (AMI)として利用可能であり、AWS VPC内でElastic Compute Cloud (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として許可される最小の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2GBのメモリが必要です。AWS VPC内で起動されたEC2インスタンスは、VPX構成に必要な複数のインターフェース、インターフェースごとの複数のIPアドレス、およびパブリックIPアドレスとプライベート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
- クライアント向けサブネット (VIP)
- バックエンド向けサブネット (SNIP)

NetScalerは、AWSへの標準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ースを推奨しています。

AWSは現在、マルチIP機能をAWS VPC内で実行されているインスタンスにのみ提供しています。VPC内のVPXインスタンスは、EC2インスタンスで実行されているサーバーのロードバランシングに使用できます。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を含む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リージョンあたり最大5つのVPCインスタンスを作成できます。ユーザーは、Amazonの申請フォーム「Amazon VPC Request」を提出することで、より高いVPC制限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インスタンスには、以下のライセンスオプションが利用可能です。

- 無料 (無制限)
- 時間単位
- 年間

持ち込みライセンス

無料トライアル (AWSマーケットプレイスで21日間無料のすべてのNetScaler VPX-AWSサブスクリプション提供)。

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。詳細については、「[AWS上に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。

ハイブリッドおよびマルチクラウド展開のためのNetScalerグローバル負荷分散

NetScalerハイブリッドおよびマルチクラウド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、ユーザーはハイブリッドクラウド、マルチクラウド、およびオンプレミス展開の複数のデータセンターにわたってアプリケーショントラフィックを分散できます。NetScalerハイブリッドおよびマルチクラウドGSLBソリューションは、既存のセットアップ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セットアップを管理するのに役立ちます。また、オンプレミス環境のユーザーは、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のみをクラウドにルーティングし、ほとんどのトラフィックをオンプレミスで処理できます。NetScalerハイブリッドおよびマルチクラウドGSLBソリューションは、単一の統合コンソールから地理的な場所をまたいでNetScalerインスタンスを管理および監視することも可能にします。

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、ユーザーパートナーや顧客のニーズを満たすために異なるインフラストラクチャを使用することで、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。複数のクラウドアーキテクチャを使用すると、ユーザーは使用した分だけ支払うため、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ため、アプリケーションをより適切にスケールリングできます。また、各プロバイダーの最高のサービスを利用するために、あるクラウドから別のクラウドへ迅速に切り替える機能も提供します。

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

NetScaler GSLBノードはDNS名前解決を処理します。これらのGSLBノードのいずれも、任意のクライアントロケーションからDNS要求を受信できます。DNS要求を受信したGSLBノードは、構成された負荷分散方法によって選択された負荷分散仮想サーバーIPアドレスを返します。メトリック（サイト、ネットワーク、および永続性メトリック）は、NetScaler独自の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ノードで構成されたモニターは、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、親子トポロジでは、GSLBノードと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、以下の前提条件を設けています。

- ユーザーが既存の負荷分散セットアップを持っている場合、それは稼働中であること。
- 各NetScaler GSLBノードにSNIPアドレスまたはGSLBサイトIPアドレスが構成されていること。このIPアドレスは、他のデータセンターとメトリックを交換する際のデータセンターソースIPアドレスとして使用されます。
- 各NetScaler GSLBインスタンスに、DNSトラフィックを受信するためのADNSまたはADNS-TCPサービスが構成されていること。
- 必要なファイアウォールとセキュリティグループがクラウドサービスプロバイダーで構成されていること。

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設定を行う必要があります。AWSの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、MEPトラフィック交換のために、ADNSサービス/DNSサーバーIPアドレス用にポート53を、GSLBサイトIPアドレス用にポート3009を開放する必要があります。ロードバランシングノードでは、アプリケーションのトラフィックを受信するために適切なポートを開放する必要があります。例えば、HTTPトラフィックを受信するためにポート80を、HTTPSトラフィックを受信するためにポート443を開放する必要があります。NetScalerエージェントとNetScaler ADM間のNITRO通信のためにポート443を開放してください。

動的ラウンドトリップタイムGSLBメソッドの場合、設定された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ために、ユーザーはポート53を開放する必要があります。UDPまたはTCPプローブはSNIPのいずれかを使用して開始されるため、この設定はサーバー側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

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をこのセクションで説明します。

他のロードバランシングソリューションとの互換性

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、さまざまなロードバランシングソリューションをサポートしています。

注:

NetScaler以外のロードバランシングソリューションは、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、かつ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メソッド

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。
 - 最小接続数: 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。
 - 最小パケット数: クライアント要求は、過去14秒間に受信したパケット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。
- メトリックに基づかないGSLBメソッド
 - ラウンドロビン: クライアント要求は、ロードバランサーのリストの最上位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。そのロードバランサーは、リストの最下位に移動します。
 - 送信元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。

- 近接性ベースのGSLBメソッド

- 静的近接性: クライアント要求は、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。
- ラウンドトリップタイム (RTT): このメソッドは、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時間遅延）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。

ロードバランシングメソッドの詳細については、[ロードバランシングアルゴリズム](#)を参照してください。

GSLBトポロジ

NetScalerハイブリッドおよびマルチクラウドGSLBソリューションは、アクティブ/パッシブトポロジと親子トポロジをサポートしています。

- アクティブ/パッシブトポロジ - 障害点から保護することで、ディザスタリカバリを提供し、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した場合、パッシブデータセンターが稼働状態になります。GSLBアクティブ/パッシブトポロジの詳細については、[ディザスタリカバリのためのGSLBの構成](#)を参照してください。
- 親子トポロジ - お客様がメトリックベースのGSLBメソッドを使用してGSLBおよびLBノードを構成し、LB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、親サイトと子サイト間のメトリック交換はメトリック交換プロトコル (MEP) を介して行われるため、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のサポート

NetScalerハイブリッドおよびマルチクラウドGSLBソリューションもIPv6をサポートしています。

監視

NetScalerハイブリッドおよびマルチクラウドGSLBソリューションは、セキュア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とGSLBの設定が同じNetScalerインスタンス上にある場合、または親子トポロジが使用されている場合は、モニターの設定はオプションです。

パーシステンス

NetScalerハイブリッドおよびマルチクラウドGSLBソリューションは、以下をサポートしています。

- 送信元IPアドレスベースのパーシステンスセッション。これにより、同じクライアントからの複数のリクエストが、設定されたタイムアウト期間内に到着した場合、同じサービスに転送されます。クライアントが別のリクエストを送信する前にタイムアウト値が期限切れになった場合、セッションは破棄され、クライアントの次のリクエストに対して新しいサーバーを選択するために、設定された負荷分散アルゴリズムが使用されます。
- スピルオーバーパーシステンス。これにより、プライマリの負荷がしきい値を下回った後でも、バックアップ仮想サーバーは受信したリクエストの処理を継続します。詳細については、「[スピルオーバーの設定](#)」を参照してください。
- サイトパーシステンス。これにより、GSLBノードは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、以降のすべてのDNSリクエストに対して、選択されたデータセンターのIPアドレスを転送します。設定されたパーシステンスがDOWN状態の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以降のリクエストに対してパーシステントになります。

NetScaler ADM StyleBookを使用した構成

お客様は、NetScaler ADMのデフォルトのマルチクラウドGSLB StyleBook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、LBノードStyleBook用のデフォルトのマルチクラウドGSLB StyleBook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の子サイトであるNetScaler負荷分散ノードを構成できます。このStyleBookは、ユーザーが親子トポロジでLBノードを構成したい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を使用して個別に構成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で提供されているマルチクラウドGSLB StyleBook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。ワークフロー図のステップは、図の後に詳細に説明されています。



クラウド管理者として、次のタスクを実行します。

1. NetScaler Cloudアカウントにサインアップします。

NetScaler ADMの使用を開始するには、NetScaler Cloud会社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にログインした後、**NetScaler Application Delivery Management** タイルで**Manage**をクリックして、ADMサービスを初めてセットアップします。
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。

ユーザーは、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有効にするために、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。管理対象インスタンスでLBおよびGSLB構成を設定できるように、各リージョンにエージェントをインストールします。LBおよびGSLB構成は単一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、[Getting Started](#)を参照してください。

4. Microsoft AWS クラウドまたはオンプレミスデータセンターにロードバランサーを展開します。

ユーザーがクラウドおよびオンプレミスに展開するロードバランサーの種類に応じて、それらを適切にプロビジョニングします。たとえば、ユーザーはAmazon Web Services (AWS) 仮想プライベートクラウドおよびオンプレミスデータセンターに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、その他のリソースを構成することにより、NetScalerインスタンスを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インスタンスの展開方法の詳細については、以下のドキュメントを参照してください。

- [NetScaler VPX on AWS](#)。
- [NetScaler VPXスタンドアロンインスタンスの構成](#)。

5. セキュリティ構成を実行します。

ARMおよび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を構成し、ユーザーインスタンスとサブネットのインバウンドおよび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、監視したい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LBにNetScaler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で

NetScalerインスタンスを追加する方法の詳細については、[Getting Started](#)を参照してください。

7. デフォルトのNetScaler ADM StyleBookを使用して、GSLBおよびLB構成を実装します。

- Multi-cloud 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。
- ロードバランシング構成を実装します。(管理対象インスタンスにすでにLB構成がある場合、ユーザー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ロードバランシングのためにインスタンスを手動で構成します。インスタンスを手動で構成する方法の詳細については、「[基本的なロードバランシングのセットアップ](#)」を参照してください。
- StyleBookを使用します。ユーザーは、NetScaler ADM StyleBook (HTTP/SSLロードバランシングStyleBookまたはHTTP/SSLロードバランシング (モニター付き) StyleBook) 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の詳細については、「[StyleBook](#)」を参照してください。

8. 次のいずれかのケースでGSLB親子トポロジを構成するには、LBノード用のMulti-cloud GSLB StyleBook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LBノードを構成し、LB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。

NetScaler LBノードでGSLBを構成するためのStyleBookの使用

お客様は、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LBノードを構成し、LBノードが別のNetScalerインスタンスに展開されている場合に、**LBノード用のMulti-cloud GSLB StyleBook**を使用できます。

ユーザーは、このStyleBookを使用して、既存の親サイトに子サイトを追加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、子サイトの数だけ、このStyleBookから構成 (構成パック) を作成します。StyleBookは、子サイトにGSLB構成を適用します。ユーザーは最大1024個の子サイトを構成できます。

注:

親サイトを構成するには、Multi-cloud GSLB StyleBookを使用します。

このStyleBookは次の仮定に基づいています。

- SNIPアドレスまたはGSLBサイトIPアドレスが構成されていること。
- 必要なファイアウォールとセキュリティグループがクラウドサービスプロバイダーで構成されていること。

LBノード用のマルチクラウドGSLB StyleBookを使用して親子トポロジで子サイトを構成する

1. **Applications > Configuration > Create New** に移動します。
2. **Applications > Configuration** に移動し、**Create New** をクリックします。

StyleBook は、ユーザーがこの StyleBook で定義されているすべてのパラメーターの値を入力できるユーザーインターフェイスページとして表示されます。

注記:

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。

3. 次のパラメーターを設定します。

- **Application Name**。子サイトを作成する GSLB サイトに展開されている GSLB アプリケーションの名前を入力します。
- **Protocol**。ドロップダウンリストボックスから、展開されている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 Health Check** (オプション)
- **Health Check Type**。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 アドレスの健全性をチェックするために使用されるプローブのタイプを選択します。
- **Secure Mode**。(オプション) SSL ベースのヘルスチェックが必要な場合は、このパラメーターを有効にするために **Yes** を選択します。

- **HTTP Request。** (オプション) ユーザーが HTTP をヘルスチェックタイプとして選択した場合、VIP アドレスをプローブするために使用される完全な HTTP リクエストを入力します。
- **List of HTTP Status Response Codes。** (オプション) ユーザーが HTTP をヘルスチェックタイプとして選択した場合、VIP が正常なときに HTTP リクエストへの応答で预期される HTTP ステータスコードのリストを入力します。

4. 親サイトの構成。

- 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。** 親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際にソース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ノードにすでに構成されているものとします。
 - **サイトのパブリックIPアドレス。** (オプション) サイトのIPアドレスがNATされている場合、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。

5. 子サイトの構成。

- 子サイト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。** 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) サイトのIPアドレスがNATされている場合、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。

6. アクティブなGSLBサービスの構成 (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、アプリケーションが展開されているサイトでローカルGSLBサービスのリストを構成できます。
 - **サービスIP。** このサイトのロードバランシング仮想サーバーの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、それにNATされたパブリックIPアドレスがある場合は、そのパブリック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サービスが配置されているサイトの名前を入力します。

7. **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でGSLBインスタンスとして構成されているNetScalerインスタンスを選択します。
8. 「**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、この構成を検査、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 Machine Images (AMI) として利用できます。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ユーザーは規約に同意し、AWS Marketplace製品を購読する必要があります。MarketplaceのNetScaler VPXの各エディションで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。これらのテンプレートは、NetScaler VPXの推奨される展開アーキテクチャをコード化したり、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、特定の生産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、または強化できます。ほとんどのテンプレートでは、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（例：リリース12.0-56.20）およびエディション（例：NetScaler VPX Platinum Edition - 10 Mbps）またはNetScaler BYOLに固有のAMI IDが含まれています。CloudFormationテンプレートで異なるバージョンの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、こちらにあります: [NetScaler AWS CloudFormation Master](#)。

CFT 3 NIC展開

このテンプレートは、2つのアベイラビリティゾーン用に3つのサブネット（管理、クライアント、サーバー）を持つVPCを展開します。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展開します。このテンプレートは、2つのNetScalerインスタンスでアベイラビリティゾーンをまたぐHAペアも作成します。プライマリには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、セカンダリには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が作成されます。このCFTによって作成されるすべてのリソース名は、スタック名のtagNameでプレフィックスが付けられます。

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に関連付けられたプライマリ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に関連付けられたプライマリVPXインスタンスのプライベートIPアドレス
- PrimaryCitrixADCSNIP - SNIPに関連付けられたプライマリ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)
- SecondaryCitrixADCManagementURL2 - 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に関連付けられたセカンダリVPXインスタンスのプライベートIPアドレス
- SecondaryCitrixADCSNIP - SNIPに関連付けられたセカンダリ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に入力する際、CFT内の任意のパラメータに対する ***** は、それが必須フィールドであることを意味します。例えば、**VPC ID*** は必須フィールドです。

以下の前提条件を満たす必要があります。CloudFormationテンプレートは、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を必要とします。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、AWS Marketplace製品の利用規約に同意し、サブスクライブする必要があります。

以下も存在する必要があります。

- キーペア
- 3つの未割り当てEIP
- プライマリ管理
- クライアントVIP
- セカンダリ管理

AWSでNetScaler VPXインスタンスをプロビジョニングする方法の詳細については、「AWSでのNetScaler VPXインスタンスのプロビジョニング」([/ja-jp/netScaler-console-service/hybrid-multi-cloud-deployments/provisioning-vpx-aws](#))を参照してください。

スタイルブックを使用してGSLBを構成する方法については、「StyleBookを使用したGSLBの構成」([/ja-jp/netScaler-console-service/hybrid-multi-cloud-deployments/hybrid-and-multi-cloud-solution/configure-glb-using-stylebooks](#))を参照してください。

前提条件

AWSでVPXインスタンスを作成する前に、ユーザーは以下のものがあることを確認する必要があります。

- Amazon Web Services (AWS) Virtual Private Cloud (VPC) でNetScaler VPX AMIを起動するためのAWSアカウント。ユーザーは[Amazon](#)で無料でAWSアカウントを作成できます。
- ユーザーがAWSサービスとリソースへのアクセスを安全に制御するためのAWS Identity and Access Management (IAM) ユーザーアカウント。IAMユーザーアカウントの作成方法の詳細については、トピック「[IAMユーザーの作成 \(コンソール\)](#)」を参照してください。

スタンドアロンおよび高可用性展開の両方で、IAMロールは必須です。IAMロールには、以下の権限が必要です。

- ec2:DescribeInstances
- ec2:DescribeNetworkInterfaces
- ec2:DetachNetworkInterface
- ec2:AttachNetworkInterface

- ec2:StartInstances
- ec2:StopInstances
- ec2:RebootInstances
- ec2:DescribeAddresses
- ec2:AssociateAddress
- ec2:DisassociateAddress
- オートスケーリング:*
- SNS:*
- SQS:*
- iam:SimulatePrincipalPolicy
- iam:GetRole

NetScaler CloudFormation テンプレートを使用する場合、IAM ロールは自動的に作成されます。このテンプレートでは、すでに作成されている IAM ロールを選択することはできません。

注:

ユーザーが GUI を介して 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構成するためのプロンプトが表示されます。権限がすでに構成されている場合は、このプロンプトを無視してください。

- 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には、AWS CLI が必要です。詳細については、「[AWS Command Line Interface とは](#)」を参照してください。ユーザーは、ネットワークインターフェースタイプを SR-IOV に変更するためにも AWS CLI を必要とします。

GSLB の前提条件

NetScaler GSLBサービスグループの前提条件には、機能するAWS環境と、セキュリティグループ、Linuxウェブサーバー、AWS内のNetScaler ADC、Elastic IP、およびElastic Load Balancerを設定するための知識と能力が含まれます。

GSLB DBSサービス統合には、AWS ELBロードバランサーインスタンス向けにNetScalerバージョン12.0.57が必要です。

現在サポートされているVPXモデル、AWSリージョン、インスタンスタイプ、およびサービスに関する最新情報については、[VPX-AWSサポートマトリックス](#)を参照してください。

その他のリソース

[ハイブリッドおよびマルチクラウド展開向けのNetScaler ADM GSLB。](#)

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

注:

高可用性セットアップにおける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以降で利用可能です。

AWS上にNetScaler VPXインスタンスを作成した後、AWS CLIを使用して、仮想アプライアンスを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できます。

NetScaler VPX AWS Marketplace Editionsの3Gおよび5Gを除くすべてのNetScaler 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は有効になっていません。

構成を開始する前に、以下のトピックをお読みください。

- [前提条件](#)
- [制限事項と使用ガイドライン](#)

このセクションには、以下のトピックが含まれています。

- インターフェイスタイプをSR-IOVに変更する
- 高可用性セットアップでSR-IOVを構成する

インターフェイスタイプをSR-IOVに変更する

show interface summary コマンドを実行して、ネットワークインターフェイスのデフォルト構成を確認できます。

例1: 次のCLIスクリーンショットは、NetScaler VPX AWS Marketplace Editions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。



例2: 以下のCLIスクリーンショットは、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。



インターフェースタイプを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>](http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/sriov-networking.html) を参照してください。

インターフェースタイプをSR-IOVに変更するには

1. AWSで実行中の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ースでSR-IOVを有効にするには、AWS CLIで次のコマンドを入力します。

```
$ aws ec2 modify-instance-attribute --instance-id \<instance\_id\> --sriov-net-support simple
```

3. SR-IOVが有効になっているか確認するには、AWS CLIで次のコマンドを入力します。

```
$ aws ec2 describe-instance-attribute --instance-id \<instance\_id\> --attribute sriovNetSupport
```

例3: AWS CLIを使用して、ネットワークインターフェースタイプが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サポートは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ー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: 次のスクリーンショットは、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 は SR-IOV が有効になっています。

ネットワークインターフェイス SR-IOV 有効化済み(/en-us/vpx/media/aws-sriov-4.png)

これらの手順で、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は完了です。

高可用性セットアップでの SR-IOV の構成

NetScaler リリース 12.0 ビルド 57.19 以降では、SR-IOV インターフェイスで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が手動で、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された場合、高可用性セットアップにアタッチされている 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
- オートスケーリング:*
- エスエヌエス:*
- sqs:すべて
- IAM:SimulatePrincipalPolicy
- IAM:GetRole

デフォルトでは、NetScaler バージョン 12.0 57.19 用の Citrix CloudFormation テンプレートは、必要な権限を IAM ロールに自動的に追加します。

注:

SR-IOV インターフェースを使用する高可用性セットアップでは、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で拡張ネットワーキングを使用するように NetScaler VPXインスタンスを構成する

AWS上にNetScaler VPXインスタンスを作成した後、AWS CLIを使用して、仮想アプライアンスを [Enhanced Networking](#)と[AWS Elastic Network Adapter \(ENA\)](#)を使用するように構成できます。

AWS ENAと組み合わせることで、拡張ネットワーキングは、より高い帯域幅、より高い1秒あたりのパケット数（PPS）パフォーマンス、および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。

構成を開始する前に、以下のトピックをお読みください。

- [前提条件](#)
- [制限事項と使用ガイドライン](#)

ENAが有効なインスタンスでは、以下の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、アベイラビリティゾーン間で移動できます。

AWS上のNetScaler VPXインスタンスをアップグレードする

AWSで稼働しているNetScaler VPXのEC2インスタンスタイプ、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できます。特定の種類のアップグレードでは、Citrixはダウンタイムを最小限に抑えるために高可用性構成方式を使用することを推奨します。

注:

- NetScalerソフトウェアリリース10.1.e-124.1308.e以降のNetScaler VPX AMI（ユーティリティライセンスとカスタマーライセンスの両方を含む）は、M1およびM2インスタンスファミリーをサポートしていません。
- VPXインスタンスサポートの変更により、10.1.e-124またはそれ以降のリリースから10.1.123.xまたはそれ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しいAMIの起動は必要なく、現在のNetScaler AMIインスタンスでアップグレードを実行できます。新しいNetScaler AMIインスタンスにアップグレードしたい場合は、高可用性構成方式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

NetScaler VPXインスタンスがリリース10.1.e-124.1308.e以降で稼働している場合、AWSコンソールからEC2インスタンスタイプを次のように変更できます。

1. VPXインスタンスを停止します。
2. AWSコンソールからEC2インスタンスタイプを変更します。
3. インスタンスを起動します。

10.1.e-124.1308.eより前のリリースでEC2インスタンスタイプを変更する場合も、インスタンスタイプをM3に変更したい場合を除いて、上記の手順を使用できます。その場合は、まず標準のNetScalerアップグレード手順に従ってNetScalerソフトウェアを10.1.e-124またはそれ以降のリリースにアップグレードし、その後、上記の手順に従う必要があります。

AWS上の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をアップグレードする

ソフトウェアエディション（例: StandardからPremiumエディションへのアップグレード）またはスループット（例: 200 Mbpsから1000Mbpsへのアップグレード）をアップグレードするには、その方法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を使用する (Bring-Your-Own-License)

顧客ライセンスを使用している場合、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インスタンスにインストールできます。Citrix Webサイトからのライセンスのダウンロードとインストールに関する詳細については、「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（時間単位料金のユーティリティライセンス）

AWSは、有料インスタンスの直接アップグレードをサポートしていません。有料のNetScaler VPXインスタンスのソフトウェアエディションまたはスループットをアップグレードするには、目的のライセンスと容量を持つ新しいAMIを起動し、古いインスタンス構成を新しいインスタンスに移行します。これは、このページの (#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NetScaler高可用性構成を使用することで実現できます。

AWS上のNetScaler VPXインスタンスのシステムソフトウェアをアップグレードする

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、[NetScalerアプライアンスのアップグレードとダウングレード](#)で標準のNetScalerアップグレード手順に従ってください。

10.1.e-124.1308.eより前の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、次にインスタンスタイプをM3に次のように変更します。

1. VPXインスタンスを停止します。
2. AWSコンソールからEC2インスタンスタイプを変更します。
3. インスタンスを開始します。

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

新しいNetScaler AMIインスタンスへのアップグレードの高可用性方法を使用するには、次のタスクを実行します。

- AWS Marketplaceから、目的のEC2インスタンスタイプ、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で新しいインスタンスを作成します。
- 古いインスタンス（アップグレード対象）と新しいインスタンスの間で高可用性を構成します。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が構成されると、古いインスタンスからの構成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HAフェイルオーバーを強制します。その結果、新しいインスタンスがプライマリになり、トラフィックの受信を開始します。
- 古いインスタンスを停止し、再構成するか、AWSから削除します。

前提条件と考慮事項

- AWS上の2つのNetScaler 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いることを確認してください。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に高可用性ペアを展開する](#)」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に、まったく同じセキュリティグループとサブネットで作成する必要があります。
- 高可用性セットアップには、両方のインスタンスについて、ユーザーのAWS Identity and Access Management (IAM) アカウントに関連付けられたアクセスキーとシークレットキーが必要です。VPXインスタンスの作成時に正しいキー情報が使用されない場合、HAセットアップは失敗します。VPXインスタンスのIAMアカウントの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 1-click launchは、アクセスキーとシークレットキーを入力として受け付けられないため、使用できません。
- 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NetScaler 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NetScaler 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。

```
add ha node 30 192.0.2.30  
  
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。

```
add ha node 10 192.0.2.10  
  
Done
```

次の点に注意してください。

- HAセットアップでは、古いインスタンスがプライマリノードであり、新しいインスタンスがセカンダリノードです。
- NSIP IP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。そのため、アップグレード後、新しいインスタンスの管理IPアドレスは以前のものと異なります。
- HA同期後、新しいインスタンスの `nsroot` アカウントパスワードは古いインスタンスのものに設定されます。

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、「[Deploy a high availability pair on AWS](#)」を参照してください。

2. HAフェイ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。

```
force HA failover
```

フェイルオーバーを強制した結果、古いインスタンスのENIは新しいインスタンスに移行され、トラフィックは新しいインスタンス（新しいプライマリノード）を経由して流れます。古いインスタンス（新しいセカンダリノード）は再起動します。

次の警告メッセージが表示された場合は、Nと入力して操作を中止してください。

```
[WARNING]:Force Failover may cause configuration loss, peer health not optimum. Reason(s):

HA version mismatch

HA heartbeats not seen on some interfaces

Please confirm whether you want force-failover (Y/N)?
```

2つのVPXインスタンスのシステムソフトウェアがHA互換ではないため、警告メッセージが表示されます。その結果、強制フェイルオーバー中に古いインスタンスの構成を新しいインスタンスに自動的に同期することはできません。

この問題の回避策は次のとおりです。

- 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>`

例: `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構成を保存します。

- `save config`

- 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イルオーバーを強制し、警告メッセージに対して「Y」と入力して強制フェイルオーバー操作を確定します。

- `force ha failover`

例:

```
> force ha failover

WARNING]:Force Failover may cause configuration loss, peer health not
optimum.

Reason(s):

HA version mismatch

HA heartbeats not seen on some interfaces

Please confirm whether you want force-failover (Y/N)? Y
```

3. HA構成を削除して、2つのインスタンスがHA構成にない状態にします。まずセカンダリノードからHA構成を削除し、次にプライマリノードからHA構成を削除します。

2つのNetScaler VPXインスタンス間のHA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次のように入力します。

```
> remove ha node \<nodeID\>

> 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に高可用性ペアを展開する](#)」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。

```
> remove ha node 30  
  
Done  
  
> save config  
  
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。

```
> remove ha node 10  
  
Done  
  
> save config  
  
Done
```

AWS上のVPXインスタンスのトラブルシューティング

Amazon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。ネットワークが接続されている場合にのみデバッグできます。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上のAWS MarketplaceライセンスのNetScaler VPXインスタンス（時間料金制のユーティリティライセンス）のサポートを提供しています。サポートケースを申請するには、AWSアカウント番号とサポートPINコードを見つけて、NetScalerサポートに電話してください。氏名とメールアドレスも尋ねられます。サポートPINを見つけるには、VPX GUIにログインし、「System」ページに移動します。

以下は、サポートPINが表示されているシステムページの例です。



AWS よくある質問

- **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動作します。暗号化されたボリュームの詳細については、以下の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良の方法は何ですか？**

以下のいずれかの方法で、AWSにNetScaler VPXインスタンスをプロビジョニングできます。

- AWS MarketplaceのAWS CloudFormationテンプレート (CFT)
- NetScaler ADM
- AWS クイックスタート
- GitHub にある Citrix® AWS CFTs
- GitHubのCitrix Terraformスクリプト
- ギットハブにあるシトリックスアンシブルプレイブック
- AWS EC2起動ワークフロー

使用する自動化ツールに基づいて、リストされているいずれかのオプションを選択できます。

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方法？**

AWS で NetScaler VPX インスタンスをアップグレードするには、システムソフトウェアをアップグレードするか、([/ja-jp/vpx/13-1/deploy-aws/upgrade-vpx-aws.html](#)) の手順に従って新しい 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方法は、([/ja-jp/citrix-application-delivery-management-service/networks/configuration-jobs/how-to-upgrade-adc-instances.html](#)) の手順に従って ADM サービスを使用することです。

- **AWS における NetScaler VPX の HA フェイルオーバー時間はどのくらいですか？**

- AWS アベイラビリティゾーン内での NetScaler VPX の HA フェイルオーバーには、約 3 秒かかります。

- AWS アベイラビリティゾーン間での 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サブスクリプションのお客様には、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、「Select for Software」サービスが提供されます。

- **(/ja-jp/vpx/13-1/deploy-aws/vpx-ha-eip-different-aws-zones.html) の展開では、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 を持つ複数のアプリケーションが複数の EIP にマッピングされている場合、複数の IPSet が必要です。したがって、HA フェイルオーバー中に、EIP のすべてのプライマリ VIP マッピングはセカンダリ (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の高可用性で INC モードが有効になっているのはなぜですか？**

アベイラビリティゾーン間の HA ペアは異なるネットワークにあります。HA 同期の場合、ネットワーク構成は同期されてはなりません。これは、HA ペアで INC モードを有効にすることで実現されます。

- **1つのアベイラビリティゾーンにある HA ノードは、同じ VPC 内にある場合、別の利用可能ゾーンにあるバックエンドサーバーと通信できますか？**

はい、同じ VPC 内の異なるアベイラビリティゾーンにあるサブネットは、SNIP 経由でバックエンドサーバーサブネットを指す追加のルートを追加することで到達可能です。たとえば、AZ1 の ADC の SNIP サブネットが 192.168.3.0/24 で、AZ2 のバックエンドサーバーサブネットが 192.168.6.0/24 の場合、AZ1 に存在する NetScaler アプライアンスに 192.168.6.0 255.255.255.0 192.168.3.1 としてルートを追加する必要があります。

- **(/ja-jp/vpx/13-1/deploy-aws/vpx-ha-eip-different-aws-zones.html) と (/ja-jp/vpx/13-1/deploy-aws/vpx-ha-pip-different-aws-zones.html) の展開は連携して機能しますか？**

はい、両方の構成を同じ HA ペアに適用できます。

- **プライベート IP を使用した異なるゾーン間での高可用性展開において、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すべきルートテーブルをどのように認識しますか？**

セカンダリノードはプライマリ NIC を認識しており、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、/var パーティションのサイズはどのくらいですか？ ディスクスペースを増やすにはどうすればよいですか？**

ディスクイメージを小さく保つため、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/` ディレクトリのスペースを増やしたい場合は、追加のディスクを接続します。`/var` のサイズを増やすには、現在、追加のディスクを接続し、重要なコンテンツを新しいディスクにコピーした後、`/var` へのシンボリックリンクを作成する必要があります。

- **いくつかのパケットエンジンがアクティブ化され、vCPUに割り当てられますか？**

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に固定されておらず、PE以外のvCPUのいずれかで実行される可能性があります。AWSによると、C5.9xlargeは36vCPUインスタンスで72 GBのメモリを搭載しています。プールライセンスを使用すると、NetScaler VPXインスタンスは最大のPE数でデプロイされます。この場合、19個の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で実行されます。

- **ADCに適したAWSインスタンスを決定する方法は？**

- a. スループット、PPS、SSL要件、平均パケットサイズなど、ユースケースと要件を理解してください。
- b. VPX帯域幅提供やvCPUベースのライセンスなど、要件を満たす適切なADC提供とライセンスを選択してください。
- c. 選択した提供内容に基づいて、AWSインスタンスを決定してください。

例:

5 Gbpsライセンスは5つのデータパケットエンジンを有効にします。したがって、vCPU要件は6（管理用に5+1）です。しかし、6 vCPUインスタンスは利用できません。そのため、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でそのスループットに到達するのに十分です。例えば、5 Gbpsライセンスの最大PE割り当てを有効にするには、5 Gbps帯域幅ライセンス用にm5.2xlargeを選択する必要があります。しかし、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場合、m5.xlargeインスタンス自体を使用して5 Gbpsのスループットを得られる可能性があります。

AWSインスタンス(/en-us/vpx/media/aws-instance.png)

- **AWS で ADC には 3 つの NIC と 3 つのサブネットのデプロイが必須ですか？**

3つのNICと3つのサブネットは推奨されるデプロイであり、それぞれ管理、クライアント、サーバーネットワーク用です。このデプロイにより、トラフィック分離とVPXパフォーマンスが向上します。2つのNICと2つのサブネット、および1つのNICと1つのサブネットが他の利用可能なオプションです。AWSで複数のNICがサブネットを共有すること、たとえば

2 つの NIC と 1 つのサブネットのデプロイは推奨されません。このシナリオは、非対称ルーティングのようなネットワークの問題を引き起こす可能性があります。詳細については、[AWS でのネットワークインターフェース設定のベストプラクティス](#) を参照してください。

- **AWS の ENA ドライバーは、インスタンスのネットワーク機能に関わらず、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) の報告される速度は、選択されたインスタンスタイプに関わらず、しばしば 1Gbps (1/1) と表示されます。これは、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しているわけではないためです。従来のネットワークインターフェースとは異なり、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。真のネットワークパフォーマンスは、主にインスタンスタイプとサイズによって決定されます。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タイプと現在のネットワーク負荷によって大幅に異なる可能性があります。

Microsoft AzureへのNetScaler VPXインスタンスの展開

Microsoft Azure Resource Manager (ARM) にNetScaler VPXインスタンスを展開すると、ビジネスニーズを達成するために以下の両方の機能セットを使用できます。

- Azureクラウドコンピューティング機能
- NetScalerの負荷分散およびトラフィック管理機能

NetScaler VPX インスタンスは、ARM にスタンドアロンインスタンスとして、またはアクティブ/スタンバイモードで高可用性ペアとして展開できます。

Microsoft AzureにNetScaler VPXインスタンスを展開する方法は2つあります。

- Azure Marketplace経由。NetScaler VPX仮想アプライアンスは、Microsoft Azure Marketplaceでイメージとして利用できます。
- GitHubで入手可能なNetScaler Azure Resource Manager (ARM) JSONテンプレートを使用。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 Stackは、ハードウェアとソフトウェアを統合したプラットフォームであり、Microsoft Azureパブリッククラウドサービスをローカルデータセンターで提供し、組織がハイブリッドクラウドを構築できるようにします。NetScaler VPXインスタンスをMicrosoft Azure Stackに展開できるようになりました。

注：

Azureは、Azure外部からの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アドレスが割り当てられている仮想マシン (VM) のNICにアタッチされた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。

前提条件

AzureにNetScaler VPXインスタンスを展開する前に、いくつか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関する知識。詳細については、[Azureの用語](#)を参照してください。
- NetScalerアプライアンスに関する知識。NetScalerアプライアンスの詳細については、[NetScaler](#)を参照してください。
- NetScaler ネットワークの知識。(/ja-jp/citrix-adc/13-1/networking.html)トピックを参照してください。

Azure での NetScaler VPX インスタンスの動作

オンプレミス展開では、NetScaler VPX インスタンスには少なくとも3つのIPアドレスが必要です。

- NSIPアドレスと呼ばれる管理IPアドレス
- サーバーファームとの通信用のサブネットIP (SNIP) アドレス
- クライアント要求を受け入れるための仮想サーバーIP (VIP) アドレス

詳細については、(/ja-jp/vpx/13-1/deploy-vpx-on-azure/network-architecture-vpx-azure.html)を参照してください。

注:

NetScaler VPX インスタンスは、Intel および AMD プロセッサ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つ以上の仮想化コアと2GB以上のメモリを持つ任意のインスタンスタイプに展開できます。システム要件の詳細については、[NetScaler VPX データシート](#)を参照してください。

Azure 展開では、NetScaler VPX インスタンスをAzureに3つの方法で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
- シングルNICシングルIP

必要に応じて、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

この展開タイプでは、VPXインスタンスに複数のネットワークインターフェース（NIC）を接続できます。任意のNICには、1つ以上のIP構成（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。

詳細については、以下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を持つ高可用性セットアップを構成する
- PowerShellコマンドを使用して、複数のIPアドレスとNICを持つ高可用性セットアップを構成する

注:

Azure環境でのMAC移動やインターフェースミュートを避けるため、NetScaler VPXインスタンスのデータインターフェースごとに（タグなしで）VLANを作成し、AzureのNICのプライマリIPをバインドすることをお勧めします。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、1つのネットワークインターフェース（NIC）に、静的または動的なパブリックおよびプライベートIPアドレスが割り当てられた複数のIP構成が関連付けられています。詳細については、以下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

シングルNICシングルIP

この展開タイプでは、1つのネットワークインターフェース（NIC）が単一のIPアドレスに関連付けられており、そのIPアドレスは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を構成する](#)を参照してください。

注:

シングルIPモードはAzure展開でのみ利用可能です。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おけるNetScaler VPXインスタンスでは利用できません。

NetScaler VPX ライセンス

Azure 上の NetScaler VPX インスタンスには、有効なライセンスが必要です。Azure で実行されている NetScaler VPX インスタンスで利用可能なライセンスオプションは次のとおりです。

- **独自のライセンスを持ち込む (BYOL):** BYOL オプションを使用するには、次の手順に従います。
 - 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。
 - 生成された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:** このライセンスモデルでは、利用可能なライセンスプールからライセンスをチェックアウトし、不要になったときにチェックインし直すことができます。詳細については、「[NetScaler VPX 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。

注:

- Azure 上の NetScaler VPX インスタンスでは、Marketplace サブスクリプションライセンスはサポートされなくなりました。
- 正しい NetScaler VPX ライセンスを有効にするには、NetScaler VPX インスタンスで設定変更を行う前にウォームリスタートを実行してください。

VPX のパフォーマンスと推奨される Azure インスタンスタイプ

目的の 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 のパフォーマンス	AZURE インスタンス タイプ		
	VPX 1 NIC/2 NIC	VPX 3 NIC	VPX 最大8 NIC
最大200 Mbps	Standard_D2s_v5	Standard_D8s_v5	Standard_D16_v5
最大1 Gbps	Standard_D4s_v5	Standard_D8s_v5	Standard_D16_v5
最大5 Gbps	Standard_D8ds_v5	Standard_D8ds_v5	Standard_D16_v5
最大10 Gbps	Standard_D8s_v5	Standard_D8s_v5	Standard_D16_v5

注:

NetScalerバージョン13.1-59.x以降では、Microsoft Azure Network Adapter (MANA) がサポートされています。詳細については、[Microsoft Documentation](#)を参照してください。

留意事項

- Azureは最大10 GbpsのVPXスループットをサポートしています。詳細については、[NetScaler VPXデータシート](#)を参照してください。
- スループットが1 Gbpsを超える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アクセラレーテッドネットワークを有効にする必要があります。この目的のために、アクセラレーテッド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。アクセラレーテッドネットワークの構成の詳細については、[Azureアクセラレーテッド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。
- NetScaler VPX仮想マシンをいつでもシャットダウンして一時的に割り当て解除する必要がある場合は、仮想マシンの作成時に静的内部IPアドレスを割り当ててください。静的内部IPアドレスを割り当てない場合、Azureは再起動のたびに仮想マシンに異なるIPアドレスを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可能性があります。
- Citrix Virtual Apps and Desktopsの展開では、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。このモードでは、ICA0n1y 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、ライセンスのないNetScaler VPXインスタンスで完全に機能します。

- SmartAccessモード。このモードでは、ICAonly VPN仮想サーバーパラメーターがOFFに設定されます。SmartAccessモードは、ライセンスのないNetScaler VPXインスタンスで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機能を構成するには、NetScaler VPXインスタンスにPremiumライセンスを適用する必要があります。

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でIPv6アドレスをサポートします。AzureクラウドのNetScaler VPXスタンドアロンインスタンスで、IPv6アドレスをVIPおよび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、以下の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](#)
- [Azure仮想ネットワーク内のIPv4アプリケーションにIPv6を追加する - Azure CLI](#)
- [アドレスの種類](#)

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でのIPv6展開は、現在、Azureバックエンドの自動スケーリングをサポートしていません。
- NetScaler VPX HA展開ではIPv6はサポートされていません。

制限事項

ARMでNetScaler VPXロードバランシングソリューションを実行すると、以下の制限が課されます。

- Azureアーキテクチャは、以下のNetScaler機能のサポートに対応していません。
 - 無償ARP (GARP)
 - L2モード

- タグ付きVLAN
 - ダイナミックルーティング
 - 仮想MAC
 - USIP
 - クラスタリング
- 3 Gbpsを超えるスループットを持つ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1秒あたりのSSLトランザクションなどの他の機能は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が生成するデプロイIDは、ARMではユーザーに表示されません。ARMでNetScaler VPXアプライアンスをデプロイするために、そのデプロイIDを使用することはできません。

Azure 用語

NetScaler VPX Azure 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 Load Balancer – 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受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で定義された仮想マシン間で分散されます。ロードバランサーは、外部またはインターネットに面している場合もあれば、内部の場合もあります。
2. Azure Resource Manager (ARM) – ARM は Azure のサービスのため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 と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 – これらは、負荷が分散される仮想マシン NIC (NIC) に関連付けられた IP アドレスです。
4. BLOB - Binary Large Object – Azure ストレージに保存できるファイルや画像などの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 アドレスは、トラフィックのインGRESSとして機能します。
6. インスタンスレベルパブリック IP (ILPIP) – ILPIP は、仮想マシンまたはロールインスタンスが配置されてい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

れている VIP (仮想 IP) の代わりになるものではありません。むしろ、仮想マシンまたはローカ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 の略) と呼ばれていました。

7. インバウンド NAT ルール – ロードバランサー上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ています。
 8. IP-Config - 個々の NIC に関連付けられた IP アドレスペア (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 アドレスは NULL にすることができます。各 NIC には、最大 255 個の IP-Config を関連付けることができます。
 9. ロードバランシングルール – 指定されたフロントエンド 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ピングするルールプロパティです。ロードバランサーリソースの単一の定義で、複数のロードバランシングルールを定義できます。各ルールは、仮想マシンに関連付けられたフロントエンド IP とポート、およびバックエンド IP とポートの組み合わせを反映します。
- 
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ています。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のいずれか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、およびVPNゲートウェイを使用してネットワークをAzureに拡張する際のオンプレミスネットワーク内での通信に使用されます。プライベートIPアドレスを使用すると、Azureリソースは、インターネットから到達可能なIPアドレスを使用せずに、VPNゲートウェイまたはExpressRoute回線を通じて、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デプロイモデルでは、プライベートIPアドレスは、仮想マシン、内部ロードバランサー (ILB)、アプリケーションゲートウェイなどのAzureリソースに関連付けられています。
 12. プローブ – これは、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確認するために使用されるヘルスプローブを含みます。特定の仮想マシンが一定期間ヘルスプローブに応答しない場合、その仮想マシンはトラフィック処理から除外されます。プローブを使用

すると、仮想インスタンスの健全性を追跡できます。ヘルスプローブが失敗した場合、仮想インスタンスは自動的にローテーションから除外されます。

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ています。
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的領域。価格設定、リージョンサービス、およびオファertypeはリージョンレベルで公開されます。リージョンは通常、数百マイル離れた別のリージョンとペアになり、リージョンペアを形成します。リージョンペアは、ディザスタリカバリーおよび高可用性シナリオのメカニズムとして使用できます。一般的に「ロケーション」とも呼ばれます。
15. リソースグループ - Resource Manager内のコンテナで、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すべてのリソースを含めることも、論理的にグループ化されたリソースのみ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は、Azure Storage内のAzure BLOB、キュー、テーブル、およびファイルサービスへのアクセスを提供します。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の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ーのソフトウェア実装。複数の仮想マシンを同じハードウェア上で同時に実行できます。Azureでは、さまざまなサイズの仮想マシンが利用可能です。
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します。これは、サブスクリプション専用のAzureクラウドの論理的な分離です。このネットワーク内で、IPアドレスブロック、DNS設定、セキュリティポリシー、およびルートテーブルを完全に制御できます。また、VNetをさらにサブネットに分割し、Azure IaaS仮想マシンとクラウドサービス (PaaSロールインスタンス) を起動することもできます。さらに、Azureで利用可能な接続オプションのいずれかを使用して、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。要するに、Azureが提供するエンタープライズ規模のメリットを享受しつつ、IPアドレスブロックを完全に制御してネットワークをAzureに拡張できます。



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

Azure Resource Manager (ARM) では、NetScaler VPX仮想マシン (VM) は仮想ネットワーク内に存在します。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ースを作成し、VPXインスタンスにアタッチできます。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

ができます。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インスタンスへのリクエスト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インスタンスの展開におけ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、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アドレスなどのネットワーク構成の詳細が含まれています。

ARMでは、単一のNICと単一のIPアドレスで展開されたVM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便利です。

- パブリックIP (PIP) アドレスは、NetScaler® VMの仮想NICに直接構成されたインターネットに面した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、VMに構成された内部IPアドレスです。ルーティングできません。
- 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アドレスを介してNetScalerサービスにアクセスし、リクエストがNetScaler VPX™ VMのNICまたはAzureロードバランサーに到達すると、VIPは内部IP (NSIP) と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からの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と統合され、VMで構成されたサービスに応じて、適切な種類のトラフィックをNICの適切なポートに選択的に送信します。

次の図は、ARMでプロビジョニングされたNetScaler VPXインスタンスを介して、クライアントからサーバーへトラフィックがどのように流れるかを示しています。



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。VMレベルでこの直接PIPを使用する場合、ネットワークトラフィックを傍受するためのインバウンドおよびアウトバウンドルール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リクエストは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、トラフィックをVPXインスタンスの内部IPアドレスに転送します。

次の図は、AzureがNetScaler内部IPアドレスをマッピングするためにネットワークアドレス変換をどのように実行するかを示しています。



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。インバウンドおよびアウトバウンドルールが定義されると、パブリックHTTPポート80はクライアント要求が受信され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 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は、PIPをポート10080の内部IPアドレス10.x.x.xにマッピングするためにネットワークアドレス変換を実行し、クライアント要求を転送します。

注：

高可用性構成のNetScaler VPX VMは、ロードバランシングトラフィックを制御するためにインバウンドルールが定義された外部または内部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まずこれらのロードバランサーによって傍受され、ロードバランサーに定義されたバックエンドプール、NATルール、およびヘルスプローブを持つロードバランシングルールに従ってトラフィックが転送されます。

ポート使用ガイドライン

NetScaler VPXインスタンスの作成時、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さらに多くのインバウンドおよびアウトバウンドルールを設定できます。各インバウンドおよびアウトバウンドルールは、パブリックポートと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のルールを設定する前に、使用できるポート番号に関する以下のガイドラインに注意してください。

1. NetScaler VPXインスタンスは以下のポートを予約します。インターネットからのリクエストにパブリックIPアドレスを使用する場合、これらのポート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公開されるサービスで標準ポート（例：ポート443）を使用したい場合は、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。標準ポートは、このVIPサービス用にNetScalerで構成された別のポートにマッピングされます。

例えば、VIPサービスはVPXインスタンス上でポート8443で実行されているものの、パブリックポート443にマッピングされている場合があります。そのため、ユーザーがパブリックIP経由でポート443にアクセスすると、そのリクエストはプライベートポート8443に転送されます。

2. パブリックIPアドレスは、パッシブFTPやALGなど、ポートマッピングが動的に開かれるプロトコルをサポートしていません。
3. 高可用性は、Azureロードバランサーに構成された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インスタンスは2つのIPアドレス（パブリックIPアドレス（PIP）と内部IPアドレス）に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はルーティングできません。VPXでVIPを設定するには、内部IPアドレスと利用可能な任意の空きポートを使用します。PIPを使用してVIPを設定しないでください。

NetScaler VPXスタンドアロンインスタンスを構成する

仮想マシンを作成し、他のリソースを構成することで、Azure Resource Manager (ARM) ポータルで単一のNetScaler VPXインスタンスをスタンドアロンモードでプロビジョニングできます。

開始する前に

以下があ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
- マイクロソフト アジュール エスディーケー
- マイクロソフト アジュール パワーシェル

Microsoft Azure Portal ページで、ユーザー名とパスワードを入力して Azure Resource Manager ポータルにログオンします。

注:

ARMポータルでは、あるペインの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。デバイスを構成するには、あるペインから別のペインに移動します。

構成手順の概要

1. リソースグループを構成する
2. ネットワークセキュリティグループを構成する
3. 仮想ネットワークとそのサブネットを構成する
4. ストレージアカウントを構成する
5. 可用性セットを構成する
6. NetScaler VPXインスタンスを構成する。

リソースグループを構成する

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デプロイ、管理、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. **作成** をクリックします。

リソースグループを構成する(/en-us/vpx/media/conf-resource-group-1.png)

ネットワークセキュリティグループを構成する

仮想ネットワーク内の送受信トラフィックを制御するためのインバウンドおよび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、単一の仮想マシンに対するセキュリティルールを定義できるだけでなく、仮想ネットワークサブネットに対するセキュリティルールも定義できます。

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。
 - 名前 - セキュリティグループの名前を入力します
 - リソースグループ - ドロップダウンリストからリソースグループを選択します

注：

正しい場所が選択されていることを確認してください。ドロップダウンリストに表示されるリソースのリストは、場所によって異なります。



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、サービスにセキュリティと分離のレイヤーを提供します。同じ仮想ネットワークの一部であるVMとサービスは、相互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には、次の手順を実行します。

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、デプロイモードが **Resource Manager** 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** ペインで、次の値を入力し、次に **作成** をクリックします。
 - 仮想ネットワークの名前
 - アドレス空間 - 仮想ネットワーク用に予約された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番目のサブネットはこのステップの後半で作成します)
 - サブネットアドレス範囲 - サブネットの予約済みIPアドレスブロックを入力します

- リソースグループ - ドロップダウンリスト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構成する

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. **+サブネット**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
- アドレス範囲 - 2番目のサブネットの予約済み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. **作成**をクリックします。



ストレージアカウントを構成する

ARM IaaSインフラストラクチャストレージには、BLOB、テーブル、キュー、ファイル形式でデータを保存できるすべてのサービスが含まれています。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。

すべてのデータを保存するためのストレージアカウントを作成します。

1. **+New > Data + Storage > Storage account**をクリックします。

2. **Create storage account**ペインで、次の詳細を入力します。

- アカウント名
- デプロイモード - 必ず**Resource Manager**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**General purpose**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**Locally redundant storage**を選択します
- リソースグループ - ドロップダウンリスト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。



可用性セットを構成する

可用性セットは、計画的または計画外のメンテナンスの場合でも、少なくとも1つのVMが稼働し続けることを保証します。同じ「可用性セット」内の2つ以上のVMは、冗長サービスを実現するために異なる障害ドメインに配置されます。

1. **+New**をクリックします。
2. MARKETPLACEペインで**See all**をクリックし、**Virtual Machines**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**Availability set**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」ペインで、次の詳細を入力します。
 - セットの名前
 - リソースグループ - ドロップダウンリストから新しく作成したリソースグループを選択します
5. 「作成」をクリックします。



NetScaler VPXインスタンスを構成する

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、インスタンスが配置される必要なサブネットを持つ仮想ネットワークを作成していることを確認してください。VMプロビジョニング中に仮想ネットワークを作成できますが、異なるサブネットを作成する柔軟性はありません。

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サーバーとVPN接続を構成します。

注:

Citrixは、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、およびその他のエンティティを作成して、プロビジョニング中にネットワーク情報が利用できるようにすることを推奨します。

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ポータルで任意のエントリをすばやく見つけるには、Azure Marketplaceの検索ボックスにエントリの名前を入力し、\<Enter\>キーを押すこともできます。検索ボックスに「NetScaler」と入力して、NetScalerイメージを見つけます。

VPXインスタンスの構成(/en-us/vpx/media/config-netscaler-vpx-instance-1-citrix.png)

注:

最新のイメージを選択してください。NetScaler®イメージには、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 Bring Your Own License**ページで、ドロップダウンリストから**Resource Manager**を選択し、**Create**をクリックします。

VPXインスタンス1の構成(/en-us/vpx/media/config-netscaler-vpx-instance-2.png)

5. **Create virtual machine**ペインで、仮想マシンを作成するために各セクションで必要な値を指定します。構成を保存するには、各セクションで**OK**をクリックします。

基本:

- 名前 - NetScaler VPXインスタンスの名前を指定します
- VMディスクタイプ - ドロップダウンメニューからSSD（デフォルト値）またはHDDを選択します
- ユーザー名とパスワード - 作成したリソースグループ内のリソースにアクセスするためのユーザー名とパスワードを指定します
- 認証タイプ - SSH公開キーまたはパスワードを選択します
- リソースグループ - ドロップダウンリストから作成したリソースグループを選択します

ここでリソースグループを作成することもできますが、Citrix®はAzure Resource ManagerのResource groupsからリソースグループを作成し、ドロップダウンリストからそのグループを選択することを推奨しています。

注:

Azure stack環境では、基本パラメーターに加えて、以下のパラメーターを指定します。

- Azure Stack ドメイン
- Azure Stack 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VMディスクの種類（SDDまたはHDD）に応じて、ディスクサイズが表示されます。

- 要件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。

設定:

- デフォルト（標準）のディスクタイプを選択します
- ストレージアカウント - ストレージアカウントを選択します
- 仮想ネットワーク - 仮想ネットワークを選択します
- サブネット - サブネットアドレスを設定します
- パブリックIPアドレス - IPアドレス割り当てのタイプを選択します
- ネットワークセキュリティグループ - 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および送信ルールが構成されていることを確認してください。
- 可用性セット - ドロップダウンメニューボックスから可用性セットを選択します

概要:

構成設定が検証され、概要ページに検証結果が表示されます。検証に失敗した場合、概要ページに失敗の理由が表示されます。該当するセクションに戻り、必要に応じて変更してください。検証が成功した場合は、**OK**をクリックします。

購入:

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。

高可用性展開の場合、アクティブ/スタンバイ構成で展開するために、同じアベイラビリティセットおよび同じリソースグループにNetScaler VPXの2つの独立した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複数のIPアドレスを構成する

このセクションでは、Azure Resource Manager (ARM) で、複数のIPアドレスを持つスタンドアロンのNetScaler 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ドキュメントの「[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の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。

ユースケース

このユースケースでは、スタンドアロンのNetScaler VPXアプライアンスが、仮想ネットワーク (VNET) に接続された単一のNICで構成されています。NICには3つのIP構成 (ipconfig) が関連付けられており、それぞれが表に示すように異なる目的を果たします。

IP構成	関連付けられているもの	目的
IPコンフィグ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	管理トラフィックを処理する
IPコンフィグ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	クライアント側トラフィックを処理する
IPコンフィグ3	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します

注:

`IPConfig-3` は、どのパブリックIPアドレスとも関連付けられていません。

図: トポロジ

ユースケースの視覚的な表現を以下に示します。



注:

複数NIC、複数IPのAzure NetScaler VPX展開では、プライマリ（最初の）NICのプライマリ（最初の） `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、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。

始める前に

始める前に、次のリンクの手順に従ってVPXインスタンスを作成してください。

NetScaler VPXスタンドアロンインスタンスの構成

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インスタンスの複数の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アドレスを設定するには：

1. VMにIPアドレスを追加する
2. NetScalerが所有するIPアドレスを設定する

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」と入力して、**仮想マシン**をクリックします。
2. **仮想マシンブレード**で、IPアドレスを追加するVMをクリックします。表示される仮想マシンブレードで**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。VMの作成時に割り当てられた既存のIP構成である**ipconfig1**が表示されます。このユースケースでは、ipconfig1に関連付けられているIPアドレスが静的であることを確認してください。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、**追加**を作成します。



IP構成の追加ウィンドウで、**名前**を入力し、割り当て方法を**静的**として指定し、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:

静的プライベートIPアドレスを追加する前に、IPアドレスの利用可能性を確認し、そのIPアドレスがNICが接続されているのと同じサブネットに属していることを確認してください。



次に、**必要な設定を構成**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。

既定では、パブリックIPは動的です。VMが常に同じパブリックIPアドレスを使用するには、静的パブリックIP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、割り当ての下で「**静的**」をクリックします。その後、「**OK**」をクリックします。

パブリックIPアドレスの作成(/en-us/vpx/media/create_pip2.png)

注:

割り当て方法を静的に設定した場合でも、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アドレスのプールから割り当てられます。

ipconfig3 の IP 構成をもう 1 つ追加するには、次の手順に従います。パブリック IP は必須ではありません。

IP構成を表示(/en-us/vpx/media/showipconfig.png)

ステップ2: NetScalerが所有するIPアドレスを構成する

GUIまたはコマンド `add ns ip` を使用して、NetScalerが所有するIPアドレスを構成します。詳細については、「[NetScalerが所有する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を持つ高可用性セットアップを構成する

Microsoft Azureの展開では、2つのNetScaler VPXインスタンスの高可用性構成は、Azure Load Balancer (ALB) を使用して実現されます。これは、ALB上でヘルスプローブを構成することで達成され、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して、各VPXインスタンスを監視します。

このセットアップでは、プライマリノードのみ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はそのインスタンスにデータトラフィックを送信し始めます。プライマリインスタンスが2回連続でヘルスプローブを逃した場合、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には、新しいプライマリがヘルスプローブに応答し始め、ALBはそれ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13秒になる可能性があります。

Azure上で、複数のNICを持つNetScaler VPXインスタンスのペアをアクティブ/パッシブ高可用性 (HA) セットアップで展開できます。各NICは複数のIPアドレスを含むことができます。

複数のNICを持つ高可用性展開には、以下のオプションが利用可能です。

- Azureアベイラビリティセットを使用した高可用性
- Azureアベイラビリティ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ゾーンの詳細については、Azureドキュメントの「[Linux仮想マシンの可用性を管理する](#)」を参照してください。

アベイラビリティセットを使用した高可用性

アベイラビリティセットを使用した高可用性セットアップは、以下の要件を満たす必要があります。

- HA独立ネットワーク構成 (INC)構成
- ダイレクトサーバーリターン (DSR) モードの Azure ロードバランサー (ALB)

すべてのトラフィックはプライマリノードを経由します。セカンダリノードは、プライマリノードが故障するまでスタンバイモードのままです。

注:

Azureクラウド上でNetScaler VPX高可用性展開を機能させるには、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 Load Balancer (ALB) はそのフローティングPIPを提供し、フェイルオーバー時に自動的にセカンドノードに移動されます。

図: Azure Availability Set 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/パッシブデプロイでは、ALBフロントエンドのパブリックIP（PIP）アドレスが、各VPXノードのVIPアドレスとして追加されます。HA-INC構成では、VIPアドレスはフローティングであり、SNIPアドレスはインスタンス固有です。

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、デフォルトの3つのサブネットと6つのNIC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の要件に合わせて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ペアをデプロイ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®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、HA-INCモードでVPXインスタンスのペアを迅速かつ効率的にデプロイできます。このテンプレートは、3つのサブネットと6つのNICを持つ2つのノードを作成します。サブネットは管理、クライアント、サーバー側のトラフィック用であり、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。

NetScaler HAペアテンプレートは、[Azure Marketplace](#)で入手できます。

次の手順を完了して、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 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デプロイとライセンスを選択し、**続行**をクリックします。

ライセンス付き Azure HA(/en-us/vpx/media/adc-ha-license-citrix.png)

4. ページが表示されます。リソースグループを作成し、 を選択します。

テンプレートの基本(/en-us/vpx/media/template-basics.png)

5. ページが表示されます。詳細を入力し、 を選択します。

一般設定(/en-us/vpx/media/monitoring-metrics.png)

注記:

デフォルトでは、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 Azure Active 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。新しく作成した AAD アプリケーションに共同作成者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 アプリケーションと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. 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、**完了** を選択します。

ネットワーク設定(/en-us/vpx/media/template-networksettings.png)

7. ページが表示されます。構成を確認し、必要に応じて編集します。**完了** を選択して確定します。
8. ページが表示されます。**完了** 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、しばらく時間がかかる場合があります。完了後、Azure ポータルで **完了** 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 ns-vpx0 と ns-vpx1 として表示されます。

HA セットアップで、セキュリティルールやポートの追加作成など、さらに変更が必要な場合は、Azure ポータルから実行できます。

Azure ポータル(/en-us/vpx/media/node1and2-12-1.png)

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** を使用してロードバランシング仮想サーバーを構成する必要があります。ALB PIP をを見つけるには、ALB \> を選択します。

ALBフロントエンドパブリックIP(/en-us/vpx/media/alb-frontend-ip.png)

ロードバランシング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：

以下のリンクは、HA展開と仮想サーバー構成に関する追加情報を提供します。

- 異なるサブネットでの高可用性ノードの構成(/ja-jp/citrix-adc/13-1/system/high-availability-introduction/configuring-high-availability-nodes-different-subnets.html)

- 基本的なロードバランシングのセットアップ(/ja-jp/citrix-adc/13-1/load-balancing/load-balancing-setup.html)

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を持つ高可用性セットアップを構成する(/ja-jp/vpx/13-1/deploy-vpx-on-azure/configure-ha-pair-using-powershell.html)
- Azureでのアクティブ/スタンバイHA展開におけるGSLBの構成(/ja-jp/vpx/13-1/deploy-vpx-on-azure/configuring-gslb-active-standby-ha-deployment-azure.html)

可用性ゾーンを使用した高可用性

Azure可用性ゾーンは、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、ネットワークを提供し、回復性を高めます。特定のAzureリージョンのみが可用性ゾーンをサポートしています。詳細については、Azureドキュメントの「[Azureの可用性ゾーンとは]」を参照してください。

図：Azure可用性ゾーンを使用した高可用性展開アーキテクチャの例

可用性ゾーンを使用したHA展開(/en-us/vpx/media/high-availability-azure-availability-zones.png)

Azure Marketplaceで入手可能な「NetScaler 13.0 HA using Availability Zones」というテンプレートを使用して、高可用性モードでVPXペアを展開できます。

Azure可用性ゾーンを使用して、テンプレートを起動し、高可用性VPXペアを展開するには、次の手順を完了します。

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。

可用性ゾーンテンプレート(/en-us/vpx/media/availability-zones-template.png)

2. デプロイタイプがResource Manager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いることを確認してください。可用性ゾーンをサポートするリージョンに関する詳細については、Azureドキュメントの[Azureの可用性ゾーンとは](#)を参照してください。

可用性ゾーンリージョン(/en-us/vpx/media/availability-zone-region.png)

4. **一般設定**ページが表示されます。詳細を入力し、**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**ページが表示されます。構成を確認し、必要に応じて編集します。**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、しばらく時間がかかる場合があります。完了後、**リソースグループ**を選択すると、Azureポータルで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できます。高可用性ペアはns-vpx0とns-vpx1として表示されます。また、**場所**列の下に場所が表示されます。

可用性ゾーンリソースグループ(/en-us/vpx/media/availability-zones-vm-pair.png)

HAセットアップで、セキュリティルールやポートの追加作成などのさらなる変更が必要な場合は、Azureポータルから実行できます。

Azure Monitorでメトリックを使用してインスタンスを監視する

Azure Monitor 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、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で実行されるNetScaler VPXリソースをリアルタイムで監視します。収集されたデータにアクセスするには、**Metrics Explorer**を使用できます。詳細については、[Azure Monitor メトリックの概要](#)を参照してください。

注意事項

- Azure Marketplace オファーを使用してAzureにNetScaler VPXインスタンスをデプロイする場合、メトリックサービスはデフォルトで無効になっています。
- メトリックサービスはAzure CLIではサポートされていません。
- メトリックは、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 (インバウンドおよびアウトバウンド) で利用できます。

Azure Monitor でメトリックを表示する方法

インスタンスのAzure Monitorでメトリックを表示するには、次の手順を実行します。

1. **Azure Portal > Virtual Machines**にログインします。
2. プライマリノードである仮想マシンを選択します。
3. **Monitoring**セクションで、**Metrics**をクリックします。
4. **Metric Namespace**ドロップダウンメニューから、**NetScaler**をクリックします。

5. **Metrics** ドロップダウンメニューの **All metrics** の下で、表示したいメトリックをクリックします。
6. **Add metric** をクリックして、同じグラフに別のメトリックを表示します。グラフオプションを使用してグラフ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を持つ高可用性セットアップを構成する

Azureで、複数のNICを持つNetScaler VPXインスタンスのペアをアクティブ-パッシブ高可用性（HA）セットアップで展開できます。各NICには複数の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独立ネットワーク構成（INC）構成
- ダイレクトサーバーリターン（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経由します。セカンダリノードは、プライマリノードが故障するまでスタンバイモードのままです。

注：

AzureクラウドでNetScaler VPXの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 Load Balancer（ALB）は、そのフローティングPIPを提供し、フェールオーバー時に自動的にセカンド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ノードのVIPアドレスとして追加されます。HA-INC構成では、VIPアドレスはフローティングであり、SNIPアドレスはインスタンス固有です。

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インスタンスにのみトラフィックをリダイレクトします。したがって、HAセットアップでは、プライマリノードはヘルスプローブに応答しますが、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回連続でヘルスプローブを逃した場合、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。フェールオーバー時には、新しいプライマリがハ

ルスプローブに応答を開始し、ALBはトラフィックをそれにリダイレクトします。標準のVPX高可用性フェールオーバー時間は3秒です。トラフィック切り替えにかかる合計フェールオーバー時間は最大13秒になる可能性があります。

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、デフォルトの3つのサブネットと6つのNIC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ペアをデプロイ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が含まれており、各NICは異なるサブネットにデプロイされます。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で構成されており、これには1つのパブリックIP (pip1) と1つのプライベートIP (12.5.2.24) が含まれます。 nic1; Mgmtsubnet=12.5.2.0/24	管理IPはIPConfig5で構成されており、これには1つのパブリックIP (pip3) と1つのプライベートIP (12.5.2.26) が含まれます。 nic4;Mgmtsubnet=12.5.2.0/24
構成されているLBルールとポートはHTTP (80)、SSL (443)、ヘルスプローブ (9000) です	クライアント側IPはIPConfig3で構成されており、これには1つのプライベートIP (12.5.1.27) が含まれます。 nic2; FrontEndsubnet=12.5.1.0/24	クライアント側IPはIPConfig7で構成されており、これには1つのプライベートIP (12.5.1.28) が含まれます。 nic5;FrontEndsubnet=12.5.1.0/24
-	サーバー側IPはIPConfig4で構成されており、これには1	サーバー側IPはIPConfig8で構成されており、これには1

ALB	VPX1	VPX2
	つのプライベートIP (12.5.3.24) が含まれます。 nic3;BackendSubnet=12.5.3.0/24	つのプライベートIP(12.5.3.28)、nic6、BackendSubnet=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) です。	-

パラメータ設定

このシナリオでは、以下のパラメータ設定が使用されます。

```
$locName= "South 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"
```

デプロイを完了するには、PowerShellコマンドを使用して以下の手順を実行します。

1. 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する
2. ネットワークセキュリティグループを作成し、ルールを追加する
3. 仮想ネットワークと3つのサブネットを作成する
4. パブリックIPアドレスを作成する
5. VPX1のIP構成を作成する
6. VPX2のIP構成を作成する
7. VPX1のNICを作成する
8. VPX2のNICを作成する
9. VPX1を作成
10. VPX2を作成
11. ALBを作成

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

$rule2 = New-AzureRmNetworkSecurityRuleConfig -Name $rule2Name
-Description "Allow HTTPS" -Access Allow -Protocol Tcp -Direction Inbound
-Priority 110

-SourceAddressPrefix Internet -SourcePortRange * -DestinationAddressPrefix
* -DestinationPortRange 443

$rule3 = New-AzureRmNetworkSecurityRuleConfig -Name $rule3Name
-Description "Allow SSH" -Access Allow -Protocol Tcp -Direction Inbound
-Priority 120

-SourceAddressPrefix Internet -SourcePortRange * -DestinationAddressPrefix
* -DestinationPortRange 22
```

```
$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。

```
$frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name  
$frontEndSubnetName -AddressPrefix $frontEndSubnetRange (this parameter  
value should be as per your requirement)
```

```
$mgmtSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name $mgmtSubnetName  
-AddressPrefix $mgmtSubnetRange
```

```
$backendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name  
$backEndSubnetName -AddressPrefix $backEndSubnetRange
```

```
$vnet =New-AzureRmVirtualNetwork -Name $vNetName -ResourceGroupName  
$rgName -Location $locName -AddressPrefix $vNetAddressRange -Subnet  
$frontendSubnet,$backendSubnet, $mgmtSubnet
```

```
$subnetName ="frontEndSubnet"
```

```
$subnet1=$vnet.Subnets|?{$_.Name -eq $subnetName}
```

```
$subnetName="backEndSubnet"
```

```
$subnet2=$vnet.Subnets|?{$_.Name -eq $subnetName}
```

```
$subnetName="mgmtSubnet"
```

```
$subnet3=$vnet.Subnets|?{$_.Name -eq $subnetName}
```

パブリックIPアドレスを作成します。

```
$pip1=New-AzureRmPublicIpAddress -Name $pubIPName1 -ResourceGroupName  
$rgName -DomainNameLabel $domName1 -Location $locName -AllocationMethod  
Dynamic
```

```
$pip2=New-AzureRmPublicIpAddress -Name $pubIPName2 -ResourceGroupName  
$rgName -DomainNameLabel $domName2 -Location $locName -AllocationMethod  
Dynamic
```

```
$pip3=New-AzureRmPublicIpAddress -Name $pubIPName3 -ResourceGroupName
```



```
$rgName -DomainNameLabel $domName3 -Location $locName -AllocationMethod  
Dynamic
```

VPX1のIP構成を作成します。

```
$IpConfigName1 = "IPConfig1"  
  
$IPAddress = "12.5.2.24"  
  
$IPConfig1=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName1  
-Subnet $subnet3 -PrivateIpAddress $IPAddress -PublicIpAddress $pip1  
-Primary  
  
$IPConfigName3="IPConfig-3"  
  
$IPAddress="12.5.1.27"  
  
$IPConfig3=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName3  
-Subnet $subnet1 -PrivateIpAddress $IPAddress -Primary
```

```
$IPConfigName4 = "IPConfig-4"
```

```
$IPAddress = "12.5.3.24"
```

```
$IPConfig4 = New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName4  
-Subnet $subnet2 -PrivateIpAddress $IPAddress -Primary
```

VPX2のIP構成を作成します。

```
$IpConfigName5 = "IPConfig5"
```

```
$IPAddress="12.5.2.26"
```

```
$IPConfig5=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName5  
-Subnet $subnet3 -PrivateIpAddress $IPAddress -PublicIpAddress $pip2  
-Primary
```

```
$IPConfigName7="IPConfig-7"
```

```
$IPAddress="12.5.1.28"
```

```
$IPConfig7=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName7  
-Subnet $subnet1 -PrivateIpAddress $IPAddress -Primary
```

```
$IPConfigName8="IPConfig-8"
```

```
$IPAddress="12.5.3.28"
```

```
$IPConfig8=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName8  
-Subnet $subnet2 -PrivateIpAddress $IPAddress -Primary
```

VPX1のNICを作成します。

```
$nic1=New-AzureRmNetworkInterface -Name $nicName1 -ResourceGroupName  
$rgName -Location $locName -IpConfiguration $IpConfig1  
-NetworkSecurityGroupId $nsg.Id
```

```
$nic2=New-AzureRmNetworkInterface -Name $nicName2 -ResourceGroupName  
$rgName -Location $locName -IpConfiguration $IpConfig3  
-NetworkSecurityGroupId $nsg.Id
```

```
$nic3=New-AzureRmNetworkInterface -Name $nicName3 -ResourceGroupName  
$rgName -Location $locName -IpConfiguration $IpConfig4  
-NetworkSecurityGroupId $nsg.Id
```

VPX2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、次のサブステップが含まれます。

- VM構成オブジェクトを作成
- 資格情報、OS、およびイメージを設定
- NICを追加する
- OSディスクを指定し、VMを作成する

```
$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
```

```
$osVhdUri=$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  
$locName
```

VPX2を作成する。

```
...  
  
$suffixNumber=2  
  
$vmName=$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 $nic4.Id  
-Primary
```

```
$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic5.Id
```

```
$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic6.Id
```

```
$osDiskName=$vmName + "-" + $osDiskSuffix2
```

```
$osVhdUri=$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 $locName
```

```
...
```

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。

```
...
```

```
$nic1.IPConfig
```

```
$nic2.IPConfig
```



```
$nic3.IPConfig
```

```
$nic4.IPConfig
```

```
$nic5.IPConfig
```

```
$nic6.IPConfig
```

```
...
```

Azureロードバランサー (ALB) を作成する。

この手順には、次のサブステップが含まれます。

- フロントエンドIP構成を作成する
- ヘルスプローブを作成する
- バックエンドアドレスプールを作成する
- ロードバランシングルール (HTTPおよびSSL) を作成する
- フロントエンドIP構成、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を作成する
- 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ペアを正常にデプロイした後、各VPXインスタンスにログオンして、HA-INC、SNIP、およびVIPアドレスを設定します。

1. HAノードを追加するには、次のコマンドを入力します。

```
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として追加します。

```
add nsip privateIPofNIC2 255.255.255.0 -type SNIP add nsip privateIPofNIC5
255.255.255.0 -type SNIP
```

3. プライマリノードに、ALBのフロントエンドIPアドレス (パブリックIP) を持つロードバランシング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([/ja-jp/vpx/13-1/deploy-vpx-on-azure/configuring-gslb-active-standby-ha-deployment-azure.html](https://docs.citrix.com/en-us/netScaler/vpx/13-1/deploy-vpx-on-azure/configuring-gslb-active-standby-ha-deployment-azure.html))

ALBのフローティングIP無効モードでAzure上にNetScaler高可用性ペアを展開する

複数のNICを持つNetScaler VPXインスタンスのペアを、Azure上にアクティブ/パッシブ高可用性 (HA) セットアップで展開できます。各NICは多数のIPアドレスを含むことができます。

アクティブ/パッシブ展開には以下が必要です。

- HA独立ネットワーク構成 (INC)
- Azure Load Balancer (ALB) と以下:
 - フローティングIP有効モードまたはDirect Server Return (DSR) モード
 - フローティングIP無効モード

ALBフローティングIPオプションの詳細については、[Azure 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上にVPXペアをアクティブ/パッシブHAセットアップで展開したい場合は、([/ja-jp/vpx/13-1/deploy-vpx-on-azure/configure-ha-pair-using-powershell.html](#))を参照してください。

フローティングIP無効モードでのALBを使用したHA展開アーキテクチャ

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ースのプライベートIPアドレスが、各VPXインスタンスにVIPアドレスとして追加されます。HA-INCモードで構成し、VIPアドレスはIPSetを使用して共有され、SNIPアドレスはインスタンス固有になります。すべてのトラフィックはプライマリインスタンスを経由します。セカンダリインスタンスは、プライマリインスタンスが失敗するまでスタンバイモードになります。

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

Azure上にNetScaler VPXインスタンスを展開する前に、以下の情報に精通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 の用語](#)を参照してください。
- NetScaler アプライアンスの動作。詳細については、[NetScaler のドキュメント](#)を参照してください。
- NetScaler のネットワーク。詳細については、[ADC のネットワーク](#)を参照してください。

- Azure ロードバランサーとロードバランシングルール構成の詳細については、[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 無効モードで Azure に VPX HA ペアを展開する方法

HA および ALB 展開手順の概要を次に示します。

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) を展開します。
2. 両方のインスタンスにクライアント NIC とサーバー 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ロードバランシングルールを持つ ALB を展開します。
4. NetScaler GUI を使用して、両方のインスタンスで HA 設定を構成します。

ステップ 1. Azure に 2 つの VPX インスタンスを展開します。

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)。

NetScaler バージョンの選択(/en-us/vpx/media/alb-floating-ip-select-adc-version.png)

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。

「ADCライセンスモードの選択」 (/en-us/vpx/media/alb-floating-ip-adc-licensing-mode.png)

「仮想マシンの作成」ページが開きます。

3. 展開を成功させるために、各タブ (基本、ディスク、ネットワーク、管理、監視、詳細設定、タグ) で必要な詳細をすべて入力します。

「仮想マシンの作成」 (/en-us/vpx/media/alb-floating-ip-create-vm.png)

4. **ネットワーク**タブで、管理、クライアント、サーバーの各NIC用に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。または、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NICとサーバーNICはVM作成後に作成され、アタッチされます。NICネットワークセキュリティグループについては、次のいずれかを実行できます。

- **詳細設定**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:

VMの展開が完了した後も、ネットワークセキュリティグループの設定を変更できます。

「仮想マシン - ネットワークタブ」 (/en-us/vpx/media/alb-floating-ip-subnets-nsg.png)

5. 「次へ: **確認 + 作成** >」をクリックします。

検証が成功したら、基本設定、VM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。

「仮想マシンの検証に成功しました」 (/en-us/vpx/media/alb-floating-ip-validation-successful.png)

6. 展開が完了したら、**リソースに移動**をクリックして構成の詳細を確認します。

「リソースに移動」 (/en-us/vpx/media/alb-floating-ip-go-to-resource.png)

同様に、2番目のNetScaler VPXインスタンスを展開します。

ステップ2. 両方のインスタンスにクライアントNICとサーバーNICを追加します。

注:

NICを追加するには、まずVMを停止する必要があります。Azureポータルで、停止したい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**停止済み**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は、**詳細設定**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**を選択して新しいものを作成できます。



サーバー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、サーバーNIC）がすべて接続されています。

セカンダリインスタンスにNICを追加するには、上記の手順を繰り返します。

両方のインスタンスでNICを作成して接続したら、**概要 > 開始**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の受信規則でポートを介したトラフィックを許可する必要があります。これは、NetScaler VPXインスタンスを構成する際にロードバランシング仮想サーバーを作成するために後で使用されます。

ステップ3. フローティングIPモードが無効になっているロードバランシングルールを持つALBを展開します。

ALB の構成を開始するには、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展開します。



注:

NetScaler VM にアタッチされているすべてのパブリック IP は、ALB と同じ SKU を持っている必要があります。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 構成**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 VM のクライアント NIC を追加します。



5. **受信規則**タブで、**ロードバランシング規則の追加**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。既存のヘルスプローブを作成または使用します。フローティング IP オプションは**無効**に設定する必要があります。



6. 「**確認と作成**」をクリックします。検証が完了したら、「**作成**」をクリックします。

ロードバランサーの検証が完了しました(/en-us/vpx/media/alb-floating-ip-lb-validation-passed.png)

ステップ 4. NetScaler GUIを使用して、両方のNetScaler VPXインスタンスでHA設定を構成します。

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。

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設定します。

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. 「**Configuration > System > High Availability > Nodes**」に移動し、「**Add**」をクリックします。
3. 「**Remote Node IP address**」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. 「**Turn on INC (Independent Network Configuration) mode on self node**」チェックボックスをオンにします。
5. 「**作成**」をクリックします。

高可用性プライマリインスタンス(/en-us/vpx/media/alb-floating-ip-ha-primary-node.png)

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. 「**Configuration > System > High Availability > Nodes**」に移動し、「**Add**」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、プライマリインスタンスの管理NICのプライベートIPアドレスを入力します (例: 10.4.1.4)。
4. 「自己ノードで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. 「作成」をクリックします。

高可用性セカンダリインスタンス(/en-us/vpx/media/alb-floating-ip-ha-secondary-node.png)

続行する前に、「ノード」ページでセカンダリインスタンスの「同期状態」が「成功」と表示されていることを確認してください。

「

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資格情報を持つことになります。

高可用性同期状態(/en-us/vpx/media/alb-floating-ip-ha-sync-state.png)

ステップ2. 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. 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. 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用に構成されたネットマスクを入力します。
 - b. 「IPタイプ」フィールドで、ドロップダウンメニューから「仮想IP」を選択します。
 - c. 「作成」をクリックします。
3. プライマリSNIPアドレスを追加するには、次の手順に従います。
 - a. プライマリインスタンスのサーバー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ーサブネット用に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。

4. セカンダリVIPアドレスを追加するには、次の手順に従います。

- 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用に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。

プライマリVIPとSNIPを追加(/en-us/vpx/media/alb-floating-ip-primary-vip-snip.png)

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 > IPs > IPv4s**に移動し、**Add**をクリックします。

2. セカンダリVIPアドレスを追加するには、次の手順に従います。

- 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用に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. セカンダリSNIPアドレスを追加するには、次の手順に従います。

- a. セカンダリインスタンスのサーバー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ーサブネット用に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。

セカンダリVIPとSNIPを追加(/en-us/vpx/media/alb-floating-ip-secondary-vip-snip.png)

ステップ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. **System > Network > IP Sets > Add** に移動します。
2. IPセット名を追加し、「**Insert**」をクリックします。
3. 「**IPv4s**」ページから、仮想IP（セカンダリVIP）を選択し、「**Insert**」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。

プライマリIPセットを作成(/en-us/vpx/media/alb-floating-ip-create-ip-set.png)

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. IPセット名を追加し、「**Insert**」をクリックします。
3. 「**IPv4s**」ページから、仮想IP（セカンダリVIP）を選択し、「**Insert**」をクリックします。
4. 「**作成**」をクリックしてIPセットを作成します。

セカンダリIPセットの作成(/en-us/vpx/media/alb-floating-ip-create-ip-set-secondary.png)

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。

ステップ4. プライマリインスタンスにロードバランシング仮想サーバーを追加します。

1. **構成 > トラフィック管理 > ロードバランシング > 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)、ポートに必要な値を入力します。
3. 「**詳細**」をクリックします。「**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「**IPセット**」を選択し、「**ステップ3**」で作成したIPセットを指定します。
4. 「**OK**」をクリックして、ロードバランシング仮想サーバーを作成します。

ロードバランシング仮想サーバーの作成(/en-us/vpx/media/alb-floating-ip-create-lb-vserver.png)

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。

1. **構成 > トラフィック管理 > ロードバランシング > 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、ポートに必要な値を入力し、「**OK**」をクリックします。

ステップ6. プライマリインスタンスで、サービスまたはサービスグループをロードバランシング仮想サーバーにバインドします。

1. **構成 > トラフィック管理 > ロードバランシング > 仮想サーバー**に移動します。
2. 手順4で構成したロードバランシング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、ロードバランシング仮想サーバーサービスバインディング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。

サービスをバインド(/en-us/vpx/media/alb-floating-ip-service-binding.png)

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あった場合に、すべての構成が失われます。

手順8. 構成を確認します。

フェイルオーバー後もALBフロントエンドIPアドレスに到達可能であることを確認してください。

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到達可能であ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 > 強制フェイルオーバー**に移動します。

強制フェイルオーバー(/en-us/vpx/media/alb-floating-ip-force-failover.png)

4. フェイルオーバー後も、以前使用したALBフロントエンドIPを介してバックエンドサーバーに到達可能であることを確認してください。

Azureアクセラレーテッド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

アクセラレーテッドネットワークは、シングルルートI/O仮想化（SR-IOV）仮想機能（VF）NICを仮想マシンに有効にし、ネットワークパフォーマンスを向上させます。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重いワークロードで使用できます。NICでアクセラレーテッド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。

アクセラレーテッドネットワークには、次の利点があります。

- 低遅延
- より高い1秒あたりのパケット数（pps）パフォーマンス
- スループットの向上
- ジッターの低減

- CPU使用率の低下

注：

Azure アクセラレーテッドネットワークは、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サイズがAzure アクセラレーテッドネットワークの要件を満たしていることを確認してください。
- いずれかのNICでアクセラレーテッドネットワークを有効にする前に、VM（個別または可用性セット内）を停止してください。

制限事項

アクセラレーテッドネットワークは、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、[サポートされているインスタンスタイプ](#)を参照してください。

高速ネットワークでサポートされるNIC

Azure は、高速ネットワーク用に SR-IOV モードの Mellanox ConnectX3、ConnectX4、および ConnectX5 NIC を提供します。

NetScaler VPX インターフェイスで高速ネットワーク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 インターフェイスとバンドルします。

注:

NetScaler VPX は、リリース 13.1 ビルド 37.x 以降で ConnectX5 NIC をサポートしています。

インターフェイスを VM に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、「[高速ネットワークでネットワークインターフェイスを作成する](#)」を参照してください。

VM 上の既存のインターフェイスで高速ネットワークを有効にする方法の詳細については、「[VM 上の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 アベイラビリティセットまたはアベイラビリティ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 portal](#) にログインし、**Azure Marketplace** に移動します。



2. **Azure Marketplace** から、**NetScaler** を検索します。



3. ライセンスとともに非 FIPS 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。「**パラメーター**」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。



5. 「**次へ: VM 構成 >**」をクリックします。

「**VM 構成**」ページで、以下を実行します。

- a. パブリック IP ドメイン名サフィックスを構成します。
- b. 「**Azure Monitoring Metrics**」を有効または無効にします。
- c. 「**Backend Autoscale™**」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。

「**ネットワークと追加設定**」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。

「**高速ネットワーク**」セクションでは、管理インターフェース、クライアントインターフェース、およびサーバーインターフェースに対して、高速ネットワーク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: 確認 + 作成 >」をクリックします。

検証が成功した後、基本設定、VM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。必要な構成でAzureリソースグループが作成されるまで、しばらく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**を選択して構成の詳細を確認します。



9. アクセラレーテッドネットワークの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。アクセラレーテッドネットワークのステータスは、各NICに対して**有効**または**無効**として表示されます。



Azure PowerShellを使用してアクセラレーテッドネットワークを有効にする

VM作成後にアクセラレーテッドネットワークを有効にする必要がある場合は、Azure PowerShellを使用して有効にできます。

注:

Azure PowerShellを使用してアクセラレーテッドネットワークを有効にする前に、VMを停止してください。

Azure PowerShellを使用してアクセラレーテッドネットワークを有効にするには、以下の手順を実行します。

1. **Azure portal**に移動し、右上の**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>
```

アクセラレーテッドネットワークのパラメータは、次のいずれかの値を受け入れます。

- **True:** 指定されたNICでアクセラレーテッドネットワークを有効にします。
- **False:** 指定されたNICでアクセラレーテッドネットワークを無効にします。

特定のNICでアクセラレーテッドネットワークを有効にするには:

```
az network nic update --name citrix-adc-vpx-nic01-0 --accelerated-
networking true --resource-group rsgp1-aan
```

特定のNICでアクセラレーテッドネットワークを無効にするには:

```
az network nic update --name citrix-adc-vpx-nic01-0 --accelerated-
networking false --resource-group rsgp1-aan
```

3. デプロイ完了後のアクセラレーテッドネットワークのステータスを確認するには、**VM > Networking** に移動します。

次の例では、アクセラレーテッドネットワークが **有効** になっていることがわかります。



次の例では、アクセラレーテッドネットワークが **無効** になっていることがわかります。



NetScalerのFreeBSDシェルを使用してインターフェース上のアクセラレーテッドネットワークを確認するには

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テッド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」コマンド出力です。「50/n」はMellanox ConnectX3 NICのVFインターフェースを示します。0/1と1/1はNetScaler VPXインスタンスのPVインターフェースを示します。PVインターフェース (1/1) とCX3 VFインターフェース (50/1) の両方が同じMACアドレス (00:22:48:1c:99:3e) を持っていることがわかります。これは、2つのインターフェースがバンドルされていることを示しています。

**ConnectX4 NICの例:**

次の例は、Mellanox ConnectX4 NICの「ifconfig」コマンド出力です。「100/n」はMellanox ConnectX4 NICのVFインターフェースを示します。0/1、1/1、および1/2はNetScaler VPXインスタンスのPVインターフェースを示します。PVインターフェース（1/1）とCX4 VFインターフェース（100/1）の両方が同じMACアドレス（00:0d:3a:9b:f2:1d）を持っていることがわかります。これは、2つのインターフェースがバンドルされていることを示しています。同様に、PVインターフェース（1/2）とCX4 VFインターフェース（100/2）も同じMACアドレス（00:0d:3a:1e:d2:23）を持っています。

**ADC CLIを使用してインターフェースで高速ネットワークを検証するには****ConnectX3 NICの例:**

次のshow interfaceコマンド出力は、PVインターフェース1/1が仮想機能50/1（SR-IOV VF NIC）と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MACアドレスは同じです。高速ネットワークが有効になった後、1/1インターフェースのデータは、ConnectX3インターフェースである50/1インターフェースのデータパスを介して送信されます。PVインターフェース（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インターフェース（50/1）の「show interface」出力はPVインターフェース（1/1）を指しています。

**ConnectX4 NICの例:**

次のshow interfaceコマンド出力は、PVインターフェース1/1が仮想機能100/1（SR-IOV VF NIC）と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MACアドレスは同じです。高速ネットワークが有効になった後、1/1インターフェースのデータは、ConnectX4インターフェースである100/1インターフェースのデータパスを介して送信されます。PVインターフェース（1/1）の「show interface」出力がVF（100/1）を指していることがわかります。同様に、VFインターフェース（100/1）の「show interface」出力はPVインターフェース（1/1）を指しています。

**NetScalerで注意すべき点**

- PVインターフェースは、必要なすべての操作のプライマリまたはメインインターフェースと見なされます。設定はPVインターフェースでのみ実行する必要があります。
- VFインターフェースに対するすべての「set」操作は、以下を除いてブロックされます。

- インターフェースを有効にする
- インターフェースを無効にする
- インターフェースをリセットする
- 統計をクリアする

注:

Citrix®は、VFインターフェースに対して操作を実行しないことを推奨します。

- `show interface` コマンドを使用して、PVインターフェースとVFインターフェース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における動的なNICの削除と、削除されたNICの再アタッチ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活動のために、アクセラレーテッドネットワークのSR-IOV VF NICを削除する場合があります。Azure VMからNICが削除されると、NetScaler VPXインスタンスはインターフェースの状態を「リンクダウン」と表示し、トラフィックは仮想インターフェースのみを通過します。削除されたNICが再アタッチされると、VPXインスタンスは再アタッチされた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インターフェースがVLANにバインドされている場合、関連付けられたアクセラレーテッドVFインターフェースもPVインターフェースと同じVLANにバインドされます。この例では、PVインターフェース（1/1）がVLAN（20）にバインドされています。PVインターフェース（1/1）とバンドルされているVFインターフェース（100/1）もVLAN 20にバインドされています。

例:

1. VLANを作成します。

```
add vlan 20
```

2. VLANをPVインターフェー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 : L0/1
```

```
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:

アクセラレーテッドVFインターフェースでは、VLANバインディング操作は許可されていません。

```
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

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モードのVPXインスタンスペアを迅速かつ効率的に展開できます。図1に示すように、Azure内部ロードバランサー(ILB)は、フロントエンドに内部またはプライベートIPアドレスを使用します。このテンプレートは、3つのサブネットと6つのNICを持つ2つのノードを作成します。サブネットは管理、クライアント、サーバー側のトラフィック用であり、各サブネットは各デバイスの異なるNICに属します。

図1: 内部ネットワーク内のクライアント向けNetScaler HAペア



図2に示すように、NetScaler HAペアがファイアウォールの背後にある場合にも、この展開を使用できます。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を備えたNetScaler HAペア



イントラネットアプリケーション用のNetScaler HAペアテンプレートは、[Azure portal](#)で入手できます。

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ペアを展開するには、次の手順を実行します。

1. Azureポータルから、**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。**パラメーター**タブで、リージョン、管理者ユーザー名、管理者パスワード、ライセンスの種類 (**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 : レビュー + 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、しばらく時間がかかる場合があります。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およびADC-VPX-1として表示されます。

HAセットアップで、より多くのセキュリティルールやポートの作成などの追加の変更が必要な場合は、Azureポータルから実行できます。

必要な構成が完了すると、以下のリソースが作成されます。

HA ILB リソースグループ(/en-us/vpx/media/ha-ilb-resource-group.png)

4. **ADC-VPX-0** および **ADC-VPX-1** ノードにログオンして、以下の構成を検証します。

- 両方のノードの NSIP アドレスは、管理サブネット内にある必要があります。
- プライマリ (ADC-VPX-0) およびセカンダリ (ADC-VPX-1) ノードには、2つの SNIP アドレスが表示されるはずです。1つの 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-0 および ADC-VPX-1 VM の SNIP アドレスは、同じサブネット内であっても異なります。これは、両方が同じである従来のオンプレミス ADC HA 展開とは異なります。VPX ペアの SNIP が異なるサブネットにある展開、または VIP が SNIP と同じサブネットにない場合は常に、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) で

プライマリノードでの IP CLI の表示(/en-us/vpx/media/primary-node1.png)

プライマリノードでの高可用性ノード CLI の表示(/en-us/vpx/media/primary-node2.png)

セカンダリノード (ADC-VPX-1) で

セカンダリノードでの IP CLI の表示(/en-us/vpx/media/secondary-node1.png)

セカンダリノードでの高可用性ノード CLI の表示(/en-us/vpx/media/secondary-node2.png)

5. プライマリノードとセカンダリノードが起動し、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) で、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) アドレスを使用して、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、「[サンプル構成](#)」セクションを参照してください。
 6. ADC Azure ロードバランサーのプライベート 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** の構成ページでは、ARMテンプレートのデプロイにより、LBルール、バックエンドプール、およびヘルスプローブが作成されます。



- LBルール (LbRule1) は、デフォルトでポート80を使用します。



- ルールを編集してポート443を使用するようにし、変更を保存します。

注記：

セキュリティ強化のため、Citrix®はLB仮想サーバーまたはゲートウェイ仮想サーバーにSSLポート443を使用することを推奨します。



ADCにVIPアドレスを追加するには、次の手順を実行します。

1. **Azure Load Balancer > Frontend IP configuration** 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. **フロントエンドIPアドレスの追加** 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、動的または静的IPアドレスを割り当てて、**追加** 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。新しいロードバランシングルールを作成し、それをフロントエンドIPアドレスに関連付けます。



4. **Azure Load Balancer** 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構成** に適用されたLBルールが表示されます。



構成例

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構成するには、プライマリノード（ADC-VPX-0）で次のコマンドを実行します。構成はセカンダリノード（ADC-VPX-1）に自動的に同期されます。

ゲートウェイ構成例

```
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
```

負荷分散構成例

```
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
```

これで、ILBの内部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、負荷分散またはVPN仮想サーバーにアクセスできます。

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:

以下のリンクは、HA展開と仮想サーバー構成に関する追加情報を提供します。

- [異なるサブネットでの高可用性ノードの構成](#)
- [基本的な負荷分散のセットアップ](#)

関連リソース:

- [PowerShellコマンドを使用した複数のIPアドレスとNICによる高可用性セットアップの構成](#)
- [Azureでのアクティブ/スタンバイHA展開におけるGSLBの構成](#)

インターネットに公開されるアプリケーション向け 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

インターネットに公開されるアプリケーション向けの標準テンプレートを使用すると、HA-INC モードでVPXインスタンスのペアを迅速かつ効率的に展開できます。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。このテンプレートは、3つのサブネットと6つのNICを持つ2つのノードを作成します。サブネットは、管理、クライアント、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。

インターネットに公開されるアプリケーション向けのNetScaler HAペアテンプレートは、[Azure Marketplace](#)で入手できます。

Azure可用性セットまたは可用性ゾーンを使用してテンプレートを起動し、高可用性VPXペアを展開するには、次の手順を実行します。

1. Azure Marketplaceで「**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展開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。「**パラメーター**」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。



5. 「**次へ : VM構成 >**」をクリックします。



6. 「**VM構成**」ページで、次を実行します。

- パブリックIPドメイン名サフィックスを構成します
- **Azure Monitoring Metrics** を有効または無効にします。
- **Backend Autoscale™** を有効または無効にします。

7. **次へ: ネットワークと追加設定 >** をクリックします。

ネットワークと追加設定 (/en-us/vpx/media/network-and-additional-settings-alb-citrix.png)

8. **ネットワークと追加設定**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。

ブート診断 (/en-us/vpx/media/boot-diagnostics-citrix.png)

9. 次へ: **確認と作成** > をクリックします。
10. 基本設定、VM構成、ネットワークと追加設定を確認し、**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、しばらく時間がかかる場合があります。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確認します。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。

HAセットアップで、追加のセキュリティルールやポートの作成など、さらに変更が必要な場合は、Azureポータルから実行できます。

必要な構成が完了すると、以下のリソースが作成されます。

HAインターネットアプリケーションリソースグループ (/en-us/vpx/media/ha-internet-application-resource-group.png)

11. **citrix-adc-vpx-0** と **citrix-adc-vpx-1** ノードにログオンして、以下の構成を検証する必要があります。

- 両方のノードのNSIPアドレスは、管理サブネット内にある必要があります。
- プライマリ (citrix-adc-vpx-0) およびセカンダリ (citrix-adc-vpx-1) ノードでは、2つのSNIPアドレスが表示されるはずですが、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との通信に使用されます。

注:

HA-INCモードでは、従来のオンプレミスADC高可用性展開では両方とも同じであるのとは異なり、citrix-adc-vpx-0 と citrix-adc-vpx-1 VM のSNIPアドレスは異なります。

プライマリノード (citrix-adc-vpx-0) で



セカンダリノード (citrix-adc-vpx-1) で



12. プライマリノードとセカンダリノードがUPし、同期ステータスが**SUCCESS**になったら、ALB 仮想サーバーのパブリックIPアドレスを使用して、プライマリノード (citrix-adc-vpx-0) でロードバランシング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、「[サンプル構成](#)」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 > フロントエンドIP構成**に移動します。

14. 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、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追加します。

15. アクセスしたいALBポートを構成し、指定され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、ロードバランシング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、ロードバランシング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仮想サーバーとロードバランシング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。

ゲートウェイ設定例

```
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
```

ロードバランシング設定例

```
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
```

これで、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、ロードバランシング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。

ロードバランシング仮想サーバーの設定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:

次のリンクは、HA展開と仮想サーバー構成に関連する追加情報を提供します。

- [仮想サーバーの作成](#)
- [基本的なロードバランシングのセットアップ](#)

Azureの外部および内部ロードバランサーを同時に使用して高可用性セットアップを構成する

Azure上の高可用性ペアは、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の外部および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、以下の2つのオプションがあります。

- NetScaler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仮想サーバーは、IPセット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。

外部および内部ロードバランサーの両方を同時に使用してAzure上で高可用性ペアを構成するには、以下の手順を実行します。

ステップ1と2ではAzureポータルを使用します。ステップ3と4ではNetScaler VPX GUIまたはCLIを使用します。

ステップ1. Azureロードバランサー（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）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セットアップの構成の詳細については、[複数のIPアドレスとNICを使用した高可用性セットアップの構成](#)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、[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](#)を参照してください。

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で外部ロードバランサーを作成した場合は、ここで内部ロードバランサーを作成し、その逆も同様です。

- 内部ロードバランサーを作成するには、ロードバランサーの種類として**Internal**を選択します。**サブネットフィールド**には、NetScaler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アドレスを提供することを選択できます。そうでない場合は、動的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、ロードバランサーの種類として**Public**を選択し、ここでパブリックIPアドレスを作成します。



- Azure Load Balancerを作成したら、**フロントエンド IP 構成**に移動し、ここに表示されているIPアドレスをメモします。このIPアドレスは、手順3でADC 負荷分散仮想サーバーを作成する際に使用する必要があります。

フロントエンド IP 構成(/en-us/vpx/media/frontend-ip-configuration.png)

- Azure Load Balancer 構成**ページでは、ARM テンプレートの展開により、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が支援されます。
- 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。
- ヘルスプローブを作成します (TCP、ポート 9000)。
- 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 (Web アプリケーションのユースケース) 用の LB ルールを 1 つ。このルール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 トラフィック用の別の LB ルール。プロセスは HTTP トラフィックと同じです。

手順 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:

手順 2 で作成した追加のロードバランサーに関連付けられている、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セットアップ](#)を参照してください。

ステップ 4: ステップ 3 の代替として、IPセットを使用してILB用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IPセットを追加します。

```
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仮想サーバーがIPセット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 Solution に NetScaler VPX インスタンスをインストールする

Azure VMware Solution (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 クラスターを含むプライベートクラウドを提供します。最小初期デプロイは3つのホストですが、追加のホストは一度に1つずつ、クラスターあたり最大16ホストまで追加できます。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、および NSX-T が含まれています。

Azure 上の VMware Cloud (VMC) を使用すると、必要な数の ESX ホストで 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 の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ーフェイスを提供します。これは ESX ベースの NetScaler VPX デプロイ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 Solution を示しています。管理者は、Azure VMware Solution 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、Windows Jumpbox から AVS の Web ベースの vCenter および NSX-T Manager 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 VM 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 ホストクラスター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 Jumpbox 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 Solution の外部にあるが同じ 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 Solution 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、NetScaler VPX インスタンスをインターネット経由でサービスを提供するように構成できます。

AVS アーキテクチャ(/en-us/vpx/media/avs-architecture.png)

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、以下を実行してください。

- Azure VMware Solution とその前提条件の詳細については、[Azure VMware Solution documentation](#) を参照してください。
- Azure VMware Solution のデプロイの詳細については、[Deploy an Azure VMware Solution private cloud](#) を参照してください。
- Azure VMware Solution にアクセスして管理するための Windows Jumpbox VM の作成の詳細については、[Access an Azure VMware Solution private cloud](#) を参照してください。
- Windows Jumpbox VM 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、[Add a network segment in Azure VMware Solution](#) を参照してください。
- VPX ライセンスファイルを取得します。

- 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仮想コンピューティングリソース

コンポーネント	要件
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 ハードウェアがバージョン 7 以降にアップグレードされている場合、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。
ディスク容量	20 GB

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です。

VPX仮想アプライアンスを実稼働環境で使用する場合、全メモリ割り当てを予約する必要があります。

OVF Tool 1.0 のシステム要件

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 Tool 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 Tool のインストールに関するシステム要件

コンポーネント	要件
オペレーティングシステム	VMware の詳細な要件については、 http://kb.vmware.com/ で「OVF Tool

コンポーネント	要件
	User Guide」 PDF ファイルを検索してください。
CPU	750 MHz 以上、1 GHz 以上を推奨
RAM	1 GB 以上、2 GB を推奨
NIC	100 Mbps 以上の NIC

OVF のインストールに関する情報については、<http://kb.vmware.com/> で「OVF Tool User Guide」 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 NetScaler VPX インスタンスセットアップパッケージは、Open Virtual Machine (OVF) 形式標準に準拠しています。ファイルはCitrixのウェブ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てください。「新規ユーザー」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てください。

ログイン後、Citrix®のホームページから次のパスに移動します。

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てください。

- NSVPX-ESX-**<リリース番号>**-**<ビルド番号>**-disk1.vmdk (例: NSVPX-ESX-13.0-79.64-disk1.vmdk)
- NSVPX-ESX-**<release number>**-**<build number>**.ovf (例: NSVPX-ESX-13.0-79.64.ovf)
- NSVPX-ESX-**<リリース番号>**-**<ビルド番号>**.mf (例: NSVPX-ESX-13.0-79.64.mf)

Azure VMware Solution をデプロイする

1. [Microsoft Azure portal](#) にログインし、**Azure Marketplace** に移動します。
2. **Azure Marketplace** から、**Azure VMware Solution** 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ーを作成するには、最低3つのESXiホストを選択します。
- **アドレスブロック** 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**については、CIDR範囲がオンプレミスまたは他のAzureサブネット（仮想ネットワーク）、あるいはゲートウェイサブネットと重複しないようにしてください。
- 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をExpressRouteでルーティングするために使用されます。

プライベートクラウドを作成する(/en-us/vpx/media/avs-create-private-cloud.png)

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、**戻る**をクリックします。

プライベートクラウドの設定を確認する(/en-us/vpx/media/review-private-cloud-settings.png)

6. **作成**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時間かかる場合があります。

プライベートクラウドをプロビジョニングする(/en-us/vpx/media/provision-private-cloud.png)

7. 作成されたプライベートクラウドを確認するには、**リソースに移動**をクリックします。

AVSプライベートクラウド(/en-us/vpx/media/avs-private-cloud.png)

注記

このリソースにアクセスするには、ジャンプボックスとして機能するWindowsのVM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Azure仮想マシン (VM) をデプロイする方法を示します。VMが動作していることを確認するには、VMにRDP接続し、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。

AVS リソースの作成(/en-us/vpx/media/avs-create-resource.png)

2. **Microsoft Windows 10**を検索し、**作成**をクリックします。

Microsoft Windows 10 の作成(/en-us/vpx/media/avs-ms-windows10.png)

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選択します。残りのデフォルト設定はそのままにし、ページの最下部にある**確認 + 作成**ボタンを選択します。

仮想マシンの作成(/en-us/vpx/media/avs-create-vm.png)

4. 検証が実行された後、ページの最下部にある**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 VM に移動します。Windows VM のパブリック IP アドレスを使用して、RDP で接続します。

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、その後サインオンします。

Mac から Windows VM に接続するには、Microsoft Remote Desktop 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 を実行している 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 Solution プライベートクラウドで、**管理**の下にある**ID**を選択します。vCenter 資格情報をメモしておきます。

vCenter 資格情報(/en-us/vpx/media/vcenter-credentials.png)

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 Solution プライベートクラウドの vCenter 資格情報を使用して、VMware vSphere にログインします。



4. vSphere クライアントで、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 ポータルにアクセスする](#)」を参照してください。

AzureポータルでNSX-Tセグメントを作成する

AzureポータルのAzure VMware SolutionコンソールからNSX-Tセグメントを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ており、これらのセグメント上のワークロードは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とvCenterに表示されます。

1. Azure VMware Solutionプライベートクラウドで、**ワークロードネットワーク**の下にある**セグメント > 追加**を選択します。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ース用に3つの個別のセグメントを作成できます。

2. Azure VMware Solutionプライベートクラウドで、**管理**の下にある**ID**を選択します。NSX-T Managerの資格情報をメモしておきます。

3. NSX-T WebクライアントURLを入力して、VMware NSX-T Managerを起動します。

4. NSX-T Managerで、**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。サブネットも確認できます。


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 Cloud に NetScaler VPX インスタンスをインストールする

VMware Software-Defined Data Center (SDDC) をインストールして構成したら、SDDC を使用して VMware Cloud 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 で利用可能なメモリ量によって異なります。

VMware Cloud に NetScaler VPX インスタンスをインストールするには、Windows Jumpbox VM で次の手順を実行します。

1. NetScaler ダウンロードサイトから、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ダウンロードします。
2. Windows Jumpbox で VMware SDDC を開きます。
3. **ユーザー名とパスワード**のフィールドに管理者資格情報を入力し、**ログイン**をクリックします。
4. **ファイルメニュー**で、**OVF テンプレートの展開**をクリックします。

5. **OVF テンプレートの展開**ダイアログボックスの**ファイルから展開**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へ移動し、.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、OVF を変更して E1000 の代わりに VMXNET3 インターフェイスを使用します。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ており、Azure VMware Solution 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ているネットワークを、VMware SDDC で構成したネットワークにマッピングします。**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NetScaler VPXインスタンスを選択し、右クリックメニューから**Power On**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. vSphereクライアントからNetScaler VMに接続されました。



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. `show ns ip` コマンドを使用してADC構成を検証できます。



Azure VMware Solution で NetScaler VPX スタンドアロンインスタンスを構成する

インターネットに接続するアプリケーション向けに、Azure VMware Solution (AVS) で 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 Solution 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。これは、AVS の外部で、ただし同じ Azure 仮想ネットワーク内で 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プロビジョニングすることで実現できます。ロードバランサーを構成して、AVS サービス内の NetScaler VPX インスタンスの VIP にアクセスします。



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 Solution とその前提条件の詳細については、「[Azure VMware Solution のドキュメント](#)」を参照してください。
- Azure VMware Solution の展開の詳細については、「[Azure VMware Solution プライベートクラウドの展開](#)」を参照してください。
- Azure VMware Solution にアクセスして管理するための Windows Jump Box VM の作成の詳細については、「[Azure VMware Solution 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 Jump Box VM 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、「[Azure VMware Solution でネットワークセグメントを追加する](#)」を参照してください。
- VMware Cloud 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 Cloud 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 上に NetScaler VPX スタンドアロンインスタンスを構成する

NetScaler ロードバランサーを使用して、インターネットに接続するアプリケーション向けに AVS 上の NetScaler VPX スタンドアロンインスタンスを構成するには、次の手順に従います。

1. Azure クラウドに NetScaler VPX インスタンスを展開します。詳細については、「[NetScaler VPX スタンドアロンインスタンスを構成する](#)」を参照してください。

注記:

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てください。

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 の VIP 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

Azure ロードバランサーを使用して、インターネットに接続するアプリケーション向けに AVS 上の NetScaler VPX スタンドアロンインスタンスを構成するには、次の手順に従います。

1. Azure クラウドで Azure Load Balancer インスタンスを構成します。詳細については、[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アドレスをロードバランサーの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>
```

注記:

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てください。

Azure VMware Solution 上で NetScaler VPX 高可用性セットアップを構成する

インターネットに公開されるアプリケーション向けに、Azure VMware Solution (AVS) 上で 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。これは、AVS の外部で、ただし同じ Azure 仮想ネットワーク内に 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プロビジョニングすることで実現できます。ロードバランサーを構成して、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します。

AVS 上の VPX HA デプロイメント(/en-us/vpx/media/vpx-ha-on-avs.png)

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、以下の Azure の前提条件をお読みください。

- Azure VMware Solution とその前提条件の詳細については、[Azure VMware Solution ドキュメント](#)を参照してください。
- Azure VMware Solution のデプロイの詳細については、[Azure VMware Solution プライベートクラウドのデプロイ](#)を参照してください。
- Azure VMware Solution にアクセスして管理するための Windows Jump box VM の作成の詳細については、[Azure VMware Solution 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 Jump box VM 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、[Azure VMware Solution でネットワークセグメントを追加する](#)を参照してください。

構成手順

インターネットに公開されるアプリケーション向けに、AVS で NetScaler VPX 高可用性セットアップを構成するには、以下の手順に従ってください。

1. VMware Cloud 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 Cloud 上に 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 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するアプリケーションから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。

NetScaler VPX HAペアでAzureルートサーバーを構成する

AzureルートサーバーをNetScaler VPXインスタンスと構成して、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。NetScaler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てい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アドレスを使用しています。

プライマリADCインスタンスのIP構成例:

```
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構成例:

```
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

AzureにNetScaler VPXインスタンスを展開する前に、次の情報に精通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の用語](#)」を参照してください。
- Azure Route Serverの概要。詳細については、「[Azure Route Serverとは](#)」を参照してください。
- NetScalerアプライアンスの動作。詳細については、「[NetScalerのドキュメント](#)」を参照してください。
- NetScalerのネットワーク。詳細については、「[ADC Networking](#)」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ポータルを使用してルートサーバーを作成および構成する](#)」を参照してください。

以下の例では、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（例：10.0.3.4、10.0.3.5）を取得します。

ルートサーバーの作成(/en-us/vpx/media/azure-route-server.png)

2. Azureポータル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。

以下の例では、ピアを追加する際に、1/1インターフェース上のADC SNIP (10.0.1.4および10.0.1.5) とASN (400および500) が使用されます。

ピアの設定(/en-us/vpx/media/ars-peers.png)

3. HA構成用に2つのNetScaler VPXインスタンスを追加します。

以下の手順を完了します。

- a. 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。
- b. 両方のインスタンスにクライアントNICとサーバーNICを追加します。
- c. NetScaler GUIを使用して、両方のインスタンスでHA設定を構成します。

4. プライマリADCインスタンスで動的ルーティングを構成します。

構成例:

```
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
```

5. セカンダリADCインスタンスで動的ルーティングを構成します。

構成例:

```
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
```

6. VTYシェルインターフェースで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構成の検証](#)を参照してください。

```
show ip bgp neighbors
```

7. 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は、LB仮想サーバーにアクセスできるようになります。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。

Azure Autoscale™ 設定を追加する

クラウドでのアプリケーションの効率的なホスティングには、アプリケーションの需要に応じてリソースを簡単かつ費用対効果の高い方法で管理することが含まれます。需要の増加に対応するには、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。需要が減少した場合は、アイドル状態のリソースによる不要なコストを回避す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 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面倒です。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と、必要に応じたリソースのスケールアップおよびスケールダウンのプロセスを自動化する必要があります。

Azure 上での VPX マルチIPスタンドアロンおよび高可用性展開には、Azure 仮想マシン スケールセット (VMSS) と Autoscale を使用できます。

Azure VMSS および Autoscale 機能と統合された NetScaler VPX インスタンスは、次の利点を提供します。

- 負荷分散と管理: 需要に応じてサーバーを自動的にスケールアップおよびスケールダウンするように構成します。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にある VMSS Autoscale 設定を自動検出します。負荷を分散するために VMSS Autoscale 設定を選択できます。これは、VPX インスタンス上で 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。

VPX Azure オートスケール(/en-us/vpx/media/vpx-azure-auto-scale.png)

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 スケール セットのドキュメント](#)
- Microsoft Azure の仮想マシン、クラウドサービス、および Web アプリにおけるオートスケールの概要(<https://docs.microsoft.com/ja-jp/azure/monitoring-and-diagnostics/monitoring-overview-autoscale>)

始める前に

1. Azure 関連の使用ガイドラインをお読みください。詳細については、「[Microsoft Azure に NetScaler VPX インスタンスを展開する](#)」を参照してください。
2. 要件 (スタンドアロンまたは高可用性展開) 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持つ 1 つ以上の NetScaler VPX インスタンスを作成します。
3. 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信します。
4. NetScaler VPXインスタンスがデプロイされているのと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デプロイされている場合、以下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は同じAzureサブスクリプション内にある必要があります。
 - 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されている必要があります。

既存のVMSS構成がない場合は、次のタスクを完了してください。

- a) VMSSを作成します
- b) VMSSでAutoscaleを有効にします
- c) VMSS Autoscale設定でスケールインおよびスケールアウトポリシーを作成します

詳細については、[Azure仮想マシン スケール セットでのAutoscaleの概要](#)を参照してください。

- リソースにアクセスできるAzure Active 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。新しく作成したAADアプリケーションに共同作成者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アプリケーションとサービスプリンシパルを作成する](#)を参照してください。

NetScaler VPXインスタンスにVMSSを追加する

GUIを使用して、ワンクリックでAutoscale設定をVPXインスタンスに追加できます。VPXインスタンスにAutoscale設定を追加するには、次の手順を完了してください。

- VPXインスタンスにログオンします。
- NetScaler VPXインスタンスに初めてログオンすると、「資格情報の設定」ページが表示されます。Autoscale機能が動作するために必要なAzure資格情報を追加します。



- デフォルトのクラウドプロファイルページで、次の例に示す詳細を入力し、「作成」をクリックします。



クラウドプロファイル作成時の留意事項

- 仮想サーバーのIPアドレスは、VPXインスタンスで利用可能な空きIPアドレスから自動入力されます。詳細については、[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で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に入力されます。詳細については、[Azure仮想マシン スケール セットでのオートスケールの概要](#)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は、サーバーがそのプロトコルとポートでリッスンしていること、およびサービスグループに正しいモニターをバインドしていることを確認してください。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オートスケーリングの場合、クラウドプロファイルを作成した後、証明書がないためロードバランス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ーまたはサービスグループに手動でバインドできます。

初回ログオン後、クラウドプロファイルを作成する場合は、GUIで **System > Azure > Cloud Profile** に移動し、**Add** 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は、NetScalerロードバランシング仮想サーバーと、オートスケーリンググループのサーバーをメンバー（サーバー）とする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、Azureで同じVMSSを使用して、異なるサービス（異なるポートを使用）に対して異なるクラウドプロファイルを作成できます。これにより、NetScaler VPXインスタンスは、パブリッククラウドで同じオート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ポータルでオートスケール関連情報を表示するには、**All service > Virtual machine scale set > Select Virtual machine scale set > Scaling** に移動します。

Azureでアンマネージドディスクからマネージドディスクに移行する

NetScaler VPXインスタンスまたはVMSSがアンマネージドディスクでデプロイされている場合、Azureクラウドの変更に対応するため、マネージドディスクへの移行が必要です。移行ガイドと関連するFAQについては、以下のAzureドキュメントを参照してください。

- [Linux仮想マシンをアンマネージドディスクからマネージドディスクに移行する](#)
- [FAQ](#)

参照

NetScaler Application Delivery and Managementを使用してMicrosoft AzureでNetScaler VPXの自動スケーリングに関する情報については、[NetScaler ADMを使用したAzure自動スケーリング](#)を参照してください。

NetScaler VPX展開のためのAzureタグ

Azureクラウドポータルでは、リソースに名前:値のペア（例: Dept: Finance）でタグ付けして、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を横断してリソースを分類し、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展開におけるAzureタグの仕組み

Azure Cloudに展開されたNetScaler VPXの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ース（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を分散する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にクエリを実行します。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持つ利用可能なリソースを取得します。適切なタグを持つVMまたはNICが追加または削除されるたびに、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から自動的に追加または削除します。

タグの仕組み(/en-us/vpx/media/vpx-azure-tags-citrix.png)

開始する前に

NetScaler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のいずれかに割り当てることができます。

Azureタグの作成(/en-us/vpx/media/azure-tags-citrix.png)

Azureタグの追加の詳細については、Microsoftドキュメント「[Azureリソースを整理するためのタグの使用](#)」を参照してください。

注:

Azureタグ設定を追加するためのADC CLIコマンドは、数字またはアルファベットで始まり、その他のキーボード文字ではないタグ名とタグ値のみをサポートします。

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できます。これにより、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。次の手順に従います。

1. VPX GUIから、**Configuration > Azure > Cloud Profile**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. 以下のフィールドに値を入力します。

- 名前: プロファイルの名前を追加します
- 仮想サーバーIPアドレス: 仮想サーバーIPアドレスは、VPXインスタンスで利用可能な空きIPアドレスから自動入力されます。詳細については、「[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VMまたはNICに割り当てた名前を入力します。
- Azureタグ値: Azureポータルで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、ポーリング期間は60秒で、これは最小値です。要件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リッスン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. 「作成」をクリックします。

タグ付けされたVMまたはNIC用に、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表示するには、VPX GUIから**Traffic Management > Load Balancing > Virtual Servers**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

Azureタグのクラウドプロファイルを作成するには、NetScaler CLIで次のコマンドを入力します。

```
add cloud profile `<profile name>` -type azuretags -vServerName `< vserver name>` -serviceType HTTP -IPAddress `<vserver IP address>` -port 80 -serviceGroupName `<service group name>` -boundServiceGroupSvcType HTTP -vsrvbindsvcpport 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トラフィック用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
-vsvrbindsvcpport 80 -azureTagName myTagName -azureTagValue myTagValue
-azurePollPeriod 60
```

Done

クラウドプロファイルを表示するには、 `show cloudprofile` と入力します。

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。

```
show cloudprofile
```

```
1)   Name:  MyTagCloudProfile Type:  azuretags           VServerName:
      MyTagVServer ServiceType:  HTTP           IPAddress:  52.178.209.133 Port:
      80           ServiceGroupName:  MyTagsServiceGroup
      BoundServiceGroupSvcType:  HTTP

      Vsvrbindsvcpport:  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で作成されたクラウドプロファイルを削除します。

```
> rm cloudprofile MyTagCloudProfile
```

```
Done
```

トラブルシューティング

問題: ごくまれに、「rm cloud profile」CLIコマンドが、削除されたクラウドプロファイルに関連付けられたサービスグループとサーバーを削除できないことがあります。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数秒前にコマンドが発行された場合に発生します。

解決策: 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。

```
#> rm servicegroup <serviceName>
```

また、残りのサーバーごとに次のCLIコマンドを入力して、残りのサーバーをそれぞれ削除します。

```
#> rm server <name>
```

問題: CLIを使用してAzureタグ設定をVPXインスタンスに追加すると、ウォームリブート後にHAペアノードでrain_tagsプロセスが実行され続けます。

解決策: ウォームリブート後、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移動します。

```
#> 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が正常であるにもかかわらず、VPXインスタンスによって到達不能と報告され、DOWN状態になることがあります。 **解決策:** 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てください。タグ付きNICの場合、これはNICのIPアドレスです。タグ付きVMの場合、これはVMのプライマリIPアドレスです。VM/NICが別のAzure VNetにある場合は、VNetピアリングが有効になっていることを確認してください。

NetScaler VPXインスタンスでGSLBを構成する

グローバルサーバー負荷分散 (GSLB) 用に構成されたNetScalerアプライアンスは、WANにおける障害点から保護することで、アプリケーションのディザスタリカバリと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に誘導するか、障害が発生した場合に稼働中のデータセンターに誘導することで、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つのサイトにある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つのステップを実行します:

1. 各サイトで複数のNICと複数のIPアドレスを持つVPXインスタンスを作成する。
2. VPXインスタンスでGSLBを有効にする。

注:

複数のNICとIPアドレスの構成の詳細については、以下を参照してください: [PowerShellコマンドを使用してスタンドアロンモードのNetScaler VPXインスタンスに複数のIPアドレスを構成する](#)

シナリオ

このシナリオには、サイト1とサイト2の2つのサイトが含まれます。各サイトには、複数のNIC、複数のIPアドレス、およびGSLBで構成されたVM (VM1とVM2) があります。

図。 2つのサイト (サイト1とサイト2) にわたって実装されたGSLBセットアップ。



このシナリオでは、各VMには、NIC 0/1、1/1、および1/2の3つのNICがあります。各NICは、複数のプライベートIPアドレスとパブリックIPアドレスを持つことができます。NICは次の目的で構成されています。

- NIC 0/1: 管理トラフィック用
- NIC 1/1: クライアント側トラフィック用
- NIC 1/2: バックエンドサーバーと通信するため

このシナリオで各NICに構成されているIPアドレスについては、[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

以下に、このドキュメントにおけるこのシナリオのパラメーター設定例を示します。

```
$location="West Central US"

$vmnetName="NSVPX-vnet"

$RGName="multiIP-RG"

$prmsStorageAccountName="multiipstorageacctnt"

$avSetName="MultiIP-avset"

$vmSize="Standard_DS3_V2"
```

注:

VPXインスタンスの最小要件は、2 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
```

VMの作成

PowerShellコマンドを使用して、複数のNICと複数のIPアドレスを持つVM1を作成するには、手順1～10に従います。

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. 可用性セットの作成
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 構成オブジェクトの作成
8. 資格情報を取得し、VM の OS プロパティを設定する
9. NIC の追加
10. OS ディスクを指定して VM を作成する

VM1 を作成するためのすべての手順とコマンドを完了した後、これらの手順を繰り返して、VM2 をそれに固有のパラメータで作成します。

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
```

可用性セットの作成

```
$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 を作成

```
$nic1Name=$nicNamePrefix + $suffixNumber + "-Mgmt"

$ipAddress1=$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 を作成

```
$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 を作成

```
$nic3Name=$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構成オブジェクトを作成

```
$vmName=$vmNamePrefix

$vmConfig=New-AzureRMVMConfig -VMName $vmName -VMSize $vmSize
-AvailabilitySetId $avSet.Id
```

資格情報を取得し、OSプロパティを設定する

```
$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を追加

```
$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ディスクを指定し、VMを作成する

```
$osDiskName=$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コマンドを使用してマルチNIC VMを作成する」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でVM2を作成します。

IP構成の詳細

以下のIPアドレスが使用されます。

表1. VM1で使用される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) として構成済み
1/1	10.0.1.10	PIP2	SNIP/GSLB サイト IP として構成済み
-	10.0.1.11	-	LB サーバー IP として構成済み。パブリック IP は必須ではありません
1/2	10.0.2.10	-	サービスへのモニタープローブ送信のためにSNIPとして構成されます。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 として構成
-	20.0.1.11	-	LBサーバーIPとして構成されています。パブリックIPは必須ではありません
1/2	20.0.2.10	-	サービスにモニタープローブを送信するためのSNIPとして構成

NIC	内部IP	パブリックIP (PIP)	説明
			成されています。パブリックIPは必須ではありません

このシナリオのサンプル構成を次に示します。これらは、VM1およびVM2用のNetScaler VPX CLIを介して作成されたIPアドレスと初期LB構成を示しています。

VM1の構成例を次に示します。

```
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の構成例を次に示します。

```
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サイトとその他の設定を構成する

以下のトピックで説明されてい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。

グローバルサーバー負荷分散

VM1およびVM2でのGSLB構成例を次に示します。

```
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上で動作するNetScaler VPXインスタンスにGSLBを設定しました。

ディザスタリカバリ

ディザスタとは、自然災害または人為的な事象によって引き起こされるビジネス機能の突然の中断です。ディザスタはデータセンターの運用に影響を与え、その後、ディザスタサイトで失われたリソースとデータは完全に再構築および復元される必要があります。データセンターでのデータの損失またはダウンタイムは致命的であり、事業継続性を崩壊させます。

今日、顧客が直面している課題の1つは、DRサイトをどこに設置するかを決定することです。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の障害に関係なく、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可能性のある理由は次のとおりです。

- 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、企業は自社システムの拡張から時間とリソースを解放できます。
- 多くの自動化されたオーケストレーションにより、より迅速なリカバリが可能になります。
- 継続的なデータ保護または継続的なスナップショットを提供することでデータをレプリケートし、あらゆる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する、顧客が多くの異なる種類のコンプライアンスおよびセキュリティ制御を必要とするユースケースをサポートします。これにより、独自のものを構築するよりも、必要なコンプライアンスを達成しやすくなります。

GSLB用に構成されたNetScalerは、最も負荷の低い、または最もパフォーマンスの高いデータセンターにトラフィックを転送します。アクティブ/アクティブ設定と呼ばれるこの構成は、パフォーマンスを向上させるだけでなく、設定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、即座のディザスタリカバリも提供します。これにより、NetScalerは顧客の貴重な時間と費用を節約します。

ディザスタリカバリのためのマルチNICマルチIP（3NIC）展開

セキュリティ、冗長性、可用性、容量、スケーラビリティが重要となる本番環境に展開する場合、顧客は3NIC展開を使用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要な懸念事項ではありません。

ディザスタリカバリのためのシングルNICマルチIP（1NIC）展開

顧客は、以下の理由により、非本番環境に展開する場合に1NIC展開を使用する可能性があります。

- テストのために環境をセットアップしている場合、または本番展開の前に新しい環境をステージングしている場合。
- クラウドに直接、迅速かつ効率的に展開する。
- 単一サブネット構成のシンプルさを求める一方で。

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

Azure上のアクティブ/スタンバイHA展開でグローバルサーバー負荷分散 (GSLB) を3つのステップで構成できます。

1. 各GSLBサイトに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を使用した高可用性セットアップの構成](#)を参照してください。

2. Azure Load Balancer (ALB) をフロントエンドIPアドレスとルールで構成し、GSLBおよびDNSトラフィックを許可します。

このステップには、以下のサブステップが含まれます。これらのサブステップを完了するために使用される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. 以下の負荷分散ルールを作成します。

```
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をステップcで作成したLBルールに関連付けます。
 - e. HAペアの両方のノードのNIC 1/1のネットワークセキュリティグループを更新し、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、サイト1とサイト2の2つのサイトが含まれます。各サイトには、複数のNIC、複数のIPアドレス、およびGSLBで構成されたHAペア (HA1とHA2) があります。

図: Azure上のアクティブ/スタンバイHA展開におけるGSLB



このシナリオでは、各VMには3つのNIC (NIC 0/1、1/1、1/2) があります。これらのNICは以下の目的で構成されています。

NIC 0/1: 管理トラフィックを処理するため

NIC 1/1: クライアント側のトラフィックを処理するため

NIC 1/2: バックエンドサーバーと通信するため

パラメータ設定

以下はALBのパラメータ設定例です。必要に応じて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、GSLBおよびDNSトラフィックを許可するルールを使用してALBを構成する

ステップ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
```

ステップ2. LBルールを作成し、既存のALBを更新する。

```
$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を有効にする

これで、各ALB（ALB 1とALB 2）に2つのフロントエンドIPアドレスがあります。1つのIPアドレスはLB仮想サーバー用で、もう1つはGSLBサイトIP用です。

HA 1には以下のフロントエンドIPアドレスがあります。

- FrontEndIPofALB1（LB仮想サーバー用）
- PIPFORGSLB1（GSLB IP）

HA 2には以下のフロントエンド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、次のコマンドが使用されます。

```
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](#)

[グローバルサーバー負荷分散](#)

Azure で NetScaler GSLB を展開する

需要の増加に伴い、地域のお客様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用している企業は、Azure クラウドを使用してグローバルに拡張および展開したいと考えています。ネットワーク管理者の側で NetScaler を使用すると、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 を使用すると、近接性に応じてオンプレミスまたはクラウドのリソースに到達できます。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が得られます。

DBS の概要

NetScaler GSLB は、クラウドロードバランサーにドメインベースサービス (DBS) を使用すること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な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/アクティブ環境で GSLB DBS を実装できます。DBS は、DNS 検出から Microsoft Azure 環境のバックエンドリソースのスケーリングを可能にします。このセクションでは、Azure Autoscale 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) を使用したドメイン名ベースのサービス

GSLB DBS は、ユーザー ALB の FQDN を使用して GSLB サービスグループを動的に更新し、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めます。この機能を構成するために、ユーザーは Citrix® ADC を ALB にポイントして、Azure 内の異なるサーバーに動的にルーティングします。これにより、Azure 内でインスタンスが作成および削除されるたびに Citrix ADC を手動で更新する必要がなくなります。GSLB サービスグループの Citrix ADC DBS 機能は、DNS 対応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。

次の図は、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、機能する Microsoft Azure 環境と、Linux Web サーバー、Azure 内の NetScaler アプライアンス、パブリック IP アドレス、および Azure ロード バランサー (ALB) を構成するための知識と能力が含まれます。

- GSLB DBS サービス統合には、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-  
xxxxxxxxxxx.australiaeast.cloudapp.azure.com  
  
> add gslb serviceGroup sydney_sg HTTP -autoscale DNS -siteName sydney  
  
> bind gslb serviceGroup sydney_sg sydney_server 80  
  
...
```

Azureコンポーネントの構成

1. ユーザーAzureポータルにログインし、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ロードバランサーを作成します。

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用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を使用して、ロードバランシング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: ステップ1で作成されたNetScaler
- ヘルスプローブ: ステップ4で作成
- セッション永続性: なし



NetScaler GSLBドメインベースサービスの構成

GSLBが有効な環境で、オートスケールするADCのドメインベースサービスを有効にするために必要な設定を以下にまとめます。

- [トラフィック管理設定](#)
- [GSLB設定](#)

トラフィック管理設定

注:

DBSサービスグループのALBドメインを解決するために、NetScalerをネームサーバーまたは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 nameServer](#)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. **Add** をクリックしてサーバーを作成し、ALBのAzureにおける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. Azureの2番目のリソースから2番目のALBを追加するには、手順2を繰り返します。

GSLB設定

1. **Add** ボタンをクリックしてGSLBサイトを設定します。
2. GSLBサイトを設定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、サイトを設定する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。特定のIPアドレスが外部ファ

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、パブリックIPアドレスが必要です。サイトを親サイトとして構成し、**Trigger Monitors** 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 **Metric Exchange**、**Network Metric Exchange**、および **Persistence Session Entry Exchange** の3つのチェックボックスを必ずオンにしてください。

Trigger monitor を **MEPDOWN** に設定することをお勧めします。詳細については、[Configure a GSLB Service Group](#)を参照してください。



3. 「作成」をクリックします。
4. 「Traffic Management > GSLB > Service Groups」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. サービスグループを設定するための詳細を指定します

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、作成した該当するサイトを選択します。オートスケールモードをDNSとして設定し、状態とヘルス監視のチェックボックスをオンにしてください。「**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。実行ガイドの冒頭で設定した該当するALBを選択します。ポート80経由でトラフィックが流れ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のバインディングには、ALBから受信した2つのインスタンスが入力されます。



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仮想サーバーを設定するには、「Traffic Management > GSLB > Virtual Servers」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. GSLB仮想サーバーを設定するための詳細を指定します。

サーバーに名前を付け、DNSレコードタイプをAに設定し、サービスタイプをHTTPに設定し、「作成後に有効にする」と「AppFlow®ロギング」の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仮想サーバーが作成されたら、**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、前の手順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なし**をクリックして、GSLB仮想サーバーのドメインバインディングを設定します。FQDNを設定してバインドします。その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. ロードバランシングサービスを設定するための詳細を指定します。

サービス名を追加し、**新しいサーバー**をクリックして、ADNSサーバーの**IPアドレス**を入力します。ユーザーADNSがすでに設定されている場合は、**既存のサーバー**を選択し、ドロップダウンメニューからユーザーADNSを選択できます。プロトコルがADNSであり、トラフィックがポート53を介して流れるように設定されていることを確認してください。



17. **メソッド**を**最小接続**に、**バックアップメソッド**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GSLB仮想サーバーが「稼働中」と表示されていることを確認します。



その他のリソース

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([/ja-jp/citrix-application-delivery-management-service/hybrid-multi-cloud-deployments/glb-solution.html](https://ja-jp.citrix-application-delivery-management-service/hybrid-multi-cloud-deployments/glb-solution.html))

NetScaler VPX インスタンスを Azure Local に展開する

NetScaler VPX™ は、Azure Local (Azure Stack ハイパーコンバージドインフラストラクチャ (HCI)) のバージョン 22H2、23H2、および 24H2 への展開をサポートしています。NetScaler VPX インスタンスをスタンドアロンの Azure Local ホストに展開するか、Azure Local インスタンス (旧 Azure Stack HCI クラスタ) で高可用性 (HA) ペアとして設定できます。Azure Local は、組織が独自のデータセンターで Azure サービスを実行できるようにするハイブリッドクラウドプラットフォームです。NetScaler VPX を Azure Local に展開することで、高度なロードバランシング、アプリケーション配信、およびセキュリティ機能がオンプレミスで提供され、Azure クラウドサービスとのシームレスな統合が保証されます。このソリューションは、NetScaler が Azure エコシステムの利点を活用してアプリケーションをローカルで最適化および保護できるため、低遅延、データ主権、または規制順守を必要とする企業にとって有益です。

NetScaler VPX インスタンスを Azure Local に展開するための前提条件

- お使いの環境が、以下の NetScaler VPX バージョンおよび Azure Local バージョンの要件を満たしていることを確認してください。

- サポートされている NetScaler VPX バージョン: 13.1-61.x 以降
- サポートされている Azure Local OS ビルド: 25398.1965、26100.7171、および 20349.3692

Azure Local のリリースバージョンに関する詳細については、[Microsoft ドキュメント](#) を参照してください。

- 専用の Azure Local ノードにアクセスできること、および Windows Admin Center がインストールされ、このノードに適切に接続されていることを確認してください。
- NetScaler 仮想ハードディスク (VHD) イメージを Citrix 公式ウェブサイトまたはライセンスされたソースからダウンロードしてください。イメージがお使いの環境と互換性があることを確認してください。

Windows Admin Center から VHD を使用して NetScaler VPX インスタンスを Azure Local に展開する方法

Windows Admin Center から VHD を使用して NetScaler VPX インスタンスを Azure Local に展開するには、次の手順に従います。

1. Windows Admin Center を使用して、NetScaler VHD ファイルを Azure Stack HCI ノードにアップロードします。

2. Windows Admin Center で新しい仮想マシン (VM) を作成し、名前、Generation 1、CPU、メモリ、仮想スイッチなどの VM 設定を構成し、アップロードした VHD を選択します。
3. 新しく作成した NetScaler VM を起動し、その起動ステータスを監視します。
4. Hyper-V マネージャーを使用して NetScaler インスタンスコンソールにアクセスし、ログインして IP アドレスを割り当てます。
5. NetScaler VPX インスタンスに SSH を使用してリモートで接続し、さらなるセットアップと管理を行います。
6. NetScaler 管理インターフェースへの接続を確認し、必要な設定を完了してデプロイメントを検証します。

ステップ1. Windows Admin Centerを使用して、NetScaler VHDファイルをAzure Localマシンにアップロードします。

1. Windows Admin Centerを開き、Azure Localマシンに接続します。
2. ツール > ファイルとファイル共有に移動します。
3. 目的のボリュームを参照し、VHDファイルをアップロードしたい特定のフォルダーに移動します。
4. アップロードをクリックし、コンピューターからVHDファイルを選択します。
5. アップロードプロセスが完了するまで待ちます。完了したら、選択したフォルダーにVHDファイルが表示されていることを確認します。

Azureファイル共有(/en-us/vpx/media/azure-local-file-sharing.png)

ステップ2. 新しい仮想マシンを作成し、VM設定を構成します。

1. Windows Admin Centerで、**仮想マシン**に移動します。
2. **追加**をクリックし、次に**新規**を選択します。
3. VMの名前を入力し、**Generation 1**を選択します。
4. NetScalerの要件とデプロイメントのニーズを満たすように、仮想プロセッサ (CPU) 、メモリ、および仮想スイッチを構成します。
5. **ストレージ設定**で、**既存の仮想ハードディスクを使用**を選択し、アップロードしたVHDイメージを選択します。
6. VMの作成を完了します。

VMの作成(/en-us/vpx/media/azure-local-create-new-vm.png)

ステップ 3. NetScaler VMを起動します。

1. 設定が完了したら、前のステップで作成したNetScaler仮想マシンを起動します。

VMの起動(/en-us/vpx/media/azure-local-start-vm.png)

2. Windows Admin CenterでVMのステータスを監視し、正常に起動することを確認します。

VMの監視(/en-us/vpx/media/azure-local-monitor-vm.png)

ステップ 4. ネットワークを構成し、Hyper-Vマネージャーを使用してリモートアクセスを設定します。

1. NetScalerインスタンスが起動したら、Hyper-Vマネージャーでコンソールを開きます。

ネットワークの構成(/en-us/vpx/media/azure-local-configure-network.png)

2. Azure LocalノードのHyper-VコンソールからNetScalerにログインします。

Hyper-Vコンソール(/en-us/vpx/media/azure-local-assign-ip-address.png)

3. IPアドレスを割り当てて保存します。
4. SSHを使用してNetScalerにリモート接続し、その他の設定と管理を行います。

Azureローカル検証(/en-us/vpx/media/azure-local-validate.png)

ステップ 5. NetScaler VPX の展開を検証します。

1. 設定されたIPアドレスを使用してNetScaler GUIまたはCLIにログインし、NetScaler VPX の構成を続行します。
2. NetScaler の構成が完了していること、およびVPXインスタンスが期待どおりに機能していることを確認します。

Azure に NetScaler® Web App Firewall を展開する

NetScaler Web App Firewall は、最新のアプリケーションに最先端の保護を提供するエンタープライズグレードのソリューションです。これは、ウェブサイト、ウェブアプリケーション、API などの公開資産に対する脅威を軽減します。IPレピュテーションベースのフィルタリング、ボット対策、OWASP Top 10 アプリケーション脅威保護、レイヤー7 DDoS保護などが含まれています。また、認証、強力なSSL/TLS暗号、TLS 1.3、レート制限、および書き換えポリシーを適用するオプションも含まれています。基本的小および高度なWAF保護の両方を使用することで、NetScaler Web App Firewall は比類のない使いやすさでアプリケーションを包括的に保護します。数分で稼働を開始できます。さらに、動的プロファイリングと呼ばれる自動学習モデルを使用することで、ユーザーの貴重な時間を節約します。保護されたアプリケーションの動作を自動的に学習することで、開発者がアプリケーションを展開および変更しても、そのアプリケーションに適応します。PCI-

DSS、HIPAA など、すべての主要な規制基準および機関への準拠を支援します。当社の CloudFormation テンプレートを使用すると、迅速に稼働を開始することがこれまでになく簡単になります。自動スケーリングにより、ユーザーはトラフィックが増加してもアプリケーションが保護されたままであることを確信できます。

NetScaler Web App Firewall は、顧客サーバーと顧客ユーザーの間で、通常は顧客企業のルーターまたはファイアウォールの背後にある、レイヤー 3 ネットワークデバイスまたはレイヤー 2 ネットワークブリッジとしてインストールできます。詳細については、「[NetScaler Web App Firewall の概要](#)」を参照してください。

NetScaler Web App Firewall の展開戦略

1. Web アプリケーションファイアウォールを展開する目的は、どのアプリケーションまたは特定のデータが最大のセキュリティ保護を必要とし、どのアプリケーションが脆弱性が低く、どのアプリケーションでセキュリティ検査を安全にバイパスできるかを評価することです。これにより、ユーザーは最適な構成を考案し、トラフィックを分離するための適切なポリシーとバインドポイントを設計するのに役立ちます。たとえば、ユーザーは、画像、MP3 ファイル、動画などの静的 Web コンテンツに対するリクエストのセキュリティ検査をバイパスするポリシーを構成し、動的コンテンツに対するリクエストに高度なセキュリティチェックを適用する別のポリシーを構成したい場合があります。ユーザーは、複数のポリシーとプロファイルを使用して、同じアプリケーションの異なるコンテンツを保護できます。
2. 展開のベースラインを設定するには、仮想サーバーを作成し、テストトラフィックを流して、ユーザーシステムを流れるトラフィックの速度と量を把握します。
3. Web アプリケーションファイアウォールを展開します。NetScaler ADM と Web Application Firewall StyleBook を使用して、Web アプリケーションファイアウォールを構成します。詳細については、このガイドの以下の StyleBook セクションを参照してください。
4. NetScaler Web App Firewall と OWASP Top Ten を実装する。

Web アプリケーションファイアウォールの 3 つの保護機能は、一般的な種類の Web 攻撃に対して特に効果的であり、他のどの保護機能よりも一般的に使用されています。したがって、これらは初期展開で実装する必要があります。それらは次のとおりです。

- **HTML クロスサイトスクリプティング:** スクリプトが配置されている Web サイトとは異なる Web サイトのコンテンツにアクセスまたは変更しようとするスクリプトについて、リクエストとレスポンスを検査します。このチェックがそのようなスクリプトを検出すると、リクエストまたはレスポンスを宛先に転送する前にスクリプトを無害化するか、接続をブロックします。
- **HTML SQL インジェクション:** フォームフィールドデータを含むリクエストを検査し、SQL コマンドを SQL データベースに挿入しようとする試みを検出します。このチェックが挿入され

た SQL コードを検出すると、リクエストをブロックするか、リクエストを Web サーバーに転送する前に挿入された SQL コードを無害化します。

注:

構成で次の条件が適用されるように、Web App Firewall が正しく構成されていることを確認してください。

```
>- If users enable the HTML Cross-Site Scripting check or the HTML SQL Injection check (or both).
```

```
>
```

```
>- User protected websites accept file uploads or contain Web forms that can contain large POST body data.
```

このケースを処理するように Web アプリケーションファイアウォールを構成する方法の詳細については、「アプリケーションファイアウォールの構成: [Web App Firewall の構成](#)」を参照してください。

- **バッファオーバーフロー:** Web サーバーでバッファオーバーフローを引き起こそうとする試みを検出するためにリクエストを検査します。

Webアプリケーションファイアウォールの設定

NetScaler Web App Firewall がすでに有効になっており、正しく機能していることを確認してください。NetScaler Web App Firewall は、Web Application Firewall StyleBook を使用して構成することをお勧めします。ほとんどのユーザーは、Web Application Firewall を構成する最も簡単な方法であると考えており、間違いを防ぐように設計されています。GUI とコマンドラインインターフェイスはどちらも経験豊富なユーザー向けであり、主に既存の構成を変更したり、高度なオプションを使用したりすることを目的としています。

SQLインジェクション

NetScaler Web App Firewall の HTML SQLインジェクションチェックは、ユーザーアプリケーションのセキュリティを侵害する可能性のある不正なSQLコードのインジェクションに対する特別な防御を提供します。NetScaler Web App Firewall は、リクエストペイロード内のインジェクションされたSQLコードを次の3つの場所で検査します。1) POSTボディ、2) ヘッダー、3) クッキー。詳細については、[HTML SQLインジェクションチェック](#)を参照してください。

クロスサイトスクリプティング

HTMLクロスサイトスクリプティング（クロスサイトスクリプティング）チェックは、ユーザーリクエストのヘッダーとPOSTボディの両方を検査し、クロスサイトスクリプティング攻撃の可能性を検出します。クロスサイトスクリプトが検出された場合、攻撃を無害化するためにリクエストを変更（変換）するか、リクエストをブロックします。詳細については、[HTMLクロスサイトスクリプティングチェック](#)を参照してください。

バッファオーバーフローチェック

バッファオーバーフローチェックは、Webサーバーでバッファオーバーフローを引き起こそうとする試みを検出します。Web Application Firewall がURL、クッキー、またはヘッダーが設定された長さよりも長いことを検出した場合、バッファオーバーフローを引き起こす可能性があるため、リクエストをブロックします。詳細については、[バッファオーバーフローチェック](#)を参照してください。

仮想パッチ/シグネチャ

シグネチャは、既知の攻撃からユーザーのWebサイトを保護するタスクを簡素化するための、特定の構成可能なルールを提供します。シグネチャは、オペレーティングシステム、Webサーバー、Webサイト、XMLベースのWebサービス、またはその他のリソースに対する既知の攻撃のコンポーネントであるパターンを表します。事前設定された豊富な組み込みルールまたはネイティブルールは、パターンマッチングの力を適用して攻撃を検出し、アプリケーションの脆弱性から保護する、使いやすいセキュリティソリューションを提供します。詳細については、[シグネチャ](#)を参照してください。

NetScaler Web App Firewall は、シグネチャの**自動および手動更新**の両方をサポートしています。シグネチャを最新の状態に保つために、**自動更新**を有効にすることもお勧めします。



これらのシグネチャファイルはAWS環境でホストされており、最新のシグネチャファイルを取得するために、ネットワークファイアウォールからNetScaler IPアドレスへのアウトバウンドアクセスを許可することが重要です。リアルタイムトラフィックの処理中にNetScalerにシグネチャを更新しても影響はありません。

アプリケーションセキュリティ分析

アプリケーションセキュリティダッシュボードは、ユーザーアプリケーションのセキュリティステータスを包括的に表示します。たとえば、セキュリティ違反、シグネチャ違反、脅威インデックスなどの主要なセキュリティメトリックが表示されます。アプリケーションセキュリティダッシュ

ボードには、検出されたNetScalerに対するSYN攻撃、スモールウィンドウ攻撃、DNSフラッド攻撃などの攻撃関連情報も表示されます。

注記:

アプリケーションセキュリティダッシュボードのメトリックを表示するには、ユーザーが監視したいNetScalerインスタンスでAppFlow for Security insightを有効にする必要があります。

アプリケーションセキュリティダッシュボードでNetScalerインスタンスのセキュリティメトリックを表示するには：

1. 管理者資格情報を使用してNetScaler ADMにログインします。
2. **Applications > App Security Dashboard** に移動し、DevicesリストからインスタンスのIPアドレスを選択します。

ユーザーは、グラフにプロットされたバブルをクリックすることで、Application Security Investigatorで報告された不一致についてさらにドリルダウンできます。

ADMでの集中学習

NetScaler Web App Firewallは、SQLインジェクションやクロスサイトスクリプティング（XSS）などの悪意のある攻撃からユーザーのWebアプリケーションを保護します。データ侵害を防ぎ、適切なセキュリティ保護を提供するために、ユーザーは脅威と攻撃に関するリアルタイムの実行可能なデータをトラフィックで監視する必要があります。場合によっては、報告された攻撃が誤検知である可能性があり、それらは例外として提供される必要があります。

NetScaler ADMの集中学習は、WAFがユーザーのWebアプリケーションの動作（通常の活動）を学習できるようにする反復パターンフィルターです。監視に基づいて、エンジンはHTTPトラフィックに適用される各セキュリティチェックに対して、推奨されるルールまたは例外のリストを生成します。

学習エンジンを使用して緩和ルールを展開する方が、必要な緩和策として手動で展開するよりもはるかに簡単です。

学習機能を展開するには、ユーザーはまずユーザーのNetScalerでWeb Application Firewallプロファイル（セキュリティ設定のセット）を設定する必要があります。詳細については、「[Web App Firewallプロファイルの作成](#)」を参照してください。

NetScaler ADMは、各セキュリティチェックの例外（緩和策）のリストを生成します。管理者として、NetScaler ADMで例外のリストを確認し、展開するかスキップするかを決定できます。

NetScaler ADMのWAF学習機能を使用すると、次のことができます。

- 以下のセキュリティチェックで学習プロファイルを構成します。
 - バッファオーバーフロー
 - HTMLクロスサイトスクリプティング

注：

ロケーションのクロスサイトスクリプトの制限はFormFieldのみです。

- HTML SQLインジェクション

注：

HTML SQLインジェクションチェックの場合、ユーザーはNetScalerで `set -sqlinjectionTransformSpecialChars ON` と `set -sqlinjectiontype sqlspclcharorkeywords` を設定する必要があります。

- NetScaler ADMで緩和ルールを確認し、必要なアクション（展開またはスキップ）を実行するかどうかを決定します。
- メール、Slack、ServiceNowを通じて通知を受け取ります。
- ダッシュボードを使用して緩和の詳細を表示します。

NetScaler ADMでWAF学習を使用するには：

1. 学習プロファイルを構成する: [学習プロファイルの構成](#)
2. 緩和ルールを表示する: [緩和ルールとアイドルルールの表示](#)
3. WAF学習ダッシュボードを使用する: [WAF学習ダッシュボードの表示](#)

StyleBooks

StyleBooksは、ユーザーアプリケーションの複雑なNetScaler構成を管理するタスクを簡素化します。StyleBookは、ユーザーがNetScaler構成を作成および管理するために使用できるテンプレートです。ここでは、ユーザーは主にWebアプリケーションファイアウォールを展開するために使用されるStyleBookに関心があります。StyleBooksの詳細については、[StyleBooks](#)を参照してください。

セキュリティインサイト分析

インターネットに公開されているWebおよびWebサービスアプリケーションは、攻撃に対してますます脆弱になっています。アプリケーションを攻撃から保護するために、ユーザーは過去、現在、および差し迫った脅威の性質と範囲、攻撃に関するリアルタイムで実行可能なデータ、および対策に関する推奨事項を可視化する必要があります。Security Insightは、ユーザーがユーザーアプリケーションのセキュリティステータスを評価し、ユーザーアプリケーションを保護するための是正措置を講じるのに役立つ単一ペインソリューションを提供します。詳細については、「[Security Insight](#)」を参照してください。

セキュリティ侵害に関する詳細情報の取得

ユーザーは、アプリケーションに対する攻撃のリストを表示し、攻撃の種類と重大度、ADCインスタンスによって実行されたアクション、要求されたリソース、および攻撃元に関する洞察を得たい場合があります。

たとえば、ユーザーはMicrosoft Lyncに対する攻撃がいくつかブロックされたか、どのようなリソースが要求されたか、および攻撃元のIPアドレスを特定したい場合があります。

Security Insightダッシュボードで、**Lync > Total Violations**をクリックします。テーブルで、**Action Taken**列ヘッダーのフィルターアイコンをクリックし、**Blocked**を選択します。



要求されたリソースに関する情報については、**URL**列を確認してください。攻撃元に関する情報については、**Client IP**列を確認してください。

ログ式の詳細を表示

NetScalerは、Application Firewallプロファイルで構成されたログ式を使用して、ユーザー企業内のアプリケーションに対する攻撃に対処します。**Security Insight**では、ユーザーはADCインスタンスが使用するログ式に対して返された値を表示できます。これらの値には、リクエストヘッダー、リクエストボディなどが含まれます。ログ式の値に加えて、ユーザーは、ADCインスタンスが攻撃に対処するために使用したApplication Firewallプロファイルで定義されたログ式の名前とコメントも表示できます。

前提条件:

ユーザーが以下を実行していることを確認します。

- Application Firewallプロファイルでログ式を構成します。詳細については、「[Application Firewall](#)」を参照してください。
- NetScaler ADMでログ式ベースのSecurity Insights設定を有効にします。次の手順を実行します。

- **Analytics > Settings**に移動し、**Enable Features for Analytics**をクリックします。
- 「分析機能の有効化」 ページで、「ログデータベースの **Security Insight 設定**」セクションの「**Security Insight を有効にする**」を選択し、「OK」をクリックします。



たとえば、ユーザー企業内のMicrosoft Lyncへの攻撃に対してADCインスタンスが実行したアクションについて、ADCインスタンスによって返されたログ式の値を表示したい場合があります。

「**Security Insight ダッシュボード**」で、「**Lync**」>「**合計違反**」に移動します。「アプリケーションサマリー」テーブルで、URLをクリックして、「**違反情報**」ページで、ログ式の名前、コメント、およびそのアクションに対してADCインスタンスによって返された値を含む、違反の完全な詳細を表示します。



構成を展開する前に、安全指数を決定します。ユーザーがADCインスタンスにセキュリティ構成を展開した後、セキュリティ侵害が発生しますが、ユーザーは展開する前にセキュリティ構成の有効性を評価したい場合があります。

たとえば、ユーザーはIPアドレス10.102.60.27のADCインスタンス上のSAPアプリケーションの構成の安全指数を評価したい場合があります。

「**Security Insight ダッシュボード**」の「**デバイス**」の下で、ユーザーが構成したADCインスタンスのIPアドレスをクリックします。脅威指数と攻撃の総数がともに0であることがわかります。脅威指数は、アプリケーションに対する攻撃の数と種類の直接的な反映です。攻撃がゼロであることは、アプリケーションが脅威にさらされていないことを示します。



「**Sap**」>「**安全指数**」>「**SAP_Profile**」をクリックし、表示される安全指数情報を評価します。



アプリケーションファイアウォールの概要で、ユーザーはさまざまな保護設定の構成ステータスを表示できます。設定がログに設定されている場合、または設定が構成されていない場合、アプリケーションには低い安全指数が割り当てられます。



セキュリティ違反

インターネットに公開されているWebアプリケーションは、攻撃に対して劇的に脆弱になっています。NetScaler ADMを使用すると、実用的な違反の詳細を視覚化して、アプリケーションを攻撃から保護できます。

アプリケーションセキュリティ違反の詳細を表示

インターネットに公開されているWebアプリケーションは、攻撃に対して劇的に脆弱になっています。NetScaler ADMを使用すると、ユーザーは実行可能な違反の詳細を視覚化して、アプリケーションを攻撃から保護できます。**セキュリティ > セキュリティ違反** に移動すると、次の単一ページソリューションが表示されます。

- ネットワーク、ボット、WAFなどのカテゴリに基づいて、アプリケーションセキュリティ違反にアクセスします。
- アプリケーションを保護するための是正措置を講じます。

NetScaler ADMでセキュリティ違反を表示するには、以下を確認してください。

- ユーザーはNetScalerのプレミアムライセンスを所有する必要があります（WAFおよびBOT違反の場合）。
- ユーザーは、負荷分散またはコンテンツスイッチング仮想サーバーにライセンスを適用する必要があります（WAFおよびBOTの場合）。詳細については、「[仮想サーバーでのライセンス管理](#)」を参照してください。
- ユーザーはさらに設定を有効にできます。詳細については、NetScaler製品ドキュメントの「設定」セクションにある手順を参照してください: [設定](#)。

違反カテゴリ

NetScaler ADMを使用すると、ユーザーは[すべての違反](#)で利用可能な違反を表示できます。

設定

違反については、メトリクスコレクターが有効になっていることを確認してください。デフォルトでは、NetScalerでメトリクスコレクターが有効になっています。詳細については、「[インテリジェントアプリ分析の構成](#)」を参照してください。

高度なセキュリティ分析を有効にする

- ネットワーク > インスタンス > **NetScaler** に移動し、インスタンスタイプを選択します。例: MPX。
- NetScalerインスタンスを選択し、**アクションの選択リスト**から**分析の構成**を選択します。
- 仮想サーバーを選択し、**Enable Analytics** をクリックします。
- **Enable Analytics** ウィンドウで:
 - **Web Insight** を選択します。ユーザーが Web Insight を選択すると、読み取り専用の **Advanced Security Analytics** オプションが自動的に有効になります。

注:

Advanced Security Analytics オプションは、プレミアムライセンスのADCインスタンスでのみ表示されます。

- トランスポートモードとして **Logstream** を選択します。
- 式はデフォルトでtrueです。
- **OK** をクリックします。



Webトランザクション設定を有効にする

- **Analytics > Settings** に移動します。

Settings ページが表示されます。

- **Enable Features for Analytics** をクリックします。
- **Web Transaction Settings** で、**All** を選択します。



- 「**Ok**」 をクリックします。

セキュリティ違反ダッシュボード

セキュリティ違反ダッシュボードでは、ユーザーは以下を表示できます。

- すべてのNetScalerおよびアプリケーションで発生した合計違反。合計違反は、選択した期間に基づいて表示されます。



- 各カテゴリの合計違反。



- 影響を受けたADCの合計、影響を受けたアプリケーションの合計、および合計発生数と影響を受けたアプリケーションに基づく上位の違反。



違反の詳細については、「[すべての違反](#)」を参照してください。

ボットインサイト

NetScalerでBOTインサイトを設定します。詳細については、「[ボット](#)」を参照してください。

ボットの表示

仮想サーバーをクリックして、**アプリケーションサマリー**を表示します。



1. アプリケーションの概要詳細を以下のように提供します。

- **平均RPS** – 仮想サーバーで受信した1秒あたりのボットトランザクションリクエスト（RPS）の平均を示します。
- **重大度別ボット** – 重大度に基づいて最も多くのボットトランザクションが発生したことを示します。重大度は「**緊急**」、「**高**」、「**中**」、「**低**」に分類されます。

例えば、仮想サーバーに重大度「**高**」のボットが11770件、重大度「**緊急**」のボットが1550件ある場合、NetScaler ADMは「**緊急 1.55 K**」を「**重大度別ボット**」の下に表示します。

- **最大のボットカテゴリ** – ボットカテゴリに基づいて最も多くのボット攻撃が発生したことを示します。

例えば、仮想サーバーにブロックリストに登録されたボットが8000件、許可リストに登録されたボットが5000件、レート制限超過ボットが10000件ある場合、NetScaler ADMは「**レート制限超過 10 K**」を「**最大のボットカテゴリ**」の下に表示します。

- **最大の地理的ソース** – 地域に基づいて最も多くのボット攻撃が発生したことを示します。

例えば、仮想サーバーでサンタクララで5000件、ロンドンで7000件、バンガロールで9000件のボット攻撃が発生した場合、NetScaler ADMは「**バンガロール 9 K**」を「**最大の地理的ソース**」の下に表示します。

- **平均ボットトラフィック率** – 人間とボットの比率を示します。

2. マップビューで場所に基づいてボット攻撃の重大度を表示します。

3. ボット攻撃の種類（良好、不良、すべて）を表示します。

4. 設定された対応するアクションとともに、ボット攻撃の合計数を表示します。例えば、以下のように設定した場合：

- IPアドレス範囲（192.140.14.9～192.140.14.254）をブロックリストボットとして、これらのIPアドレス範囲のアクションとしてドロップを選択した場合
- IP範囲（192.140.15.4～192.140.15.254）をブロックリストボットとして、これらのIP範囲のアクションとしてログメッセージの作成を選択した場合

このシナリオでは、NetScaler ADMは以下を表示します。

- 。ブロックリストに登録されたボットの合計
- 。ドロップ済みのボットの合計
- 。ログのボットの合計

CAPTCHAボットの表示

ウェブページでは、CAPTCHAは、受信トラフィックが人間からのものか、自動化されたボットからのものかを識別するように設計されています。NetScaler ADMでCAPTCHAアクティビティを表示するには、ユーザーはNetScaler ADMインスタンスでIPレピュテーションおよびデバイスフィンガープリント検出技術のボットアクションとしてCAPTCHAを設定する必要があります。詳細については、[ボット管理の構成](#)を参照してください。

NetScaler ADMがボットインサイトに表示するCAPTCHAアクティビティは次のとおりです。

- **CAPTCHA試行回数超過** – ログイン失敗後に試行されたCAPTCHAの最大回数を示します。
- **CAPTCHAクライアントミュート** – これらのリクエストが以前にCAPTCHAチャレンジで悪質なボットとして検出されたため、ドロップまたはリダイレクトされたクライアントリクエストの数を示します。
- **人間** – 人間ユーザーによって実行されたCAPTCHAエントリを示します。
- **無効なCAPTCHA応答** – NetScalerがCAPTCHAチャレンジを送信したときに、ボットまたは人間から受信した不正なCAPTCHA応答の数を示します。



ボットトラップの表示

NetScaler ADMでボットトラップを表示するには、NetScalerでボットトラップを設定する必要があります。詳細については、[ボット管理の構成](#)を参照してください。



ボットトラップを識別するために、ウェブページでスクリプトが有効になっています。このスクリプトは人間からは隠されていますが、ボットからは隠されていません。NetScaler ADMは、このスクリプトがボットによってアクセスされたときに、ボットトラップを識別して報告します。

仮想サーバーをクリックし、**Zero Pixel Request** を選択します。

(/en-us/vpx/media/image-vpx-azure-appsecurity-deployment-53.png)

ボットの詳細を表示

詳細については、**Bot Category** の下のボット攻撃タイプをクリックしてください。

選択したCAPTCHAカテゴリの攻撃時間やボット攻撃の合計数などの詳細が表示されます。

(/en-us/vpx/media/image-vpx-azure-appsecurity-deployment-54.png)

ユーザーは棒グラフをドラッグして、ボット攻撃とともに表示する特定の時間範囲を選択することもできます。

(/en-us/vpx/media/image-vpx-azure-appsecurity-deployment-55.png)

ボット攻撃の追加情報を取得するには、クリックして展開します。

(/en-us/vpx/media/image-vpx-azure-appsecurity-deployment-56.png)

- **インスタンスIP** – NetScalerインスタンスのIPアドレスを示します。
- **合計ボット数** – その特定の時間に発生したボット攻撃の合計数を示します。
- **HTTPリクエストURL** – CAPTCHAレポート用に構成されているURLを示します。
- **国コード** – ボット攻撃が発生した国を示します。
- **地域** – ボット攻撃が発生した地域を示します。
- **プロファイル名** – ユーザーが設定時に入力したプロファイル名を示します。

高度な検索

ユーザーは検索テキストボックスと期間リストを使用して、ユーザーの要件に応じてボットの詳細を表示することもできます。ユーザーが検索ボックスをクリックすると、検索ボックスには以下の検索候補リストが表示されます。

- **インスタンス IP** – NetScaler インスタンスの IP アドレス。
- **クライアント IP** – クライアントの IP アドレス。
- **ボットタイプ** – Good または Bad などのボットタイプ。
- **重大度** – ボット攻撃の重大度。
- **実行されたアクション** – ドロップ、アクションなし、リダイレクトなど、ボット攻撃後に実行されたアクション。
- **ボットカテゴリ** – ブロックリスト、許可リスト、フィンガープリントなどのボット攻撃のカテゴリ。カテゴリに基づいて、ユーザーはボットアクションを関連付けることができます。
- **ボット検出** – ユーザーがNetScalerで設定したボット検出タイプ（ブロックリスト、許可リストなど）。
- **場所** – ボット攻撃が発生した地域/国
- **リクエスト URL** – ボット攻撃の可能性がある URL

ユーザーは、ユーザー検索クエリで演算子を使用して、ユーザー検索の焦点を絞り込むこともできます。たとえば、すべての悪質なボットを表示したい場合：

- 検索ボックスをクリックし、**ボットタイプ**を選択します
- もう一度検索ボックスをクリックし、演算子 **=** を選択します
- 検索ボックスを再度クリックし、「**Bad**」を選択します
- 「**Search**」をクリックして結果を表示します

(/en-us/vpx/media/image-vpx-azure-appsecurity-deployment-57.png)

異常に高いリクエストレート

ユーザーは、アプリケーションとの間で送受信されるトラフィックを制御できます。ボット攻撃は、異常に高いリクエストレートを実行する可能性があります。たとえば、ユーザーがアプリケーションを1分あたり100リクエストを許可するように構成し、350リクエストを観測した場合、それはボット攻撃である可能性があります。

「**異常に高いリクエストレート**」インジケーターを使用すると、ユーザーは、アプリケーションが受信した異常なリクエストレートを分析できます。

(/en-us/vpx/media/image-vpx-azure-appsecurity-deployment-65.png)

「**イベントの詳細**」では、ユーザーは以下を表示できます。

- 影響を受けるアプリケーション。2つ以上のアプリケーションが違反の影響を受けている場合、ユーザーはリストからアプリケーションを選択することもできます。
- すべての違反を示すグラフ
- 違反発生時刻
- 違反の検出メッセージ。受信した合計リクエスト数と、予想されるリクエスト数よりも過剰に受信したリクエストの割合を示します。
- アプリケーションからの予想されるリクエストレートの許容範囲

ボット検出

NetScalerボット管理システムは、さまざまな手法を使用して、受信するボットトラフィックを検出します。これらの手法は、ボットの種類を検出するための検出ルールとして使用されます。

NetScaler GUI を使用したボット管理の設定

ユーザーは、アプライアンスで機能を有効にすることで、NetScaler ボット管理を設定できます。詳細については、「[ボット検出](#)」を参照してください。

IP レピュテーション

IP レピュテーションは、不要なリクエストを送信する IP アドレスを識別するツールです。IP レピュテーションリストを使用すると、評判の悪い IP アドレスからのリクエストを拒否できます。

NetScaler GUI を使用した IP レピュテーションの設定

この設定は、ボット IP レピュテーション機能の前提条件です。詳細については、「[IP レピュテーション](#)」を参照してください。

ボットシグネチャの自動更新

ボット静的シグネチャ技術は、良好なボットと悪意のあるボットのリストを含むシグネチャルックアップテーブルを使用します。詳細については、「[シグネチャの自動更新](#)」を参照してください。

NetScaler Web App Firewall と OWASP トップ 10 – 2021

Open Web Application Security Project (OWASP) は、Web アプリケーションセキュリティに関する 2021 年版 OWASP Top 10 を発表しました。このリストは、最も一般的な Web アプリケーションの脆弱性を文書化しており、Web セキュリティを評価するための優れた出発点となります。このセクションでは、これらの欠陥を軽減するために NetScaler Web App Firewall を構成する方法について説明します。WAF は、NetScaler (Premium Edition) の統合モジュールとして、またあらゆるアプライアンスで利用できます。

OWASP Top 10 の完全なドキュメントは、[OWASP Top Ten](#) で入手できます。

OWASP TOP 10 – 2021	NETSCALER WEB APP FIREWALL の機能
A1:2021 アクセス制御の不備	AAA、NetScaler の AAA モジュール内の認証セキュリティ機能、フォーム保護、Cookie 改ざん保護、StartURL、および ClosureURL
A2:2021 - 暗号化の失敗	クレジットカード保護、セーフコマース、クッキープロキシ、およびクッキー暗号化
A3:2021- インジェクション	インジェクション攻撃防止 (SQLまたはOSコマンドインジェクション、XPathインジェクション、LDAPインジェクションなどのカスタムインジェクション)、自動更新シグネチャ機能

OWASP TOP 10 – 2021	NETSCALER WEB APP FIREWALL の機能
A5:2021 セキュリティ設定ミス	WSIチェック、XMLメッセージ検証、XML SOAP障害フィルタリングチェックを含むこの保護
A6:2021 - 脆弱性と古いコンポーネント	脆弱性スキャンレポート、アプリケーションファイアウォールテンプレート、およびカスタムシグネチャ
A7:2021 - 識別と認証の失敗	AAA、クッキー改ざん防止、クッキープロキシ、クッキー暗号化、CSRFタグ付け、SSLの使用
A8:2021 – ソフトウェアとデータの整合性の失敗	XMLセキュリティチェック、GWTコンテンツタイプ、カスタムシグネチャ、JSONおよびXML用のXPath
A9:2021 – セキュリティログと監視の失敗	ユーザー設定可能なカスタムロギング、NetScaler Management and Analytics System

A1:2021 不適切なアクセス制御

認証済みユーザーに許可されていることに対する制限が、適切に適用されていないことがよくあります。攻撃者はこれらの欠陥を悪用して、他のユーザーのアカウントへのアクセス、機密ファイルの表示、他のユーザーのデータの変更、アクセス権の変更など、不正な機能やデータにアクセスできます。

NetScaler Web App Firewall の保護機能

- すべてのアプリケーショントラフィックに対する認証、認可、監査をサポートするAAA機能により、サイト管理者はADCアプライアンスでアクセス制御を管理できます。
- ADCアプライアンスのAAAモジュール内の認可セキュリティ機能により、アプライアンスは、保護されたサーバー上のどのコンテンツに各ユーザーがアクセスすることを許可すべきかを検証できます。
- フォームフィールドの一貫性: オブジェクト参照がフォームの隠しフィールドとして保存されている場合、フォームフィールドの一貫性を使用することで、これらのフィールドが後続のリクエストで改ざんされていないことを検証できます。
- クッキープロキシとクッキーの一貫性: クッキー値に保存されているオブジェクト参照は、これらの保護機能で検証できます。

- URLクローズによる開始URLチェック: 事前定義されたURLの許可リストへのユーザーアクセスを許可します。URLクローズは、ユーザーセッション中に有効な応答で検出されたすべてのURLのリストを作成し、そのセッション中にそれらへのアクセスを自動的に許可します。

A2:2021 - 暗号化の失敗

多くのWebアプリケーションやAPIは、金融、医療、PIIなどの機密データを適切に保護していません。攻撃者は、このような保護が不十分なデータを盗んだり改ざんしたりして、クレジットカード詐欺、個人情報盗難、その他の犯罪を行う可能性があります。機密データは、保存時または転送時の暗号化などの追加保護なしに侵害される可能性があり、ブラウザとの間で交換する際には特別な注意が必要です。

NetScaler Web App Firewall の保護機能

- Web Application Firewallは、クレジットカード情報などの機密データが漏洩するのを防ぎます。
- 機密データは、露出を避けるためにSafe Commerce保護で安全なオブジェクトとして構成できます。
- クッキー内の機密データは、クッキープロキシとクッキー暗号化によって保護できます。

A3:2021 - インジェクション

SQL、NoSQL、OS、LDAPインジェクションなどのインジェクションの欠陥は、信頼できないデータがコマンドまたはクエリの一部としてインタープリターに送信されたときに発生します。攻撃者の悪意のあるデータは、インタープリターをだまして意図しないコマンドを実行させたり、適切な認可なしにデータにアクセスさせたりする可能性があります。

XSSの脆弱性は、アプリケーションが適切な検証やエスケープなしに信頼できないデータを新しいウェブページに含めたり、HTMLやJavaScriptを作成できるブラウザAPIを使用してユーザー提供データで既存のウェブページを更新したりするたびに発生します。XSSにより、攻撃者は被害者のブラウザでスクリプトを実行し、ユーザーセッションを乗っ取ったり、ウェブサイトを改ざんしたり、ユーザーを悪意のあるサイトにリダイレクトしたりすることができます。

NetScaler Web App Firewallの保護機能

- SQLインジェクション防止機能は、一般的なインジェクション攻撃から保護します。XPathやLDAPを含むあらゆる種類のインジェクション攻撃から保護するために、カスタムインジェクションパターンをアップロードできます。これはHTMLとXMLの両方のペイロードに適用されます。

- 自動更新シグネチャ機能により、インジェクションシグネチャが常に最新の状態に保たれます。
- フィールド形式保護機能により、管理者は任意のユーザーパラメータを正規表現に制限できます。たとえば、郵便番号フィールドに整数のみ、または5桁の整数のみを含めるように強制できます。
- フォームフィールドの一貫性は、送信された各ユーザーフォームをユーザーセッションフォームシグネチャに対して検証し、すべてのフォーム要素の有効性を保証します。
- バッファオーバーフローチェックは、URL、ヘッダー、およびCookieが適切な制限内にあることを確認し、大規模なスクリプトやコードを挿入しようとする試みをブロックします。
- XSS保護は、一般的なXSS攻撃から保護します。許可されたタグと属性のデフォルトリストを変更するために、カスタムXSSパターンをアップロードできます。ADC WAFは、許可されたHTML属性とタグのホワイトリストを使用してXSS攻撃を検出します。これはHTMLとXMLの両方のペイロードに適用されます。
- ADC WAFは、OWASP XSS Filter Evaluation Cheat Sheetに記載されているすべての攻撃をブロックします。
- フィールド形式チェックは、攻撃者が不適切なWebフォームデータを送信するのを防ぎ、これは潜在的なXSS攻撃となる可能性があります。
- フォームフィールドの一貫性。

A5:2021 - セキュリティ設定ミス

セキュリティ設定ミスは、最も一般的に見られる問題です。これは通常、安全でないデフォルト設定、不完全または場当たりの設定、オープンなクラウドストレージ、誤って設定されたHTTPヘッダー、機密情報を含む詳細なエラーメッセージの結果として発生します。すべてのオペレーティングシステム、フレームワーク、ライブラリ、およびアプリケーションは、安全に構成されるだけでなく、タイムリーにパッチが適用され、アップグレードされる必要があります。

多くの古いまたは不適切に構成されたXMLプロセッサは、XMLドキュメント内の外部エンティティ参照を評価します。外部エンティティは、ファイルURIハンドラを使用した内部ファイルの開示、内部ファイル共有、内部ポートスキャン、リモートコード実行、およびサービス拒否攻撃に使用される可能性があります。

NetScaler Web App Firewallの保護機能

- アプリケーションファイアウォールによって生成されたPCI-DSSレポートは、ファイアウォールデバイスのセキュリティ設定を文書化します。

- スキャンツールからのレポートは、セキュリティの誤設定を処理するためにADC WAFシグネチャに変換されます。
- NetScaler Web App Firewall Web Application Firewallは、Cenzic、IBM AppScan (EnterpriseおよびStandard)、Qualys、TrendMicro、WhiteHat、およびカスタム脆弱性スキャンレポートをサポートしています。
- XMLベースのアプリケーションを攻撃するために適応できる一般的なアプリケーション脅威（クロスサイトスクリプティング、コマンドインジェクションなど）の検出とブロックに加えて。
- NetScaler Web App Firewall Web Application Firewallには、XML固有の豊富なセキュリティ保護機能が含まれています。これには、SOAPメッセージとXMLペイロードを徹底的に検証するためのスキーマ検証、および悪意のある実行可能ファイルやウイルスを含む添付ファイルをブロックするための強力なXML添付ファイルチェックが含まれます。
- 自動トラフィック検査方法は、アクセスを目的としたURLおよびフォームに対するXPathインジェクション攻撃をブロックします。
- NetScaler Web App Firewall Web Application Firewallは、外部エンティティ参照、再帰的展開、過剰なネスト、および長い属性や多数の属性と要素を含む悪意のあるメッセージなど、さまざまなDoS攻撃も阻止します。

A6:2021 - 脆弱で古いコンポーネント

ライブラリ、フレームワーク、その他のソフトウェアモジュールなどのコンポーネントは、アプリケーションと同じ権限で実行されます。脆弱なコンポーネントが悪用された場合、そのような攻撃は深刻なデータ損失やサーバー乗っ取りを引き起こす可能性があります。既知の脆弱性を持つコンポーネントを使用するアプリケーションやAPIは、アプリケーションの防御を弱め、さまざまな攻撃や影響を可能にする可能性があります。

NetScaler Web App Firewallの保護

- サードパーティコンポーネントを最新の状態に保つことをお勧めします。
- ADCシグネチャに変換された脆弱性スキャンレポートは、これらのコンポーネントを仮想的にパッチ適用するために使用できます。
- これらの脆弱なコンポーネントで利用可能なアプリケーションファイアウォールテンプレートを使用できます。
- これらのコンポーネントを保護するために、カスタムシグネチャをファイアウォールにバインドできます。

A7:2021 – 認証の不備

認証およびセッション管理に関連するアプリケーション機能は、多くの場合、誤って実装されており、攻撃者がパスワード、キー、またはセッショントークンを侵害したり、他の実装上の欠陥を悪用して、一時的または永続的に他のユーザーのIDを乗っ取ったりすることを可能にします。

NetScaler Web App Firewallの保護

- NetScaler AAAモジュールは、ユーザー認証を実行し、バックエンドアプリケーションにシングルサインオン機能を提供します。これはNetScaler AppExpertポリシーエンジンに統合されており、ユーザーおよびグループ情報に基づいたカスタムポリシーを可能にします。
- SSLオフロードおよびURL変換機能を使用することで、ファイアウォールは、ネットワークスニффイングによるセッショントークンの盗難を防ぐために、サイトが安全なトランスポート層プロトコルを使用するのを支援することもできます。
- クッキープロキシとクッキー暗号化を使用することで、クッキーの盗難を完全に軽減できます。

A8:2021 - ソフトウェアとデータの整合性の不備

安全でないデシリアライゼーションは、多くの場合、リモートコード実行につながります。デシリアライゼーションの欠陥がリモートコード実行に至らない場合でも、リプレイ攻撃、インジェクション攻撃、権限昇格攻撃などの攻撃を実行するために使用される可能性があります。

NetScaler Web App Firewallの保護

- カスタム署名によるJSONペイロード検査。
- XMLセキュリティ：XMLサービス拒否（xDoS）、XML SQLおよびXPathインジェクション、クロスサイトスクリプティング、フォーマットチェック、WS-I基本プロファイル準拠、XML添付ファイルのチェックから保護します。
- フィールド形式チェック、Cookieの一貫性、およびフィールドの一貫性を使用できます。

A9:2021 - セキュリティログと監視の不備

不十分なロギングと監視は、インシデント対応との統合が欠けているか非効果的であることと相まって、攻撃者がシステムをさらに攻撃し、永続性を維持し、より多くのシステムに侵入し、データを改ざん、抽出、または破壊することを可能にします。ほとんどの侵害調査では、侵害の検出にかかる時間は200日を超え、通常は内部プロセスや監視ではなく、外部の当事者によって検出されることが示されています。

NetScaler Web App Firewallの保護

- セキュリティチェックまたは署名に対してログアクションが有効になっている場合、生成されるログメッセージは、アプリケーションファイアウォールがウェブサイトとアプリケーションを保護している間に観測したリクエストとレスポンスに関する情報を提供します。
- アプリケーションファイアウォールは、悪意のあるリクエストの発信元であるIPアドレスに対応する場所を特定するために、組み込みのADCデータベースを使用する利便性を提供します。
- デフォルト形式 (PI) 式を使用すると、アプリケーションファイアウォールが生成するログメッセージにキャプチャする特定のデータを追加するオプションを使用して、ログに含まれる情報を柔軟にカスタマイズできます。
- アプリケーションファイアウォールはCEFログをサポートしています。

参照

- [HTML SQLインジェクションチェック](#)
- [XML SQLインジェクションチェック](#)
- [コマンドラインを使用したHTMLクロスサイトスクリプティングチェックの設定](#)
- [XMLクロスサイトスクリプティングチェック](#)
- [コマンドラインを使用したバッファオーバーフローセキュリティチェックの設定](#)
- [署名オブジェクトの追加または削除](#)
- [署名オブジェクトの設定または変更](#)
- [署名オブジェクトの更新](#)
- [Snortルール統合](#)
- [ボット検出](#)
- [Microsoft AzureへのNetScaler VPXインスタンスの展開](#)

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

状況によっては、NetScaler Gateway Plug-inで接続するユーザーは、NetScaler Gatewayアプライアンスに一意のIPアドレスを必要とします。グループに対してアドレスプール (IPプーリングとも呼ばれます) を有効にすると、NetScaler Gatewayアプライアンスは各ユーザーに一意のIPアドレスエイリアスを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に展開された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段階の手順に従います。

- アドレスプールで使用するプライベートIPアドレスをAzureに登録する
- NetScaler Gatewayアプライアンスでアドレスプールを構成する

AzureポータルでプライベートIPアドレスに登録する

Azureでは、複数のIPアドレスを持つNetScaler VPXインスタンスを展開できます。VPXインスタンスにIPアドレスを追加する方法は2つあります。

a. VPXインスタンスのプロビジョニング中

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の複数のIPアドレスを構成する](#)を参照してください。

PowerShellコマンドを使用してVPXインスタンスのプロビジョニング中に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のNetScaler 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。

b. VPXインスタンスの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後、NetScaler Gatewayアプライアンスでアドレスプールとして構成するプライベートIPアドレスをAzureポータルに登録するには、次の手順に従います。

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。登録したいIIPが属す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、次に **追加** をクリックします。



3. 以下の例に示すように、必要な詳細を入力し、**OK**をクリックします。

IIPの詳細(/en-us/vpx/media/details-iip.png)

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アドレスは登録されている必要があります。

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

Azure 環境では、NetScaler VPX 仮想アプライアンスを複数の NIC で展開できます。各 NIC は複数の IP アドレスを持つことができます。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 アドレスを持つ NetScaler VPX インスタンスを展開する方法について説明します。マルチ NIC およびマルチ IP 展開には、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パブリック IP とプライベート IP のペアの IP アドレスを指します。詳細については、[Azure 用語](#) セクションを参照してください。

ユースケース

このユースケースでは、単一の NIC が仮想ネットワーク (VNET) に接続されています。NIC は、次の表に示すように、3 つの IP 構成に関連付けられています。

IP 構成	関連付け
IP構成-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
IP構成-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
IP構成-3	静的プライベート IP アドレス

注:

IPConfig-3 はどのパブリック IP アドレスとも関連付けられていません。

図: トポロジ

ユースケースの視覚的な表現を次に示します。

単一モードの構成(/en-us/vpx/media/config-multiip-singlemode.png)

注:

マルチNIC、マルチIPのAzure NetScaler VPX展開では、プライマリ（最初の）NICのプライマリ（最初の） `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のNetScaler VPX仮想アプライアンスに複数のIPアドレスを構成するために必要な手順の概要を次に示します。

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. 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インスタンスを作成する
10. NIC構成を確認する
11. VPX側の構成を確認する

スクリプト

パラメータ

このドキュメントのユースケースにおけるパラメータ設定の例を以下に示します。

```
$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" (This parameter creates a VM with up to four  
NICs.)
```

注:

VPXインスタンスの最小要件は、2 vCPUと2 GBのRAMです。

```
$publisher="Citrix"
```

```
$offer="netscalervpx110-6531" (You can use different offers.)
```

```
$sku="netscalerbyol" (According to your offer, the SKU can be different.)
```

```
$version="latest"
```

```
$pubIPName1="PIP1"
```

```
$pubIPName2="PIP2"
```

```
$domName1="multiipvpx1"
```

```
$domName2="multiipvpx2"
```

```
$vmNamePrefix="VPXMultiIP"
```

```
$osDiskSuffix="osmultiipalbdiskdb1"
```

```
**Network Security Group (NSG)-related information**:
```

```
$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. ルールを追加します。トラフィックを処理するポートには、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 "Allow HTTPS" -Access Allow -Protocol Tcp -Direction
Inbound -Priority 110 -SourceAddressPrefix Internet -SourcePortRange *
-DestinationAddressPrefix * -DestinationPortRange 443 $rule3=New-
AzureRmNetworkSecurityRuleConfig -Name $rule3Name -Description "Allow SSH"
-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="frontEndSubnet" $subnet1=$vnet.Subnets|?{$_.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確認してください。

IPアドレスの割り当て方法は、動的または静的です。

7. IP構成の割り当て

このユースケースでは、IPアドレスを割り当てる前に以下の点を考慮してください。

- IPConfig-1はVPX1のsubnet1に属します。
- IPConfig-2はVPX1のsubnet 1に属します。
- IPConfig-3はVPX1のsubnet 1に属します。

注:

複数のIP構成をNICに割り当てる場合、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。

```
$IPAddress1="11.6.1.27"
```

```
$IPConfig1=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName1
-Subnet $subnet1 -PrivateIpAddress $IPAddress1 -PublicIpAddress $pip1 -
Primary
```

```
$IPAddress2="11.6.1.28"
```

```
$IPConfig2=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName2
-Subnet $subnet1 -PrivateIpAddress $IPAddress2 -PublicIpAddress $pip2
```

```
$IPAddress3="11.6.1.29"
```

```
$IPConfig3=New-AzureRmNetworkInterfaceIpConfig -Name $IPConfigName3
-Subnet $subnet1 -PrivateIpAddress $IPAddress3 -Primary
```

サブネット要件を満たす有効なIPアドレスを使用し、その可用性を確認してください。

8. NICの作成

```
$nic1=New-AzureRmNetworkInterface -Name $nicName1 -ResourceGroupName $rgName
-Location $locName -IpConfiguration $IpConfig1,$IpConfig2,$IPConfig3
-NetworkSecurityGroupId $nsg.Id
```

9. NetScaler VPXインスタンスの作成

1. 変数を初期化します。

```
$suffixNumber = 1 $vmName = $vmNamePrefix + $suffixNumber
```

2. VM構成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、およびイメージを設定します。

```
$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
```

4. NICを追加します。

```
$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic1.Id -Primary
```

注:

複数のNICを持つ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。そのため、そのNICをVPXインスタンスに追加する際には、「-Primary」を付加する必要があります。

5. OSディスクを指定し、VMを作成します。

```
$osDiskName=$vmName + "-" + $osDiskSuffix1
$osVhdUri=$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 $locName
```

10. NIC構成を確認する

VPXインスタンスが起動した後、次のコマンドを使用して、VPX NICの `IPConfigs` に割り当てられたIPアドレスを確認できます。

```
$nic.IPConfig
```

11. VPX側の構成を確認する

NetScaler VPXインスタンスが起動すると、プライマリNICのプライマリ `IPconfig` 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、要件に応じて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
```

これで、スタンドアロンモードのNetScaler VPXインスタンスに複数のIPアドレスを設定しました。

Azure展開用の追加PowerShellスクリプト

このセクションでは、Azure PowerShellで以下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のNetScaler VPXペアのプロビジョニング
- 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のNetScaler VPXペアのプロビジョニング

PowerShellコマンドを使用して実行できる構成については、以下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を持つ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 VPXインスタンスに複数の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、またはグループとして管理したいリソースのみを含めることができます。ここで指定された場所は、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ためのすべてのコマンドが同じリソースグループを使用していることを確認してください。

```
$rgName="<resource group name>" $locName="<location name, such as West US>"  
New-AzureRmResourceGroup -Name $rgName -Location $locName
```

例:

```
$rgName = "ARM-VPX"  
  
$locName = "West US"  
  
New-AzureRmResourceGroup -Name $rgName -Location $locName
```

2. ストレージアカウントを作成する

小文字と数字のみを含む、ストレージアカウントの一意的な名前を選択します。

`$saName="<storage account name>"` `$saType="<storage account type>"`、次のいずれかを指定します: `Standard_LRS`、`Standard_GRS`、`Standard_RAGRS`、または `Premium_LRS` `New-AzureRmStorageAccount -Name $saName -ResourceGroupName $rgName -Type $saType -Location $locName`

例:

```
$saName="vpxstorage"  
  
$saType="Standard_LRS"
```

```
New-AzureRmStorageAccount -Name $saName -ResourceGroupName $rgName
-Type $saType -Location $locName
```

3. 可用性セットを作成する

可用性セットは、メンテナンス中などのダウンタイム中に仮想マシンが利用可能であることを維持するのに役立ちます。可用性セットで構成されたロードバランサーは、アプリケーションが常に利用可能であることを保証します。

```
$avName="<availability set name>"
```

```
New-AzureRmAvailabilitySet -Name $avName -ResourceGroupName $rgName -Location
$locName
```

4. 仮想ネットワークを作成する

サブネットが以前に作成されていない場合は、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-AzureRmVirtualNetworkSubnetConfig
-Name backendSubnet -AddressPrefix $BackendAddressPrefix New-
AzureRmVirtualNetwork -Name TestNet -ResourceGroupName $rgName -Location
$locName -AddressPrefix $VnetAddressPrefix -Subnet $FrontendSubnet,
$BackendSubnet
```

例:

```
$FrontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
frontendSubnet -AddressPrefix $FrontendAddressPrefix

$BackendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
backendSubnet -AddressPrefix $BackendAddressPrefix

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、そのNICをNetScaler VPXインスタンスに関連付けます。上記のプロシージャで作成されたフロントエンドサブネットはインデックス0、バックエンドサブネットはインデックス1です。次に、以下の3つの方法のいずれかで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 と DNS ラベルを持つ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、コマンドを使用して利用可能かどうかを確認してください:

```
Test-AzureRmDnsAvailability -DomainQualifiedName $domName -Location $locName
```

```
$nic = New-AzureRmNetworkInterface -Name $nicName -ResourceGroupName $rgName
-Location $locName -SubnetId $vnet.Subnets[$subnetIndex].Id -PublicIpAddressId
$pip.Id
```

例:

```
$nicName="frontendNIC"
```

```
$domName="vpxazure"
```

```
$pip = New-AzureRmPublicIpAddress -Name $nicName -ResourceGroupName  
$rgName -DomainNameLabel $domName -Location $locName -AllocationMethod  
Dynamic
```

```
$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

VM に追加するプライベート (静的) IP アドレスが、指定されたサブネットと同じ範囲であることを確認してください。

```
$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. 仮想オブジェクトを作成する

```
$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イメージを取得する

```
$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にログインするために使用される資格情報を提供します

```
$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:

```
$pubName="citrix"
```

Citrix®のすべてのオファーを表示するには、次のコマンドを使用します。

```
Get-AzureRMVMImageOffer -Location $locName -Publisher $pubName | Select  
Offer
```

```
$offerName="netscalervpx110-6531"
```

特定のオファー名についてパブリッシャーが提供するSKUを知るには、次のコマンドを使用します。

```
Get-AzureRMVMImageSku -Location $locName -Publisher $pubName -Offer  
$offerName | Select Skus
```

8. 仮想マシンを作成する

```
$diskName="<name identifier for the disk in Azure storage, such as OSDisk>"
```

例:


```

$diskName="dynamic"

$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

```

マーケットプレイスにあるイメージから仮想マシンを作成する場合、VMプランを指定するには次のコマンドを使用します。

```
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する

Azureユーザー資格情報を使用してAzureRmAccountにログオンします。

1. リソースグループを作成する

ここで指定された場所は、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。ロードバランサーの作成に使用されるすべてのコマンドが同じリソースグループを使用していることを確認してください。

```
$rgName="<resource group name>"
```

```
$locName="<location name, such as West US>"
```

```
New-AzureRmResourceGroup -Name $rgName -Location $locName
```

例:

```
$rgName = "ARM-LB-NS"
```

```
$locName = "West US"
```

```
New-AzureRmResourceGroup -Name $rgName -Location $locName
```

2. ストレージアカウントを作成する

小文字と数字のみを含む、ストレージアカウントの一意的な名前を選択してください。

```
$saName="<storage account name>"
```

```
$saType="<storage account type>"、いずれかを指定してください: Standard_LRS、  
Standard_GRS、Standard_RAGRS、または Premium_LRS
```

```
New-AzureRmStorageAccount -Name $saName -ResourceGroupName $rgName -Type  
$saType -Location $locName
```

例:

```
$saName="vpxstorage"
```

```
$saType="Standard_LRS"
```

```
New-AzureRmStorageAccount -Name $saName -ResourceGroupName $rgName
-Type $saType -Location $locName
```

3. 可用性セットを作成する

可用性セットで構成されたロードバランサーは、アプリケーションが常に利用可能であることを保証します。

```
$avName="<availability set name>"
```

```
New-AzureRmAvailabilitySet -Name $avName -ResourceGroupName $rgName -Location
$locName
```

4. 仮想ネットワークを作成する

サブネットが以前に作成されていない場合は、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パラメーターの値を選択してください。

このステップで以前に作成した仮想ネットワークに、フロントエンドとバックエンドのサブネットを割り当てます。

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である必要があります。

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 である必要があります。以下同様です。

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、ロードバランスされた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の値の可用性を確認してください。

```
$FIPName = "ELBFIP"
```

```
$frontendIP1 = New-AzureRmLoadBalancerFrontendIpConfig -Name $FIPName  
-PublicIpAddress $publicIP1  
  
$BEPool = "LB-backend-Pool"  
  
$beaddresspool1= New-AzureRmLoadBalancerBackendAddressPoolConfig -Name  
$BEPool
```

6. ヘルスプローブを作成する

ポート9000、間隔5秒のTCPヘルスプローブを作成します。

```
$healthProbe = New-AzureRmLoadBalancerProbeConfig -Name HealthProbe  
-Protocol Tcp -Port 9000 -IntervalInSeconds 5 -ProbeCount 2
```

7. ロードバランシングルールを作成する

ロードバランシングする各サービスに対してLBルールを作成します。

例：

以下の例を使用して、HTTPサービスをロードバランシングできます。

```
$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ルールを作成する

ロードバランシングしないサービスに対してNATルールを作成します。

例えば、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：

Protocol-FrontEndPort-BackendPortのトリプレットは、2つのNATルールで同じであってはいません。

```
$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を作成する

すべてのオブジェクト（NATルール、ロードバランサールール、プローブ構成）をまとめて追加して、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、それぞれのNICを各VPXインスタンスに関連付けます

a) VPX1が関連付けられたNIC1

例：

```
$nicName="NIC1"
```

```
$lbName="ELB"

$bePoolIndex=0

\* Rule indexes starts from 0.

$natRuleIndex=0

$subnetIndex=0

\* Frontend subnet index

$lb=Get-AzureRmLoadBalancer -Name $lbName -ResourceGroupName $rgName

$nic1=New-AzureRmNetworkInterface -Name $nicName -ResourceGroupName
$rgName -Location $locName -Subnet $vnet.Subnets\[ $subnetIndex\]
-LoadBalancerBackendAddressPool $lb.BackendAddressPools\[ $bePoolIndex\]
-LoadBalancerInboundNatRule $lb.InboundNatRules\[ $natRuleIndex\]
```

b) VPX2が関連付けられたNIC2

例：

```
$nicName="NIC2"

$lbName="ELB"
```

```

$bePoolIndex=0

$natRuleIndex=1

\* Second Inbound NAT (SSH) rule we need to use

`$subnetIndex=0

\* Frontend subnet index

$lb=Get-AzureRmLoadBalancer -Name $lbName -ResourceGroupName $rgName

$nic2=New-AzureRmNetworkInterface -Name $nicName -ResourceGroupName
$rgName -Location $locName -Subnet $vnet.Subnets[$subnetIndex\]
-LoadBalancerBackendAddressPool $lb.BackendAddressPools[$bePoolIndex\]
-LoadBalancerInboundNatRule $lb.InboundNatRules[$natRuleIndex\]

```

11. NetScaler VPXインスタンスを作成する

同じリソースグループと可用性セットの一部として2つのNetScaler VPXインスタンスを作成し、それを外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:


```
$vmName="VPX1"
```

```
$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:

```
$vmName="VPX2"

$vmSize="Standard\_A3"
```

```
$avSet=Get-AzureRmAvailabilitySet -Name $avName -ResourceGroupName  
$rgName  
  
$vm2=New-AzureRmVMConfig -VMName $vmName -VMSize $vmSize  
-AvailabilitySetId $avset.Id  
  
$cred=Get-Credential -Message " Type Credentials which will be used to  
login to VPX instance "  
  
$vm2=Set-AzureRmVMOperatingSystem -VM $vm2 -Linux -ComputerName $vmName  
-Credential $cred -Verbose  
  
$vm2=Set-AzureRmVMSourceImage -VM $vm2 -PublisherName $pubName -Offer  
$offerName -Skus $skuName -Version "latest"  
  
$vm2=Add-AzureRmVMNetworkInterface -VM $vm2 -Id $nic2.Id  
  
$diskName="dynamic"  
  
$storageAcc=Get-AzureRmStorageAccount -ResourceGroupName $rgName -Name  
$saName  
  
$osDiskUri1=$storageAcc.PrimaryEndpoints.Blob.ToString() + "vhds2/" +  
$diskName + ".vhd"
```

```
$vm2=Set-AzureRmVMOSDisk -VM $vm2 -Name $diskName -VhdUri $osDiskUri1
-CreateOption fromImage
```

```
Set-AzureRmVMPlan -VM $vm2 -Publisher $pubName -Product $offerName
-Name $skuName
```

```
New-AzureRmVM -ResourceGroupName $rgName -Location $locName -VM $vm2
```

12. 仮想マシンを構成する

両方のNetScaler VPXインスタンスが起動したら、SSHプロトコルを使用して両方のNetScaler VPXインスタンスに接続し、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: 両方のNetScaler VPXインスタンスのコマンドラインで、このコマンドを実行します。

```
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のNetScaler VPXペアのプロビジョニング

Azureユーザー資格情報を使用してAzureRmAccountにログインします。

1. リソースグループを作成する

ここで指定された場所は、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すべてのコマンドが同じリソースグループを使用していることを確認してください。

```
$rgName="\<resource group name\>"
```

```
$locName="\<location name, such as West US\>"
```

```
New-AzureRmResourceGroup -Name $rgName -Location $locName
```

例:

```
$rgName = "ARM-LB-NS"

$locName = "West US"

New-AzureRmResourceGroup -Name $rgName -Location $locName
```

2. ストレージアカウントを作成する

小文字と数字のみを含む、ストレージアカウントの一意的な名前を選択してください。

```
$saName="<storage account name>"
```

`$saType="<storage account type>"`、次のいずれかを指定します: `Standard_LRS`、`Standard_GRS`、`Standard_RAGRS`、または `Premium_LRS`

```
New-AzureRmStorageAccount -Name $saName -ResourceGroupName $rgName -Type $saType -Location $locName
```

例:

```
$saName="vpxstorage"

$saType="Standard_LRS"

New-AzureRmStorageAccount -Name $saName -ResourceGroupName $rgName -Type $saType -Location $locName
```

3. 可用性セットを作成する

可用性セットで構成されたロードバランサーは、アプリケーションが常に利用可能であることを保証します。

```
$avName="<availability set name>"
```

```
New-AzureRmAvailabilitySet -Name $avName -ResourceGroupName $rgName -Location  
$locName
```

4. 仮想ネットワークを作成する

サブネットが以前に作成されていない場合は、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"
```

```
$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パラメーター値を選択してください。

このステップで以前に作成した仮想ネットワークに、フロントエンドとバックエンドのサブネットを割り当てます。

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\$vnnet.Subnets\[0\].Idである必要があります。

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\$vnnet.Subnets\[1\].Idである必要があります、以降同様です。

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

ロードバランシングを行わないサービスに対してNATルールを作成します。

```
$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とバックエンドポートを使用します。

7. ヘルスプローブを作成する

ポート9000、間隔5秒のTCPヘルスプローブを作成します。

```
$healthProbe = New-AzureRmLoadBalancerProbeConfig -Name "HealthProbe"
" -Protocol tcp -Port 9000 -IntervalInSeconds 5 -ProbeCount 2
```

8. ロードバランシングルールを作成する

ロードバランシングする各サービスに対してLBルールを作成します。

例：

以下の例を使用して、HTTPサービスをロードバランシングできます。

```
$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とバックエンドポートを使用してください。

9. ロードバランサーエンティティを作成する

すべてのオブジェクト（NATルール、ロードバランサールール、プローブ構成）をまとめて追加して、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、それぞれの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ある必要があります。

```
$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 VPX 2用です。パラメータ `Private IPAddress` は、要件に応じて任意のプライベートIPを持つことができます。

11. NetScaler VPXインスタンスを作成する

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、それら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：

```
$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:

```
$vmName="VPX2"

$vmSize="Standard_A3"
```

```
$avSet=Get-AzureRmAvailabilitySet -Name $avName -ResourceGroupName  
$rgName  
  
$vm2=New-AzureRmVMConfig -VMName $vmName -VMSize $vmSize  
-AvailabilitySetId $avset.Id  
  
$cred=Get-Credential -Message " Type Credentials which will be used to  
login to VPX instance "  
  
$vm2=Set-AzureRmVMOperatingSystem -VM $vm2 -Linux -ComputerName $vmName  
-Credential $cred -Verbose  
  
$vm2=Set-AzureRmVMSourceImage -VM $vm2 -PublisherName $pubName -Offer  
$offerName -Skus $skuName -Version "latest"  
  
$vm2=Add-AzureRmVMNetworkInterface -VM $vm2 -Id $backendnic2.Id  
  
$diskName="dynamic"  
  
$storageAcc=Get-AzureRmStorageAccount -ResourceGroupName $rgName -Name  
$saName  
  
$osDiskUri1=$storageAcc.PrimaryEndpoints.Blob.ToString() + "vhds2/" +  
$diskName + ".vhd"
```

```
$vm2=Set-AzureRmVMOSDisk -VM $vm2 -Name $diskName -VhdUri $osDiskUri1
-CreateOption fromImage
```

```
Set-AzureRmVMPlan -VM $vm2 -Publisher $pubName -Product $offerName
-Name $skuName
```

```
New-AzureRmVM -ResourceGroupName $rgName -Location $locName -VM $vm2
```

12. 仮想マシンを構成する

両方のNetScaler VPXインスタンスが起動したら、SSHプロトコルを使用して両方のNetScaler VPXインスタンスに接続し、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、同じ構成コマンドのセットを実行します。

b) アクティブ-パッシブ: 両方のNetScaler VPXインスタンスのコマンドラインで、このコマンドを実行します。

```
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 よくある質問

- 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、オンプレミスのアップグレード手順と異なりますか？

いいえ。Microsoft AzureクラウドのNetScaler VPXインスタンスは、標準のNetScaler VPXアップグレード手順を使用して、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の手順のいずれか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、Microsoft Azureクラウド用のNetScaler VPXイメージを使用してください。

NetScaler VPXアップグレードビルドをダウンロードするには、**NetScaler downloads > NetScaler Firmware**に移動します。

- AzureでホストされているNetScaler VPXインスタンスで観測されるMAC移動とインターフェースミュートを修正する方法？

AzureのマルチNIC環境では、デフォルトで、すべてのデータインターフェースでMAC移動とインターフェースミュートが発生する可能性があります。Azure環境でMAC移動とインターフェースミュートを回避するには、CitrixはNetScaler VPXインスタンスのデータインターフェースごとに（タグなしで）VLANを作成し、AzureのNICのプライマリIPをバインドすることを推奨します。

詳細については、[CTX224626](#)の記事を参照してください。

Google Cloud Platform に NetScaler VPX インスタンスを展開する

Google Cloud Platform (GCP) に NetScaler VPX インスタンスを展開できます。GCP の VPX インスタンスを使用すると、GCP の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 の負荷分散およびトラフィック管理機能を利用できます。GCP に VPX インスタンスをスタンドアロンインスタンスとして展開できます。シングル NIC とマルチ NIC の両方の構成がサポートされています。

サポートされる機能

使用されるライセンス/バージョンタイプに基づいて、すべての Premium、Advanced、および Standard 機能が GCP でサポートされています。

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 vCPU と 4 GB の RAM が必要です。

注意事項

展開を開始する前に、以下の GCP 固有の点について考慮してください。

- インスタンス作成後、ネットワークインターフェイスを追加または削除することはできません。
- マルチ NIC 展開の場合、各 NIC 用に個別の VPC ネットワークを作成します。1 つの NIC は 1 つのネットワークのみに関連付けることができます。
- 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イスを持つインスタンスには、最低 4 vCPU が必要です。

- IP フォワーディングが必要な場合は、インスタンスの作成時および NIC の構成時に IP フォワーディングを有効にする必要があります。

シナリオ: 複数のNIC、複数のIPを持つスタンドアロンVPXインスタンスを展開する

このシナリオでは、GCPにNetScaler VPXスタンドアロンインスタンスを展開する方法を説明します。このシナリオでは、複数のNICを持つスタンドアロンVPXインスタンスを作成します。このインスタンスは、バックエンドサーバー（サーバーファーム）と通信します。



以下の目的のために3つのNICを作成します。

NIC	目的	関連するVPCネットワーク
NIC 0	管理トラフィックを処理します (NetScaler IP)	管理ネットワーク
NIC 1	クライアント側トラフィックを処理します (VIP)	クライアントネットワーク
NIC 2	バックエンドサーバーと通信します (SNIP)	バックエンドサーバーネットワーク

以下の間で必要な通信ルートを設定します。

- VPXインスタンスとバックエンドサーバー。
- 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。

展開手順の概要

1. 3つの異なるNIC用に3つのVPCネットワークを作成します。
2. ポート22、80、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

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。

ステップ1. 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 > ネットワーキング > VPCネットワーク > VPCネットワークを作成**にログインします。スクリーンショットに示されているように、必要なフィールドをすべて入力し、**作成**をクリックします。

シナリオ用のVPCネットワーク



同様に、クライアント側およびサーバー側のNIC用に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にある必要があります。このシナリオでは asia-east1 です。

ステップ2. ポート22、80、443のファイアウォールルールを作成します。

各VPCネットワークに対して、SSH (ポート22)、HTTP (ポート80)、および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。



ステップ3. VPXインスタンスを作成します。

1. GCPコンソールにログオンします。
2. [GCP Marketplace](#)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ます。



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. 展開フォームを完了し、**展開**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. 展開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、スタンドアロンVPXインスタンスの展開

このシナリオでは、GCPでシングルNICを持つNetScaler VPXスタンドアロンインスタンスを展開する方法を説明します。この展開を実現するために、エイリアスIPアドレスが使用されます。

シングルNICによるGCPスタンドアロン展開(/en-us/vpx/media/deploy-vpx-gcp-standalone-with-single-nic.png)

単一のNIC (NIC0) を作成し、以下の目的で使用します。

- 管理ネットワークで管理トラフィック (NetScaler IP) を処理します。
- クライアントネットワークで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でバックエンドサーバー (SNIP) と通信します。

以下の間で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。
- 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。

展開手順の概要

1. NIC0 用のVPCネットワークを作成します。
2. ポート22、80、443のファイアウォールルールを作成します。
3. シングルNICを持つ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. インスタンス上の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。

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。

ステップ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 > ネットワーキング > 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、**作成**をクリックします。



ステップ2. ポート22、80、443のファイアウォールルールを作成します。

VPCネットワーク用にSSH (ポート22)、HTTP (ポート80)、および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。



ステップ3. シングルNICでインスタンスを作成します。

シングル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. **Compute**の下で、**Compute Engine**にカーソルを合わせ、**Images**を選択します。
3. イメージを選択し、**Create Instance**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. **Management, security, disks, networking**ウィンドウから**Networking**タブをクリックします。
6. **Network interfaces**の下で、**Edit**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. **Network interfaces**ウィンドウの**Network**の下で、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。**External IP addresses**の下で、**Create IP address**をクリックします。
9. **Reserve a static address**ウィンドウで、名前と説明を追加し、**Reserve**をクリックします。
10. **Create**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はVMインスタンスの下に表示されます。

ステップ 4. VPX インスタンスにエイリアス IP アドレスを追加します。

VIP および SNIP アドレスとして使用するために、2 つのエイリアス IP アドレスを VPX インスタンスに割り当てます。

注:

VIP または SNIP を構成するために、VPX インスタンスのプライマリ内部 IP アドレスを使用しないでください。

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ース**ウィンドウで、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、エイリアス IP アドレスを入力します。



4. **完了**をクリックし、次に**保存**をクリックします。
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。



ステップ 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。

「IPv4アドレスを追加」 (/en-us/vpx/media/adsc-add-ipv4-address.png)

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:

- VMインスタンスのVPCサブネット用に構成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:

- VMインスタンスのVPCサブネット用に構成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。

ステップ6. 負荷分散仮想サーバーを追加します。

1. NetScaler GUIで、**Configuration > Traffic Management > Load Balancing > Virtual Servers** に移動し、「**Add**」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。

「負荷分散仮想サーバーを作成」 (/en-us/vpx/media/create-lb-vserver-adc.png)

ステップ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、**Configuration > Traffic Management > Load Balancing > Services** に移動し、**Add** をクリックします。
2. サービス名、IPアドレス、プロトコル、ポートに必要な値を入力し、**OK** をクリックします。

ステップ8. サービス/サービスグループをインスタンス上のロードバランシング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 > 仮想サーバー** に移動します。
2. **ステップ6** で設定したロードバランシング仮想サーバーを選択し、**Edit** をクリックします。
3. **Service and Service Groups** ウィンドウで、**No Load Balancing Virtual Server Service Binding** 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、**Bind** をクリックします。

GCPにVPXインスタンスをデプロイした後の注意点

- ユーザー名 `nsroot` とパスワードとしてインスタンスIDを使用して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、通常の `show techsupport` の代わりにコマンド `shell /netscaler/showtech_cloud.pl` を実行します。
- GCPコンソールからNetScaler VMを削除した後、関連するNetScaler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。

```
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 VPXインスタンスには有効なライセンスが必要です。GCPで実行されているNetScaler VPXインスタンスで利用可能なライセンスオプションは次のとおりです。

BYOL (Bring your own license): BYOLオプションを使用するには、次の手順に従います。

- NetScaler®ウェブ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。
- 生成された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:** このライセンスモデルでは、利用可能なライセンスプールからライセンスをチェックアウトし、不要になったときにチェックインし直すことができます。詳細については、「[NetScaler VPX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。

注:

GCP上のNetScaler VPXインスタンスでは、サブスクリプションベースのライセンスはサポートされなくなりました。

GCPでサポートされているNetScaler VPX製品

次の表に、GCPでサポートされているNetScaler VPX製品を示します。

サポートされているVPX製品
NetScaler VPX - お客様ライセンス
NetScaler VPX FIPS - お客様ライセンス

サポートされているGCPマシンタイプファミリー

マシンタイプファミリー	最小マシンタイプ
汎用マシン	e2-ミディアム、e2-スタンダード-2、e2-ハイメモリ-2、n1-スタンダード-2、n1-ハイメモリ-2、n2-スタンダード-2、n2-ハイメモリ-2、n2d-スタンダード-2、n2d-ハイメモリ-2
コンピューティング最適化マシン	c2-standard-4, c2d-standard-2, c2d-highmem-2

NetScaler VPXインスタンスをデプロイするためのGDMテンプレート

NetScaler VPX Google Deployment Manager (GDM) テンプレートを使用して、GCP に VPX インスタンスをデプロイできます。詳細については、[NetScaler GDM Templates](#)を参照してください。

NetScaler マーケットプレイスイメージ

GDMテンプレート内のイメージを使用して、NetScalerアプライアンスを起動できます。

次の表に、GCPマーケットプレイスで利用可能なイメージを示します。

リリース	イメージ名	イメージの場所
13.1	citrix-adc-vpx-express-13-1-53-17	projects/citrix-master-project/global/images/citrix-adc-vpx-express-13-1-53-17
13.1	シトリックス-エーディー シー-バイピーエックス-200- エンタープライズ-13-1-53-17	projects/citrix-master-project/global/images/citrix-adc-vpx-200-enterprise-13-1-53-17
13.1	シトリックス-エーディー シー-バイピーエックス-1000- プラチナム-13-1-53-17	projects/citrix-master-project/global/images/citrix-adc-vpx-1000-platinum-13-1-53-17

リリース	イメージ名	イメージの場所
13.1	シトリックス-エーディー シー-バイピーエックス-5000- プラチナ-13-1-53-17	projects/citrix-master-project/ global/images/citrix-adc- vpx-5000- platinum-13-1-53-17
13.1	シトリックス-エーディー シー-バイピーエックス-ビー ワイオーエル-13-1-53-17	projects/citrix-master-project/ global/images/citrix-adc-vpx- byol-13-1-53-17

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

- [Google Cloud Platform に VPX 高可用性ペアを展開する](#)

Google Cloud Platform で VPX 高可用性ペアを展開する

Google Cloud Platform (GCP) 上で2つのNetScaler 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。一方のインスタンスをプライマリノードとして、もう一方をセカンダリノードとして構成すると、プライマリノードが接続を受け入れ、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。何らかの理由でプライマリノードが接続を受け入れられなくなった場合、セカンダリ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のいずれかに配置できます。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
- クライアント向けサブネット (VIP)
- バックエンド向けサブネット (SNIP、MIP など)

Citrix® は、標準の VPX インスタンスに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、次の方法で展開できます。

- [外部静的 IP アドレスを使用する](#)
- [プライベート IP アドレスを使用する](#)
- [プライベート IP アドレスを持つ単一 NIC VM を使用する](#)

GCP で VPX 高可用性ペアを展開するための GDM テンプレート

NetScaler Google Deployment Manager (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上のVPX高可用性ペアに対する転送ルールサポート

転送ルールを使用して、GCP上にVPX高可用性ペアを展開できます。

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は一致している必要があります。

例:

`us-east1` リージョンに高可用性ペアがあり、プライマリVPXは `us-east1-b` ゾーンに、セカンダリVPXは `us-east1-c` ゾーンにあります。プライマリVPXには、`us-east1-b` ゾーンのターゲットインスタンスで転送ルールが構成されています。フェイルオーバー時に転送ルールを更新するには、`us-east1-c` ゾーンのセカンダリVPXにターゲットインスタンスを構成します。

制限事項

バックエンドでターゲットインスタンスが構成されている転送ルールのみが、VPX高可用性展開でサポートされます。

Google Cloud Platformで外部静的IPアドレスを持つVPX高可用性ペアを展開する

GCPで外部静的IPアドレスを使用してVPX高可用性ペアを展開できます。プライマリノードのクライアントIPアドレスは、外部静的IPアドレスにバインドされている必要があります。フェイル

オーバーが発生すると、トラフィックを再開するために外部静的IPアドレスがセカンダリノードに移動されます。

静的外部IPアドレスは、解放を決定するまでプロジェクト用に予約される外部IPアドレスです。サービスにアクセスするためにIPアドレスを使用する場合、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。

開始する前に

- [Google Cloud PlatformにNetScaler VPXインスタンスを展開する](#)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展開にも適用されます。
- GCPプロジェクトで**Cloud Resource Manager 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。

```
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",

]

```

- 管理インターフェース以外のインターフェースでエイリアスIPアドレスを構成している場合は、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、フェイルオーバー時に新しいプライマリに更新するために、[GCPでのVPX高可用性ペアの転送ルールサポート](#)に記載されている制限事項と要件をお読みください。

Google Cloud PlatformでVPX HAペアを展開する方法

HA展開の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。例: Asia-east。
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。これらは同じゾーンまたは異なるゾーンに配置できます。例: Asia east-1aおよびAsia east-1b。
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 HA 設定を構成します。

ステップ 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。Citrix® は、管理 NIC、クライアント NIC、およびサーバー 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推奨しています。

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 > ネットワーク > VPC ネットワーク > VPC ネットワークを作成** にログインします。
2. 必要なフィールドに入力し、**作成** をクリックします。

詳細については、[Google Cloud Platform に NetScaler VPX インスタンスを展開する](#) の「**VPC ネットワークを作成する**」セクションを参照してください。

ステップ 2. 2 つの VPX インスタンスを作成する

シナリオ: 複数の NIC、複数の IP を持つスタンドアロン VPX インスタンスを展開する 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 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ース > 外部 IP** に移動し、**IP アドレスを作成** 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、静的外部 IP アドレスは古いプライマリから新しいプライマリに移動してアタッチされます。詳細については、Google Cloud ドキュメント [静的外部 IP アドレスの予約](#) を参照してください。

VPX インスタンスを構成したら、VIP および SNIP アドレスを構成できます。詳細については、[NetScaler が所有する IP アドレスの構成](#) を参照してください。

ステップ 3. 高可用性を構成する

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HAを構成できます。

GUIを使用したHAの構成

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. ユーザー名 **nsroot** と、GCPコンソールからのノードのインスタンスIDをパスワードとして使用して、インスタンスにログインします。
2. **Configuration > System > High Availability > Nodes**に移動し、**Add**をクリックします。

3. **Remote Node IP address**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. **Turn on INC (Independent Network Configuration) mode on self node**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。

1. ユーザー名 **nsroot** と、GCPコンソールからのノードのインスタンスIDをパスワードとして使用して、インスタンスにログオンします。
2. **Configuration > System > High Availability > Nodes**に移動し、**Add**をクリックします。
3. **Remote Node IP address**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. **Turn on INC (Independent Network Configuration) mode on self node**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**ページで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。

セカンダリノードの同期状態(/en-us/vpx/media/sync-state-success-external-static-ip.png)

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つことになります。

ステップ 2。両方のノードで仮想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 > IPs > IPv4s**に移動し、**Add**をクリックします。
2. プライマリVIPアドレスを追加するには、次の手順に従います。
 - a. プライマリインスタンスのクライアント向けインターフェースの内部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用に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。

3. プライマリSNIPアドレスを追加するには、次の手順に従います。
 - a. プライマリインスタンスのサーバー向けインターフェースの内部IPアドレスと、プライマリインスタンスのサーバーサブネット用に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。
4. 以下の手順に従って、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向けインターフェースの内部IPアドレスと、VMインスタンス内のクライアントサブネット用に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。

プライマリノードでVIPとSNIPを作成する(/en-us/vpx/media/primary-vip-snip.png)

セカンダリノードで、以下の手順を実行します。

1. **System > Network > IPs > IPv4s**に移動し、**Add**をクリックします。
2. 以下の手順に従って、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向けインターフェースの内部IPアドレスと、VMインスタンス内のクライアントサブネット用に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. 以下の手順に従って、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ー向けインターフェースの内部IPアドレスと、セカンダリインスタンス内のサーバーサブネット用に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。

セカンダリノードでVIPとSNIPを作成する(/en-us/vpx/media/secondary-vip-snip.png)

ステップ 3. IP セットを追加し、両方のインスタンスのセカンダリ VIP に IP セットをバインドします。

プライマリノードで、次の手順を実行します。

1. **System > Network > IP Sets > Add** に移動します。
2. IP セット名を追加し、**Insert** をクリックします。

3. **IPv4s** ページから、仮想 IP (セカンダリ VIP) を選択し、**Insert** 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。

プライマリ IP セットを作成する(/en-us/vpx/media/primary-ip-set.png)

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. IP セット名を追加し、**Insert** をクリックします。
3. **IPv4s** ページから、仮想 IP (セカンダリ VIP) を選択し、**Insert** 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 IP セットを作成する(/en-us/vpx/media/secondary-ip-set.png)

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負荷分散仮想サーバーを追加します。

1. 「**構成 > 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)、ポートに必要な値を入力します。



3. **More** をクリックします。 **IP Range IP Set Settings** 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. 「**構成 > 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、ポートに必要な値を入力し、**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. **構成 > 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. **ステップ 4** で構成した負荷分散仮想サーバーを選択し、**Edit** 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ディングなし** をクリックします。
4. **ステップ 5** で構成したサービスを選択し、**Bind** をクリックします。

構成を保存します。強制フェイルオーバー後、セカンダリが新しいプライマリになります。古いプライマリVIPの外部静的IPは、新しいセカンダリVIPに移動します。

CLIを使用して高可用性を構成する

ステップ1. 両方のインスタンスでINCモードの高可用性を設定します。

プライマリノードで、次のコマンドを入力します。

```
add ha node 1 <sec_ip> -inc ENABLED
```

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 <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. 両方のノードに仮想IPとサブネットIP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_vip> <subnet> -type VIP
```

```
add ns ip <secondary_vip> <subnet> -type VIP
```

```
add ns ip <primary_snip> <subnet> -type SNIP
```

primary_vip は、プライマリインスタンスのクライアント向けインターフェースの内部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向けインターフェースの内部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ー向けインターフェースの内部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 <secondary_vip> <subnet> -type VIP

add ns ip <secondary_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向けインターフェースの内部IPアドレスを指します。

`secondary_snip` は、セカンダリインスタンスのサーバー向けインターフェースの内部IPアドレスを指します。

ステップ3. 両方のインスタンスでIPセットを追加し、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、次のコマンドを入力します。

```
add ipset <ipsetname>

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。

```
add ipset <ipsetname>

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ーを追加します。

次のコマンドを入力します。


```
add <server_type> vserver <vserver_name> <protocol> <primary_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。

```
add service <service_name> <service_ip_address> <protocol> <port>
```

ステップ6. プライマリインスタンスで、サービス/サービスグループをロードバランシング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、インスタンスを再起動した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にアタッチされた外部IPアドレスが、フェイ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リクエストを行い、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 > 強制フェイルオーバー** 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。フェイルオーバー後、外部IPアドレスはセカンダリのクライアントNICに移動しているはずです。

3. 外部IPにcURLリクエストを発行し、再び到達可能であることを確認します。

Google Cloud PlatformでプライベートIPアドレスを持つシングルNIC VPX高可用性ペアを展開する

GCPでプライベートIPアドレスを使用して、シングルNIC VPX高可用性ペアを展開できます。クライアントIP（VIP）アドレスは、プライマリノードでエイリアスIPアドレスとして構成する必要があります。フェイルオーバー時には、トラフィックを再開するためにクライアント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範囲として構成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。

開始する前に

- 「[Google Cloud PlatformにNetScaler VPXインスタンスを展開する](#)」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**Cloud Resource Manager API**を有効にします。
- インスタンスの作成時に、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以下のIAM権限があることを確認してください。

```
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にインターネットアクセスがない場合は、VPCサブネットで**Private 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、フェイルオーバー時に新しいプライマリに更新するための「[GCP上のVPX高可用性ペアに対する転送ルールサポート](#)」に記載されている制限事項と要件をお読みください。

Google Cloud PlatformでVPX高可用性ペアを展開する方法

シングルNICでHAペアを展開する手順の概要を以下に示します。

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。これらは同じゾーンまたは異なるゾーンに配置できます。例えば、アジア東部-1aとアジア東部-1bです。
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HA設定を構成します。

ステップ 1. 1つのVPCネットワークを作成する

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 console > Networking > VPC network > Create VPC Network** にログインします。
2. 必須フィールドに入力し、**作成** をクリックします。

詳細については、[Google Cloud Platform に NetScaler VPX インスタンスを展開する](#) 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。

ステップ 2. 2つのVPXインスタンスを作成する

[シナリオ: シングルNIC、スタンドアロンVPXインスタンスを展開する](#) に記載されているステップ 1からステップ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アドレスはプライマリノードにのみ割り当て、サーバーエイリアスIPアドレスはプライマリノードとセカンダリノードに割り当てます。VIPまたはSNIPを構成するために、VPXインスタンスの内部IPアドレスを使用しないでください。

クライアントおよびサーバーエイリアス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** 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範囲の追加** をクリックし、サーバーエイリアスIPアドレスを入力します。



サーバーエイリアス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、クライアントエイリアスIPアドレスは古いプライマリから移動され、新しいプライマリにアタッチされます。

VPXインスタンスを設定したら、仮想 (VIP) およびサブネットIP (SNIP) アドレスを設定できます。詳細については、「[NetScalerが所有するIPアドレスの設定](#)」を参照してください。

ステップ3. 高可用性を構成する

Google Cloud Platformでインスタンスを作成したら、NetScaler GUIまたはCLIを使用して高可用性を構成できます。

GUIを使用して高可用性を構成する

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. ユーザー名 **nsroot** と、GCPコンソールからのノードのインスタンスIDをパスワードとして使用してインスタンスにログインします。
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、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. 「**Turn on INC (Independent Network Configuration) mode on self node**」チェックボックスをオンにします。
5. 「**作成**」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. ユーザー名 **nsroot** と、GCP コンソールからのノードのインスタンス ID をパスワードとして使用して、インスタンスにログインします。
2. 「**Configuration > System > High Availability > Nodes**」に移動し、「**Add**」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. 「**Turn on INC (Independent Network Configuration) mode on self node**」チェックボックスをオンにします。
5. 「**作成**」をクリックします。

続行する前に、「**ノード**」ページで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。



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、セカンダリノードはプライマリノードと同じログイン資格情報を持つようになります。

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 > IPs > IPv4s**」に移動し、「**Add**」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:

- a. プライマリVMインスタンスのVPCサブネット用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。
- 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:

- a. プライマリVMインスタンスのVPCサブネット用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。
- 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。

プライマリのクライアントおよびサーバーIPアドレスを作成する(/en-us/vpx/media/gcp-single-nic-create-ipv4-address.png)

セカンダリノードで、次の手順を実行します。

1. 「**System > Network > IPs > IPv4s**」に移動し、「**Add**」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:

- a. プライマリVMインスタンスのVPCサブネット用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。
- 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:

- a. セカンダリVMインスタンスのVPCサブネット用に構成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。
- 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。

セカンダリのクライアントおよびサーバーIPアドレスを作成する(/en-us/vpx/media/gcp-single-nic-create-ipv4-address-secondary.png)

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. **構成 > 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。

- 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入力し、**OK**をクリックします。

LB仮想サーバクライアントエイリアス(/en-us/vpx/media/gcp-single-nic-lb-vserver-client-alias.png)

ステップ4。 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。

- 構成 > トラフィック管理 > 負荷分散 > サービス > 追加**に移動します。
- 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入力し、**OK**をクリックします。

ステップ5。 プライマリノードで、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。

- 構成 > トラフィック管理 > 負荷分散 > 仮想サーバー**に移動します。
- ステップ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。
- 「サービスとサービスグループ」タブで、「ロードバランシングなしの仮想サーバサービスバインディング」をクリックします。
- 「**手順4**」で設定したサービスを選択し、「**バインド**」をクリックします。

手順6。 設定を保存します。

強制フェールオーバー後、セカンダリが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) は新しいプライマリに移動します。

CLI を使用して高可用性を構成する

手順1。 NetScaler CLI を使用して、両方のインスタンスで高可用性を「**INC Enabled**」モードで設定します。

プライマリノードで、次のコマンドを入力します。

```
add ha node 1 <sec_ip> -inc ENABLED
```

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 <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。

手順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インスタンスでクライアントサブネット用に構成されたエイリアスIPアドレスとネットマスクを入力します。

```
add ns ip <primary_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インスタンスでクライアントサブネット用に構成されたエイリアスIPアドレスとネットマスクを入力します。

```
add ns ip <secondary_server_alias_ip> <subnet> -type SNIP
```

注:

VMインスタンスでサーバーサブネット用に構成されたエイリアスIPアドレスとネットマスクを入力します。

ステップ 3. プライマリノードに仮想サーバーを追加します。

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. プライマリノードで、サービスまたはサービスグループをロードバランシング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、インスタンスを再起動した後に設定が失われます。

Google Cloud PlatformでプライベートIPアドレスを使用してVPX高可用性ペアを展開する

GCPでプライベートIPアドレスを使用してVPX高可用性ペアを展開できます。クライアントIP (VIP) は、プライマリノードでエイリアスIPアドレスとして構成する必要があります。フェイルオーバー時には、トラフィックを再開するためにクライアント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。

開始する前に

- [Google Cloud PlatformへのNetScaler VPXインスタンスの展開](#)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**Cloud Resource Manager API**を有効にします。
- インスタンスの作成時に、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.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で外部IPアドレスを構成している場合は、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。

```
REQUIRED_INSTANCE_IAM_PERMS = [  
  
  "compute.addresses.use"  
  
  "compute.instances.addAccessConfig",  
  
  "compute.instances.deleteAccessConfig",  
  
  "compute.networks.useExternalIp",
```

```
"compute.subnetworks.useExternalIp",  
  
]
```

- VMにインターネットアクセスがない場合は、管理サブネットでは**Private 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高可用性ペアに対する転送ルールサポート](#)に記載されている制限事項と要件を読み、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でVPX高可用性ペアを展開する方法

高可用性展開の手順の概要を以下に示します。

1. 同じリージョンにVPCネットワークを作成します。例: Asia-east。
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。これらは同じゾーンまたは異なるゾーンに配置できます。例: Asia east-1aおよびAsia east-1b。
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。

ステップ 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。Citrix® は、管理 NIC、クライアント NIC、およびサーバー 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 > ネットワーク > VPC ネットワーク > VPC ネットワークを作成**にログインします。
2. 必須フィールドに入力し、**作成**をクリックします。

詳細については、[Google Cloud Platform に NetScaler VPX インスタンスを展開する](#)の「**VPC ネットワークを作成する**」セクションを参照してください。

ステップ 2. 2 つの VPX インスタンスを作成する

シナリオ: 複数の NIC、複数の IP を持つスタンドアロン VPX インスタンスを展開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:

プライマリノードにクライアントエイリアス IP アドレスを割り当てます。VIP を構成するために VPX インスタンスの内部 IP アドレスを使用しないでください。

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**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**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。

クライアントエイリアスIPアドレス(/en-us/vpx/media/client-alias-ip-address.png)

ネットワークインターフェイス(/en-us/vpx/media/network-interface-client.png)

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、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アタッチされます。

VPXインスタンスを構成した後、仮想IP (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、「[NetScalerが所有するIPアドレスの構成](#)」を参照してください。

手順3. 高可用性を構成する

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を使用して高可用性を構成する

手順1. 両方のノードでINC有効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. ユーザー名 **nsroot** と、GCPコンソールから取得したノードのインスタンスIDをパスワードとして使用して、インスタンスにログオンします。
2. **Configuration > System > High Availability > Nodes** に移動し、**Add** をクリックします。
3. **Remote Node IP address** 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. **Turn on INC (Independent Network Configuration) mode on self node** 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。

1. ユーザー名 **nsroot** と、パスワードとして GCP コンソールからのノードのインスタンス ID を使用してインスタンスにログオンします。
2. **Configuration > System > High Availability > Nodes** に移動し、**Add** をクリックします。
3. **Remote Node IP address** 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. **Turn on INC (Independent Network Configuration) mode on self node** チェックボックスを選択します。
5. **Create** をクリックします。

さらに進む前に、**Nodes** ページで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。



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、セカンダリノードはプライマリノードと同じログオン資格情報を持つようになります。

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 > IPs > IPv4s** に移動し、**Add** 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:
 - a. VM インスタンスのクライアントサブネット用に構成されたエイリアス 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. 「**作成**」をクリックします。
3. サーバーIP (SNIP) アドレスを作成するには:
 - a. プライマリインスタンスのサーバー側インターフェースの内部IPアドレスと、サーバーサブネット用に構成されたネットマスクを入力します。
 - b. 「**IPタイプ**」フィールドで、ドロップダウンメニューから「**サブネットIP**」を選択します。

- c. 「作成」をクリックします。

プライマリのクライアントおよびサーバーIPアドレスを作成します(/en-us/vpx/media/create-ipv4-address.png)

セカンダリノードで、次の手順を実行します。

1. 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:
 - 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用に構成されたエイリアスIPアドレスとネットマスクを入力します。
 - b. 「IPタイプ」フィールドで、ドロップダウンメニューから「サブネットIP」を選択します。
 - c. 「作成」をクリックします。
3. サーバーIP (SNIP) アドレスを作成するには:
 - a. セカンダリインスタンスのサーバー側インターフェースの内部IPアドレスと、サーバーサブネット用に構成されたネットマスクを入力します。
 - b. 「IPタイプ」フィールドで、ドロップダウンメニューから「サブネットIP」を選択します。
 - c. 「作成」をクリックします。

セカンダリのクライアントおよびサーバーIPアドレスを作成(/en-us/vpx/media/create-ipv4-address-secondary.png)

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. 構成 > トラフィック管理 > ロードバランシング > 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) 、ポートに必要な値を入力し、「OK」をクリックします。

LB仮想サーバークライアントエイリアス(/en-us/vpx/media/lb-vserver-client-alias.png)

ステップ4. 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. 構成 > トラフィック管理 > ロードバランシング > 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、ポートに必要な値を入力し、「OK」をクリックします。

ステップ5. プライマリノードで、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. **構成 > トラフィック管理 > ロードバランシング > 仮想サーバー** に移動します。
2. **ステップ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ディングなし**」をクリックします。
4. **ステップ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。

強制フェイルオーバー後、セカンダリが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) とサーバーエイリアスIP (SNIP) は、新しいプライマリに移動します。

CLI を使用して高可用性を構成する

ステップ 1. NetScaler CLI を使用して、両方のインスタンスで **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、次のコマンドを入力します。

```
add ha node 1 <sec_ip> -inc ENABLED
```

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 <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:

VM インスタンスでクライアントサブネット用に構成されたエイリアス 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ー側インターフェースの内部 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 インスタンスでクライアントサブネット用に構成されたエイリアス IP アドレスとネットマスクを入力します。

```
add ns ip <secondary_snip> <subnet> -type SNIP
```

secondary_snip は、セカンダリインスタンスのサーバー側インターフェースの内部 IP アドレスを指します。

注:

VM インスタンスでサーバーサブネット用に構成された IP アドレスとネットマスクを入力します。

ステップ 3. プライマリノードに仮想サーバーを追加します。

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. プライマリノードで、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。

次のコマンドを入力します。

```
bind <server_type> vservers <vservers_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、インスタンスを再起動した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

Google Cloud VMware Engine (GCVE) は、専用のベアメタル Google Cloud Platform インフラストラクチャから構築された vSphere クラスター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つのホストですが、追加のホストは一度に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、および NSX-T が含まれています。

GCVE を使用すると、Google Cloud Platform 上に、必要な数の ESX ホストを備えた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 のデプロイをサポートしています。GCVE はオンプレミスの 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 NetScaler VPX デプロイメントとまったく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 Web ベースの vCenter および 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) とサーバー VM 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 ホストクラスター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。

GCVE アーキテクチャ(/en-us/vpx/media/gcve-on-gcp-architecture.png)

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 Engine とその前提条件の詳細については、「[Google Cloud VMware Engine documentation](#)」を参照してください。
- Google Cloud VMware Engine のデプロイの詳細については、「[Deploy a Google Cloud VMware Engine private cloud](#)」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、「[Access an Google Cloud VMware Engine private cloud](#)」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、「[Add a network segment in Google Cloud VMware Engine](#)」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、「[Licensing overview](#)」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されている必要があります。

VMware クラウドのハードウェア要件

以下の表は、VMware SDDCが各VPX nCore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ています。

表1. NetScaler 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	要件
メモリ	2 GB
仮想CPU (vCPU)	2

コンポーネント	要件
仮想ネットワークインターフェイス	VMware SDDCでは、VPXハードウェアがバージョン7以降にアップグレードされている場合、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。
ディスク容量	20 GB

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です。

VPX仮想アプライアンスを本番環境で使用する場合、メモリ割り当て全体を予約する必要があります。

OVF Tool 1.0 のシステム要件

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 Tool をインストールするための最小システム要件を示します。

表 2. OVF Tool インストールの最小システム要件

コンポーネント	要件
オペレーティングシステム	VMware の詳細な要件については、 http://kb.vmware.com/ で「OVF Tool User Guide」PDF ファイルを検索してください。
CPU	最小 750 MHz、1 GHz 以上を推奨
RAM	最小 1 GB、2 GB を推奨
NIC	100 Mbps 以上の NIC

OVF のインストールについては、<http://kb.vmware.com/> で「OVF Tool User Guide」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、Open Virtual Machine (OVF) 形式標準に準拠しています。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てください。新規ユーザーリンクをクリックし、手順に従って新しいCitrixアカウントを作成してください。

ログイン後、Citrix®ホームページから次のパスに移動します。

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-\\<リリース番号\\>-\\<ビルド番号\\>-disk1.vmdk (例: NSVPX-ESX-13.0-79.64-disk1.vmdk)
- NSVPX-ESX-\\<release number\\>-\\<build number\\>.ovf (例: NSVPX-ESX-13.0-79.64.ovf)
- NSVPX-ESX-\\<リリース番号\\>-\\<ビルド番号\\>.mf (例: NSVPX-ESX-13.0-79.64.mf)

Google Cloud VMware Engineのデプロイ

1. [GCVEポータル](#)にログインし、ホームに移動します。



2. 新しい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ーを作成するには、少なくとも3つのESXiホストを選択します。
- vSphere/vSanサブネットCIDR範囲フィールドには、/22アドレス空間を使用します。
- HCXデプロイネットワークCIDR範囲フィールドには、/26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他のGCPサブネット（仮想ネットワーク）と重複しないことを確認してください。

「GCVEプライベートクラウド」 (/en-us/vpx/media/gcve-private-cloud.png)

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。

「プライベートクラウド設定の確認」 (/en-us/vpx/media/gcve-review-private-cloud-settings.png)

5. 「作成」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時間かかる場合があります。
6. 作成されたプライベートクラウドを確認するには、「リソース」に移動します。

「プライベートクラウドの作成に成功」 (/en-us/vpx/media/gcve-successful-private-cloud.png)

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、以下のドキュメントを参照してください。

- [VPNゲートウェイ](#)
- [VPNを使用した接続](#)

プライベートクラウドの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。

「GCVEリソース」 (/en-us/vpx/media/gcve-resources.png)

2. vCenterの資格情報をメモします。

「GCVE vCenter資格情報」 (/en-us/vpx/media/gcve-vcenter-credentials.png)

3. **LAUNCH VSPHERE CLIENT**をクリックするか、**VSPHERE MANAGEMENT NETWORK**に移動して**vCenter Server Appliance FQDN**をクリックし、vSphereクライアントを起動します。



4. この手順のステップ2に記載されているvCenterの資格情報を使用して、VMware vSphereにログインします。



5. vSphereクライアントで、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セグメントを作成する

Google Cloud VMware EngineコンソールのNSX ManagerからNSX-Tセグメントを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ており、これらのセグメント上のワークロードは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で、**Summary > NSX-T login info**の下にある**View**を選択します。



2. NSX-Tの資格情報をメモします。



3. **VSPHERE MANAGEMENT NETWORK**に移動し、**NSX Manager FQDN**をクリックしてNSX Managerを起動します。



4. この手順のステップ2に記載されている資格情報を使用して、NSX Managerにログインします。

NSX Manager 資格情報(/en-us/vpx/media/gcve-nsx-manager-credentials.png)

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、**ネットワーキング > DHCP** に移動します。ネットワーキングダッシュボードには、サービスが 1 つのティア 0 と 1 つのティア 1 ゲートウェイを作成することが示されます。
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 IP アドレス範囲を指定します。
12. **エッジクラスター** を選択します。
13. **保存** をクリックして DHCP サービスを作成します。

DHCP サービス(/en-us/vpx/media/gcve-dhcp-service.png)

14. サーバー DHCP 範囲について、手順 6 から 13 を繰り返します。

サーバー DHCP 範囲(/en-us/vpx/media/gcve-server-dhcp-range.png)

15. 2 つの別々のセグメントを作成します。1 つはクライアントおよび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、**ネットワーキング > セグメント** に移動します。
17. 「**セグメントの追加**」をクリックします。

セグメントの追加(/en-us/vpx/media/gcve-add-segment.png)

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. 「**接続済みゲートウェイ**」リストで、「**Tier1**」を選択してTier-1ゲートウェイに接続します。
20. 「**トランスポートゾーン**」リストで、「**TZ-OVERLAY | Overlay**」を選択します。
21. 「**サブネット**」列に、サブネット範囲を入力します。最後のオクテットを **.1** としてサブネット範囲を指定します。例: 10.12.2.1/24。

セグメントの詳細(/en-us/vpx/media/gcve-segment-details.png)

22. 「**DHCP設定**」をクリックし、「**DHCP範囲**」フィールドに値を入力します。

DHCP設定(/en-us/vpx/media/gcve-set-dhcp-config.png)

23. 「**適用**」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。保存(/en-us/vpx/media/gcve-save.png)

セグメントの作成に成功しました(/en-us/vpx/media/gcve-successful-segment.png)

25. サーバーセグメントについても、ステップ17から24を繰り返します。
26. VMを作成する際に、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す。

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 クラウドに NetScaler VPX インスタンスをインストールする

GCVE にプライベートクラウドをインストールして構成した後、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利用可能なリソースの量によって異なります。

プライベートクラウドに NetScaler VPX インスタンスを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次の手順を実行します。

1. NetScaler ダウンロードサイトから、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. **ユーザー名とパスワード**のフィールドに管理者資格情報を入力し、**ログイン**をクリックします。
4. **ファイルメニュー**で、**OVF テンプレートの展開**をクリックします。
5. **OVF テンプレートの展開**ダイアログボックスの**ファイルから展開**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に移動し、.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、OVF を変更して E1000 の代わりに VMXNET3 インターフェイスを使用します。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ており、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ているネットワークを、NSX-T Manager で構成したネットワークにマッピングします。 **OK** をクリックします。



7. VMware クラウドに仮想アプライアンスのインストールを開始するには、**完了** をクリックします。



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 NetScaler VPX インスタンスを選択し、右クリックメニューから **Power On** を選択します。 **Launch Web Console** タブをクリックして、コンソールポートをエミュレートします。



9. これで vSphere クライアントから NetScaler VM に接続されました。



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. `show ns ip` コマンドを使用して ADC 構成を確認できます。



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後、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。

新しいリージョンでパブリック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、**Network > REGIONAL SETTINGS > Add Region**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を割り当てます。このとき、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てください。

エッジサービスの割り当て(/en-us/vpx/media/gcve-assign-edge-services.png)

4. パブリックIPサービスは、選択した地域で数分以内に有効になります。

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。

GCVEパブリックIP(/en-us/vpx/media/gcve-public-ips.png)

2. パブリックIPの名前を入力します。地域を選択し、IPが使用されるプライベートクラウドを選択します。
3. パブリックIPをマッピングしたい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イスのプライベートIP**になります。
4. **送信**をクリックします。パブリックIPの割り当て(/en-us/vpx/media/gcve-allocate-public-ip.png)
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、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GCPオートスケーリングサービスを追加

クラウドでのアプリケーションの効率的なホスティングには、アプリケーションの需要に応じて、リソースを簡単かつ費用対効果の高い方法で管理することが求められます。増大する需要に対応するためには、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。需要が減少した場合は、リソースの利用率が低いこと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。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の使用状況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面倒です。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と、必要に応じたリソースのスケールアップおよびスケールダウンのプロセスを自動化する必要があります。

GCPオートスケーリングサービスと統合されたNetScaler VPXインスタンスは、以下の利点を提供します。

- **負荷分散と管理:** 需要に応じてサーバーを自動的にスケールアップおよびスケールダウンするように構成します。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るようにします。仮想IPアドレスとサブネットIPアドレスはVPXインスタンス上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、サーバーの負荷分散を行います。
- **ネットワーク可用性の向上:** VPXインスタンスは以下をサポートします。
 - 同じ配置グループ上のバックエンドサーバー
 - 異なるゾーン上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 VPXインスタンスでGCPオートスケーリングサービスがどのように動作するかを示しています。



開始する前に

NetScaler VPXインスタンスでオートスケーリングの使用を開始する前に、以下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 VPXインスタンスを作成します。
 - NetScaler VPXインスタンスの作成方法の詳細については、「[Google Cloud PlatformにNetScaler VPXインスタンスを展開する](#)」を参照してください。

- HAモードでVPXインスタンスを展開する方法の詳細については、「[Google Cloud Platform にVPX高可用性ペアを展開する](#)」を参照してください。

- GCPプロジェクトで**Cloud Resource Manager 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.zones.list",

    "compute.instanceGroupManagers.list",

    "compute.instanceGroupManagers.get"

]
```

- オートスケーリングを設定するには、以下が構成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
 - オートスケーリングポリシー

NetScaler VPXインスタンスにGCPオートスケーリングサービスを追加する

GUIを使用して、ワンクリックでVPXインスタンスにオートスケーリングサービスを追加できます。VPXインスタンスにオートスケーリングサービスを追加するには、次の手順を実行します。

1. **nsroot** の資格情報を使用して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、**Create**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**フィールドには、インスタンスに関連付けられているすべてのIPアドレスが自動的に入力されます。
- **Autoscale™ Group**は、GCPアカウントで構成されているマネージドインスタンスグループから事前に入力されます。
- **Autoscale Group Protocol**と**Autoscale Group Port**を選択する際は、サーバーが構成されたプロトコルとポートでリッスンしていることを確認してください。サービスグループに正しい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。
- **Graceful**チェックボックスはサポートされていないため、オフにします。

注:

SSLプロトコルタイプのオートスケーリングの場合、クラウドプロファイルを作成すると、証明書がないためロードバランス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ーまたはサービスグループに手動でバインドできます。

3. 初回ログオン後、クラウドプロファイルを作成したい場合は、GUIで**System > Google Cloud Platform > Cloud Profile**に移動し、**Add**をクリックします。



Create Cloud Profile設定ページが表示されます。



クラウドプロファイルは、マネージドインスタンスグループのサーバーをメンバーとして、NetScalerロードバランシング仮想サーバーと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で同じマネージドインスタンスグループを使用して、異なるサービス（異なるポートを使用）に対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で同じオート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インスタンスにおけるVIPスケーリングのサポート

NetScalerアプライアンスはクライアントとサーバーの間に配置され、クライアントのリクエストとサーバーの応答がそれを通して通過します。一般的なインストールでは、アプライアンスに構成された仮想サーバーが、クライアントが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するために使用する接続ポイントを提供します。デプロイメント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、インスタンス上の各インターフェースが異なるVPCに接続されるように制限されています。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をまたがることができます。さらに、GCPは以下の制限を課しています。

- パブリックIPアドレスの数とNICの数は1対1でマッピングされます。1つのNICには1つのパブリックIPアドレスのみを割り当てることができます。
- より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。

例えば、n1-standard-2インスタンスは2つのNICしか持てず、追加できるパブリックVIPは2つに制限されます。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インスタンスでパブリック仮想IPアドレスのより高いスケーリングを実現するには、VIPアドレスをインスタンスのメタデータの一部として構成できます。NetScaler VPXインスタンスは、GCPが提供する転送ルールを内部的に使用して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。VIPアドレスをメタデータの一部として構成した後、転送ルールの作成に使用されるのと同じIPを使用してLB仮想サーバーを構成できます。したがって、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に関する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。

注意点

- Googleは、仮想IP転送ルールごとに一部追加料金を請求します。実際の費用は、作成されたエントリの数によって異なります。関連費用はGoogleの料金ドキュメントで確認できます。
- 転送ルールはパブリックVIPにのみ適用されます。デプロイメントでプライベートIPアドレスをVIPとして必要とする場合は、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アドレスで異なるプロトコルを持つ新しいロードバランシング仮想サーバーを追加することもできます。

開始する前に

- NetScaler VPXインスタンスはGCPにデプロイされている必要があります。
- 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以下の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.get"  
  
    "compute.targetInstances.use",  
  
    ]
```

- GCPプロジェクトで**Cloud Resource Manager 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、GCPサービスアカウントに以下の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、GCPサービスアカウントに以下の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ロールがない場合、サービスアカウントにサービスアカウントユーザーロールを追加する必要があります。

NetScaler VPXインスタンスでのVIPスケーリングのための外部IPアドレスの構成

1. Google Cloud Console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. 以下を入力して**カスタムメタデータ**を更新します。

- キー = vips
- 値 = 以下のJSON形式で値を指定します。

```
{ "外部予約済みIPの名前": , }
```

GCP は以下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:

```
{ "external-ip1-name": , "external-ip2-name": }
```

この例では、NetScaler VPX インスタンスは、各 IP、プロトコルペアに対して内部的に1つの転送ルールを作成します。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。この例は、メタデータエントリに対していくつの転送ルールが作成されるかを理解するのに役立ちます。

以下の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 および ユーディーピー
- c. external-ip2-name と アイシーエムピー
- d. external-ip2-name と エーエイチ

注:

HAモードでは、カスタムメタデータはプライマリインスタンスにのみ追加する必要があります。フェイルオーバー時には、カスタムメタデータは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アドレスを持つロードバランシング仮想サーバーを設定する

ステップ 1. ロードバランシング仮想サーバーを追加します。

1. **構成 > トラフィック管理 > ロードバランシング > 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にVIPとして追加される転送ルールの外部IPアドレス)、およびポートにそれぞれ必要な値を入力し、**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. **構成 > トラフィック管理 > ロードバランシング > 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、およびポートにそれぞれ必要な値を入力し、**OK**をクリックします。



ステップ 3. サービスまたはサービスグループをロードバランシング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. **ステップ 1** で設定したロードバランシング仮想サーバーを選択し、**Edit** 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、**ロードバランシング仮想サーバーサービスバインディングなし** をクリックします。

ロードバランシング仮想サーバーサービスバインディングなし(/en-us/vpx/media/lb-service-binding.png)

4. **ステップ 3** で設定したサービスを選択し、**Bind** をクリックします。

サービスバインディング(/en-us/vpx/media/bind-service-citrix.png)

5. 設定を保存します。

GCP上のVPXインスタンスのトラブルシューティング

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。デバッグは、ネットワークが接続されている場合にのみ可能です。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てください。

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、NetScaler®サポートに電話してください。氏名とメールアドレスの提供を求められます。サポートPINを見つけるには、VPX GUIにログオンし、**System**ページに移動してください。

これは、サポートPINが表示されているシステムページの例です。

GCPテクニカルPIN(/en-us/vpx/media/gcp-technical-pin.png)

NetScaler VPXインスタンスでのジャンボフレーム

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。ジャンボフレームは、標準のIP MTUサイズである1500バイトよりも効率的に大きなファイルを転送できます。

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できます。

- ジャンボからジャンボ。アプライアンスはジャンボフレームとしてデータを受信し、ジャンボフレームとして送信します。
- 非ジャンボからジャンボ。アプライアンスは通常のフレームとしてデータを受信し、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。アプライアンスはジャンボフレームとしてデータを受信し、通常のフレームとして送信します。

詳細については、「[NetScaler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、以下の仮想化プラットフォームで実行されているNetScaler VPXアプライアンスで利用できます。

- ヴィエムウェア ESX
- Linux-KVMプラットフォーム
- シトリックス® XenServer®
- アマゾン ウェブ サービス (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。ジャンボフレームとそのユースケースの詳細については、「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ユースケースは、VPXアプライアンスにも適用されます。

VMware ESX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

VMware ESX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ースまたはチャネルのMTUを1501～9000の範囲の値に設定します。MTUサイズを設定するには、CLIまたはGUIを使用します。VMware ESXで実行されているNetScaler VPXアプライアンスは、最大9000バイトのIPデータを含むジャンボフレームの送受信のみをサポートしています。
2. VMware ESXサーバーの対応する物理インターフェースで、管理アプリケーションを使用して同じMTUサイズを設定します。VMware ESXの物理インターフェースでの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ースまたはチャネルのMTUを1501～9216の範囲の値に設定します。MTUサイズを設定するには、NetScaler VPX CLIまたはGUIを使用します。
2. Linux-KVMサーバーの対応する物理インターフェースで、管理アプリケーションを使用して同じMTUサイズを設定します。Linux-KVMの物理インターフェースでのMTUサイズの設定の詳細については、[<http://www.linux-kvm.org/>](http://www.linux-kvm.org/)を参照してください。

Citrix XenServer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. **Networking**タブで、ネットワーク - network 0/1/2を選択します。
4. **Properties**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、「[NetScaler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

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、「[NetScaler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ってください。

よくある質問

以下のセクションでは、Citrix® Application Delivery Controller™ (ADC) VPXに基づいて、よくある質問を分類するのに役立ちます。

- 機能(#feature-and-functionality)
- 暗号化(#encryption)
- 価格とパッケージ(#pricing-and-packaging)
- (#Citrix ADC VPX Expressと90日間の無料トライアル)
- ハイパーバイザー(#hypervisor)
- キャパシティプランニングまたはサイジング(#capacity-planning-or-sizing)
- システム要件(#system-requirements)
- その他の技術的なよくある質問(#other-technical-faqs)

機能

NetScaler VPXとは何か？

NetScaler VPXは、業界標準サーバーにインストールされたハイパーバイザー上でホストできる仮想ADCアプライアンスです。

NetScaler VPXには、ADCアプライアンスと同様に、すべてのWebアプリケーション最適化機能が含まれるか？

はい。NetScaler VPXには、すべての負荷分散、トラフィック管理、アプリケーション高速化、アプリケーションセキュリティ（NetScaler GatewayおよびCitrix Application Firewallを含む）、およびオフロード機能が含まれています。NetScalerの機能の完全な概要については、[Application delivery your way](#)を参照してください。

NetScaler VPXでCitrix Application Firewallを使用する場合、何か制限はありますか？

NetScaler VPX上のCitrix Application Firewallは、NetScalerアプライアンスと同様のセキュリティ保護を提供します。Citrix Application Firewall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アプライアンス上のNetScaler Gatewayに違いはありますか？

機能的には、両者は同じです。NetScaler VPX上のNetScaler Gatewayは、NetScalerソフトウェアリリース9.1で利用可能なすべてのNetScaler Gateway機能をサポートしています。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。

ハイパーバイザー上で実行できるという明白な違いを除いて、NetScaler VPXはNetScaler物理アプライアンスとどのように異なりますか？

お客様が動作の違いを認識する主な領域は2つあります。1つ目は、NetScaler VPXが多くの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アプライアンスが独自のL2ネットワーク機能を取り入れているのに対し、NetScaler VPXはL2ネットワークサービスをハイパーバイザーに依存していることです。一般的に、これはNetScaler VPXの展開方法を制限するものではありません。物理NetScalerアプライアンスで構成されている特定のL2機能は、基盤となるハイパーバイザーで構成する必要がある場合があります。

NetScaler VPXはアプリケーションデリバリー市場でどのような役割を果たしますか？

NetScaler VPXは、以下の方法でアプリケーションデリバリー市場の状況を変えます。

- 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アプリケーションに適用されます。
- NetScaler VPXは、顧客が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することを可能にします。NetScaler VPXは、仮想化された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に使用できるだけでなく、Webアプリケーションデリバリー自体を、どこに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も可能にします。IT組織は、Webアプリケーションデリバリー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のみを使用する場合には実用的ではない新しい展開アーキテクチャを可能にします。NetScaler VPXと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、それぞれのアプリケーションの個々のニーズに合わせて使用できます。データセンターのエッジでは、NetScaler MPXアプライアンスが、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、グローバルロードバランシングなどの大量のネット

ワーク全体のタスクを処理します。高性能なNetScaler MPXアプライアンスと展開が容易なNetScaler VPX仮想アプライアンスを組み合わせることで、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もたらし、同時にデータセンター全体のコストを削減します。

NetScaler VPXは、当社のCitrixデリバリーセンター戦略にどのように適合しますか？

NetScaler VPXの利用により、Citrixデリバリーセンターの提供全体が仮想化されたサービスとして利用可能になります。Citrixデリバリーセンター全体は、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の恩恵を受けます。これは、ほぼすべての環境に迅速に展開でき、どこからでも一元的に管理できます。統合された仮想化アプリケーションデリバリー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、およびWebアプリケーションを提供できます。

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アプライアンスと同じSSLパフォーマンスは提供しません。NetScaler VPXは、毎秒最大750の新規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たサードパーティ製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。サードパーティ製SSLカードをサポートしても、NetScaler VPXを特定のハードウェア実装に関連付けることはできません。これにより、組織がデータセンター内のどこにでもNetScaler VPXを柔軟にホストする能力が大幅に低下します。NetScaler VPXが提供するよりも多くのSSLスループットが必要な場合は、NetScaler MPXアプライアンスを使用する必要があります。

NetScaler VPXは、物理NetScalerアプライアンスと同じ暗号化方式をサポートしていますか？

VPXは、ECDSAを除くすべての暗号化方式を物理NetScalerアプライアンスと同様にサポートしています。

NetScaler VPXのSSLトランザクションスループットはどのくらいですか？

SSLトランザクションスループットについては、[NetScaler VPXデータシート](#)を参照してください。

価格とパッケージ

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と同様です。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帯域幅ティアを選択します。NetScaler VPXは、Standard、Advanced、Premiumエディションで利用できます。NetScaler VPXは、10 Mbps（VPX 10）から100 Gbps（VPX 100G）までを提供します。詳細については、NetScaler VPXデータシートを参照してください。

NetScaler VPXはすべてのハイパーバイザーで同じ価格ですか？

はい。

すべてのハイパーバイザーでVPXに同じNetScaler SKUが使用されますか？

はい。

NetScaler VPXライセンスをあるハイパーバイザーから別のハイパーバイザー（たとえば、VMwareからHyper-Vへ）に移動できますか？

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ファミリーエディションの両方をアップグレードできます。両方の種類のアップグレードに対応するアップグレードSKUが利用可能です。

NetScaler VPXを高可用性ペアで展開する場合、いくつかのライセンスが必要ですか？

NetScaler物理アプライアンスと同様に、NetScalerの高可用性構成には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日間無料トライアル

NetScaler VPX Expressには、すべてのNetScaler標準機能が含まれていますか NetScaler Gatewayと、Citrix Virtual Apps（旧XenApp）Web InterfaceおよびXMLブローカーのロードバランシングが含まれていますか？

はい。NetScaler VPX Expressには、NetScaler Standardの全機能が含まれています。NetScalerリリース12.0–56.20以降、CitrixはVPX Expressの動作を変更しました。

NetScaler VPX Expressの機能：NetScaler GatewayとCitrix Virtual Apps Web InterfaceおよびXMLブローカーのロードバランシング？

NetScalerリリース12.0–56.20以降、VPX ExpressはGateway機能を除くNetScaler Standard Editionの機能セットを提供します。12.0–56.20リリース以前は、VPX ExpressにはStandard Editionのすべての機能が含まれていました。

NetScaler VPX Expressのライセンス要件？

新しいNetScaler VPX Expressリリース（12.0–56.20以降）では、VPX Expressは無料で、インストールにライセンスファイルは不要で、コミットメントもありません。すでにVPX Expressライセンスをお持ちの場合は、以前のVPX Expressの動作が維持されます。VPX Expressのライセンスファイルが削除され、12.0–56.20以降のリリースが使用された場合、新しい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では、いいえ。ライセンスも有効期限もありません。すでにVPX Expressライセンスをお持ちの場合、ライセンスはダウンロードから1年後に期限切れになります。

NetScaler VPX Expressには、5つの無料のNetScaler Gateway同時ライセンスが含まれていますか？

はい、VPX Expressライセンスをお持ちの場合です。

お客様がダウンロードできるNetScaler VPX Expressの数に制限はありますか？

5つです。

NetScaler VPX Expressは、NetScaler MPXアプライアンスと同じ暗号化方式をサポートしていますか？

一般提供されているNetScalerアプライアンスでサポートされているすべての強力な暗号化方式は、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申請できますか？

いいえ。テクニカルサポートケースを申請するには、VPX-10、VPX-200、VPX-1000、VPX-3000などのリテール版NetScaler VPXライセンスが必要です。ただし、NetScaler VPX Expressユーザーは、NetScaler VPXナレッジセンターを利用したり、Z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助けを求めたりすることができます。

NetScaler VPX Expressをリテール版にアップグレードできますか？

はい。必要なリテール版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VMwareバージョンをサポートしていますか？

NetScaler VPXは、バージョン3.5以降の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にいくつかの仮想ネットワークインターフェースを割り当てることができますか？

NetScaler VPXには、最大10個の仮想ネットワークインターフェースを割り当てることができます。

vSphereからNetScaler VPXコマンドラインにアクセスするにはどうすればよいですか？

VMware vSphereクライアントは、コンソールタブを介して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できます。SSHまたはTelnetクライアントでNetScaler VPXのNSIPアドレスを使用できます。

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、任意のブラウザのアドレスフィールドにNetScaler VPXのNSIP（例: `http://NSIP address`）を入力します。

同じVMware ESXにインストールされた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、可能ですが推奨されません。ハードウェア障害が発生した場合、両方のNetScaler VPXインスタンスに影響が及びます。

2つの異なるVMware ESXシステムで実行されている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。高可用性セットアップでは推奨されます。

VMwareの場合、NetScaler VPXでインターフェース関連イベントはサポートされていますか？

いいえ。インターフェース関連イベントはサポートされていません。

VMwareの場合、NetScaler VPXでタグ付きVLANはサポートされていますか？

はい。NetScalerのタグ付きVLANは、リリース11.0以降のNetScaler VPXでサポートされています。詳細については、[NetScalerドキュメント](#)を参照してください。

VMwareの場合、NetScaler VPXでリンクアグリゲーションとLACPはサポートされていますか？

いいえ。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レベルで構成する必要があります。

NetScaler VPXのドキュメントにはどのようにアクセスしますか？

ドキュメントはNetScaler VPX GUIから入手できます。ログイン後、「Documentation」タブを選択してください。

キャパシティプランニングまたはサイジング

NetScaler VPXでどのようなパフォーマンスが期待できますか？

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パワーが異なることを考慮すると、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、より高いパフォーマンスが得られます（ライセンスで許可されている最大値まで）。一方、低速な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、インバウンドトラフィックのみか、それともインバウンドとアウトバウンドの両方のトラフィックか？

NetScaler VPXの帯域幅制限は、リクエストトラフィックかレスポンストラフィックかに関わらず、NetScalerへのインバウンドトラフィックのみに適用されます。これは、NetScaler VPX-1000（例えば）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ています。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は、リクエストトラフィックとレスポンストラフィックと同じではありません。NetScalerにとって、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に入ってくる）です。

同じサーバーで複数のNetScaler VPXインスタンスを実行できるか？

はい。ただし、物理サーバーがホスト上で実行されている総ワークロードをサポートするのに十分なCPUとI/O容量を持っていることを確認してください。そうしないと、NetScaler VPXのパフォーマンスに影響が出る可能性があります。

物理サーバーで複数のNetScaler VPXインスタンスが実行されている場合、NetScaler VPXインスタンスあたり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のハードディスクスペース、および2つのvCPUを割り当てる必要があります。重要な展開の場合、システムがメモリ制約のある環境で動作するため、VPXに2 GBのRAMを推奨しません。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。推奨は4 GB RAMまたは8 GB RAMです。

注:

NetScaler VPXは、レイテンシに敏感な高性能仮想アプライアンスです。期待されるパフォーマンスを提供するには、アプライアンスはvCPU予約、メモリ予約、ホスト上でのvCPUピンニングを必要とします。また、ホストでは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へのアクセス時の遅延、pit boss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。

同じサーバーでNetScaler VPXと他のアプリケーションをホストできますか？

はい。例えば、NetScaler VPX、Citrix Virtual Apps Web Interface、Citrix Virtual Apps XML Broker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、物理ホストが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とI/O容量を備えていることを確認してください。

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は向上しますか？

はい、NetScaler VPXインスタンスが追加のvCPUのライセンスを取得している場合、CPUコアを追加することでNetScaler VPXのパフォーマンスを向上させることができます。NetScaler VPXは、構成とパフォーマンスティアに応じて、最大20 vCPU（41 Gbps～100 Gbpsのパフォーマンスの場合）をサポートできます。vCPUを増やすと、特に高性能なシナリオでスループットの向上に役立ちます。ただし、パフォーマンスへの影響は、ネットワークドライバ（PCIパススルーやSR-IOVなど）や特定のワークロードなどの要因にも依存します。異なるVPXパフォーマンスティアでサポートされるvCPUの数については、[NetScaler VPXデータシート](#)を参照してください。

NetScaler VPXはアイドル状態なのに、なぜCPUを90%以上消費しているように見えるのですか？

これは通常の動作であり、NetScalerアプライアンスも同様の動作を示します。NetScaler VPXのCPU使用率の真の範囲を確認するには、NetScaler CLIでstat CPUコマンドを使用するか、NetScaler GUIからNetScaler VPXのCPU使用率を表示します。NetScalerパケット処理エンジンは、処理すべき作業がない場合でも常に「作業を探しています」。そのため、CPUを制御し、それを解放しないようにあらゆることを行います。NetScaler VPXのみがインストールされたサーバーでは、（ハイパーバイザーの観点から）NetScaler VPXがCPU全体を消費しているように見えます。「NetScaler内部」から（CLIまたはGUIを使用して）CPU使用率を見ると、NetScaler VPXのCPU容量が使用されている状況がわかります。

システム要件

NetScaler VPXの最小ハードウェア要件は何ですか？

次の表は、NetScaler VPXの最小ハードウェア要件を説明しています。

種類	要件
プロセッサ	VPXプラットフォームのプロセッサ要件については、 NetScaler VPXでサポートされるプロセッサ の表を参照してください。
メモリ	最小2 GB。ただし、4 GBを推奨します。
ディスク	最小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	最小100 Mbpsですが、1 Gbpsを推奨します。
NIC	お使いのハイパーバイザーと互換性のあるNICを使用してください。詳細については、「 NetScaler VPXでサポートされるNIC 」を参照してください。

注:

- 重要な展開では、NetScaler VPXには4 GBのメモリが推奨されます。2 GBのメモリでは、NetScaler VPXは非常にメモリ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。
- 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ます。

システム要件の詳細については、「[NetScaler VPXデータシート](#)」を参照してください。

Intel 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によって実行される機密性の高いCPU命令や特権CPU命令をハイパーバイザーにトラップします。これにより、ハイパーバイザー上でのゲストOS（NetScaler VPXの場合はBSD）のホスティングが簡素化されます。

VT-xはどのくらい一般的ですか？

多くのサーバーでは、BIOS設定で仮想化支援機能（VT-xやAMD-Vなど）がデフォルトで無効になっています。NetScaler VPXを実行できないと結論付ける前に、BIOS構成を確認してください。仮想化サポートが無効になっている場合は、サーバーがNetScaler VPXのような仮想化アプリケーションを適切に実行できるように、BIOSで有効にする必要がある場合があります。

NetScaler VPX の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は基盤となるハイパーバイザーと互換性のある任意のサーバーで実行できます。サポートされているプラットフォームの包括的なリストについては、ハイパーバイザーのHCLを参照してください。

NetScaler VPX はどのバージョンの NetScaler OS に基づいていますか？

NetScaler VPX は NetScaler 9.1 以降のリリースに基づいています。

NetScaler VPX は BSD 上で動作しますが、BSD Unix がインストールされたサーバー上でネイティブに実行できますか？

いいえ。NetScaler VPX の実行にはハイパーバイザーが必要です。詳細なハイパーバイザーのサポートについては、[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 FAQ

複数の NIC を持つ物理サーバー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 の場合、静的リンクアグリゲーションはサポートされており、4つのチャンネルと7つの仮想インターフェースという制限があります。VMware の場合、NetScaler VPX 内では静的リンクアグリゲーションはサポートされていませんが、VMware レベルで構成できます。

VPX で MAC ベースの転送 (MBF) は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点がありますか？

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。ハイパーバイザーは基本的に、NetScaler VPX から受信したすべてのパケットを外部に、またその逆も同様に切り替えます。

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、install ns または GUI の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スペースはどのように割り当てられますか？変更できますか？

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保存するためのスペースを含む、保守性ニーズに対応するために 20 GB のサイズに設定されています。より小さなディスクイメージを生成することは可能ですが、現在その予定はありません。`/flash` と `/var` は両方とも同じディスクイメージ内にあります。これらは互換性の目的で別々のファイルシステムとして保持されています。

以下の値は、NetScaler VPX インスタンス上の特定のディレクトリに割り当てられたディスク容量を表します。

- /flash は 965M
- /var は 14G

メモリ割り当ての詳細な推奨事項については、[NetScaler VPX データシート](#)を参照してください。

NetScaler VPXインスタンスの容量を増やすために新しいハードドライブを追加できますか？

はい。NetScalerリリース13.1ビルド21.x以降では、2番目のディスクを追加することで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番目のディスクはコアファイルとログの保存に使用されます。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続けます。

注記：

データの損失を避けるため、NetScalerアプライアンスをダウングレードする際は外部バックアップを取得してください。

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ドキュメント](#)

注記：

Azureに展開された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てください。詳細については、[ローカル一時ディスクがない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：

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動作する一部のスクリプトが、以下の条件下で失敗する可能性があります。

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するために、「link」シェルコマンドを使用した場合。

そのようなコマンドはすべて、シンボリックリンクを使用するために「ln -s」に置き換える必要があります。また、失敗するスクリプトもそれに応じて修正してください。

NetScaler VPXのビルド番号付けと他のビルドとの相互運用性に関して、何を期待できますか？

NetScaler VPXは、9.1. Cl (classic) および 9.1. Nc (nCore) リリースと同様のビルド番号付けを持っています。例えば、9.1_97.3.vpx、9.1_97.3.nc、および 9.1_97.3.cl のようになります。

NetScaler VPXは、NetScalerアプライアンスとの高可用性セットアップの一部になり得ますか？

サポートされていない構成です。

NetScaler VPXで表示されるすべてのインターフェースは、ハイパーバイザー上のインターフェース数に直接関係していますか？

いいえ。ハイパーバイザー上に物理NICが1つしかない場合でも、NetScaler VPX構成ユーティリティを介して最大7つのインターフェース（VMwareの場合は10）を追加できます。

NetScaler VPX上の準仮想化（PV）インターフェースが、基盤となる物理インターフェースが10Gであるにもかかわらず、なぜ1Gインターフェース（1/x）として表示されるのか 最大スループットは1 Gbpsに制限されるのか？

NetScaler VPXでは、すべての準仮想化NIC（VMXNET3など）は、背後にある物理NICの速度に関係なく、1/x として表示されます。これは予期される動作であり、スループットが1 Gbpsに制限されることを意味するものではありません。実際のスループットは基盤となる物理NICの容量によって決定されるため、10G NICに支えられたPVインターフェースは1 Gbpsをはるかに超える速度を提供できます。

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のアクティブインスタンスを移動できるか？

NetScaler VPXはXenMotionまたは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 12.1リリース以降でサポートされています。

fix

This is the content for the sample heading topic. It is hidden from the main table of contents but remains accessible via links.