

NetScaler VPX 14.1

Contents

NetScaler VPX サポートマトリックス	31
VMware ESX ハイパーバイザー上のVPXインスタンス	31
XenServerまたはCitrix Hypervisor™上のVPXインスタンス	33
Microsoft Hyper-V 上の VPX インスタンス	33
Azure Local 上の VPX インスタンス	34
Red Hat® OpenShift® Virtualization (RHOCPU) 上の VPX インスタンス	34
Nutanix AHV上のVPXインスタンス	35
Nutanix AHV 上の Citrix Ready® VPX (KVM) の非推奨化	35
汎用 KVM 上の VPX インスタンス	36
パブリッククラウド上のVPXインスタンス	37
ハイパーバイザーでサポートされるVPX機能	37
パブリッククラウドでサポートされているVPX機能	40
サポートされているブラウザ	42
NetScaler VPX™でサポートされているプロセッサ	42
NetScaler® VPX でサポートされている NIC	43
その他の参考資料	44
VMware ESX、Linux KVM、およびCitrix Hypervisor上でのNetScaler VPXのパフォーマンスを最適化する	45
VMware ESXハイパーバイザー上のNetScaler VPXインスタンス	45
ESXホストでの推奨構成	45
E1000ネットワークインターフェイスを備えたNetScaler VPX	46
pNICあたり2つのvNIC展開	47
VMXNET3ネットワークインターフェイスを備えたNetScaler VPX	48
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	49
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	50
VMware ESXi ハイパーバイザーの使用ガイドライン	50

パケットエンジンCPU使用率を制御するコマンド	52
Linux-KVMプラットフォーム上のNetScaler VPXインスタンス	53
KVMのパフォーマンス設定	53
PVネットワークインターフェースを備えたNetScaler VPX	57
SR-IOVおよびFortville PCIeパススルーネットワークインターフェースを備えたNetScaler VPX	59
Citrix Hypervisors 上の NetScaler VPX インスタンス	61
Citrix Hypervisors のパフォーマンス設定	61
SR-IOV ネットワークインターフェースを備えた NetScaler VPX	61
準仮想化インターフェースを備えたNetScaler VPX	62
NetScaler VPXディスク容量の増加のサポート	63
NetScaler VPXでプライマリディスクサイズを動的に増やす	63
セカンダリディスクを追加する	64
NetScaler VPXのプライマリディスクサイズを手動で増やす	65
NetScaler VPX で追加の管理 CPU コアを構成する	66
CLI を使用して追加の管理 CPU コアを構成する	66
GUI を使用して追加の管理 CPU コアを構成する	67
統計と監視	68
追加の管理CPUのトラブルシューティング	68
クラウドでのNetScalerアプライアンスの初回起動時にNetScaler VPX構成を適用する	69
ユーザーデータとは	69
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	69
AWS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	69
AWS CLI を使用してプリブートユーザーデータを提供する	70
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	71
Azure CLI を使用してプリブートユーザーデータを提供する	71
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	71
gcloud CLI を使用してプリブートユーザーデータを提供する	72

プリブートユーザーデータ形式	72
NetScaler構成	73
ユーザースクリプト	75
ライセンス	78
ブートストラップ	79
デフォルトのブートストラップ構成	79
ブートストラップ構成のカスタマイズ	80
方法1：インターフェースの詳細のみを指定するカスタムブートストラップ	81
AWS のカスタムブートストラップの例	81
Azure のカスタムブートストラップの例	81
GCPのカスタムブートストラップの例	82
方法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	82
AWSのカスタムブートストラップの例	83
Azureのカスタムブートストラップの例	83
GCPのカスタムブートストラップの例	84
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提供するカスタムブートストラップ	84
AWSのカスタムブートストラップの例	85
Azure のカスタムブートストラップの例	87
GCP のカスタムブートストラップの例	89
AWSおよびAzureにおけるNICの接続と切断の影響	90
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	92
NetScaler CLIを使用してPEモードを設定する	92
クラウドでのNetScalerアプライアンスの初回起動時にPEモード設定を適用する	93
パブリッククラウド上のNetScaler VPX向けに同時マルチスレッディングを構成する	94
SMTサポートマトリックス	95
制限事項	95

SMTの構成	96
トラブルシューティング	97
NetScaler 健全性チェッカーツール	99
AWS 用 NetScaler 健全性チェッカーツール	99
NetScaler CLI を使用して健全性チェッカーツールを実行する	100
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	101
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	101
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	101
Citrix Hypervisor/XenServer に NetScaler VPX インスタンスをインストールする	103
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	103
Hypervisor のハードウェア要件	103
XenCenterのシステム要件	105
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	105
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イスを使用するように構成する	106
制限事項	107
Intel 82599 NICの制限事項	107
Intel X710 10GおよびIntel XL710 40G NICの制限事項	107
Intel 82599 NIC の前提条件	108
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	108
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	109
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	110
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	112
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	112
SR-IOVインターフェイスでVLANを構成する	113

その他の参照	113
VMware ESX に NetScaler VPX インスタンスをインストールする	114
前提条件	114
VMware ESXのハードウェア要件	115
VMware vSphereクライアントのシステム要件	116
OVF Tool 1.0のシステム要件	117
NetScaler VPX セットアップファイルのダウンロード	117
VMware ESX に NetScaler VPX インスタンスをインストールする	118
VMware vMotionを使用してNetScaler VPXインスタンスを移行する	119
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	120
VMXNET3ネットワークインターフェイスの受信リングサイズを設定する	123
VMXNET3ネットワークインターフェイスでリングサイズを設定するには	124
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	124
制限事項	124
前提条件	125
ESXハイパーバイザー上のNetScaler VPXをSR-IOVモードでIntel QATを使用してSSLアクセラレーションを行うように構成する	130
前提条件	131
制限事項	131
Intel QATを使用するためのホスト環境をセットアップする	131
注意事項	131
vSphere Web Clientを使用してQAT VFをVPXに割り当てる	132
展開について	132
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	133
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	133
概要	133

前提条件	133
ホスト上でパススルーデバイスを構成する	134
NetScaler VPXインスタンスでパススルーデバイスを設定する	134
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	135
ESXハイパーバイザーでプリブートユーザーデータを提供する方法	136
CD/DVD ISOを使用してユーザーデータを提供する	136
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	138
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	142
VMware Cloud on AWSにNetScaler VPXインスタンスをインストールする	146
前提条件	146
VMware Cloudのハードウェア要件	146
OVF Tool 1.0のシステム要件	147
NetScaler VPXセットアップファイルのダウンロード	147
VMware Cloud に NetScaler VPX インスタンスをインストールする	148
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	150
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	150
Microsoftサーバーのハードウェア要件	151
NetScaler VPXセットアップファイルをダウンロードする	151
MicrosoftサーバーにNetScaler VPXインスタンスをインストールする	152
Hyper-V上でのNetScaler VPXインスタンスの自動プロビジョニング	153
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	158
制限事項と使用ガイドライン	158
一般的な推奨事項	158
制限事項	158
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	159
ソフトウェア要件	159
ゲストVMのハードウェア要件	159

ネットワーク要件	160
ソースインターフェイスとモード	160
ソースインターフェイスのプロパティ	161
プロミスキュースモード	164
必要なモジュール	164
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	165
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	166
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	169
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	170
CDROM 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	170
Virtual Machine Manager を使用して NetScaler VPX インスタンスにインターフェイスを追加する	171
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	172
制限事項	173
前提条件	174
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	176
SR-IOVインターフェースで静的LA/LACPを構成する	177
SR-IOVインターフェースでのVLANの構成	177
KVMハイパーバイザー上のNetScaler VPXをSR-IOVモードでIntel QATを使用してSSLアクセラレーションを行うように構成する	178
前提条件	178
制限事項	178
Intel QATを使用するためのホスト環境をセットアップする	178
注意点	180
Linux KVMハイパーバイザーに展開されたNetScaler VPXにQAT VFを割り当てる	180
展開について	181

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	181
前提条件	181
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	182
virsh プログラムを使用してNetScaler VPXインスタンスにインターフェースを追加する	186
NetScaler VPXゲストVMを管理する	187
Virtual Machine Managerを使用してVPXゲストVMを管理する	188
virsh プログラムを使用してNetScaler VPXゲストVMを管理する	189
KVM上の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	190
前提条件	191
DPDKをインストールする	191
OVSをビルドしてインストールする	192
OVSブリッジを作成する	192
物理インターフェースをOVSブリッジに接続する	192
vhost-user ポートをOVSデータパスに接続する	193
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	194
注意点	210
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	211
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	213
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	213
Red Hat OpenShift Virtualization に NetScaler VPX をインストールする	215
システム要件	215
OpenShift Virtualization に NetScaler VPX インスタンスをインストールする	215
OpenShift VirtualizationでNetScaler VPXをアップグレードする	221
サポートされていない機能と制限	221
Nutanix AcropolisハイパーバイザーにNetScaler VPXをインストールする	222

システム要件	222
Prismを使用してNutanix AHVにNetScaler VPXインスタンスをインストールする	223
Nutanix AHV 上の NetScaler VPX™ をアップグレードする	224
サポートされていない機能と制限事項	225
Citrix Ready KVM VPX から Nutanix AHV 上で公式にサポートされている NetScaler VPX™ へ移行する	225
前提条件	225
Citrix Ready KVM VPX から Nutanix AHV 上で公式にサポートされている NetScaler VPX へのインプレースアップグレード	226
前提条件	226
アップグレードを進める前の重要な考慮事項	226
アップグレード中のデータのバックアップと保存	227
アップグレード手順	228
アップグレード後の検証	228
ノードバイノードの高可用性 (HA) 置換方法を使用した移行	228
重要な考慮事項	231
NetScaler VPX™ on Nutanix Acropolis Hypervisor - よくある質問	231
Red Hat OpenShift VirtualizationおよびNutanix AHV上でのNetScaler VPX展開に関する制限事項	37
AWS上のネットスケーラー VPX	239
はじめに	239
AWSにNetScaler VPXインスタンスを展開する	239
ハイブリッド展開	240
ライセンス	240
自動化	240
ブログ	240
ビデオ	241
顧客事例	241
ソリューション	241

サポート	241
その他の参考資料	241
AWS用語	242
AWS-VPX サポートマトリックス	244
AWS上でサポートされているVPX製品	244
サポートされているAWSリージョン	244
サポートされているAWSインスタンスタイプ	245
サポートされているAWSサービス	246
制限事項と使用ガイドライン	247
前提条件	248
NetScaler VPXインスタンスでAWS IAMロールを設定する	253
IAM ロールの作成	256
IAMポリシーシミュレーターを使用してIAMポリシーをテストする	259
その他の参照	260
AWS上のNetScaler VPXインスタンスの仕組み	260
AWS に NetScaler VPX スタンドアロンインスタンスを展開する	261
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	262
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	266
AWS CLI を使用して NetScaler VPX インスタンスを構成する	267
シナリオ: スタンドアロンインスタンス	267
NetScaler VPXライセンスのダウンロード	272
AWSサブスクリプションをBYOLに移行する方法	273
制限事項	276
異なるアベイラビリティゾーンでのサーバーの負荷分散	276
AWS での高可用性の仕組み	277
同一ゾーン内での高可用性	277
異なるゾーン間での高可用性	279
展開を開始する前に	279

トラブルシューティング	280
同じAWSアベイラビリティゾーンにVPX HAペアを展開する	280
同じゾーンにVPX HAペアを展開する方法	281
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	281
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る	281
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	282
ステップ4：両方のインスタンスでHAを構成する	283
セカンダリプライベートIP移行に基づくレガシーHAペアから新しいHAペアへの移行	284
Citrix CloudFormationテンプレートを使用して高可用性ペアを展開する	284
Amazon CloudWatch を使用してインスタンスを監視する	287
注意点	287
CloudWatchメトリックの表示方法	287
高可用性セットアップでのSR-IOVの構成	288
関連リソース	288
異なるAWSアベイラビリティゾーン間での高可用性	288
注意事項	288
その他の参照情報	289
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高可用性ペアを展開する	289
異なるAWSゾーン間でのEIPアドレスによるHAの仕組み	289
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高可用性ペアを展開する方法	90
CLIを使用して高可用性を構成する	291
GUIを使用して高可用性を構成する	291
シナリオ	292
異なるAWSゾーン間でプライベートIPアドレスを持つVPX高可用性ペアを展開する	294
AWS非共有VPCを使用したプライベートIPアドレスを持つ高可用性ペア	294

前提条件	294
AWS非共有VPCを使用したプライベートIPアドレスを持つVPX HAペアを展開する	295
CLI を使用して高可用性を構成する	297
GUI を使用して高可用性を構成する	298
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	298
前提条件	299
クロスアカウントIAMロールを作成する	301
NetScaler CLIを使用してクロスアカウントIAMロールを設定する	301
NetScaler GUIを使用してクロスアカウントIAMロールを設定する	301
シナリオ	301
HAプライベートIPソリューションのAWS Transit Gateway設定	302
トラブルシューティング	303
AWS Outposts に NetScaler VPX インスタンスを展開する	304
AWS Outposts の仕組み	304
前提条件	304
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	305
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	306
参照	307
NetScaler Web App Firewall を使用して AWS API Gateway を保護する	307
AWS API Gateway の前に NetScaler アプライアンスを展開する	307
WAFが有効なNetScalerアプライアンスを構成する	308
NetScalerの機能と性能	310
バックエンドのAWSオートスケーリングサービスを追加する	311
始める前に	312
NetScaler VPX インスタンスに AWS Autoscaling サービスを追加する	314
AWS で NetScaler GSLB を展開する	316
DBS の概要	316

ELB と DBS	316
AWS コンポーネントを構成する	317
セキュリティグループ	317
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のサービスの設定	318
展開タイプ	319
CFT展開	319
展開手順	319
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	321
セキュリティグループの構成	322
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	322
他のロードバランシングソリューションとの互換性	323
GSLBメソッド	323
GSLBトポロジ	324
IPv6のサポート	324
監視	324
永続性	324
NetScaler ADM StyleBook を使用した構成	325
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	325
StyleBookを使用してNetScaler負荷分散ノードでGSLBを構成する	327
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	328
CloudFormation テンプレートの展開	330
CFT 3-NIC デプロイ	330
災害復旧 (DR)	332
その他のリソース	332
AWS で NetScaler Web App Firewall を展開する	332
本番環境展開のための AWS 上の NetScaler Web App Firewall のアーキテクチャ	333
コストとライセンス	334

展開オプション	335
AWS クイックスタート	336
ステップ1: ユーザーのAWSアカウントにサインインする	336
ステップ2: NetScaler Web App Firewall AMIをサブスクライブする	336
ステップ3: AWSクイックスタートを起動する	337
新しいVPCにNetScaler Web App Firewallをデプロイするためのパラメータ	339
ステップ4: デプロイのテスト	348
踏み台ホストを使用してNetScaler Web App Firewall HAペアに接続する	349
NetScaler コンソール	350
NetScaler Consoleを使用してAWS上にNetScaler VPXインスタンスを展開する	350
NetScaler WebアプリケーションファイアウォールとOWASP Top 10 – 2017	350
NetScaler Webアプリケーションファイアウォール	352
Web App Firewallの展開戦略	352
ネットスケーラー コンソール	353
NetScaler Consoleアプリケーションの分析および管理機能	353
前提条件	353
制限事項と使用ガイドライン	353
技術要件	353
リソース	353
リージョン	354
キーペア	354
参照	355
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	355
インターフェースタイプをSR-IOVに変更する	356
インターフェースタイプをSR-IOVに変更するには	356
高可用性セットアップでSR-IOVを設定する	357
NetScaler VPXインスタンスをAWS ENAで拡張ネットワーキングを使用するように構成する	358

AWS 上の NetScaler VPX インスタンスをアップグレードする	358
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	359
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	359
カスタマーライセンスの使用 (Bring-Your-Own-License)	359
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)	360
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	360
NetScaler の高可用性構成を使用して新しい NetScaler AMI インスタンスにアップグレードする	360
前提条件と考慮事項	361
AWS上のVPXインスタンスのトラブルシューティング	365
AWS よくある質問	365
Microsoft Azure に NetScaler VPX インスタンスを展開する	369
前提条件	369
Azure での NetScaler VPX インスタンスの動作	370
マルチNICマルチIPアーキテクチャ	370
シングルNICマルチIPアーキテクチャ	371
シングルNICシングルIP	371
NetScaler VPXライセンス	371
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	372
注意点	373
Azure における NetScaler VPX インスタンスの IPv6 サポート	373
制限事項	374
Azure 用語	374
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	377
ネットワークアドレス変換によるトラフィックフロー	378
ポート使用ガイドライン	379
NetScaler VPXスタンドアロンインスタンスを構成する	380

開始する前に	380
構成手順の概要	380
リソースグループを構成する	381
ネットワークセキュリティグループを構成する	381
仮想ネットワークとサブネットを構成する	381
2番目のサブネットを構成する	382
ストレージアカウントを構成する	383
可用性セットを構成する	383
NetScaler VPXインスタンスを構成する	384
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	386
ユースケース	387
開始する前に	387
ステップ2: NetScalerが所有するIPアドレスを構成する	389
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	389
アベイラビリティセットを使用した高可用性	390
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	391
アベイラビリティゾーンを使用した高可用性	392
Azure Monitorでメトリックを使用してインスタンスを監視する	394
注意点	394
Azure Monitorでメトリックを表示する方法	394
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	394
PowerShellコマンドを使用してHA-INCノードを構成する	396
シナリオ: HA-INC PowerShell展開	396
パラメータ設定	396
ALBがフローティングIP無効モードの場合に、NetScaler高可用性ペアをAzureにデプロイする	414
ALBがフローティングIP無効モードの場合のHAデプロイメントアーキテクチャ	415
前提条件	415

ALB フローティング IP が無効な状態で Azure に VPX HA ペアを展開する方法	415
Azure DNS プライベートゾーン向けNetScaler®の展開	423
Azure DNS の概要	423
Azure DNS プライベートゾーンにNetScaler GSLB を使用する理由?	424
ユースケース	424
ユースケースソリューション	424
トポロジ	425
Azure DNSプライベートゾーンのNetScalerを構成する	425
前提条件	425
ソリューションの説明	426
Azureとオンプレミスのセットアップを構成する	426
Azure プライベート DNS ゾーン	426
ハブアンドスポークモデルの Azure 仮想ネットワーク (VNet A、VNet B)	426
VNet A から VNet B へのピアリング	427
VNet B から VNet A へのピアリング	427
VNet A に App サーバー、DNS フォワーダー、Windows 10 Pro クライアント、 NetScaler を展開する	427
クライアントがVNet Bにある場合、AppサーバーとDNSフォワーダーをデプロイする	429
オンプレミスにAppサーバー、DNSフォワーダー、Windows 10 Proクライアントをデプ ロイする	430
Azure仮想ネットワークでNetScalerを構成する	431
NetScaler GSLBの構成	431
ADNSサービスの追加	431
GSLBサイトの追加	431
GSLBサービスの追加	431
GSLB仮想サーバーを追加する	432
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	433

前提条件	433
制限事項	433
高速ネットワークでサポートされるNIC	434
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	434
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	435
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ワーキングを確認するには	437
ADC CLI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ワーキングを確認するには	437
NetScalerにおける注意点	438
PVインターフェースにVLANを構成する	439
Azure ILBを使用したNetScaler®高可用性テンプレートでHA-INCノードを構成する	440
構成例	443
インターネットに接続するアプリケーション向けのNetScaler®高可用性テンプレートを使用してHA-INCノードを構成する	444
サンプル構成	447
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	448
セカンダリIP構成を使用してAzureにNetScaler VPX HAペアを展開する	451
AzureセカンダリIP構成を使用してNetScaler VPX™ HAペアを展開するための前提条件	452
AzureのセカンダリIP構成を使用してNetScaler VPX HAペアをデプロイする方法	452
ソリューションテンプレートを使用してAzureにNetScaler VPX HAペアを展開する	457
NetScaler VPX Azureソリューション展開の主な機能	457
前提条件	457
マーケットプレイス オファー情報 (Bring Your Own License - BYOL)	458
ソリューションテンプレートを使用したAzureへのNetScaler VPX HAペアの展開	458
Azure VMware Solution に NetScaler VPX インスタンスをインストールする	460
前提条件	461
VMware クラウド ハードウェア要件	462

OVF Tool 1.0のシステム要件	462
NetScaler VPX セットアップファイルのダウンロード	463
Azure VMware Solution を展開する	463
Windowsを実行しているAzure仮想マシンに接続する	464
プライベートクラウド vCenter ポータルへのアクセス	465
Azure portalでNSX-Tセグメントを作成する	466
VMware Cloud に NetScaler VPX インスタンスをインストールする	466
Azure VMware Solution 上で NetScaler VPX スタンドアロンインスタンスを構成する	468
前提条件	468
NetScaler ロードバランサーを使用して、AVS 上で NetScaler VPX スタンドアロンインスタンスを構成する	468
Azure ロードバランサーを使用して AVS 上で NetScaler VPX スタンドアロンインスタンスを構成する	470
Azure VMware Solution 上でのNetScaler VPX高可用性セットアップの構成	470
前提条件	471
構成手順	471
Azure ルートサーバーをNetScaler VPX HAペアで構成する	472
前提条件	473
AzureルートサーバーをNetScaler VPX HAペアで構成する方法	473
バックエンドのAzure自動スケーリングサービスを追加する	476
開始する前に	477
仮想マシンでマネージド ID を構成する	478
VMSSをNetScaler VPXインスタンスに追加する	479
クラウドプロファイル作成時に留意すべき点	480
Azure でのアンマネージドディスクからマネージドディスクへの移行	480
参照	481
NetScaler VPX デプロイメントの Azure タグ	481
VPX デプロイメントにおける Azure タグの仕組み	481

始める前に	481
VPX GUI を使用して Azure タグ設定を追加する方法	482
VPX CLIを使用してAzureタグ設定を追加する方法	483
トラブルシューティング	484
NetScaler VPXインスタンスでGSLBを構成する	485
シナリオ	486
パラメータ	486
VMの作成	489
リソースグループを作成する	490
ストレージアカウントを作成する	490
可用性セットを作成する	490
仮想ネットワークを作成する	490
パブリックIPアドレスを作成する	491
NICを作成する	491
VM構成オブジェクトを作成	492
資格情報を取得し、OSプロパティを設定	492
NICを追加	492
OSディスクを指定してVMを作成	493
IP構成の詳細	493
GSLBサイトとその他の設定を構成する	495
ディザスタリカバリ	496
ディザスタリカバリのためのMulti-NIC Multi-IP (3-NIC) 展開	497
ディザスタリカバリのためのSingle-NIC Multi-IP (1-NIC) 展開	497
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	497
シナリオ	498
パラメータ設定	498
GSLBおよびDNSトラフィックを許可するためのフロントエンドIPアドレスとルールでALB を構成する	499

各高可用性ペアでGSLBを有効にする	501
Azure に NetScaler GSLB を展開する	502
DBS の概要	503
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	503
Azure GSLB の前提条件	503
Azure コンポーネントを設定する	504
NetScaler GSLB ドメインベースサービスを構成する	504
トラフィック管理設定	505
GSLB設定	505
その他のリソース	507
Azure Local に NetScaler VPX インスタンスを展開する	507
Azure Local に NetScaler VPX インスタンスを展開するための前提条件	507
Windows Admin Center から VM ファイルパッケージを使用して Azure Local に NetScaler VPX インスタンスを展開する	508
Windows Admin Center から VHD を使用して Azure Local に NetScaler VPX インスタンスを展開する	510
Azure 上で NetScaler® Web App Firewall を展開する	512
NetScaler Web App Firewall の展開戦略	512
Webアプリケーションファイアウォールの構成	514
SQLインジェクション	514
クロスサイトスクリプティング	514
バッファオーバーフローチェック	514
仮想パッチ/シグネチャ	514
アプリケーションセキュリティ分析	515
ADMでの集中学習	515
StyleBooks	517
セキュリティインサイト分析	517
セキュリティ侵害に関する詳細情報の取得	517

ログ式の詳細を表示	518
セキュリティ違反	519
アプリケーションセキュリティ違反の詳細を表示	519
違反カテゴリ	520
セットアップ	520
高度なセキュリティ分析を有効にする	520
Webトランザクション設定を有効にする	520
セキュリティ違反ダッシュボード	521
ボットインサイト	521
ボットの表示	521
CAPTCHAボットの表示	522
ボットトラップの表示	523
ボットの詳細を表示	523
高度な検索	524
異常に高いリクエストレート	525
ボット検出	525
IP レピュテーション	525
NetScaler Webアプリケーションファイアウォール と OWASP トップ10 – 2021	526
A1:2021 不適切なアクセス制御	527
NetScaler Web App Firewallによる保護	527
A2:2021 - 暗号化の失敗	527
NetScaler Web App Firewallによる保護	528
A3:2021 - インジェクション	528
NetScaler Web App Firewallの保護機能	528
A5:2021 - セキュリティ設定ミス	529
NetScaler Web App Firewallの保護機能	529
A6:2021 - 脆弱で古いコンポーネント	530
NetScaler Web App Firewallの保護機能	530

A7:2021 – 認証の不備	530
NetScaler Web App Firewall の保護機能	530
A8:2021 - ソフトウェアとデータの整合性の不備	531
NetScaler Web App Firewall の保護機能	531
A9:2021 - セキュリティログと監視の不備	531
NetScaler Web App Firewall の保護機能	531
参照	531
NetScaler GatewayアプライアンスのアドレスプールイントラネットIPを構成する	532
AzureポータルでプライベートIPアドレスを登録する	532
NetScaler Gatewayアプライアンスでアドレスプールを構成する	533
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の複数のIPアドレスを構成する	533
ユースケース	534
スクリプト	535
パラメータ	535
1. リソースグループの作成	538
2. ストレージアカウントの作成	538
3. 可用性セットの作成	538
4. ネットワークセキュリティグループの作成	538
5. 仮想ネットワークを作成する	539
6. パブリックIPアドレスを作成する	539
7. IP構成の割り当て	539
8. NICの作成	540
9. NetScaler VPXインスタンスの作成	540
10. NIC 構成の確認	541
11. VPX 側の構成の確認	541
Azureデプロイ用の追加PowerShellスクリプト	542
NetScaler VPXスタンドアロンインスタンスのプロビジョニング	542

Azure外部ロードバランサーを使用して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	548
Azure内部ロードバランサーを使用した高可用性セットアップでNetScaler VPXペアをプロ ビジョニングする	559
Azure上のVPXインスタンスのサポートチケットを作成する	567
Azure よくある質問	567
Google Cloud Platform に NetScaler VPX インスタンスを展開する	569
サポートされている GCP マシンタイプファミリー	569
サポートされている機能	569
制限事項	570
ハードウェア要件	570
注意事項	570
シナリオ: 複数の NIC と複数の IP を持つスタンドアロンの NetScaler VPX インスタンスをデ プロイする	570
展開手順の概要	571
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	572
展開手順の概要	573
GCP に VPX インスタンスをデプロイした後の注意点	576
NetScaler VPX ライセンス	577
GCPでサポートされているNetScaler VPX製品	577
NetScaler VPXインスタンスを展開するためのGDMテンプレート	577
リソース	577
関連情報	578
Google Cloud Platform で VPX 高可用性ペアを展開する	578
GCP に VPX 高可用性ペアを展開するための GDM テンプレート	578
GCP上のVPX高可用性ペアに対する転送ルールサポート	578
前提条件	579
制限事項	579

Google Cloud Platform上で外部静的IPアドレスを持つVPX高可用性ペアをデプロイする	579
開始する前に	579
Google Cloud PlatformでVPX HAペアをデプロイする方法	581
ステップ1. VPCネットワークを作成する	581
ステップ2. 2つのVPXインスタンスを作成する	581
ステップ3. 高可用性を構成する	582
Google Cloud PlatformでプライベートIPアドレスを持つ単一NIC VPX高可用性ペアをデプロイする	588
始める前に	589
Google Cloud PlatformにVPX高可用性ペアをデプロイする方法	590
ステップ 1. 1つのVPCネットワークを作成する	590
ステップ 2. 2つのVPXインスタンスを作成する	590
ステップ3. 高可用性を構成する	591
GUIを使用して高可用性を構成する	591
CLIを使用して高可用性を構成する	594
Google Cloud PlatformでプライベートIPアドレスを持つVPX高可用性ペアをデプロイする	596
開始する前に	596
Google Cloud PlatformでVPX高可用性ペアをデプロイする方法	598
ステップ 1. VPC ネットワークを作成する	598
ステップ 2. 2 つの VPX インスタンスを作成する	598
ステップ3. 高可用性を構成する	599
GUIを使用して高可用性を構成する	599
CLI を使用した高可用性の構成	602
Google Cloud VMware Engine に NetScaler VPX インスタンスをインストールする	604
前提条件	604
VMware クラウドのハードウェア要件	605
OVF Tool 1.0 のシステム要件	606
NetScaler VPX セットアップファイルのダウンロード	606

Google Cloud VMware Engine をデプロイする	607
プライベートクラウド vCenter ポータルにアクセスする	607
GCVE NSX-TポータルでNSX-Tセグメントを作成する	608
VMware Cloud に NetScaler VPX インスタンスをインストールする	610
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	611
バックエンドGCPオートスケーリングサービスを追加	612
開始する前に	613
GCPオートスケーリングサービスをNetScaler VPXインスタンスに追加する	615
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	616
注意点	617
始める前に	617
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	620
NetScaler VPXインスタンス上で外部IPアドレスを持つ負荷分散仮想サーバーの設定	621
GCP上のVPXインスタンスのトラブルシューティング	622
NetScaler VPXインスタンスでのジャンボフレーム	623
VMware ESXで動作するVPXインスタンスのジャンボフレームを構成する	624
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	624
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	624
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	625
よくある質問	626
機能と性能	626
NetScaler VPXとは何ですか?	626
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ていますか?	626
NetScaler VPXでCitrix Application Firewallを使用する場合の制限事項?	626
NetScaler VPX上のNetScaler GatewayとNetScalerアプライアンス上のNetScaler Gatewayの違い?	627

NetScaler VPXがハイパーバイザー上で動作するという明白な違いを除いて、NetScaler物理 アプライアンスとはどのように異なりますか？	627
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？	627
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？	628
暗号化	628
NetScaler VPXはSSLオフロードをサポートしていますか？	628
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？	628
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？	628
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？	628
価格とパッケージ	629
NetScaler VPXはどのようにパッケージ化されていますか？	629
NetScaler VPXはすべてのハイパーバイザーで同じ価格ですか？	629
すべてのハイパーバイザー上のVPXに同じNetScaler SKUが使用されますか？	629
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（例えば VMwareからHyper-Vへ）に移動できますか？	629
NetScaler VPXインスタンスはアップグレードできますか？	629
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？	629
NetScaler® VPX Expressと90日間無料トライアル	630
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？	630
NetScaler VPX Expressにはライセンスが必要ですか？	630
NetScaler VPX Expressライセンスは期限切れになりますか？	630
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？	630
NetScaler VPX Expressのテクニカルサポートケースを申請できますか？	630
NetScaler VPX Expressはリテール版にアップグレードできますか？	630
ハイパーバイザー	631

NetScaler VPXはどのVMwareバージョンをサポートしていますか？	631
VMwareにおけるVPXへの仮想ネットワークインターフェースの割り当て数？	631
vSphereからのNetScaler VPXコマンドラインへのアクセス方法？	631
NetScaler VPX GUIへのアクセス方法？	631
同じVMware ESXにインストールされた2つのNetScaler VPXインスタンスの高可用性セットアップでの構成？	631
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？	631
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？	631
VMwareの場合、NetScaler VPXでタグ付きVLANはサポートされていますか？	632
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？	632
NetScaler VPXドキュメントにはどのようにアクセスしますか？	632
キャパシティプランニングまたはサイジング	632
NetScaler VPXでどのようなパフォーマンスが期待できますか？	632
サーバーのCPU性能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？	632
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に適用されますか、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に適用されますか？	632
複数のNetScaler VPXインスタンスを同じサーバーで実行できますか？	633
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？	633
NetScaler VPXと他のアプリケーションを同じサーバーでホストできますか？	633
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？	634
NetScaler VPXはアイドル状態でもCPUの90%以上を消費しているように見えるのはなぜですか？	634
システム要件	634
NetScaler VPX の最小ハードウェア要件は何ですか？	634

Intel VT-xとは何ですか？	635
VT-xはどのくらい一般的ですか？	635
NetScaler VPXのハードウェア互換性リスト（HCL）はありますか？	636
NetScaler VPXはどのバージョンのNetScaler OSに基づいていますか？	636
NetScaler VPXはBSD上で動作するため、BSD Unixがインストールされているサーバー上でネイティブに実行できますか？	636
その他の技術的なFAQ	636
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？	636
MACベースの転送（MBF）はVPXでサポートされていますか。NetScalerアプライアンスの実装からの変更点？	636
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？	636
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？	637
NetScaler VPXインスタンスのスペース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？	637
NetScaler VPXのプライマリディスクサイズを増やすことはできますか？	638
NetScaler VPX のプライマリディスクサイズを手動で増やすにはどうすればよいですか？	638
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？	639
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できますか？	639
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？	640
NetScaler VPX上の準仮想化 (PV) インターフェースは、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ですか。これは最大スループットが1 Gbpsに制限されることを意味しますか？	640
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ますか？	640
fix	641

NetScaler VPX サポートマトリックス

このドキュメントでは、NetScaler VPX と VPX FIPS の両方の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。ただし、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は除きます。このドキュメントでは、それらの使用ガイドラインと既知の制限事項についても説明します。

VMware ESX ハイパーバイザー上のVPXインスタンス

ESXIバージョン	ESXI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
イーセスエック スアイ 9.0.2	2026/01/20	25148076	14.1-66.x以降 のビルド	10 Mbpsから 100 Gbps
ESXi 8.0 アッ プデート 3k	2026/07/29	25595708	14.1-73.x 以降 のビルド	
ESXi 8.0 アッ プデート 3j	2026/05/27	25429389	14.1-72.x 以降 のビルド	
ESXi 8.0 アッ プデート 3g	2025/07/29	24859861	14.1-47.x 以降 のビルド	
ESXi 8.0 アッ プデート 3f	2025/07/15	24784735	14.1-47.x 以降 のビルド	
ESXi 8.0 アッ プデート 3e	2025/04/10	24674464	14.1-43.x 以降 のビルド	
ESXi 8.0 アッ プデート 3d	2025/03/04	24585383	14.1-38.x 以降 のビルド	
ESXi 8.0 アッ プデート 3c	2025/01/23	24414501	14.1-29.x 以降 のビルド	
ESXi 8.0 アッ プデート 3b	2024/09/17	24280767	14.1-21.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 8.0 アップデート 3	2024/06/25	24022510	14.1-21.x 以降のビルド	
ESXi 8.0 アップデート 2c	2024/05/21	23825572	14.1-21.x 以降のビルド	
ESXi 8.0 アップデート 2b	2024/02/29	23305546	14.1-17.x 以降のビルド	
ESXi 8.0 アップデート 2	2023/09/21	22380479	14.1-17.x 以降のビルド	
ESXi 8.0 アップデート 1	2023/04/18	21495797	14.1-4.x 以降のビルド	
イーセスエックスアイ 8.0c	2023/03/30	21493926	14.1-4.x 以降のビルド	
イーセスエックスアイ 8.0	2022/10/11	20513097	14.1-4.x 以降のビルド	
ESXi 7.0 アップデート 3w	2025/07/15	24784741	14.1-47.x 以降のビルド	
ESXi 7.0 アップデート 3s	2025/03/04	24585291	14.1-29.x 以降のビルド	
ESXi 7.0 アップデート 3r	2024/12/12	24411414	14.1-29.x 以降のビルド	
ESXi 7.0 アップデート 3q	2024/05/21	23794027	14.1-21.x 以降のビルド	
ESXi 7.0 アップデート 3p	2024/04/11	23307199	14.1-17.x 以降のビルド	
ESXi 7.0 アップデート 3o	2023/09/28	22348816	14.1-12.x 以降のビルド	
ESXi 7.0 アップデート 3n	2023/07/06	21930508	14.1-8.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 3m	2023/05/03	21686933	14.1-4.x 以降のビルド	

注：

各ESXiパッチのサポートは、前述の表で指定されたNetScaler VPXバージョンで検証されており、NetScaler VPX 14.1バージョンのすべての新しいビルドに適用されます。

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

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

XENSERVERまたはCITRIX HYPERVISORバージョン	SYSID	パフォーマンス範囲
8.4、NetScaler VPX バージョン 14.1 ビルド 17.x 以降でサポート	450000	10 Mbps から 40 Gbps
8.2、NetScaler VPX バージョン 13.0 ビルド 64.x 以降でサポート		
8.0, 7.6, 7.1		

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	14.1-4.x 以降	10 Mbps から 3 Gbps まで
2022		14.1-47.x 以降	
2025		14.1-51.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	14.1-60.x 以降	450020
Azure Local OS ビルド	25398.1965、26100.7171、 および 20349.3692	

Azure Local 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。

Red Hat® OpenShift® Virtualization (RHOCPV) 上の VPX インスタンス

RHOCPV バージョン	SYSID	NETSCALER VPX バージョン	パフォーマンス範囲
4.20.5 ベアメタルシ ングルノード OpenShift (SNO) ク ラスタ	450070	14.1-66.64 以降	10 Mbps から 2 Gbps

注：

- RHOCPV上のNetScaler VPXインスタンスは、NetScalerリリース14.1-66.64以降でサポートされています。詳細については、[Red Hat OpenShift VirtualizationへのNetScaler VPXのインストール](#)を参照してください。
- RHOCPV上のNetScaler VPX FIPSは、NetScalerリリース14.1-72.x以降でサポートされています。

Nutanix AHV上のVPXインスタンス

注：

Nutanix AHV上のNetScaler VPXインスタンスは、NetScalerリリース14.1-51.x以降でサポートされています。詳細については、[Nutanix AcropolisハイパーバイザーへのNetScaler VPXのインストール](#)を参照してください。

NUTANIXバージョン	SYSID	パフォーマンス範囲
エーオーエス 7.3、エーエイチブイ 10.0.1、プリズムセントラル 7.3	450070	10 Mbps から 5 Gbps

Nutanix AHV展開でサポートされているNetScalerバージョンと検証済みの構成を確認するには、[Nutanix互換性および相互運用性マトリックス](#)を参照してください。

注：

Citrix は AOS 6.0.1 CE を使用して Nutanix AHV を認定しました。

Nutanix AHV 上の Citrix Ready® VPX (KVM) の非推奨化

Citrix は以前、[Citrix Ready パートナiership プログラム](#)を通じて、Nutanix AHV 上の古い NetScaler VPX (KVM) 展開をサポートしていました。

NetScaler リリース 14.1-66.x 以降、Citrix は Nutanix AHV 上の古い NetScaler VPX (KVM) 展開のサポートを提供しなくなりました。

重要な考慮事項:

- Nutanix AHV 上の Citrix Ready® VPX (KVM) については、以前の NetScaler バージョンから 14.1-66.x 以降のリリースへのアップグレードはサポートされていません。
- Citrix Ready VPX (KVM) イメージを使用しているお客様は、引き続きサポートを受け、最新機能にアクセスできるように、公式にサポートされている NetScaler VPX イメージとアップグレードバンドルに移行する必要があります。詳細については、[Citrix Ready KVM VPX から Nutanix AHV 上の公式にサポートされている NetScaler VPX™ への移行](#)を参照してください。

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		

留意事項：

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で認定されており、バージョン内のパッチリリースで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実行してください。
 - /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実行してください。
 - /etc/default/grub を編集し、`GRUB_CMDLINE_LINUX` 変数に `"kvm_intel.preemption_timer=0"` を追加します。
 - `"# 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。
 - 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
アジュール	450020	10 Mbpsから10 Gbps
GCP	450070	10 Mbpsから10 Gbps

ハイパーバイザーでサポートされるVPX機能

ハイパーバイザー → 機能 ↓	ゼノサーバー 上の VPX	ヴィエムウェア ア ES X上の の VP X	MICROSOFT FT HYPER- PE R-V 環境 での VP X	汎用 KV M での VP X	NU TA NIX AH V での VP X						
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	PV
Multiplex サポート	はい	はい	はい	はい	はい	はい	はい	はい	はい	はい	はい

ハイパーバイザー → 機能 ↓ ゼノサーバー上のVPX ヴィエムウェアESX上のVPX MICROSOFT HYPER-V環境でのVPX 汎用KVMでのVPX NUC TA NIX AH VでのVPX											
クラスタリングのサポート	はい	はい ¹	はい	はい ¹	はい	はい	はい	はい	はい ¹	はい	はい
VLANTagging	はい	はい	はい	はい	はい	はい	はい (2012 R2のみ)	はい	はい	はい	はい
リンクイベントの検出/HA	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	いいえ ²

ハイパーバイザー → 機能 ↓	ゼノサーバー上のVPX	ヴィエムウェアESX上のVPX	MICROSOFT HYPER-V環境でのVPX	汎用KVMでのVPX	NUMA/NIX/AHVでのVPX						
Monitor											
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	いいえ
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	いいえ ²
LA CP	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	いいえ ²
静的CLAG	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ

ハイパーバイザー → 機能 ↓	ゼノサーバー上のVPX	ヴィエムウェアESX上のVPX	MICROSOFT HYPER-V環境でのVPX	汎用KVMDでのVPX	NUCLEAR NIX AHVでのVPX						
LA CP CL AG	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	いいえ ²
ホットプラグ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ

パブリッククラウドでサポートされている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
クラスタリングのサポート	いいえ	いいえ	いいえ
VLANタギング	いいえ	いいえ	いいえ
リンクイベントの検出/HAMon	いいえ ²	いいえ ²	いいえ ²

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構成	いいえ	いいえ	いいえ
静的LA	いいえ	いいえ	いいえ
LACP	いいえ	いいえ	いいえ
静的CLAG	いいえ	いいえ	いいえ
LACP 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の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. Static 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、収束時間はルーティングプロトコルに依存します。

- モニターをスタティックルートにバインドしない場合、ルートの状態がVLANステータスに依存するため、監視対象のスタティックルート機能は失敗します。VLANステータスはリンクステータスに依存します。
- 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、高可用性の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に変更されません。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上で破棄されます。
 - 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
Citrix ハイパーバイザー	はい	はい
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
ニュータニックス AHV	はい	いいえ
AWS	はい	はい

プラットフォーム	INTEL プロセッサ	AMD プロセッサ
アジュール	はい	はい
GCP	はい	はい

NetScaler® VPX でサポートされている NIC

注:

NetScaler VPX は、サポートされている物理ネットワークアダプター上のハイパーバイザーによって提供される、VirtIO、NetVSC、XenNetFront などの準仮想化 vNIC をサポートしています。さらに、次の表に記載されている NIC ドライバーは、SR-IOV および PCI パススルー (PCI-PT) モードに対応しています。

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
シトリックス ハイパーバイザー	NA	NA	NA	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	NA	NA	該当なし	いいえ	いいえ	いいえ
KVM	いいえ	はい	はい	はい	はい	いいえ

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
ニュータニックス AHV	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ
AWS	該当なし	該当なし	該当なし	はい	該当なし	該当なし
アジュール	はい	はい	はい	該当なし	NA	NA
GCP	NA	NA	NA	NA	NA	NA

その他の参考資料

- Citrix Ready製品については、[Citrix Ready Marketplace](#)をご覧ください。
- Citrix Ready製品のサポートについては、[Citrix Readyパートナーページ](#)を参照してください。
- VMware ESXのハードウェアバージョンについては、[VMware Toolsのアップグレード](#)を参照してください。

VMware ESX、Linux KVM、およびCitrix Hypervisor上でのNetScaler VPXの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、次に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

このセクションには、構成可能なオプションと設定、およびVMware ESXハイパーバイザー上で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の詳細が含まれています。

- [ESXホストでの推奨構成](#)
- [E1000ネットワークインターフェイスを備えたNetScaler VPX](#)
- [VMXNET3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NetScaler VPX](#)

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、ESXホストにNon-uniform Memory Access (NUMA) アフィニティとCPUアフィニティを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 ホストで次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を作成すると、ESXホスト内に複数の受信 (Rx) スレッドが作成されます。これにより、pNICインターフェイスの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。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
```

- 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。次の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構成コマンドです。



NetScaler VPXの構成例:

上記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インスタンスで次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イス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イス 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 -gslb 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。

- ゲートウェイがSNIP 1.1.1.2を指す1.1.1.0/24サブネット

- ゲートウェイがSNIP 2.2.2.2を指す2.2.2.0/24サブネット

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。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ホストで複数の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インターフェイス用に構成された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全体でのパケットのパフォーマンスと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VMXNET3デバイスのESX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` および `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 NetScaler 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。

VMware ESXi ハイパーバイザーの使用ガイドライン

- NetScaler VPX インスタンスは、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 Considerations**」セクションを参照してください。以下はその抜粋です。

- CPU または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展開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、たとえば、一部の仮想マシンの電源を切るか、別のホストに移行する (または DRS に自動的に移行させる) ことで、CPU 負荷を軽減します。
- NetScaler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使用することを推奨します。ハードウェアとESXiバージョンの互換性の詳細については、[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発揮するには、アプライアンスはホスト上でvCPU予約、メモリ予約、および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、次の問題が発生する可能性があります。
 - 高可用性フェイルオーバー
 - VPXインスタンス内のCPUスパイク
 - VPX CLIへのアクセスが遅い
 - 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
- 次の2つの条件のいずれかが満たされる場合、ハイパーバイザーはオーバース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。
 - 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。

インスタンスがオーバースプロビジョニングされている場合、ハイパーバイザーのスケジューリングオーバーヘッド、バグ、またはハイパーバイザーの制限により、ハイパーバイザーはインスタンスに予約されたリソース (CPU、メモリなど) を保証でき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ホストにはスケジューリングオーバーヘッドがあり、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を返すようにホスト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トリアージするために、ハイパーバイザーベンダーに連絡してください。

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

`Set ns vpxparam` パラメータ:

-cpuyield: 割り当てられているが使用されていないCPUリソースを解放するか、解放しないか。

- **YES:** 割り当てられているが使用されていない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におけるVPX CPU使用率で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` コマンドを使用します。

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行う必要があります。

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

注:

クラスターノードを「yield=YES」に設定したい場合は、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。このパラメーターには次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にCPU0を使用します。デフォルトでは、CPU0がメインクロックソースです。

- `show ns vpxparam`

このコマンドは、現在の `vpxparam` 設定を表示します。

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

このセクションには、Linux-KVMプラットフォーム上の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、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が含まれています。

- [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ホストで次の設定を実行します。

lstopo コマンドを使用してNICのNUMAドメインを見つける:

VPXとCPUのメモリ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」はNUMAドメイン#1に紐付けられています。



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、NUMAノード #0から約10 GBのRAMが割り当てられています。



NUMAノードマッピングを変更するには、次の手順に従います。

1. ホスト上のVPXの.xmlを編集します。

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

2. 次のタグを追加します。

```
<numatune>

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

</numatune>
```

3. VPXをシャットダウンします。

4. 次のコマンドを実行します。

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

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

5. VPXの電源をオンにします。次に、ホストで `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を物理コアにピン留めします。

- 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 使用状況を表示するには、次のコマンドを入力します。

```
mpstat -P ALL 5
```

mpstat コマンドの出力(/en-us/vpx/media/mpstat-output.png)

この出力では、8 が管理 CPU であり、9~11 がパケットエンジンです。

- vCPU から pCPU へのピン留めを変更するには、2つのオプションがあります。
 - VPX の起動後、次のコマンドを使用して実行時に変更します。

```
virsh vcpupin <VPX name> <vCPU id> <pCPU number>

virsh vcpupin NetScaler-VPX-XML 0 8

virsh vcpupin NetScaler-VPX-XML 1 9

virsh vcpupin NetScaler-VPX-XML 2 10

virsh vcpupin NetScaler-VPX-XML 3 11
```

- VPX に静的な変更を加えるには、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. ホスト上の 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 の電源を入れます。次に、ホストで `virsh vcpupin <VPX name>` コマンドの出力を確認して、更新された CPU ピン留めを確認します。

ホスト割り込みのオーバーヘッドを排除します。

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

ハイパーバイザーレベルでは、ホスト割り込みは、VPX の vCPU がピン留め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。

ホスト上で実行されている VM によって行われた VM exit をを見つけるには、`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、問題を示しています。

単一の VM が存在する場合、期待される値は 30~100K です。それ以上の値は、1 つ以上のホスト割り込みベクターが同じ 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、対応するカウンターが増加します。

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発生しないようにします。

```
service irqbalance stop
```

```
service irqbalance show - To check the status
```

```
service irqbalance start - Enable if needed
```

カウンター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ースを備えた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ースを、**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) インターフェースの最適なパフォーマンスを得るには、次の手順に従います。

- PCIeスロット/NICが属するNUMAドメインを特定します。
- VPXのメモリとvCPUは、同じNUMAドメインにピン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にバインドします。

1. トラフィックが開始されたら、ホストで `top` コマンドを実行します。



2. 仮想ホストプロセス（`vhost-<pid-of-qemu>` という名前）のアフィニティを特定します。
3. 以下のコマンドを使用して、vHostプロセスを以前に特定したNUMAドメイン内の物理コアにバインドします。

```
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、以下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します。

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

```
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、以下の手順に従います。

- PCIeスロット/NICが属するNUMAドメインを特定します。
- NetScaler 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およびメモリピンニングの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>

<cputune>

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

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

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

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

</cputune>

<numatune>

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

</numatune>

</domain>
```

Citrix Hypervisors 上の NetScaler VPX インスタンス

このセクションでは、Citrix Hypervisors 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について詳しく説明します。

- [Citrix Hypervisors 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 Hypervisors のパフォーマンス設定

「xl」コマンドを使用して NIC の NUMA ドメインを検索します。

```
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。

```
xl vcpu-list
```

NetScaler VM に 8 つ以上の vCPU を割り当てます。

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

```
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 ドメインにピン留めします。
- Domain-0 vCPU を残りの CPU にバインドします。

準仮想化インターフェースを備えた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とpNICあたり1つのvNICの構成が推奨されます。

準仮想化（netfront）インターフェースの最適なパフォーマンスを実現するには、次の手順に従います。

- PCIeスロットまたはNICが属するNUMAドメインを特定します。
- VPXのメモリとvCPUを同じNUMAドメインにピン留めします。
- 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にピン留めします。

ホストスレッドをDomain-0 vCPUにピン留めする:

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

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

xl list コマンドの出力 (/en-us/vpx/media/xl-list-command.png)

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

```
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サイズ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のプライマリディスクのサイズ変更は、SDX Management Serviceを使用してはサポートされていません。

NetScaler VPXでプライマリディスクサイズを動的に増やす

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。その後の増加ごとに、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た後で、それを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

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アタッチすると、`/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した場合。

シンボリックリンクを使用するには、そのようなコマンドをすべて `ln -s` に置き換えてください。また、失敗しているスクリプトもそれに応じて更新してください。

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などのより高い値に拡張します。Azureの場合は、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の電源を入れ、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。次の例では、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: 次のコマンドを実行して、da0 MBRパーティションのサイズを変更し、10 GBの空き領域を含めます。

```
gpart resize -i 1 da0
```

8. 空き領域を最後のパーティションにマージします。

例:

```
gpart resize -i 5 da0s1
```

9. `growfs` コマンドを使用して、新しく割り当てられた空き領域を含むようにファイルシステムを拡張します。

例:

```
growfs /dev/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。

NetScaler VPX で追加の管理 CPU コアを構成する

NetScaler VPX アプライアンスは通常、構成、監視、制御操作などの管理タスクに単一の CPU コアを使用します。管理要求が著しい環境では、このデフォルト構成がパフォーマンスのボトルネックとなり、応答時間の遅延や管理操作の潜在的な遅延を引き起こします。

この機能により、管理者は管理機能専用追加の CPU コアを割り当てることができます。この機能強化により、リソースの割り当てがより効率的になり、データトラフィック処理能力を損なうことなく管理操作が改善されます。特に管理ワークロード要件が高い環境では、応答性の向上と管理パフォーマンスの向上というメリットをユーザーは享受できます。

重要な考慮事項:

- この機能は、4 つ以上の vCPU と 2 つを超えるパケットエンジンを搭載した NetScaler VPX インスタンスでのみサポートされます。
- この機能を有効にすると、データトラフィック処理に割り当てられる CPU コアが減少します。
- 変更を有効にするにはアプライアンスを再起動する必要があるため、この構成変更はメンテナンス期間中にスケジュールしてください。

CLI を使用して追加の管理 CPU コアを構成する

追加の管理 CPU コアを有効にするには、次のコマンドを使用します。

```
enable extramgmtcpu
```

追加の管理 CPU コアを無効にするには、次のコマンドを使用します。

```
disable extramgmtcpu
```

注:

構成変更を有効にするには再起動が必要です。

追加の管理 CPU コア機能が有効になっているか無効になっているかを確認するには、次のコマンドを使用します。

```
show extramgmtcpu
```



```
ConfiguredState: DISABLED
```



```
EffectiveState: DISABLED
```

このコマンドは、追加の管理 CPU コアの構成済み状態と有効な状態を返します。このコマンドは、管理者がその稼働状況を迅速に評価し、有効または無効にするためにさらなるアクションが必要かどうかを判断するのに役立ちます。

GUI を使用して追加の管理 CPU コアを構成する

1. 「**Configuration > System > Settings**」に移動します。
2. 「**Configure Extra Management CPU**」をクリックします。

3. 「**Configured State**」ドロップダウンメニューで「**ENABLED**」を選択し、「**OK**」をクリックします。

4. 変更を適用するには、「**Configuration > System**」に戻り、「**Reboot**」をクリックします。「**OK**」をクリックしてアクションを確定します。


注:

- この機能を無効にするには、「**Configured State**」ドロップダウンメニューで「**DISABLED**」を選択します。
- 高可用性展開では、追加の管理CPUは各ノードで個別に有効にする必要があります。

統計と監視

次の例は、4 vCPUシステムで追加の管理CPUを割り当てる前と後での `stat cpu` コマンド出力の変化を示しています。

追加の管理CPUを有効にする前:



追加の管理CPUを有効にした後:



追加の管理CPU使用率は、`stat system` 出力のカウンタを使用して監視できます。



追加の管理CPUのトラブルシューティング

最小4つのvCPUと少なくとも2つのパケットエンジンという前提条件が満たされていない場合、CLIコマンドの実行時にエラーメッセージが表示されます。たとえば、パケットエンジンが1つしか動作していない場合、システムは次のように表示します。

「警告: 操作は許可されていません。PEプールは追加の管理CPUを有効にするのに十分ではありません。」

デフォルトでは、NetScaler VPXは管理機能にCPUコア0を割り当てます。以下のシェルコマンドを使用して確認できます。

- `sysctl netScaler.additional_mgmt_cpu`

`netScaler.additional_mgmt_cpu = 0` は、追加の管理CPUが設定されていないことを示します。

- `sysctl netScaler.cpus_in_mgmt_pool`

`netScaler.cpus_in_mgmt_pool : CPU(s) 0` は、コア0がデフォルトの管理CPUとして設定されていることを確認します。

CPUコア3などの追加の管理CPUが設定されている場合、これらのコマンドの出力は次のように異なる場合があります。

- `sysctl netScaler.additional_mgmt_cpu` は `1` を表示し、追加の管理CPUがアクティブであることを示します。
- `sysctl netScaler.cpus_in_mgmt_pool` は `CPU(s) 0,3` を表示し、CPUコア0とコア3の両方が割り当てられていることを示します。

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

クラウド環境でNetScaler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、本書では**プリブート**段階として扱われます。したがって、ADCプールライセンスのような特定のケースでは、特定のVPXインスタンスがはるかに短い時間で起動されます。この機能は、Microsoft Azure、Google Cloud Platform、およびAWSクラウドで利用できます。

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、ユーザーデータをインスタンスに渡すオプションがあります。ユーザーデータを使用すると、一般的な自動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リブートユーザーデータを提供する方法

プリブートユーザーデータをXML形式でクラウドインスタンスに提供できます。クラウドによってユーザーデータを提供するインターフェースは異なります。

AWS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

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、**インスタンスの詳細設定 > 高度な詳細**に移動し、**ユーザーデータ**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。

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「[インスタスの起動](#)」を参照してください。

**注:**

プリブートユーザーデータ機能のAWS IMDSv2専用モードは、NetScaler VPXリリース 13.1.48.x以降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、[インスタスの実行](#)に関する AWS ドキュメントを参照してください。

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

Azure コンソールを使用してプリブートユーザーデータを提供する

Azure コンソールを使用してNetScaler VPXインスタンスをプロビジョニングする際は、**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 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、[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。

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

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。

上記の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、次の例に示すように **<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構成を指定します。

注:

<NS-CONFIG> セクションには、有効なADC CLIコマンドが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構成を指定します。

注:

<NS-CONFIG> セクションには、有効なADC CLIコマンドが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、次の図に示すように、**<NS-CONFIG>** セクションで適用された構成で起動します。



ユーザースクリプト

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

<NS-SCRIPTS> タグ内に多く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:

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

- 最初のスクリプトは「/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コンテンツが入力され、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コンテンツが入力されます。ただし、起動実行コマンド `<SCRIPT-NS-BOOTUP>` が指定されていないため、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>

```

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



ライセンス

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

標準のプールライセンスコマンドを使用して、 **<LICENSE-COMMANDS>** セクションでライセンスタイプ、容量、ライセンスサーバーなどのプール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、VPX は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すると、構成された帯域幅が VPX に適用されます。
- ライセンスのチェックアウトが失敗した場合、ライセンスはライセンスサーバーから約 10 ～ 12 分以内に取得されません。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、ライセンスサーバー (10.102.38.214) から構成済みのライセンスをチェックアウトしようとします。

ライセンスコマンド(/en-us/vpx/media/license-commands.png)

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-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。

ライセンスサーバーの表示(/en-us/vpx/media/show-license-server.png)

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>` と `<NEW-BOOTSTRAP-SEQUENCE>` タグを使用できます。このセクションは、NetScalerアプライアンスに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、このセクションでは新しいブートストラップシーケンスを提供するオプションが提供されます。

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ース割当てに従います。

- **Eth0** - 特定のNSIPアドレスを持つ管理インターフェース。
- **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ース。
- **Eth2** - 特定のSNIPアドレスを持つサーバー向けインターフェース。

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、管理インターフェース (NSIP)、クライアント向けインターフェース (VIP)、およびサーバー向けインターフェース (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>` と `<NEW-BOOTSTRAP-SEQUENCE>` タグに許可されている異なる値でのブートストラップ動作を示しています。

SKIP-DEFAULT-BOOTSTRAP	NEW-BOOTSTRAP-SEQUENCE	ブートストラップ動作
はい	はい	デフォルトのブートストラップ動作はスキップされ、 <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。

- インターフェースの詳細のみを提供する
- インターフェースの詳細をIPアドレスとサブネットマスクとともに提供する
- `<NS-CONFIG>` セクションでブートストラップ関連コマンドを提供する

方法1：インターフェースの詳細のみを指定するカスタムブートストラップ

管理インターフェース、クライアント向けインターフェース、およびサーバー向けインターフェースを指定しますが、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設定されます。

AWS のカスタムブートストラップの例

以下の例に示すように、カスタムブートストラップシーケンスを提供します。詳細については、「[クラウドインスタンスでプレブートユーザーデータを提供する方法](#)」を参照してください。Eth1 インターフェイスは管理インターフェイス (NSIP) 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. **説明タブ**で、以下の図に示すように、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

以下の例に示すように、カスタムブートストラップシーケンスを提供します。詳細については、「[クラウドインスタンスでプレブートユーザーデータを提供する方法](#)」を参照してください。Eth2 インターフェイスは管理インターフェイス (NSIP) として、Eth1 インターフェイス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 ポータル > 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

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レブートユーザーデータを提供する方法](#)を参照してください。Eth1インターフェイスは管理インターフェイス (NSIP) 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。



GCPポータルでVMインスタンスが作成された後、次のように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。

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イ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。



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。



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

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

AWSのカスタムブートストラップの例

次の例では、デフォルトのブートストラップをスキップし、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。



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)



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。

Azure管理インターフェース `method2(/en-us/vpx/media/azure-mgmt-method2.png)`

Azureクライアントインターフェース `method2(/en-us/vpx/media/azure-client-method2.png)`

Azureサーバーインターフェース `method2(/en-us/vpx/media/azure-server-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。「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 nsip メソッド2の表示(/en-us/vpx/media/azure-show-nsip-method2.png)

GCPのカスタムブートストラップの例

次の例では、ADCの新しいブートストラップシーケンスが記述されており、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 (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)

GCP メソッド2の表示(/en-us/vpx/media/gcp-method2.png)

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。

「GCP NICの詳細」 (/en-us/vpx/media/gcp-nic-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」コマンドを実行できます。

「GCP show nsipコマンド」 (/en-us/vpx/media/gcp-show-nsip-method2.png)

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提供す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提供できます。`<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。また、NSIP、デフォルト

ルート、およびNSVLANを割り当てるコマンドも提供する必要があります。さらに、使用するクラウドに関連するコマンドも提供してください。

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、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。また、NSIPを作成し、デフォルトルートを追加し、NSVLANを追加するコマンドも提供する必要があります。

「AWSカスタムブートストラップ方法3」 (/en-us/vpx/media/aws-custom-bootstrap-method3.png)

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 -cItTimeout 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 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。

「AWS eth1 方法3」 (/en-us/vpx/media/aws-eth1-method3.png)

「AWS eth0 方法3」 (/en-us/vpx/media/aws-eth0-method3.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。



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-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-B00TSTRAP>` セクションで指定された新しいブートストラップシーケンスが適用されていることを確認できます。「show route」コマンドを実行して、サブネットマスクを確認できます。



GCP のカスタムブートストラップの例

この例では、ブートストラップ関連のコマンドは `<NS-CONFIG>` セクションに記載されています。

`<NS-B00TSTRAP>` 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。



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-B00T-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。



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構成が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。



AWSおよびAzureにおけるNICの接続と切断の影響

AWSとAzureでは、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、インスタンスから切断したりするオプションが提供されています。インターフェースの接続または切断により、インターフェースの位置が変更される可能性があります。したがって、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構成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、新しく利用可能になったインターフェースのプライマリIPを管理インターフェースの位置に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構成します。その結果、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モードを設定する

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
```

設定されているPEモードの状態を表示するには、次のコマンドを実行します。

```
show cpuparam
```

例：

```
> show cpuparam
```

```
    Pemode:  CPUBOUND
```

```
Done
```

クラウドでのNetScalerアプライアンスの初回起動時にPEモード設定を適用する

クラウドでのNetScalerアプライアンスの初回起動時にPEモード設定を適用するには、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、「[クラウドでのNetScalerアプライアンスの初回起動時にNetScaler VPX設定を適用する](#)」を参照してください。

パブリッククラウド上のNetScaler VPX 向けに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ます。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コアシステムでの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)

前述の図は、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により、物理コアごとに2つの論理コアを提供します。

次の図は、SMTが無効になっている最新のCPUで実行されている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、独自の複製リソースセット、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が無効な4コア/8スレッドシステムでのNetScalerの管理およびデータプレーンのワークロード

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

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

図3。SMTが有効な4コアシステムでのNetScalerの管理およびデータプレーンのワークロード

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

SMTを有効にすると、NetScalerのパフォーマンスが次のように向上します。

- すべての物理コアでデータプレーン機能を実行します。

- 管理プレーン機能を兄弟スレッドに移動します。
- 管理プレーン機能がデータプレーン機能のパフォーマンスを損なうことを防ぐための、柔軟なリソース制限メカニズムを導入。

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	14.1-12.x以降
アジュール	ハイパースレッディングを備 えた任意のインスタンスファ ミリー（例：Ds_v4）	14.1-12.x以降
GCP	e2-インスタンス	14.1-12.x以降

注記:

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の数を実質的に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、ライセンスの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が使用されている場合、8つのvCPUを有効にするには、SMT機能で10 Gbps以上のライセンスが必要です。以前は、4つのvCPUを有効にするには1 Gbpsのライセンスで十分でした。vCPUライセンスが使用されている場合、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. 現在の構成を保存します。
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、この機能が利用可能であり、有効になっていることを示します。

```
smt_handling and smt_handling_active are set to "1"

> shell sysctl -a | grep smt_handling

netscaler.smt_handling_platform: 1

netscaler.smt_handling: 1

netscaler.smt_handling_active: 1
```

SMT機能を無効にするには、次の手順に従います。

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


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

Done

reboot

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

Done
```

3. 再起動後、NetScalerは機能が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

netscaler.smt_handling_active: 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、以下のいずれかの出力を返すことがあります。

- 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。

- お使いのNetScaler VPXが13.1-48.xまたは14.1-12.xより古いバージョンです。
- お使いのクラウドが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を超える場合）。

```
> shell sysctl -a | grep smt_handling

netscaler.smt_handling_platform: 0(indicates not supported)

netscaler.smt_handling: 0 (indicates not enabled)

netscaler.smt_handling_active: 0 (indicates not active)
```

- SMT機能はサポートされていますが、有効になっていません。

```
> shell sysctl -a | grep smt_handling

netscaler.smt_handling_platform: 1 (available)

netscaler.smt_handling: 0          (not enabled)

netscaler.smt_handling_active: 0  (not active)
```

NetScaler 健全性チェッカーツール

健全性チェッカーツールは、NetScaler の健全性とパフォーマンスを評価し、一般的な構成の問題も特定します。

注:

現在、NetScaler 健全性チェッカーツールは、AWS クラウドでのみサポートされています。

NetScaler 健全性チェッカーツールは、以下の活動も実行します。

- HA トポロジ、ネットワーク、ライセンス、および権限を検証します。
- トラブルシューティングプロセスを効率化します。
- パブリッククラウドで確認された問題の迅速な解決を可能にします。
- プレーンテキストログ、JSON、HTML など、複数の形式で結果を生成します。

AWS 用 NetScaler 健全性チェッカーツール

NetScaler 健全性チェッカーツールは、展開タイプに基づいて以下の検証をカバーします。

スタンドアロンおよび 同一ゾーン HA 展開	弾性 IP アドレスを使 用するマルチゾーン HA 展開	プライベート IP アド レスを使用するマルチ ゾーン HA 展開
• IAM 権限の検証 • ライセンス チェック • ストレージ チェック • メタデータルー トチェック • DNS解決チェッ ク	• IAM権限の検証 • インターフェー スチェック • EIPチェック • INCモード チェック • IPSetチェック	• IAM権限の検証 • インターフェー スチェック • ルートチェック • デバイスイン デックスチェッ ク • 送信元/送信先 チェック

スタンドアロンおよび 同一ゾーン HA 展開	弾性 IP アドレスを使 用するマルチゾーン HA 展開	プライベート IP アド レスを使用するマルチ ゾーン HA 展開	
• EC2エンドポイントチェック • デフォルトゲートウェイの確認 • VLAN設定の確認 • ARPの確認 • SysIDの確認 • Cloudhadaemonの確認			

NetScaler CLI を使用して健全性チェッカーツールを実行する

コマンドプロンプトで次のように入力します。

```
> Shell

> root@ns# sanitychecker -c [standalone | multizone]
```

健全性チェッカーツールを実行すると、以下のファイルが JSON および HTML 形式で生成されます。

- /var/cloudsanitychecker/results.json
- /var/cloudsanitychecker/standalone.html

これらのファイルには、実行されたチェックの詳細な結果が含まれており、潜在的な問題を特定して分析するために使用できます。

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

ベアメタルとは、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、ワークロードを個別の仮想マシンに仮想化し、仮想化の柔軟性と信頼性を得ることにあります。

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ための前提条件

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

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入手する必要があります。そのベアメタルサーバーには、NetScaler VPXインスタンスを展開する前に、Linux KVM、VMware ESX、Citrix Hypervisor、Microsoft Hyper-Vなどのサポートされているハイパーバイザーのいずれかをインストールして構成しておく必要があります。

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 インスタンスをインストールするには、Hypervisor ホストへのネットワークアクセスを持つ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を示しています。

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	仮想化支援機能 (Intel-VTまたはAMD-V) が有効になっている2つ以上の64ビットx86 CPU。NetScaler VPXインスタンスを実行するには、ハイパーバイザー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の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： ハイパーバイザーのインストールにより、ハイパーバイザーホストの制御ドメイン用に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のドキュメント](#)を参照してください。

次の表に、各nCore VPX仮想アプライアンスに対してハイパーバイザーが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インスタンスを実行するため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	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. **インポート**ダイアログボックスのインポートファイル名で、NetScaler VPXインスタンスの **.xva** イメージファイルを保存した場所に移動します。「エクスポートされたVM」オプションが選択されていることを確認し、**次へ**をクリックします。
8. 仮想アプライアンスをインストールする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、**インポート**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. **完了**をクリックしてインポートプロセスを完了します。

注:

インポート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:

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」([/ja-jp/citrix-adc/current-release/upgrade-downgrade-citrix-adc-appliance.html](http://ja-jp/citrix-adc/current-release/upgrade-downgrade-citrix-adc-appliance.html))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イ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イ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インターフェイスの一部の機能をサポートしていません。Intel 82599、Intel X710、およびIntel XL710 NIC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
- クラスタリング
- 管理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、以下を確認してください。

- Intel 82599 NIC (NIC) をホストに追加します。
- `/etc/modprobe.d/blacklist.conf` ファイルに次のエントリを追加して、`ixgbevf` ドライバ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。

オプション `ixgbe max_vfs=<number_of_VFs>`

ここで、`<number_of_VFs>` 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 を推奨します。

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

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。

注記:

args:Mac= パラメーターで使用するMACアドレスを指定します。指定しない場合、**iovirt** スクリプトはMACアドレスをランダムに生成して割り当てます。また、リンクアグリゲーションモードでSR-IOV VFを使用する場合は、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て再割り当てする必要があります。

NetScaler VPXインスタンスに割り当てられている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=\&lt;Xen\_host\_UUID\> fn=unassign\_all
args:uuid=\&lt;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

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

1. 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ネットワークを構成する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の詳細については、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の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-TOP0 ~]# 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を設定できます。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](#)

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

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 Server 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 Client 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されている必要があります。

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、次の操作を実行してください。

- 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](#)」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が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イスまたはPCIパススルーサポートについては、次のプロセッサと設定が有効になっていることを確認してください。

- Intel-VTをサポートするIntelプロセッサ
- AMD-VをサポートするAMDプロセッサ

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

SR-IOVモードでサポートされているNICは次のとおりです。

- NetScalerリリース13.1-42.x以降のMellanox ConnectX-4 NIC
- インテル 82599 NIC

次の表に、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、メモリ割り当て全体を予約する必要があります。ESXの1つのCPUコアの速度に少なくとも等しい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の詳しい要件については、 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、**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 Client を使用して VMware ESX サーバーに仮想アプライアンスをインストールできます。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 Client を起動します。
2. **IP address / Name** テキストボックスに、接続する VMware ESX サーバーの IP アドレスを入力します。
3. **ユーザー名およびパスワード** 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。
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

インスタ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、成功したことを示すポップアップウィンドウが表示されます。

- 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。
- 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。
- 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でのVLAN IDの設定に関する詳細な手順については、VMwareのドキュメントを参照してください。

VMware vMotionを使用して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 パススルーおよび SR-IOV インターフェイスで構成された仮想マシンでは vMotion 機能をサポートしていません。
- サポートされているインターフェイスは E1000 と VMXNET3 です。VPX インスタンスで vMotion を使用するには、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イ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。

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イスを使用するように構成するには：

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。



4. \<virtual_appliance\> - ダイアログボックスで、 セクションをクリックします。



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**ドロップダウンメニューで、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ています。

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、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 = 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る (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**と最大値として表示されている数値を選択します。



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



8. **新しいネットワークアダプター**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。



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必要があります。

```
> show interface summary
```

```
-----
-----
```

	Interface	MTU	MAC	Suffix
1	0/1	1500	00:0c:29:89:1d:0e	NetScaler Vir...rface, VMXNET3
2	1/1	9000	00:0c:29:89:1d:18	NetScaler Vir...rface, VMXNET3
3	1/2	9000	00:0c:29:89:1d:22	NetScaler Vir...rface, VMXNET3
4	L0/1	9000	00:0c:29:89:1d:0e	Netscaler Loopback interface

```
-----
-----
```

注記:

VMXNET3インターフェイスを4つ以上追加し、NetScaler VPXアプライアンスを再起動すると、VMware ESXハイパーバイザーが、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、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。

これはVMware ESXハイパーバイザーの制限です。

VMXNET3ネットワークインターフェイスの受信リングサイズを設定する

VMware ESX上のVMXNET3ネットワークインターフェイスの受信リングサイズを増やすことができます。リングサイズを大きくすると、トラフィックが急増したときのパケットドロップが減少します。

注記:

この機能は、リリース14.1ビルド14.x以降で利用可能です。

VMXNET3ネットワークインターフェイスでリングサイズを設定するには

コマンドプロンプトで、次のように入力します。

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

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

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

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シングルルートI/O仮想化 (SR-IOV) ネットワークインターフェイスを使用するように構成できます。

制限事項

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

- ESX VPX上のIntel 82599 10G NICを使用するSR-IOVインターフェイスでは、次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
 - スタティックリンクアグリゲーションとLACP
 - クラスタリング
 - 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
 - IPv6
- KVM VPX上のIntel 82599 10G NICを使用するSR-IOVインターフェイスでは、次の機能はサポートされていません。

- スタティックリンク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
- SR-IOV VFインターフェースのハイパーバイザー上でのVLAN設定は、`ip link` コマンドではサポートされていません

前提条件

- 次のいずれかの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ドライバーバージョン3.7.13.7.14iov以降を推奨します。
 - メラノックス ConnectX-4 ネットワークインターフェースカード
- 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。

- a. vSphere Web Clientで、ホストに移動します。
- b. **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。

- c. 物理アダプターを選択し、次に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。

- d. 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。

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

- f. **OK**をクリックします。
- g. 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のドキュメントを参照してください。

注:

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

VMware vSphere Web Client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:

1. vSphere Web Clientで、ホストとクラスタを選択します。
2. 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。手順は次のとおりです。
 - a. NetScaler VPXインスタンスの電源をオフにします。
 - 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。

VM互換性(/en-us/vpx/media/compatibility_new.png)

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

5. 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. 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数を選択します。
- b. 「**Cores per Socket**」ドロップダウンリストで、ソケット数を選択します。

c. (オプション) 「**CPU Hot Plug**」フィールドで、「**Enable CPU Hot Add**」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. 「**Reservation**」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。

「CPU設定の編集」 (/en-us/vpx/media/edit_settings_cpu_settings1.png)

e. 「**Limit**」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。

「設定CPUの編集」 (/en-us/vpx/media/edit_settings_cpu_settings2.png)

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

「CPU設定の編集」 (/en-us/vpx/media/edit_settings_cpu_settings3.png)

6. 「**Memory**」セクションで、次の設定を更新します。

- 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 = 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。

SR-IOVイメージ(/en-us/vpx/media/sr-iov_img5.jpg)

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。

SR-IOVイメージ(/en-us/vpx/media/sr-iov_img6.jpg)

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。

VM設定オプション(/en-us/vpx/media/edit_settings_vm_options.jpg)

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、以下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必要があります。

```
> show interface summary
```


	Interface	MTU	MAC	Suffix

1	0/1	1500	00:0c:29:1b:81:0b	NetScaler Virtual
	Interface			
2	10/1	1500	00:50:56:9f:0c:6f	Intel 82599 10G VF
	Interface			
3	10/2	1500	00:50:56:9f:5c:1e	Intel 82599 10G VF
	Interface			
4	10/3	1500	00:50:56:9f:02:1b	Intel 82599 10G VF
	Interface			
5	10/4	1500	00:50:56:9f:5a:1d	Intel 82599 10G VF
	Interface			
6	10/5	1500	00:50:56:9f:4e:0b	Intel 82599 10G VF
	Interface			
7	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 &lt;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
```

ESXハイパーバイザー上のNetScaler VPXをSR-IOVモードでIntel QATを使用してSSLアクセラレーションを行うように構成する

VMware ESXハイパーバイザー上のNetScaler VPXインスタンスは、Intel QuickAssist Technology (QAT) を使用してNetScaler SSLパフォーマンスを高速化できます。Intel QATを使用すると、すべての高遅延暗号処理をチップにオフロードできるため、1つ以上のホストCPUを他のタスクに解放できます。

以前は、すべてのNetScalerデータパス暗号処理は、ホストvCPUを使用してソフトウェアで行われていました。

注:

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

前提条件

- ESXホストには、1つ以上のIntel C62x (QAT) チップがプロビジョニングされています。
- NetScaler VPXはVMware ESXのハードウェア要件を満たしています。詳細については、「[VMware ESXにNetScaler VPXインスタンスをインストールする](#)」を参照してください。

制限事項

個々のVMに対して暗号化ユニットや帯域幅を予約する規定はありません。Intel QATハードウェアの利用可能なすべての暗号化ユニットは、QATハードウェアを使用するすべてのVMで共有されます。

Intel QATを使用するためのホスト環境をセットアップする

1. VMwareホストに、C62xシリーズ (QAT) チップモデル用のIntel提供のVMwareドライバーをダウンロードしてインストールします。Intelパッケージのダウンロードとインストール手順の詳細については、「[Intel QuickAssist Technology Driver for VMware](#)」を参照してください。
2. ESXホストでSR-IOVを有効にします。
3. 仮想マシンを作成します。VMを作成する際は、パフォーマンス要件を満たすために適切な数のPCIデバイスを割り当てます。

注:

各C62x (QAT) チップは、最大3つの個別のPCIエンドポイントを持つことができます。各エンドポイントはVFsの論理的な集合であり、チップの他のPCIエンドポイントと帯域幅を均等に共有します。各エンドポイントは、16個のPCIデバイスとして表示される最大16個のVFを持つことができます。これらのデバイスをVMに追加して、QATチップを使用して暗号化アクセラレーションを行うことができます。

注意事項

- VMの暗号化要件が複数のQAT PCIエンドポイント/チップを使用する場合、対称的な分散を実現するために、対応するPCIデバイス/VFをラウンドロビン方式で選択することをお勧めします。

- 選択するPCIデバイスの数は、ライセンスされたvCPUの数（管理vCPUの数を含まない）と等しくすることをお勧めします。利用可能なvCPUの数よりも多くのPCIデバイスを追加しても、必ずしもパフォーマンスが向上するとは限りません。

例:

3つのエンドポイントを持つIntel C62xチップを1つ搭載したESXホストを考えます。6つのvCPUを持つVMをプロビジョニングする際、各エンドポイントから2つのVFを選択し、それらをVMに割り当てます。このような割り当てにより、VMの暗号化ユニットの効果的かつ均等な分散が保証されます。利用可能なvCPUの合計から、デフォルトで1つのvCPUが管理プレーン用に予約され、残りのvCPUはデータプレーンPEsで利用できます。

vSphere Web Clientを使用してQAT VFをVPXに割り当てる

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

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

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

4. VMを再度パワーオンします。
5. NetScaler CLIで `stat ssl` コマンドを実行してSSLサマリーを表示し、QAT VFをVPXに割り当てた後、SSLカードを確認します。


展開について

この展開は、以下のコンポーネント仕様でテストされました。

- **NetScaler VPXのバージョンとビルド:** 14.1–8.50
- **VMware ESXiのバージョン:** 7.0.3 (ビルド 20036589)
- **VMware用Intel C62x QATドライバーのバージョン:** 1.5.1.54

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

May 24, 2018

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た物理PCIおよびPCIeデバイスに直接アクセスできます。

前提条件

- ホスト上の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:
 - インテル X710 10G NIC
 - Intel XL710デュアルポート40G NIC
 - Intel XL710シングルポート40G NIC
 - Intel XXV710デュアルポート25G NIC

ホスト上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。ホスト上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。利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. 設定する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。



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設定する

NetScaler VPXインスタンスでパススルーPCIデバイスを設定するには、次の手順に従います。

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。



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イスとPCIパススルーネットワークインターフェイスは共存できません。



5. ゲスト仮想マシンの電源をオンにします。

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ように構成する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。

プリブートユーザーデータとその形式の詳細については、「[クラウド内のNetScalerアプライアンスの初回起動時にNetScaler VPX構成を適用する](#)」を参照してください。

注:

ESXでプリ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、「[Sample- `<NS-CONFIG>` -section](#)」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リ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リ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 ドライブから ISO イメージとして仮想マシンにユーザーデータを投入できます。

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

1. プリ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 `userdata` として厳密に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。

次のサンプル構成は、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を自動的に起動しないでください。



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に接続します。



5. **新しい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> > <output-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。例えば、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="PE5TLVBRSR1CT09ULUNPTkZJRz4KICAgIDxOUy1DT05GSUc+Cg1hZGQgcml  
dGUgMC4wLjAuMCAw  
  
LjAuMCAwIDEwLjEwMi4zMCAxCiAgICA8L05TLUNPTkZJRz4KCicAgICA8TlMtQk9PVFNUUKF  
QPgog  
  
ICAgICAgICAgICA8U0tJUC1ERUZBVUxULUJPT1RTVFJBUD5ZRVM8L1NLSVAtREVGOVVmVC1
```

CT09U

U1RSQVA+CiAgICAgICAgICAgIDx0RVctQk9PVFNuUkFQLVNFUVVFTkNFP1lFUzwwTkVXLUJPT1RT

VFJBUC1TRVFVRU5DRT4KCiAgICAgICAgPE1HTVQtSU5URVJGQUFLUNPTkZJRz4KICAgICAgICAg

ICAgICAgIDxJTlRFUkZBQ0UtTlVNPiBlbGgwIDwvSU5URVJGQUFLU5VTT4KICAgICAgICAgICAg

ICAgIDxJUD4gICAgMTAuMTAyLjM4LjIxOSA8L0lQPgogICAgICAgICAgICAgICAgPFNVQk5FVC1N

QVNLPiAyNTUuMjU1LjI1NS4wIDwvU1VCTkVULU1BU0s+CiAgICAgICAgPC9NR01ULU10VEVS

RS1DT05GSuc+CiAgICA8L05TLUJPT1RTVFJBUD4KPC90Uy1QUkUtQk9PVC1DT05GSuc+Cg=

= ">

<Label>Userdata</Label>

<Description> Userdata for ESX VPX </Description>

</Property>

</ProductSection>

5. 変更されたOVFテンプレート（Productセクションを含む）をVM展開に使用します。

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> > <outuput-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="PE5TLVBRSRS1CT09ULUNPTkZJRz4KICAgIDxOUy1DT05GSUc+Cg1hZGQgcm91
dGUgMC4wLjAuMCAw

```

LjAuMC4wIDEwLjEwMi4zOC4xCiAgICA8L05TLUNPTkZJRz4KCiAgICA8TlMtQk9PVFNuUkFQpogog

ICAgICAgICAgICA8U0tJUC1ERUZBVUxULUJPT1RTVFJBUD5ZRVM8L1NLSVAtREVgQVVMVC1CT09U

U1RSQVA+CiAgICAgICAgICAgIDx0RVctQk9PVFNuUkFQLVNFUVVFTkNFP1lFUzwwTkVXLUJPT1RT

VFJBUC1TRVFRU5DRT4KCiAgICAgICAgPE1HTVQtSU5URVJGQUFLUNPTkZJRz4KICAgICAgICAg

ICAgICAgIDxJTlRFUkZBQ0UtTlVNPiBlbGgwIDwvSU5URVJGQUFLU5VTT4KICAgICAgICAgICAg

ICAgIDxJUD4gICAgMTAuMTAyLjM4LjIxOSA8L0lQPgogICAgICAgICAgICAgICAgPFNVQk5FVC1N

QVNLPiAyNTUuMjU1LjI1NS4wIDwvU1VCTkvVULU1BU0s+CiAgICAgICAgPC9NR01ULU10VEVSrkFD

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)

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

VMware Cloud (VMC) on AWS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。VMC on AWS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つ存在する必要があります。
- 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/content/citrix/en-us/13200/licenses/licenses.html>にある『*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	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インスタンスの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-`<release number>-<build number>.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. **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、 をクリックします。
3. **File** メニューで、**Deploy OVF Template** をクリックします。
4. **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、 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。 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 を選択します。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ロールが有効なWindows Serverをインストールする必要があります。Hyper-Vロールのインストール中に、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を実行するには、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イス、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 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する場合は、「[NetScaler VPXスタンドアロンアプライアンスのアップグレード](#)」を参照してください。

注:

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、次の手順を実行します。

- 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を取得します。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

次の表に、Microsoftサーバーの最小システム要件を示します。

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 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後、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。

最大8つの仮想NICを設定できます。物理NICがDOWNしている場合でも、仮想アプライアンスは仮想NICがUPしていると見なします。これは、同じホスト（サーバー）上の他の仮想アプライアンスと引き続き通信できるためです。

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 Manager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 Manager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 Manager**をクリックします。
2. ナビゲーションペインの**Hyper-V Manager**の下で、NetScaler VPXインスタンスをインストールするサーバーを選択します。
3. **アクションメニュー**で、**仮想マシンのインポート**をクリックします。

4. **仮想マシンのインポートダイアログボックスの場所**で、NetScaler VPXインスタンスのソフトウェアファイル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、Snapshots、Virtual Hard Disks、およびVirtual Machines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いることを確認してください。

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**の下にリスト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いることを確認してください。

Hyper-V上でのNetScaler VPXインスタンスの自動プロビジョニング

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アドレスなどを構成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、次の方法を使用できます。

- 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ドライブは、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ファイル名または形式が正しくありません。

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

1. インポートした仮想アプライアンスを選択し、**アクションメニュー**で**設定**を選択します。
2. **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. 右ペインで、デバイスのリストから**ネットワークアダプター**を選択します。
4. **追加**をクリックします。
5. 左ペインに****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. 右ペインで、**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、次に**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する場合は、詳細について記事[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。

ホストの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

予期せぬ動作を避けるため、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インスタンス上で動作する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。

お使いの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*
```

前述の拡張機能の**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コアを提供してください。
- プロセッサ速度に関する具体的な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カーネル用に最低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の要件に必要なスペースを計算してください。単一のNetScaler VPX VMには20GBのディスクスペースが必要です。

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ファイルで構成されています。

ネットワーク要件

NetScaler VPXは、virtIO準仮想化、SR-IOV、およびPCIパススルーネットワークインターフェイスをサポートしています。

サポートされているネットワークインターフェイスの詳細については、以下を参照してください。

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

ソースインターフェイスとモード

ソースデバイスタイプは、BridgeまたはMacVTapのいずれかです。MacVTapでは、VEPA、Bridge、Private、Pass-through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イスの種類とサポートされているトラフィックの種類を確認してください。

ブリッジ:

- Linuxブリッジ。
- 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、ホストLinux上の `Ebtables` および `iptables` 設定が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。
- 同じものを使用する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):

- ブリッジよりもパフォーマンスが優れています。
- 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するVM間通信は不可能です。

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する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、ホストをゲストVMに接続する仮想インターフェースであるVNetインターフェースで **lro** 機能を無効にする必要があります。

```
[root@localhost ~]# brctl show eth6_br
```

bridge name	bridge id	STP enabled	interfaces
eth6_br	8000.00e0ed1861ae	no	eth6
			vnet0
			vnet2

```
[root@localhost ~]#
```

上記の例では、2つの仮想インターフェースはeth6_brから派生しており、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
```

プロミスキヤスモード

次の機能が動作するためには、プロミスキヤ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

プロミスキヤスモードを有効にするには、次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:143300008 (14.3 MB)  TX bytes:1019416071 (1.0 GB)

[root@localhost ~]#
```

必要なモジュール

ネットワークパフォーマンスを向上させるには、Linuxホストにvhost_netモジュールが存在することを確認してください。vhost_netモジュールの存在を確認するには、Linuxホストで次のコマンドを実行します。

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

vhost_netがまだ実行されていない場合は、次のコマンドを入力して実行します。

```
modprobe vhost\_net
```

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。

KVM仮想化を有効にしたお好みのLinuxディストリビューションをインストールした後、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場合、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 上での一般的な 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、初期設定時にNetScaler VPXインスタンスの管理IPアドレス、ネットワークマスク、およびデフォルトゲートウェイを入力する必要はありません。

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ファイルの例:

以下はOVF XMLファイルの内容の例です。これをサンプルとしてファイルを作成できます。

```
<?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」が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ー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. 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。

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**ウィンドウの「オペレーティングシステムのインストール方法を選択してください」の下で、**既存のディスクイメージをインポート**を選択し、**次へ**をクリックします。

5. **既存のストレージパスを指定**フィールドで、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、**次へ**をクリックします。

6. **メモリとCPUの設定を選択**の下で、次の設定を選択し、**次へ**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
 - デバイスマodel — `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、「インストール開始」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イスを追加できます。

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、次の手順に従います。

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. Storage format 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イスを追加できます。

CDROM 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** を追加 をクリックします。
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、**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。



3. VPX インスタンスの電源をオンにすると、OVF ファイルで提供されているネットワーク構成で自動プロビジョニングされます (例のスクリーンショットに示すとおり)。



4. 自動プロビジョニングに失敗した場合、インスタンスはデフォルトの IP アドレス (192.168.100.1) で起動します。その場合は、初期設定を手動で完了する必要があります。詳細については、「[ADC を初めて構成する](#)」を参照してください。

Virtual Machine Manager を使用して NetScaler VPX インスタンスにインターフェイスを追加する

KVM に NetScaler VPX™ インスタンスをプロビジョニングした後、追加のインターフェイスを追加できます。

インターフェイスを追加するには、次の手順に従います。

1. KVM 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. 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 **設定** を選択します。
3. ヘッダーの  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. **ネットワーク** をクリックします。 **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. 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、Bridge または MacVTap のいずれかです。MacVTap の場合、VEPA、Bridge、Private、Pass-through 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 ホストで構成されているブリッジ名を入力します。

注:

KVM ホストに Linux ブリッジが構成され、物理インターフェースがブリッジにバインドされ、ブリッジが UP 状態になっていることを確認してください。



- iii. デバイスモデル — `virtio`。

iv. 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イスを選択します。

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、Passthroughです。インターフェイスとモードの詳細については、「ソースインターフェイスとモード」を参照してください。

v. **適用**をクリックします。

6. VPXインスタンスを自動プロビジョニングする場合は、このドキュメントの「自動プロビジョニングを有効にするための構成ドライブの追加」セクションを参照してください。それ以外の場合は、VPXインスタンス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、オートネゴシエーションなどのインターフェイスパラメータ設定はサポートされていません。

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

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

詳細については、「[NetScaler VPXでサポートされているNIC](#)」を参照してください。

このセクションでは、次の方法について説明します。

- 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
- SR-IOVインターフェイスで静的LA/LACPを構成する
- SR-IOVインターフェイスでVLANを構成する

制限事項

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インターフェイスが少なくとも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。
- ハイパーバイザーでのSRIOV VFインターフェイスのVLAN構成は、`ip link` コマンドを介して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イ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イスとして使用されている場合、ジャンボフレームはサポートされません。
- インターフェイスが切断および再接続されると、インターフェイスリストが再順序付けされません。
- 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イ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 の場合、インターフェイス名は40/X です。

- VPXインスタンスでは、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イスがサポートされます。

注

Intel X710 10G、Intel XL710 40G、および Intel X722 10G NIC が 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

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で使用するように構成する前に、次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 ネットワークインターフェースカード	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ホスト上でドライバーをブラックリスト化します。	/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。

タスク	インテル 82599 ネットワークインターフェースカード	INTEL X710、XL710、および X722 NIC
1. KVMホスト上で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> = インターフェース名。	カーネル3.8の以前のバージョンを使用している場合は、<code>/etc/modprobe.d/ixgbe</code> 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 ixgbe max_vfs=<number_of_VFs></code>。 カーネル3.8バージョン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: <code>echo <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。	カーネル3.8の以前のバージョンを使用している場合は、<code>/etc/modprobe.d/i40e.conf</code> 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 max_vfs=<number_of_VFs></code>。 カーネル3.8バージョン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: <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のKVMホストでSR-IOV VFを有効にする。



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



図4: VF を永続化する。



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

Virtual Machine Manager 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、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、 を選択します。



3. \<virtual machine on KVM\> ウィンドウで、 i アイコンを選択します。



4. ハードウェアの追加を選択します。



5. 新しい仮想ハードウェアの追加ダイアログボックスで、次の操作を行います。
 - a. PCI ホストデバイスを選択します。
 - b. 「ホストデバイス」セクションで、作成したVFを選択し、「完了」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. 作成したVF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インターフェースで静的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*
```

ここで:

- Interface_name – インターフェース名です。
- VF_id – 仮想機能IDです。

例:

なりすましチェックを無効にする(/en-us/vpx/media/config_sriov_diabale_spoof.jpg)

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構成します。詳細な手順については、「[リンクアグリゲーションの構成](#)」を参照してください。

SR-IOVインターフェースでのVLANの構成

SR-IOV VFでVLANを構成できます。詳細な手順については、「[VLANの構成](#)」を参照してください。

重要:

KVMホストにVFインターフェースのVLAN設定が含まれていないことを確認してください。

KVMハイパーバイザー上のNetScaler VPXをSR-IOVモードでIntel QATを使用してSSLアクセラレーションを行うように構成する

Linux KVMハイパーバイザー上のNetScaler VPXインスタンスは、Intel QuickAssist Technology (QAT) を使用してNetScalerのSSLパフォーマンスを高速化できます。Intel QATを使用すると、すべての高遅延暗号処理をチップにオフロードできるため、1つ以上のホストCPUを他のタスクに解放できます。

以前は、すべてのNetScaler®データパス暗号処理は、ホストvCPUを使用してソフトウェアで行われていました。

注:

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

前提条件

- Linuxホストには、マザーボードに直接統合されているか、外部PCIカードに追加されているIntel QAT C62xチップが搭載されています。

Intel QAT C62xシリーズモデル: C625、C626、C627、C628。これらのC62xモデルのみが公開鍵暗号化 (PKE) 機能を備えています。他のC62xバリエーションはPKEをサポートしていません。

- NetScaler VPXはVMware ESXのハードウェア要件を満たしています。詳細については、「[Linux KVMプラットフォームにNetScaler VPXインスタンスをインストールする](#)」を参照してください。

制限事項

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

Intel QATを使用するためのホスト環境をセットアップする

- Linuxホストに、C62xシリーズ (QAT) チップモデル用のIntel提供ドライバーをダウンロードしてインストールします。Intelパッケージのダウンロードとインストール手順の詳細については、「[Intel QuickAssist Technology Driver for Linux](#)」を参照してください。ダウンロードパッ

ページの一部としてreadmeファイルが利用可能です。このファイルには、ホストでのパッケージのコンパイルとインストールに関する手順が記載されています。

ドライバーをダウンロードしてインストールした後、次の健全性チェックを実行します。

- C62xチップの数をメモします。各C62xチップには最大3つのPCIeエンドポイントがあります。
- すべてのエンドポイントがUPであることを確認します。 `adf_ctl status` コマンドを実行して、すべてのPFエンドポイント (最大3つ) のステータスを表示します。

```
root@Super-Server:~# adf_ctl status

Checking status of all devices.

There is 51 QAT acceleration device(s) in the system

qat_dev0 - type: c6xx, inst_id: 0, node_id: 0, bsf: 0000:1a:00.0,
#accel: 5 #engines: 10 state: up

qat_dev1 - type: c6xx, inst_id: 1, node_id: 0, bsf: 0000:1b:00.0,
#accel: 5 #engines: 10 state: up

qat_dev2 - type: c6xx, inst_id: 2, node_id: 0, bsf: 0000:1c:00.0,
#accel: 5 #engines: 10 state: up
```

- すべてのQATエンドポイントでSRIOV (VFサポート) を有効にします。

```
root@Super-Server:~# echo 1 > /sys/bus/pci/devices/0000\:1a\:00.0/
srhov_numvfs

root@Super-Server:~# echo 1 > /sys/bus/pci/devices/0000\:1b\:00.0/
srhov_numvfs

root@Super-Server:~# echo 1 > /sys/bus/pci/devices/0000\:1c\:00.0/
srhov_numvfs
```

- すべてのVFが表示されていることを確認します (エンドポイントあたり16 VF、合計48 VF)。

- `adf_ctl status` コマンドを実行して、すべてのPFエンドポイント (最大3つ) と各Intel QAT チップのVFがUPであることを確認します。この例では、システムにはC62xチップが1つしかないため、合計51個のエンドポイント (3 + 48 VF) があります。



2. LinuxホストでSR-IOVを有効にします。
3. 仮想マシンを作成します。VMを作成する際は、パフォーマンス要件を満たすために適切な数のPCIデバイスを割り当てます。

注:

各C62x (QAT) チップは、最大3つの個別のPCIエンドポイントを持つことができます。各エンドポイントはVFの論理的な集合であり、チップの他のPCIエンドポイントと帯域幅を均等に共有します。各エンドポイントは最大16個のVFを持つことができ、これらは16個のPCIデバイスとして表示されます。これらのデバイスをVMに追加して、QATチップを使用して暗号化アクセラレーションを実行します。

注意点

- VMの暗号化要件が複数のQAT PCIエンドポイント/チップを使用することである場合、対称的な分散を実現するために、対応するPCIデバイス/VFをラウンドロビン方式で選択することをお勧めします。
- 選択するPCIデバイスの数は、ライセンスされたvCPUの数 (管理vCPUの数を含まない) と同じにすることをお勧めします。利用可能なvCPUの数よりも多くのPCIデバイスを追加しても、必ずしもパフォーマンスが向上するとは限りません。

例:

3つのエンドポイントを持つIntel C62xチップを1つ搭載したLinuxホストを考えます。6つのvCPUを持つVMをプロビジョニングする際、各エンドポイントから2つのVFを選択し、それらをVMに割り当てます。この割り当てにより、VMの暗号化ユニットが効果的かつ均等に分散されます。利用可能なvCPUの総数のうち、デフォルトでは1つのvCPUが管理プレーン用に予約され、残りのvCPUはデータプレーンPE用に利用できます。

Linux KVMハイパーバイザーに展開されたNetScaler VPXにQAT VFを割り当てる

1. Linux KVM仮想マシンマネージャーで、VM (NetScaler VPX) の電源がオフになっていることを確認します。

2. ハードウェアの追加 > PCIホストデバイスに移動します。

3. Intel QAT VFをPCIデバイスに割り当てます。



4. 完了をクリックします。

5. 前述の手順を繰り返して、合計vCPU数から1を引いた数まで、1つ以上のIntel QAT VFをNetScaler VPXインスタンスに割り当てます。1つのvCPUは管理プロセス用に予約されているためです。

VMあたりのQAT VF数 = vCPU数 - 1

6. VMの電源をオンにします。

7. NetScaler CLIで `stat ssl` コマンドを実行してSSLサマリーを表示し、NetScaler VPXにQAT VFを割り当てた後、SSLカードを確認します。

この例では、5つのvCPUを使用しており、これは4つのパケットエンジン (PE) を意味します。



展開について

この展開は、以下のコンポーネント仕様でテストされました。

- NetScaler VPXバージョンとビルド: 14.1–8.50
- Ubuntuバージョン: 18.04、カーネル 5.4.0-146
- Linux 用 Intel C62x QAT ドライバーバージョン: L.4.21.0-00001

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

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ー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ー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。

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

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。

PCIホスト設定(/en-us/vpx/media/pci_host_settings.jpg)

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、以下のコマンドを使用して構成を確認できます。

SR-IOVコマンドの構成(/en-us/vpx/media/config_sriov_vmm_command.jpg)

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必要があります。

インターフェイスの概要を表示(/en-us/vpx/media/show_interface_summary.jpg)

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の機能と似ています。VMゲストのステータスを変更したり（開始、停止、一時

停止など)、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ゲスト管理操作をスクリプト化するのにも役立ちます。

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
- NS-VMディスクイメージ

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。 \<ドメイン名\>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。

注記:

ドメイン名とMACアドレスは一意である必要があります。

- **source file**— ディスクイメージの絶対ソースパスを指定します。ファイルパスは絶対パスである必要があります。RAWイメージファイルまたはQCOW2イメージファイルのパスを指定できます。

RAWイメージファイルを指定したい場合は、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' />
```

絶対QCOW2ディスクイメージのソースパスを指定し、以下の例に示すようにドライバータイプを**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** — インターフェースを指定します。
- **mode** — モードを指定します。デフォルトのインターフェースは**Macvtap Bridge**です。

例: モード: MacVTap Bridge ターゲットインターフェースを `ethx`、モードをブリッジ、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='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ースです。

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
```

virsh プログラムを使用して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` ファイルで、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、インターフェース全体でMACアドレスが一意であることを確認します。
- ソースデバイス— インターフェース名を指定します。
- モード— 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` として設定し、モードをブリッジとして設定し、モデルタイプを `virtio` として設定します。

```
<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、そのMACアドレスがすべてのインターフェースで一貫であることを確認します。
- 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ているステータス（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ゲストへ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オプションを選択します。

- ゲストを起動およびシャットダウンする

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または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>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オンにして、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に接続する

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

例:

```
virsh console NetScaler-VPX
```

- ゲストを起動およびシャットダウンする。

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
```

- ゲストを再起動する

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

例：

```
virsh reboot NetScaler-VPX
```

ゲストを削除する

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`の定義を解除する必要があります。

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

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

例:

```
virsh shutdown NetScaler-VPX

virsh undefine NetScaler-VPX
```

注:

削除コマンドではディスクイメージファイルは削除されません。これは手動で削除する必要があります。

KVM上の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) で実行されているNetScaler VPXインスタンスを、Open vSwitch (OVS) とData Plane Development Kit (DPDK) を使用するように構成して、ネットワークパフォー

マンスを向上させることができます。このドキュメントでは、KVMホスト上のOVS-DPDKによって公開される `vhost-user` ポートでNetScaler VPXインスタンスを動作させる方法について説明します。

OVSは、オープンソースのApache 2.0ライセンス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。以下は、KVM上の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です。

- 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` ポートを持つ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=dppdkvhostuser -- 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>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' />
```

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

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

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

```
</cputune>
```

```
<numatune>
```

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

```
</numatune>
```

```
<resource>
```

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

```
</resource>
```

```
<os>
```

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

<boot dev='hd' />

</os>

<features>

<acpi/>

<apic/>

</features>

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

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

<vendor>Intel</vendor>

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

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

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

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

```
<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-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' />

<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

NetScalerアプライアンスの初回起動時に、KVMハイパーバイザー上で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にかかる時間を大幅に短縮できます。

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/current-release/apply-vpx-config-at-preboot.html](https://ja-jp/vpx/current-release/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. 以下の手順を使用して、Virtual Machine Managerで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。

Red Hat OpenShift Virtualization に NetScaler VPX をインストールする

NetScaler® は、Red Hat® OpenShift® Virtualization (RHOCPV) 専用の VPX イメージを提供し、展開と管理のエクスペリエンスを大幅に向上させます。RHOCPV コンソールを使用してインスタンスを展開できるようになりました。お客様は公式サポートと必要なすべてのリソースへのアクセスを受けられます。これにより、RHOCPV インフラストラクチャにシームレスに展開できます。

重要:

RHOCPV のサポートは、NetScaler リリース 14.1 ビルド 66.64 以降で利用可能です。一般提供 (GA) 後、RHOCPV イメージを使用して NetScaler VPX™ 仮想マシンを RHOCPV に展開できます。

システム要件

RHOCPV に NetScaler VPX を展開するための CPU とメモリの要件は、VPX モデルと予想されるワークロードによって異なります。

- **CPU:** 2 vCPU 以上
- **メモリ:** 4 GB RAM 以上
- **ストレージ:** 20 GB 以上のディスク容量

注:

NetScaler リリース 14.1-72.x 以降、RHOCPV 上の NetScaler VPX は 2 GB RAM でプロビジョニングできます。

OpenShift Virtualization に NetScaler VPX インスタンスをインストールする

RHOCPV コンソールに NetScaler VPX をインストールするには、この手順に従ってください。

ステップ 1. 環境を準備します。

- RHOCPU クラスターが稼働していることを確認します。
- VMを作成するための管理者アクセス権があることを確認します。
- VMを作成および管理するために、仮想化オペレーターが有効になっていることを確認します。

ステップ2. NetScaler VPXイメージを取得します。

1. **Citrix.com > Downloads > NetScaler > Virtual Appliances** に移動し、**Citrix Downloads** ページからRHOCPUと互換性のあるNetScaler VPXイメージをダウンロードします。
2. ダウンロードした「NSVPX-RHOV-`<version-build>`_lx_64.tgz」ファイルを展開して、qcow2イメージファイルを取得します。

注:

FIPS展開の場合は、代わりにFreeBSDベースのNetScaler VPX FIPSイメージをダウンロードします。このイメージは、`NSVPX-RHOCPU-<release-build>.fips_nc_64.tgz` (例: `NSVPX-RHOCPU-14.1-72.61.fips_nc_64.tgz`) という命名形式を使用します。RHOCPUでのVPX FIPSは、NetScalerリリース14.1-72.x以降でサポートされています。詳細については、[VPX FIPSアプライアンス](#) を参照してください。

ステップ3. イメージをRHOCPUブート可能ボリュームにアップロードします。

1. Red Hat OpenShiftコンソールにログインします。
2. ダッシュボードで、**Virtualization > Bootable Volumes** に移動します。
3. 右上隅にある **Add volume** をクリックし、**With Form** を選択します。
4. NetScaler VPXイメージファイルをアップロードします。



5. ボリュームの名前を入力します。
6. **Save** をクリックします。

ステップ 4. NetScaler VPX VM を作成します。

インスタンスタイプまたはYAML設定のいずれかを使用してVMを作成できます。



オプション 1. インスタンスタイプからVMを作成する:

1. **Virtualization > VirtualMachine** に移動し、**Create** をクリックします。

2. **From InstanceType** を選択します。
3. ステップ 3 で作成したブート可能ボリュームを選択します。



4. VPX モデルのVMハードウェア仕様に一致するインスタンスタイプを選択します。詳細については、[システム要件](#)を参照してください。



5. VM の名前を入力します。
6. **Create VirtualMachine** をクリックします。

オプション 2. YAML を使用してVMを作成する:

1. **Virtualization > VirtualMachine** に移動し、**Create** をクリックします。
2. **With YAML** を選択します。
3. 必要なフィールドを自分の値で編集します。
4. ブートディスクを、**ステップ 3** でインポートしたNetScalerイメージを指すように更新します。

YAMLでVMを作成(/en-us/vpx/media/rhov-create-vm-with-yaml.png)

YAMLファイルの例:

```
apiVersion: kubevirt.io/v1

kind: VirtualMachine

metadata:

  name: <vm-name>

  annotations:

  description: <VM description>

  labels:

  app: default

  os.template.kubevirt.io/rhel10: 'true'
```

```
spec:

runStrategy: Once

template:

metadata:

annotations:

  vm.kubevirt.io/flavor: small

  vm.kubevirt.io/os: rhel10

  vm.kubevirt.io/workload: server

labels:

  kubevirt.io/domain: default

  kubevirt.io/size: small

spec:

  domain:

    cpu:

      cores: <2>

      sockets: 1

      threads: 1

    devices:

      disks:

        - disk:

            bus: virtio

            name: <disk-name>
```

```
interfaces:

  - masquerade: {}

  model: virtio

  name: default

  networkInterfaceMultiqueue: true

  rng: {}

memory:

  guest: <4Gi>

hostname: <vm-name>

networks:

  - name: default

  pod: {}

terminationGracePeriodSeconds: 180

volumes:

  - name: <diskname>

    persistentVolumeClaim:

      claimName: <14.1-66.4-lx-amd64>
```

5. 作成をクリックします。

ステップ 5. NetScaler VPX VMを構成します。

1. 仮想化 > 仮想マシンに移動し、新しく作成したNetScaler VMを選択します。
2. VMが電源オフ状態であることを確認します。
3. 構成をクリックします。ネットワークセクションの構成オプションを確認し、VMインターフェース（管理とデータ）を構成します。

YAMLによるVM構成(/en-us/vpx/media/rhov-vm-configuration.png)

注:

RHOCPV GUIは、ネットワークインターフェースを追加した順序とは異なる順序で表示する場合があります。表示される順序に頼るのではなく、MACアドレスと構成によって各インターフェースを確認してください。

ステップ 6. 管理IPアドレスを構成します。

1. 初回起動時、NetScalerは管理ネットワークにデフォルトのIPアドレスを使用します。管理IPアドレスを構成するには、次の手順に従います。
2. 次のコマンドを使用して、管理IPアドレスとネットマスクを設定します。

```
set ns config -IPAddress <ip> -netmask <mask>
```

3. 以下のコマンドを使用してデフォルトルートを追加します。

```
add route <network> <netmask> <gateway>
```

4. 以下のコマンドを使用して設定を保存します。

```
save config
```

5. 変更を有効にするには、VMを再起動します。
6. VPXインスタンスが起動したら、NetScalerコンソールまたはNSIPアドレスを使用して管理インターフェイスにアクセスします。

ステップ7. 設定の最終調整。

ライセンスを適用し、ネットワーク設定を構成した後、展開の必要に応じて、ロードバランシング、SSLオフロード、アプリケーションファイアウォールなどのNetScaler機能をセットアップします。

注:

RHOCPVの展開は、ベアメタルSNOクラスターおよび汎用Uシリーズインスタンスでテストおよび認定されています。

OpenShift VirtualizationでNetScaler VPXをアップグレードする

ステップ1: RHOCPV固有のNetScalerアップグレードバンドルをダウンロードします。

1. **Citrix.com > Downloads > NetScaler > Virtual Appliances** に移動し、[Citrix Downloads](#) ページからRHOCPVと互換性のあるNetScaler VPXアップグレードバンドルをダウンロードします。

注:

RHOCPVでNetScaler VPXをアップグレードする際は、アップグレードが失敗するため、標準のNetScalerアップグレードバンドルを使用しないでください。代わりに、互換性の問題を避けるために、「build-<version-build>_lx_64.tgz」という名前のRHOCPV固有のアップグレードバンドルを必ず使用してください。

2. ダウンロードした「build-<version-build>_lx_64.tgz」ファイルをNetScaler VPX Shellの `/var/nsinstall/` ディレクトリにコピーして展開します。
3. `./installns` を実行し、再起動してNetScaler VPXをアップグレードします。

サポートされていない機能と制限

RHOCPV上のNetScaler VPX展開でサポートされていないNetScalerの機能、プロトコル、およびシステム機能のリストについては、「Red Hat OpenShift VirtualizationおよびNutanix AHVへのNetScaler VPX展開の制限事項」([/ja-jp/vpx/current-release/rhocpv-and-nutanix-limitations.html](#))を参照してください。

注:

RHOCPV上のNetScaler VPX展開の場合、インスタンスで `clear config` コマンドを実行した後、TCPプロファイルのMSS値を `1360` に設定します。より高いMSS値を設定すると、TCPパケットがネットワークレベルでドロップされ、Secure Copy Protocol (SCP) および Secure Shell (SSH) トラフィックの障害が発生します。

Nutanix Acropolisハイパーバイザーに NetScaler VPXをインストールする

NetScalerは、Nutanix Acropolis Hypervisor (AHV) 用の専用VPXイメージを提供しており、展開と管理のエクスペリエンスを大幅に向上させます。以前は、お客様は[Citrix Ready partnership](#)プログラムの下で汎用KVMハイパーバイザーイメージを使用していました。新しいイメージにより、NetScaler VPXのプロビジョニング方法が簡素化されます。標準のAHV Prism Centralを使用してインスタンスを展開できるようになりました。お客様は公式サポートを受けられ、必要なすべてのリソースにアクセスできるため、Nutanixインフラストラクチャへのシームレスな展開が可能になります。

重要:

Nutanix AHVのサポートは、NetScalerリリース14.1ビルド51.x以降で利用可能です。この専用Nutanix VPXイメージの一般提供 (GA) 開始に伴い、Citrix Ready KVM VPXイメージを使用した展開はNutanix AHVではサポートされなくなりました。お客様は、既存のCitrix Ready KVM VPX仮想マシンを、新しく公式にサポートされているNutanix VPXイメージに移行することをお勧めします。

システム要件

Nutanix AHVにNetScaler VPXを展開するためのCPUとメモリの要件は、VPXモデルと予想されるワークロードによって異なります。

- **CPU:** 2 vCPU以上。
- **メモリ:** 4 GB RAM以上。
- **ストレージ:** 20 GB以上のディスク容量。

注:

NetScalerリリース14.1-72.x以降、Nutanix AHV上のNetScaler VPXは2 GB RAMでプロビジョニングできます。

Prismを使用してNutanix AHVにNetScaler VPXインスタンスをインストールする

Nutanix AHVにNetScaler VPXをインストールするには、次の手順に従ってください。

ステップ1. 環境を準備します。

- Nutanix AHVクラスターが稼働していることを確認します。
- Nutanixの管理インターフェースであるPrism Centralへのアクセスがあることを確認します。
- Nutanix環境内でVMを作成および管理するために必要な権限があることを確認します。

ステップ2. NetScaler VPXイメージを取得します。

1. **Citrix.com > Downloads > NetScaler > Virtual Appliances** に移動し、[Citrix Downloads](#) ページからNutanix AHVと互換性のあるNetScaler VPXイメージをダウンロードします。
2. ダウンロードした「NSVPX-NUTANIX-`<version-build>`_lx_64.tgz」ファイルを展開して、qcow2イメージファイルを取得します。

ステップ3: イメージをNutanix AHVにアップロードします。

1. Nutanix Prism Webコンソールにログインします。
2. Prism Centralダッシュボードで、**Compute > Image** セクションに移動し、**Add image** をクリックします。
3. NetScaler VPXイメージファイルをアップロードし、正しいイメージタイプ（DISKイメージ）とフォーマット（qcow2）を選択していることを確認します。



4. **Save** をクリックします。

ステップ4. NetScaler VPX VMを作成します。

1. Prism Centralで、**Compute > VM** セクションに移動し、**Create VM** をクリックします。
2. 展開するVPXモデルのNetScaler®推奨事項に従って、VMハードウェア仕様を構成します。詳細については、[システム要件](#) を参照してください。



3. 「**新しいディスクの追加**」をクリックし、以下の詳細を入力します。

- **タイプ** - ディスク。
- **操作** - イメージサービスからクローン。

- **イメージ** - 前のステップでアップロードしたVPXイメージをVMのディスクとして割り当てます。



4. 「**新しいNICの追加**」をクリックし、必要な詳細を入力します。



5. 設定を保存します。

ステップ5. NetScaler VPX VMを起動します。

1. 新しく作成したVPX VMの電源を入れます。
2. Prismが提供するコンソールを通じて起動プロセスを監視し、NetScaler VPXインスタンスが正しく起動することを確認します。

ステップ6. 初期設定。

1. 起動中に、NetScalerはメニュー駆動型インターフェースを表示します。IPアドレス、サブネットマスク、ゲートウェイアドレスを設定して初期設定を行い、「**保存して続行**」オプションを選択します。
2. VPXインスタンスが起動したら、NetScalerコンソールまたはNSIPを通じてその管理インターフェースにアクセスします。



ステップ7: 設定を完了する。

ライセンスを適用し、ネットワーク設定を構成した後、デプロイメントの必要に応じて、ロードバランシング、SSLオフロード、アプリケーションファイアウォールなどのNetScaler機能をセットアップします。

Nutanix AHV 上の NetScaler VPX™ をアップグレードする

ステップ 1: Nutanix 固有の NetScaler アップグレードバンドルをダウンロードする。

1. **Citrix.com > Downloads > NetScaler > Virtual Appliances** に移動し、[Citrix Downloads](#) ページから Nutanix AHV と互換性のある NetScaler VPX アップグレードバンドルをダウンロードします。

注:

Nutanix AHV 上の NetScaler VPX をアップグレードする際は、標準の NetScaler アップグレードバンドルを使用しないでください。アップグレードが失敗します。代わりに、互換性の問題を避けるために、AHV 上の VPX 用に設計された Nutanix 固有のアップグレードバンドルを使用してください。

1. ダウンロードした「build-`<version-build>`_lx_64.tgz」ファイルを NetScaler VPX Shell の `/var/nsinstall/` ディレクトリにコピーして展開します。
2. `./installns` を実行し、再起動して NetScaler VPX をアップグレードします。

注:

Nutanix 固有の NetScaler アップグレードバンドルを使用した「Citrix Ready®」NetScaler VPX からのアップグレードはサポートされていません。

サポートされていない機能と制限事項

Nutanix AHV 上の NetScaler VPX デプロイメントでは、特定の機能、プロトコル、およびシステム機能はサポートされていません。完全なリストについては、「[Red Hat OpenShift Virtualization および Nutanix AHV 上の NetScaler VPX デプロイメントの制限事項](#)」を参照してください。

Citrix Ready KVM VPX から Nutanix AHV 上で公式にサポートされている NetScaler VPX™ へ移行する

Nutanix AHV 上の Citrix Ready® KVM ベースの NetScaler VPX は多くのお客様にご利用いただいておりますが、公式の Citrix® サポートや高度な自動化機能はありません。そのため、お客様には Citrix Ready KVM VPX から Nutanix AHV 上で公式にサポートされている NetScaler VPX への移行をお勧めします。

前提条件

移行する前に、Nutanix AHV 固有の VPX イメージの制限事項を確認し、お使いの環境が除外された機能に依存していないことを確認してください。詳細については、「[サポートされていない機能と制限事項](#)」を参照してください。

Citrix Ready KVM VPX から Nutanix AHV 上で公式にサポートされている NetScaler VPX へは、以下のいずれかの方法で移行できます。

- Citrix Ready KVM VPX から Nutanix AHV 上で公式にサポートされている NetScaler VPX へのインプレースアップグレード
- ノードバイノードの高可用性 (HA) 置換を使用した移行

Citrix Ready KVM VPX から Nutanix AHV 上で公式にサポートされている NetScaler VPX へのインプレースアップグレード

NetScaler リリース 14.1 ビルド 66.x 以降では、Citrix Ready KVM ベースの VPX インスタンスから Nutanix AHV 上で公式にサポートされている NetScaler VPX へのインプレースアップグレードを実行できます。

重要:

バージョン 14.1-66.x 以降にアップグレードする Citrix Ready VPX インスタンスは、公式にサポートされている Nutanix 固有のアップグレードバンドルを使用する必要があります。標準のアップグレードバンドルは、この移行ではサポートされていません。

前提条件

- **最小 NetScaler バージョン:** 現在の NetScaler バージョンが 14.1–56.74 以降であることを確認してください。現在のバージョンがそれより古い場合は、続行する前に標準のアップグレードバンドルを使用してアプライアンスをアップグレードしてください。



- **ハードウェア要件:** 各 vCPU につき最低 2 GB の RAM が必要です。たとえば、4 vCPU の VM には、最低 8 GB の RAM が割り当てられている必要があります。
- **制限事項の確認:** 続行する前に、[Nutanix AHV固有のVPX制限](#)を確認してください。

アップグレードを進める前の重要な考慮事項

- **VMスナップショット (必須):** アップグレードを進める前に、実行中のVMのスナップショットを作成する必要があります。これにより、設定が失われた場合にVMを復元できます。詳細については、[VMスナップショットの手動作成](#)を参照してください。
- **外部バックアップ (推奨):** 重要な構成とカスタムデータ、特に保存対象外のディレクトリにあるファイルについては、外部バックアップを作成することをお勧めします。VMスナップショットがすでに取得されている場合、この手順はオプションです。
- **データのバックアップと保存:** [アップグレード中のデータのバックアップと保存セクション](#)を確認してください。

アップグレード中のデータのバックアップと保存

- アップグレードでは、特定のディレクトリ内のカスタムファイル、証明書、署名ファイルが自動的に保持されます。これらのパス外にあるデータは保持されず、手動でのバックアップと復元が必要です。

保持されるディレクトリ:

- `/flash/nsconfig`
 - `/flash/boot`
 - `/flash/configdb`
 - `/var/tmp/*.csv`
 - `/var/AAA`
 - `/var/analytics_conf`
 - `/var/download`
 - `/var/krb`
 - `/var/learnt_data`
 - `/var/metrics_conf`
 - `/var/netScaler/gui`
 - `/var/netScaler/locdb`
 - `/var/netScaler/logon`
 - `/var/netScaler/ssl`
 - `/var/nstemplates`
 - `/var/vpn`
 - `/var/vpns`
 - `/var/clusterd/nsclusterd.conf`
- `/etc` ディレクトリにカスタマイズされた構成ファイルがある場合は、移行を進める前に、[「/etcディレクトリ内のカスタマイズされた構成ファイルのアップグレードに関する考慮事項」](#)を参照してください。
 - 以下のファイルに対して行われたカスタマイズは、移行プロセス中に保持されますが、移行前と同じように機能しない可能性があります。
 - `rc.netScaler`

- `nsbefore.sh`
- `nsafter.sh`
- `crontab`

アップグレードされたNetScalerとの互換性を確保するために、これらのカスタマイズファイルをレビューし、更新する必要があります。

アップグレード手順

1. Nutanix AHVと互換性のあるNetScaler VPXアップグレードバンドル（形式: `build-<version-build>_lx_64.tgz`）をCitrixダウンロードページからダウンロードします。
2. **Citrix.com > Downloads > NetScaler > Virtual Appliances** に移動し、[Citrix Downloads](#) ページからNutanix AHVと互換性のあるNetScaler VPXアップグレードバンドルをダウンロードします。
3. ダウンロードした `build-<version-build>_lx_64.tgz` ファイルをNetScaler VPX Shellの `/var/nsinstall/` ディレクトリにコピーし、解凍します。
4. `./installns` を実行し、プロンプトが表示されたらアプライアンスを再起動します。このプロセスに必要な複数回の再起動中にVMの電源を切らないでください。

アップグレード後の検証

1. NetScaler VPX GUI/CLIにログインし、実行中のバージョンがターゲットのアップグレードバンドルバージョンと一致することを確認します。
2. すべての仮想サービス、SSL証明書、およびポリシーを検証し、正しく機能することを確認します。

ノードバイノードの高可用性 (HA) 置換方法を使用した移行

HAノードバイノード置換方法により、Citrix Ready KVM VPXから公式にサポートされているNutanix AHV VPXイメージへのダウンタイムなしの完全サポートされたアップグレードが可能になります。この移行により、NetScalerの高度な自動化機能、完全な公式サポート、および最新化された展開プラットフォームが利用可能になります。

ステップ 1: レガシープライマリノードのHA状態をロックします。

1. Citrix Ready KVM VPXプライマリノードにログインし、次のコマンドを実行します。

```
set ha node -hastatus STAYPRIMARY
```

2. レガシーセカンダリノードで、次のコマンドを使用してプライマリノードとのHAペアリングを解除します。

```
rm ha node <primary-nsip>
```

3. 移行または交換タスクを進めている間も、プライマリノードが中断なくトラフィックを処理し続けていることを確認します。

ステップ2: 新しいセカンダリノードをデプロイします。

1. 公式のNutanix AHV 14.1-51.xイメージを使用して、新しいNetScaler VPXインスタンスをプロビジョニングします。
2. 以前レガシーセカンダリノードで使用されていた管理IPアドレスを割り当てます。
3. 初期設定を完了し、ライセンスを適用し、ネットワーク構成が正しいことを確認します。
4. 次のコマンドを実行してHAノードを結合します。

```
add ha node <primary-nsip>
```

5. 次のコマンドを使用して、HAステータスが**UP**であり、同期が成功していることを確認します。

```
show ha node status
```

ステップ3: 新しいセカンダリノードをプライマリに昇格させます。

1. 古いプライマリノード（レガシーKVM VPX）で、次のコマンドを使用してプライマリ維持ステータスを解除します。

```
unset ha node -stayprimary
```

2. 次のコマンドを使用してフェイルオーバーを強制します。

```
force ha failover
```

3. 新しくデプロイされたセカンダリノードが正常にプライマリノードになり、完全に動作していることを確認します。

```
show ha node status
```

4. 次のコマンドを使用して、新しいプライマリノードをプライマリ維持に設定します。

```
set ha node -hastatus STAYPRIMARY
```

注:

HA同期が失敗した場合は、`set rpcNode <peer-ip> -password <password>` コマンドを使用して両方のノードでRPCパスワードを手動で同期します。

5. このステップで問題が発生した場合は、古いプライマリにフェイルバックし、レガシーセカンダリノードを復元してください。サポートが必要な場合は、Citrixサポートチームにお問い合わせください。

ステップ4: レガシーセカンダリを新しいVPXに置き換えます。

1. 次のコマンドを使用して、レガシーセカンダリノードをHA構成から削除します。

```
rm ha node <legacy-secondary-nsip>
```

2. 公式のNutanix AHV 14.1-51.xイメージを使用して、別の新しいVPXインスタンスをプロビジョニングします。
3. レガシーセカンダリノードの管理IPアドレスを新しいVPXインスタンスに割り当てます。
4. セットアップとライセンスのアクティベーションを完了します。
5. 次のコマンドを使用して、この新しいVPXをHAペアに参加させます。

```
add ha node <new-primary-nsip>
```

6. 次のコマンドを使用して、HAステータスが安定しており、「UP」で同期されていることを確認します。

```
show ha node status
```

ステップ5: 検証と最終処理。

1. 両方の新しいノードでHAロールと同期ステータスを確認します。

2. すべての仮想サービス、SSL証明書、およびポリシーを検証し、正しい機能を確認します。
3. フェイルオーバーおよびフェイルバックテストを実行して、冗長性を確認します。
4. すべての構成をバックアップし、新しい環境のスナップショットを作成します。
5. 検証が成功した後、リソースを解放するためにレガシーVPXインスタンスを削除します。

重要な考慮事項

- **バックアップ:** 新しいVPX環境が完全に検証され、安定するまで、レガシーノードのバックアップを保持することが不可欠です。
- **サポートされていない機能:** 現在のデプロイが新しいNutanix AHV VPXイメージでサポートされていない機能に依存している場合、移行は推奨されません。詳細については、[サポートされていない機能と制限事項](#)を参照してください。

NetScaler VPX™ on Nutanix Acropolis Hypervisor - よくある質問

- **Nutanix AHV 上の NetScaler VPX に必要な最小メモリは何ですか？**

Nutanix AHV 上の NetScaler VPX には、最低 4 GB のメモリが必要です。

注:

NetScaler リリース 14.1-72.x 以降、Nutanix AHV 上の NetScaler VPX は 2 GB RAM でプロビジョニングできます。

- **Nutanix AHV 上の VPX は 20 GB を超えるディスク拡張をサポートしていますか？**

はい。Nutanix AHV に展開された NetScaler® VPX のディスク容量を増やすには、初期プロビジョニング段階で必要な追加ディスク容量が割り当てられていることを確認してください。Nutanix AHV は、VM のプロビジョニング後にディスクサイズを増やすことをサポートしていません。

- **Nutanix AHV でセカンダリディスクはサポートされていますか？**

はい、セカンダリディスクは NetScaler リリース 14.1-72.x 以降でサポートされています。

- **Nutanix AHV で NetScaler 変数用の sysctl はサポートされていますか？**

いいえ。同じデータを取得するには、特定の NSCLI コマンドを使用することをお勧めします。 `sysctl netscaler.version` を使用する代わりに、NetScaler CLI コマンド `show version` を使用してください。

- **Nutanix AHV 上の NetScaler のホスト名を変更するにはどうすればよいですか？**

`set hostname <hostName>` コマンドを使用してホスト名を設定できます。

- **VPX ニュータニックス環境において、再起動およびアップグレード/再インストール後も保持されるディスクパーティションはどれですか？**

再起動後も保持されるのは、以下のディレクトリ内のファイルのみです。

- `/flash/nsconfig` ディレクトリを指す `/nsconfig`
- `/var` ディレクトリ

その他のすべてのファイルは、再起動時に削除されます。

- **VPX Nutanix で誤って削除されたシステムファイル (例: `/usr/sbin` 内のバイナリファイル) を復元するにはどうすればよいですか？**

NetScaler を再起動し、起動中に「NetScaler ファームウェアバージョン 14.1 x.x を再インストール」を選択します。この操作は、保存されている設定には影響しません。

- **Nutanix AHV 上の VPX でカスタムログローテーションを設定するにはどうすればよいですか？**

Nutanix AHV では、`newsyslog.conf` はサポートされていません。代わりに、Linux の `logrotate` ユーティリティを使用してログローテーションが処理されます。これは1時間ごとに実行されます。設定ファイルは `/etc/logrotate.d/blx-syslog` にあります。

- **Nutanix AHV で cloud-init はサポートされていますか？**

はい、ただし現在サポートされているのはカスタムスクリプトのみです。ガイド付きスクリプトはまだサポートされていません。

cloud-init スクリプトの例:

- **例 1:** 空の `cloud-init` スクリプト。これにより DHCP がトリガーされ、最初のインターフェースに静的または DHCP IP アドレスが設定されている場合、それは自動的に NetScaler IP アドレスとして設定されます。

```
#cloud-config
```


- **例 2:** ホスト名を「vpx-nutanix」に設定します

```
#cloud-config  
  
fqdn: vpx-nutanix
```

- **VPX Nutanix で、最初のインターフェースに設定されているプライマリの静的/DHCP IP から NSIP を自動割り当てできますか？**

はい、ただしインスタンスの作成時に、カスタム設定でダミーの `cloud-init` 設定を指定する必要があります。

- **Nutanix AHV 上の NetScaler のドメインネームシステム (DNS) を設定するにはどうすればよいですか？**

`add dns nameserver` コマンドを使用してDNSを設定します。

- **VPX Nutanix AHVでDNSの問題をデバッグするにはどうすればよいですか？**

NetScaler for NutanixのNetScalerリリース14.1-72.x以降では、dig、nslookup、hostなどのDNSユーティリティがサポートされています。それ以前のリリースでは、`systemd-resolve` コマンドを使用してDNS解決を確認してください。

例:

```
root@ns# systemd-resolve www.google.com  
  
www.google.com: 142.251.10.105          -- link: ns2  
  
                142.251.10.147          -- link: ns2  
  
                142.251.10.104          -- link: ns2  
  
                142.251.10.103          -- link: ns2  
  
                142.251.10.99           -- link: ns2  
  
                142.251.10.106          -- link: ns2  
  
                2404:6800:4003:c0f::63  -- link: ns2
```

- **Nutanix AHV上のVPXで利用できるテキストエディタは何ですか？**

Vimがテキストエディタとして利用可能です。「vi」 コマンドはvimバイナリへのシンボリックリンクです。

- **SSHセッションを介してvimで開いたファイルから行をクリップボードにコピーするにはどうすればよいですか？**

Shiftキーを押しながら左クリックし、ドラッグしてコピーしたいテキストを選択します。

- **vimエディタにコンテンツを貼り付けるにはどうすればよいですか？**

「i」を使用して挿入モードに入り、「Shift + Insert」キーの組み合わせを使用して貼り付けます。

- **NetScalerで高可用性 (HA) を使用する場合、オーバーレイサブネット内のNutanix AHVインターフェースにIPアドレスを設定する推奨方法はどれですか？**

Nutanixオーバーレイサブネットを使用する場合、プライマリおよびセカンダリIPアドレスの両方をネットワークインターフェースに割り当てることができます。ただし、これらのIPは特定のノードのインターフェースに紐付けられており、仮想IPアドレス (VIP) またはサブネットIPアドレス (SNIP) として使用された場合、HAフェイルオーバーをサポートしません。

NetScalerで適切なHA動作を確保するには、以下のベストプラクティスに従ってください。

- Nutanixオーバーレイサブネット内の各インターフェースには、1つの (必須の) IPアドレスのみを割り当ててください。
- このIPアドレスを、HAノード間で共有される設定 (例: VIPまたはSNIP) に使用しないでください。
- このIPアドレスは、NSIPがノード固有であり、フェイルオーバーを必要としないため、NetScaler IP (NSIP) として使用できます。
- VIPとSNIPは、Nutanixインターフェースではなく、NetScalerアプライアンスで手動で構成してください。
- 正しいルーティングと通信のために、NetScalerで適切なVLAN設定を構成してください。

このアプローチを使用すると、HAの問題を回避し、ノード固有のIP構成と共有IP構成の間の明確な分離を確保できます。

- **Nutanix AHV上のNetScaler VPXでタグ付きVLANはサポートされていますか？**

はい、AHVハイパーバイザーが[ゲストVM用VLAN](#)に記載されている構成を持っている場合、NetScaler VPXはタグ付きVLANをサポートします。

- **`nsapimgr` コマンドはNutanix AHVでサポートされていますか？**

`nsapimgr` コマンドはNutanix AHVでサポートされています。ただし、呼び出しオプションは以下に限定されます。

コマンド 1	コマンド 2
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_weak_sso_type_enable</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_weak_sso_type_disable</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_sso_ntlm_use_ntdomain</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_sso_ntlm_dont_use_ntdomain</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_sso_kcd_use_spn_host</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_sso_kcd_dont_use_spn_host</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_redirect_for_artifact</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_post_for_artifact</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_enable_single_logout</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_disable_single_logout</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_enable_replay_protection</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_disable_replay_protection</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_enable_incl_ns_fix</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_disable_incl_ns_fix</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_enable_comma_sep_attr_res_no_encode</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_disable_comma_sep_attr_res_no_encode</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_enable_comma_sep_attr_res</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_disable_comma_sep_attr_res</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_dont_send_subject</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_send_subject</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_dont_send_session_index</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_send_session_index</code>
<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_dont_send_in_response</code>	<code>nsapimgr_wr.sh -ys</code> <code>call=ns_saml_send_in_response</code>

コマンド 1	コマンド 2
<pre>nsapimgr_wr.sh -ys call=ns_saml_dont_send_full_metadata</pre>	<pre>nsapimgr_wr.sh -ys call=ns_saml_send_full_metadata</pre>
<pre>nsapimgr_wr.sh -ys call=ns_rba_login_enable_debug</pre>	<pre>nsapimgr_wr.sh -ys call=ns_rba_login_disable_debug</pre>
<pre>nsapimgr_wr.sh -ys call=ns_enable_schema_checks</pre>	<pre>nsapimgr_wr.sh -ys call=ns_disable_schema_checks</pre>
<pre>nsapimgr_wr.sh -ys call=ns_aaatm_drop_options_request</pre>	<pre>nsapimgr_wr.sh -ys call=ns_aaatm_dont_drop_options_request</pre>
<pre>nsapimgr_wr.sh -ys call=ns_aaatm_302_for_bearer</pre>	<pre>nsapimgr_wr.sh -ys call=ns_aaatm_401_for_bearer</pre>
<pre>nsapimgr_wr.sh -ys call=ns_aaa_enable_api_access</pre>	<pre>nsapimgr_wr.sh -ys call=ns_aaa_disable_api_access</pre>
<pre>nsapimgr_wr.sh -ys call=enable_webview_support</pre>	<pre>nsapimgr_wr.sh -ys call=disable_webview_support</pre>
<pre>nsapimgr_wr.sh -ys call=enable_rfweb_webview_support</pre>	<pre>nsapimgr_wr.sh -ys call=disable_rfweb_webview_support</pre>
<pre>nsapimgr_wr.sh -ys call=aaatm_disable_domain_cookie</pre>	<pre>nsapimgr_wr.sh -ys call=aaatm_enable_domain_cookie</pre>
<pre>nsapimgr_wr.sh -ys call=aaa_strip_domain</pre>	<pre>nsapimgr_wr.sh -ys call=aaa_dont_strip_domain</pre>
<pre>nsapimgr_wr.sh -ys arg1=<VAL> -ys call="ns_saml_set_deflate_size"</pre>	<pre>nsapimgr_wr.sh -ys call=ns_saml_unset_deflate_size</pre>
<pre>nsapimgr_wr.sh -ys arg1=<VAL> -ys call="ns_saml_send_binding"</pre>	<pre>nsapimgr_wr.sh -ys call=ns_saml_dont_send_binding</pre>
<pre>nsapimgr_wr.sh -ys arg1=<VAL> -ys call="ns_push_set_timeout"</pre>	<pre>nsapimgr_wr.sh -ys arg1=<VAL> -ys call="ns_aaatm_cgitm_duration"</pre>

Red Hat OpenShift VirtualizationおよびNutanix AHV上でのNetScaler VPX展開に関する制限事項

このドキュメントは、Red Hat® OpenShift® Virtualization (RHOCPV) および Nutanix AHV プラットフォームに NetScaler VPX™ を展開する際に現在サポートされていない機能と能力の概要を説明します。

サポートされていないNetScalerの機能:

- 管理パーティション(/ja-jp/citrix-adc/current-release/admin-partition)
- [NextGen API](#)
- フロントエンド最適化 (FEO)(/ja-jp/citrix-adc/current-release/optimization/front-end-optimization.html)
- ジャンボフレーム(/ja-jp/citrix-adc/current-release/networking/interfaces/jumbo-frames/configuring-jumbo-frames-support-on-citrix-adc-appliance.html)
- 実行中の構成の自動保存
- レイヤー3 (L3) クラスター
- 電子メールワンタイムパスワード (OTP)
- テックサポートバンドル収集パスの変更(/ja-jp/citrix-adc/current-release/system/basic-operations#change-in-the-default-support-bundle-collection-directory-for-netscaler.html)
- IPsecプロトコル

サポートされていないコンソール機能:

- ライセンス管理ページ上のテレメトリステータス
- ファイル整合性監視(/ja-jp/netscaler-console-service/instance-advisory/file-integrity-monitoring.html)

システムの制限事項:

- プリブートユーザーデータ
- [Nutanixゲストツール](#)

- `what` ユーティリティ
- `nsapimgr` 呼び出しオプションは部分的にサポートされています。Nutanix AHVでの `nsapimgr` のサポートおよび許可されている呼び出しオプションについては、関連する[FAQ](#)を参照してください。

非推奨の機能:

- ルーティング情報プロトコル (RIP)
- 次世代ルーティング情報プロトコル (RIPng)
- 中間システム間 (IS-IS) プロトコル
- PTPプロトコル
- ファイル転送プロトコル (FTP)
- テルネット

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アプリケーションは、AWS MarketplaceでAmazon Machine Image (AMI) として利用できます。AWS上のNetScaler VPXインスタンスを使用すると、AWSのクラウドコンピューティング機能を利用し、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アプリケーション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に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\(/ja-jp/vpx/current-release/deploy-aws/vpx-ha-pip-different-aws-zones.html\)](#)
- [アクティブ-アクティブ GSLB\(/ja-jp/citrix-adc/current-release/global-server-load-balancing/deployment-types/active-active-site-deployment.html\)](#)

ハイブリッド展開

- [AWS OutpostへのNetScalerの展開\(/ja-jp/vpx/current-release/deploy-aws/deploy-vpx-on-aws-outposts.html\)](#)
- [AWSのVMCへのNetScalerの展開\(/ja-jp/vpx/current-release/install-vpx-on-vmc-on-aws.html\)](#)

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](#)
- [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を通じ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インテグレーションで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を通じてパブリッククラウドNetScalerの展開を簡素化](#)
- [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の利点を活用する](#)

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展開する](#)
- [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インスタンスのトラブルシューティング](#)を参照してください。サポートケースを申請するには、AWSアカウント番号とサポートPINコードを確認し、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 マーケットプレースのNetScaler](#)
- [NetScalerはAWSネットワーキングパートナーソリューション（ロードバランサー）の一部です](#)

- [AWS FAQ](#)

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用語集](#)を参照してください。

用語	定義
Amazonマシンイメージ (AMI)	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クラウド内の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)	クラウド内で安全で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、受信する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・インターフェース (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。

用語	定義
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。
アイデンティティおよびアクセス管理 (IAM)	AWS での ID の実行を許可または禁止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単一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	インスタンスに対して許可される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。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グループ化するためにサブネットを作成できます。

用語	定義
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内で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。
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、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。
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。

AWS-VPX サポートマトリックス

以下のセクションでは、サポートされているNetScaler VPX製品、AWSリージョン、インスタンスタイプ、およびサービスを一覧表示します。

AWS上でサポートされているVPX製品

サポートされているVPX製品
NetScaler VPX - 顧客ライセンス
NetScaler VPX FIPS - 顧客ライセンス
NetScaler VPX FIPS ENA - 顧客ライセンス

サポートされているAWSリージョン

リージョン	リージョンコード
米国西部	米国西部-1、米国西部-2
米国東部	米国東部-1、米国東部-2
アジア太平洋	ap-south-1, ap-southeast-4, ap-northeast-2, ap-southeast-1, ap-southeast-2, ap-

リージョン	リージョンコード
	northeast-1, ap-east-1, ap-northeast-3, ap-southeast-3, ap-south-2
カナダ	ca-central-1, ca-west-1
EU	eu-central-1, eu-west-1, eu-west-2, eu-west-3, eu-south-1, eu-north-1
南米	sa-east-1
イスラエル	イルセントラル-1
中東	ミーサウス-1, ミーセントラル-1
アフリカ	アフリカサウス-1
AWS GovCloud	USガブイースト-1、USガブウェスト-1
米国政府機関	us-iso-east-1、us-isob-east-1
中国	cn-north-1、cn-northwest-1

サポートされているAWSインスタンスタイプ

NetScalerは以下のAWSインスタンスタイプを推奨します。

- マーケットプレイスエディションまたは帯域幅ベースのプールライセンスには、M6iおよびC5nシリーズ。

プールライセンスまたはフレックスライセンスを使用するVPX（帯域幅ライセンス）	推奨されるAWSインスタンス
最大200 Mbps	m6i.xlarge
1～5 ギガビーピーエス	m6i.2xlarge
5～8 ギガビーピーエス	c5n.4xlarge
8～25 ギガビット/秒	c5n.9xlarge

1秒あたりのパケット数や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の担当者にご相談ください。

カテゴリ	インスタンスファミリー
汎用	M3、M4、M5、M5a、M5n、M6i、M7i、M8i、T2、T3a
コンピューティング最適化	C4、C5、C5n、C6in
メモリ最適化	R7iz
ストレージ最適化	D2

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NetScaler VPXインスタンスをプロビジョニングする際に、NetScaler VPX NITRO APIを呼び出します。
VPC and VPC ingress routing: 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ソリューションにおいて、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ソリューションにおいて、すべてのNetScaler VPXノードにトラフィックを分散します。
Cloudwatch: NetScaler VPXインスタンスのパフォーマンスとシステムパラメータを監視します。
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。

サポートされているAWSサービス

Identity and Access Management (IAM): AWSサービスおよび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。

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は、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、[AWSにNetScaler VPXインスタンスを展開する](#)のAWS用語セクションをお読みください。
- 高可用性セットアップを効果的に機能させるには、管理インターフェイスに専用の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インスタンス](#)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[シナリオ2：パブリックサブネットとプライベートサブネットを持つVPC](#)を参照してください。
- VPXインスタンスは、あるEC2インスタンスタイプから別のタイプに移動できます（例：m3.largeからm3.xlargeへ）。
- AWS上のVPXのストレージオプションについては、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で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。
- 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「インスタンスタイプごとのネットワークインターフェイスあたりのIPアドレス」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。

- NetScaler VPXインターフェイスでenableおよびdisableインターフェイスコマンドを使用しないことをお勧めします。
- VPXではIPv6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が機能するためには、**送信元/送信先チェック**が無効になっていることを確認してください。詳細については、[Elastic Network Interfaces](#)の「送信元/送信先チェックの変更」を参照してください。
- AWS上の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インスタンス上の非管理インターフェイスを介して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イ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イスを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、**モニタリング**タブの下に表示されるVPXインスタンスのvCPU使用率が、実際の使用率が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。
- ホットアドは、AWS上のNetScalerのPVおよびSRIOVインターフェイスでのみサポートされています。ENAインターフェイスを持つVPXインスタンスはホットプラグをサポートしていません。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インターフェイスを介したホットリムーブは、NetScalerのPV、SRIOV、およびENAインターフェイ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。

前提条件

AWSでVPXインスタンスを作成する前に、以下のもの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、[Creating IAM Users \(Console\)](#)を参照してください。IAMロールは、スタンドアロンデプロイと高可用性デプロイの両方で必須です。

AWSアカウントに関連付けられたIAMロールには、さまざまなシナリオに対応するために以下の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アドレスと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:

- 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 ロールに必要な特権を構成するためのプロンプトが表示されます。特権をすでに構成している場合は、プロンプトを無視してください。

- **AWS CLI:** ターミナルプログラムから AWS マネジメントコンソールが提供するすべての機能を使用するため。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イスタイプを SR-IOV に変更するためにも AWS CLI が必要です。
- **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 以降である必要があります。
- NetScaler VPX の 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、両方のメソッドを使用するように構成することも（デフォルトオプション）、IMDSv2 モードのみを使用するように構成することもできます（IMDSv1 を無効にすることで）。Citrix ADC VPX は、NetScaler VPX リリース 13.1.48.x 以降、IMDSv2 のみ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ときに各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、使用する必要があります。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アクセス許可を提供します。Amazon EC2インスタンスを起動する際、その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応するために、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"
```

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"
```

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"
```

AWSバックエンドの自動スケーリング:

```
"ec2:DescribeInstances",  
  
"autoscaling:*",
```

```
"sns:CreateTopic",  
  
"sns:DeleteTopic",  
  
"sns:ListTopics",  
  
"sns:Subscribe",  
  
"sqs:CreateQueue",  
  
"sqs:ListQueues",  
  
"sqs:DeleteMessage",  
  
"sqs:GetQueueAttributes",  
  
"sqs:SetQueueAttributes",  
  
"iam:SimulatePrincipalPolicy",  
  
"iam:GetRole"
```

注意点:

- 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。
- スタンドアロンと高可用性デプロイメントの両方で、IAM ロールは必須です。

IAM ロールの作成

この手順では、AWS バックエンドのオート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。

注記:

他の機能に対応する IAM ロールを作成するには、同じ手順に従ってください。

1. EC2 用の AWS マネジメントコンソールにログインします。
2. EC2 インスタンスページに移動し、ADC インスタンスを選択します。

3. **Actions > Security > Modify IAM role** の順に移動します。

4. **Modify IAM role** 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. **Modify IAM role** ページで、**Create new IAM role** をクリックします。

 - b. **Roles** ページで、**Create role** 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. 「権限の追加」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、すべてを削除し、使用したい機能のIAM権限を貼り付けます。

たとえば、AWSバックエンドオートスケーリング機能の次のIAM権限を貼り付けます。

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
{  
  
  "Sid": "VisualEditor0",  
  
  "Effect": "Allow",  
  
  "Action": [  
  
    "ec2:DescribeInstances",  
  
    "autoscaling:*",  
  
    "sns:CreateTopic",  
  
    "sns>DeleteTopic",  
  
    "sns:ListTopics",  
  
    "sns:Subscribe",  
  
    "sqs:CreateQueue",  
  
    "sqs:ListQueues",  
  
    "sqs>DeleteMessage",  
  
    "sqs:GetQueueAttributes",  
  
    "sqs:SetQueueAttributes",  
  
    "iam:SimulatePrincipalPolicy",  
  
    "iam:GetRole"  
  
  ],  
  
  "Resource": "*"   
  
}  
  
]  
  
}
```

提供する「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. 「Identity Access Management」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。

IDおよびアクセス管理 (/en-us/vpx/media/iam-verification.png)

- j. 名前、レビュー、作成ページで、ロールに有効な名前を付けます。

IAM情報の確認 (/en-us/vpx/media/iam-role-valid-name.png)

- k. 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-verify-create-role.png)

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。

ロールの作成を確認 (/en-us/vpx/media/iam-see-role-created.png)

7. IAMロールの更新をクリックします。

IAMロールの更新 (/en-us/vpx/media/iam-update-iam-role.png)

IAMポリシーシミュレーターを使用して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の効果を、本番環境にコミットする前に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。

IAMシミュレーター (/en-us/vpx/media/iam-simulator.png)

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。

IAMシミュレーターモード(/en-us/vpx/media/iam-simulator-mode.png)

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後、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。

その他の参照

[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上のNetScaler VPXインスタンスの仕組み

NetScaler VPXインスタンスは、AWSマーケットプレイスでAMIとして利用可能であり、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 での標準 VPX インスタンスのインストールに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負荷分散に使用できます。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インスタンス展開の例

図1は、NetScaler VPX展開を含む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へのトラフィックおよびVPCから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1つずつ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

以下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、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](#)

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介して 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. キーペアを作成する
2. 仮想プライベートクラウド (VPC) を作成する
3. サブネットを追加する
4. セキュリティグループとセキュリティルールを作成する
5. ルートテーブルを追加する
6. インターネットゲートウェイを作成する
7. NetScaler VPXインスタンスを作成する
8. ネットワークインターフェイスを追加作成してアタッチする
9. 管理NICにElastic IPをアタッチする
10. VPXインスタンスに接続する

ステップ1：キーペアを作成する。

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および起動すると、既存のキーペアを使用するか、新しいキーペアを作成するように求められます。キーペアの作成方法の詳細については、「[Amazon EC2キーペア](#)」を参照してください。

ステップ2：VPCを作成する。

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、「[Amazon VPCの開始方法](#)」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、次の点に留意してください。

- 「単一のパブリックサブネットのみを持つVPC」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では、次の種類のサブネットを少なくとも**3つ**作成することをお勧めします。
 - 管理トラフィック用のサブネットを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1つ以上のサブネット。クライアントは、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1つ以上のサブネット。サーバーは、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)、およびサブネット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したい場合があります。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、AWS Marketplaceによって生成され、Citrixが推奨する設定に基づいたデフォルトのセキュリティグループを提供します。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を開く必要があります。

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アタッチ](#)」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。

AWSリージョン(/en-us/citrix-adc/media/ec2-region.png)

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から利用可能な無料の仮想アプライアンスです。）
- 持ち込みデバイス

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
- 確認

EC2ウィザード (/en-us/citrix-adc/media/ec2-wizard.png)

ステップ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する。

VIPおよびSNIP用にネットワークインターフェイスをさらに2つ作成します。ネットワークインターフェイ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イスの作成](#)」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イスを作成したら、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。インターフェイ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、インターフェイ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イ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イスのアタッチ](#)」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た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ら、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管理コンソールから、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 Webコンソールを使用してAWSにNetScaler VPXスタンドアロンインスタンスを展開する方法の詳細については、「[シナリオ: スタンドアロンインスタンス](#)」を参照してください。

Citrix 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したり、NetScaler VPXインスタンスのペアで高可用性環境を作成したり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 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の場合、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 Command Line Interface 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説明します。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
eth0	管理トラフィック (NSIP) を処理します	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 (VIP) を処理します	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インし、**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. Create **VPC** をクリックします。

シナリオ用のVPCを作成(/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、**Next** 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。



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。



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/>](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。



このVPC用に作成したすべてのサブネットにルートテーブルが割り当てられるため、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. 「**Save**」をクリックします。

シナリオのゲートウェイを追加する(/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する。

1. AWS管理コンソールにログインし、「**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、「**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設定(/en-us/vpx/media/configure-instance-scenario.png)

6. 「ストレージの追加」ページで、デフォルトオプションを選択し、**「次へ：タグの追加」**をクリックします。
7. 「タグの追加」ページで、インスタンスの名前を追加し、**「次へ：セキュリティグループの設定」**をクリックします。
8. 「セキュリティグループの設定」ページで、デフォルトオプション（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、**「確認と起動 > 起動」**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求められます。「キーペアの選択」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、**「インスタンスの起動」**をクリックします。

シナリオのキーペアを追加(/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、**「EC2 > 実行中のインスタンス」**をクリックします。インスタンスを選択し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てください。

ステップ 5: 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。

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ースは 1 つだけでした。次に、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 VPC に追加します。

1. Amazon EC2 コンソールを <https://console.aws.amazon.com/ec2/> で開きます。
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. 「ネットワークインターフェースを作成」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は、デフォルトの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。

ネットワークインターフェースを作成(/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、その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、アタッチをクリックします。
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、アタッチをクリックします。

ネットワークインターフェースをアタッチ(/en-us/vpx/media/attach-nic-scenario.png)

ステップ6：NSIPにElastic IPをアタッチします。

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. アタッチ可能な空き EIP を確認します。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、**接続**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、および SNIP) を構成するには、「[NetScaler が所有する IP アドレスの構成](#)」を参照してください。
- NetScaler VPX アプライアンスの BYOL バージョンを構成しました。詳細については、
`<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ライセンスのダウンロード

AWS MarketplaceからNetScaler VPXの顧客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® Webサイト内のライセンスポータルを使用して、有効なライセンスを生成します。
2. ライセンスをインスタンスにアップロードします。

これが**有料**のマーケットプレイス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が自動的にアクティブ化されます。

モデル番号がVPX 5000より高い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、[NetScaler VPX - Customer Licensed](#)を使用して、同じアベイラビリティゾーンにBYOL EC2インスタンスを作成します。新しいEC2インスタンスには、ENIインターフェイスが1つだけ必要です。
2. NetScaler GUIを使用して古いEC2インスタンスのデータをバックアップするには、次の手順に従います。

- a. **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.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.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.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.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.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.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イス（NSIPアドレスがバインドされている管理インターフェイスを除く）を、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イスをある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イスを選択します。
- 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イスを新しいEC2インスタンスにアタッチします。ネットワークインターフェイスをアタッチする必要があるEC2インスタンス名を入力します。

AWSでネットワークインターフェイスをアタッチ(/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イスに対して、**Step 1**から**Step 4**を実行します。シーケンスに従い、インターフェイスの順序を維持するようにしてください。つまり、まずインターフェイス2をデタッチしてアタッチし、次にインターフェイス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. 管理インターフェイスを古い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イス（プライマリネットワークインターフェイス）上のすべてのセカンダリIPアドレス（存在する場合）を新しいEC2インスタンスに移動します。IPアドレスをあるインターフェイスから別のインターフェイスに移動するには、次の手順に従います。
 - 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（存在する場合）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オンにします。
 - i. **保存**をクリックします。
- 
6. 新しいEC2インスタンスを起動し、構成を確認します。すべての構成が移動されたら、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、逆の順序で同じ手順を実行します。

9. サブスクリプションインスタンスからBYOLインスタンスに移行した後、ライセンスが必要です。ライセンスをインストールするには、次の手順に従います。

- Citrix Webサイトのライセンスポータルを使用して、有効なライセンスを生成します。
- ライセンスをインスタンスにアップロードします。

注記:

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。正しい機能セットとパフォーマンスが自動的にアクティブ化されます。

制限事項

管理インターフェイスを新しいEC2インスタンスに移動することはできません。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インスタンスとまったく同じ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で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の負荷分散、または次の場所で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インスタンスが属する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、インスタンスをEIPs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インスタンスでSNIPを構成します。
2. サーバー側トラフィック用のパブリック向け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 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。

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。

以下の2つのルールに従ってください。

ルール1. 一方の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、以下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。この展開では、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、最初の NIC (管理 IP アドレスに対応) を除き、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 として構成されていません。
- その他のセカンダリプライベート IP アドレスは、必要に応じて作成し、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 を探し、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、次の IAM 権限を追加します。

```
"iam:GetRole" "ec2:DescribeInstances" "ec2:DescribeNetworkInterfaces"  
"ec2:AssignPrivateIpAddresses"
```

注:

`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エラスティック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。

従来の方式の場合、次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。フェールオーバー時には、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。フェールオーバープロセスでは、AWS APIは次の処理を行います。

- `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。
- 仮想サーバーがリスンしている2つのIPアドレスの中から、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。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の場合、次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/` の場所にある `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ペアを展開する

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内の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ペアとして設定できます。この場合、両方のVPXインスタンスは同じAWSゾーンの同じサブネット上にあります。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、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](#)

同じゾーンにVPX HAペアを展開する方法

同じゾーンにVPX HAペアを展開する手順の概要を以下に示します。

1. AWS上に2つのVPXインスタンスを作成し、それぞれに3つのNICを割り当てます。
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を構成します。
4. 両方のノードでHAを構成します。

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います。

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る

AWSコンソールは、構成された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ます。

ネットワークインターフェイスにプライベートIPアドレスを割り当てるには、次の手順に従います。

1. `<https://console.aws.amazon.com/ec2/>` でAmazon EC2コンソールを開きます。
2. ナビゲーションペインで、**Network Interfaces**を選択し、インスタンスにアタッチされているネットワークインターフェイスを選択します。
3. **Actions > Manage IP Addresses**を選択します。
4. 要件に基づいて、**IPv4 Addresses**または**IPv6 Addresses**を選択します。
5. IPv4アドレスの場合：
 - a. **Assign new IP**を選択します。
 - b. インスタンスのサブネット範囲内にある特定のIPv4アドレスを入力するか、フィールドを空白のままにしてAmazon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イ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イ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、インターフェイス上のセカンダリ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、インターフェイス上のプライマリ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。

設定されているIPアドレスを確認するには、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。

設定されているHAノードを確認するには、`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。

セカンダリプライベートIP移行に基づくレガシーHAペアから新しいHAペアへの移行

注：

ENI移行に基づいて動作するVPX HAペアをデプロイするレガシー方法は非推奨です。したがって、セカンダリプライベートIP移行に基づくHAペアのデプロイを使用することをお勧めします。

レガシーHAペアからセカンダリプライベートIP移行に基づく新しいHAペアへのシームレスな移行を可能にするには、以下を確認してください。

1. プライマリノードとセカンダリノードの両方に同じ数のインターフェースがあり、これらのインターフェースが同じサブネット内にある必要があります。
2. レガシー方式でプライマリプライベートIPアドレスとして構成されていたVIPおよびSNIPは、新しい方式ではセカンダリプライベートIPアドレスに移行する必要があります。
3. 新しいHA展開に必要なIAM権限は、プライマリおよびセカンダリのNetScaler®インスタンスに追加する必要があります。
4. プライマリおよびセカンダリのNetScalerインスタンスの両方を再起動します。

詳細については、「[同一ゾーン内の高可用性](#)」を参照してください。

Citrix CloudFormationテンプレートを使用して高可用性ペアを展開する

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する

注記：

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. リージョンとフルフィルメントオプションを**NetScaler VPX – Customer Licensed**として選択します。
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** を**CloudFormation Template**として選択します。
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. **Choose Action** で、**Launch CloudFormation** を選択し、**Launch** をクリックします。
Create stack ページが表示されます。
12. **Next** をクリックします。

13. 「**スタックの詳細を指定**」ページが表示されます。次の詳細を入力します。
 - 「**スタック名**」を入力します。名前は25文字以内にする必要があります。
 - 「**ネットワーク構成**」で、以下を実行します。
 - 「**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、以下を実行します。
 - 「**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、以下を実行します。
 - デフォルトでは、「**管理インターフェイスにpublicIP(EIP)を割り当てますか?**」オプションは「**いいえ**」に設定されています。
 - デフォルトでは、「**クライアントインターフェイスにpublicIP(EIP)を割り当てますか?**」オプションは「**いいえ**」に設定されています。



14. **次へ**をクリックします。

15. **スタックオプションの設定**ページが表示されます。これはオプションのページです。



16. **次へ**をクリックします。

17. **オプション**ページが表示されます。(これはオプションのページです。) **次へ**をクリックします。

18. **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オンにし、**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ら、**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イスで作成されます。
22. ユーザー名 `nsroot` とインスタンス ID をパスワードとして使用して、プライマリノードにログオンします。GUI から、**システム > 高可用性 > ノード**に移動します。NetScaler VPX は、CloudFormation テンプレートによって HA ペアで既に構成されています。
23. NetScaler VPX HA ペアが表示されます。



Amazon CloudWatch を使用してインスタンスを監視する

Amazon CloudWatchサービスを使用して、CPUとメモリの使用率、スループットなど、一連のNetScaler VPXメトリックを監視できます。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**にログオンします。
2. インスタンスを選択します。
3. **監視**をクリックします。
4. **すべてのCloudWatchメトリックを表示**をクリックします。



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. 表示したいメトリックをクリックし、期間（分、時間、日、週、月単位）を設定します。
7. 使用状況の統計を表示するには、「**グラフ化されたメトリクス**」をクリックします。「**グラフィックオプション**」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

CloudWatchのグラフ化されたメトリクス(/en-us/vpx/media/cft-graphed-metrics.png)

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い場合、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

- デプロイを開始する前に、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ゾーン内の異なるサブネットに配置することも、2つの異なるAWSアベイラビリティ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ています。

- フェールオーバーが機能するためには、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。
- 2つのインスタンスは、ハートビートに使用されるため、UDPトラフィック用にポート3003を開放している必要があります。
- 両方のノードの管理サブネットは、REST APIが機能するように、インターネットまたは内部 NATを介してAWS APIサーバーにアクセスできる必要があります。
- IAMロールには、パブリックIPまたはElastic IP (EIP) の移行にはE2権限が、プライベートIPの移行にはEC2ルートテーブル権限が必要です。

AWSアベイラビリティゾーン間で高可用性をデプロイするには、次の方法があります。

- [Elastic 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](#)

その他の参照情報

AWS 用 NetScaler Application Delivery Management (ADM) の詳細については、[NetScaler ADM エージェントを AWS にインストールする](#)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高可用性ペアを展開する

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ゾーン間でのEIPアドレスによる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が次の処理を行います。

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. 仮想サーバーがリスンしている2つのIPアドレスから、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。1つは仮想サーバーに直接アタッチされているもので、もう1つはIPセットを介してアタッチされているものです。
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。高可用性のためには、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高可用性ペアを展開する方法

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ペアを展開する手順の概要です。

1. Amazon Virtual Private Cloud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展開します。
3. 高可用性を設定します。
 - a. 両方のインスタンスでINCモードの高可用性を設定します。
 - b. 両方のインスタンスでIPセットを追加します。
 - c. 両方のインスタンスで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、[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。 高可用性を構成します。高可用性を設定するには、NetScaler VPX CLIまたはGUIを使用できます。

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モードの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 <vserver_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. プライマリノードに、ユーザー名 `nsroot` とインスタンスIDをパスワードとしてログインします。
3. GUIから、**設定 > システム > 高可用性** に移動します。「追加」をクリックします。
4. 「リモートノードIPアドレス」フィールドに、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) を有効にする」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. 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、**システム > ネットワーク > IP > 追加** に移動します。
10. IPアドレス、ネットマスク、IPタイプ (仮想IP) の必要な値を追加し、「作成」をクリックします。
11. **システム > ネットワーク > IPセット > 追加** に移動します。IPセット名を追加し、「挿入」をクリックします。
12. IPV4ページから、仮想IPを選択し、「挿入」をクリックします。「作成」をクリックしてIPセットを作成します。
13. プライマリインスタンスに仮想サーバーを追加します

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。



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にアタッチされません。

図：この図は、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



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. IPセットを追加し、両方のインスタンスでその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ゾーン間でプライベート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インスタンスを構成できます。このソリューションは、既存のマルチゾーン(/ja-jp/vpx/current-release/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/current-release/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/current-release/system/high-availability-introduction/configuring-high-availability-nodes-different-subnets.html)を参照してください。

注:

この展開は、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。この展開はAWS Transit Gatewayと互換性があります。

AWS非共有VPCを使用したプライベートIPアドレスを持つ高可用性ペア

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を使用したプライベートIPアドレスを持つ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ペアを展開するための手順の概要です。

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モードで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仮想プライベートクラウド (VPC) を作成します。

ステップ2. 2つの異なるアベイラビリティゾーンに、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、**Add route** を選択します。**Destination** には、宛先 CIDR ブロック、単一の IP アドレス、またはプレフィックスリストの ID を入力します。ゲートウェイ ID には、プライマリノードのクライアントインターフェースの ENI を選択します。

ルートテーブル - ルートの編集(/en-us/vpx/media/edit-routes.png)

注:

プライマリインスタンスのクライアント ENI で **Source/Dest Check** を無効にする必要があります。

コンソールを使用してネットワークインターフェイスの送信元/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、**Network Interfaces** を選択します。
3. プライマリクライアントインターフェイスのネットワークインターフェイスを選択し、**Actions** を選択して、**Change Source/Dest. Check** をクリックします。
4. ダイアログボックスで、**Disabled** を選択し、**Save** をクリックします。

送信元/送信先チェックの変更(/en-us/vpx/media/change-source-dst-check.png)

ステップ 5. 高可用性を構成します。高可用性を設定するには、NetScaler VPX CLI または GUI を使用できます。

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 (例: 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 モードで高可用性を設定します
2. ユーザー名 **nsroot** とインスタンス ID をパスワードとして使用して、プライマリノードにログインします。
3. **Configuration > System > High Availability** に移動し、**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 **Turn on NIC (Independent Network Configuration)** モードを選択します。
6. **Remote System Login Credential** 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. プライマリインスタンスに仮想サーバーを追加する

設定 > トラフィック管理 > 仮想サーバー > 追加 に移動します。

仮想サーバーを追加する(/en-us/vpx/media/lb-vserver-pip.png)

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。

注記：

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設定手順は、[AWS非共有VPCを使用してプライベートIPアドレスを持つ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た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": "*"

    }

  ]
```

```
}
```

注記：

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、参加者アカウントに以下のIAM権限を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。

共有VPCロールの作成(/en-us/vpx/media/shared-vpc-create-role.png)

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。

クラウドパラメータの変更(/en-us/vpx/media/shared-vpc-change-cloud-parameters.png)

2. **Configure AWS Cloud Parameters**ページで、**RoleARN**フィールドの値を入力します。

AWSクラウドパラメータの設定(/en-us/vpx/media/configure-aws-cloud-parameters.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内に、2つのアベイラビリティーゾーンに2つのVPXインスタンスが作成されます。各インスタンスには、管理用、クライアント用、バックエンドサーバー用の3つのサブネットがあります。

次の図は、AWS上のINCモードでのNetScaler VPX高可用性セットアップを示しています。VPCの一部ではないカスタムサブネット10.10.10.10がVIPとして使用されます。したがって、10.10.10.10サブネットはアベイラビリティーゾーン間で利用できます。

フェイルオーバー前のINCモードでの高可用性セットアップ(/en-us/vpx/media/before-failover-ha-pip.png)

フェイルオーバー後のINCモードでの高可用性セットアップ(/en-us/vpx/media/after-failover-ha-pip.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設定します。

プライマリノードとセカンダリノードの両方で、次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。

次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ソリューションのAWS Transit Gateway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ルートテーブル内のVIPサブネットまたはIPプールに対する静的ルートが作成され、VPCに向けられます。

AWS トランジットゲートウェイ(/en-us/vpx/media/aws-transit-gateway.png)

AWS Transit Gateway を設定するには、次の手順に従います。

1. [Amazon VPC コンソール](#)を開きます。
2. ナビゲーションペインで、**Transit Gateway ルートテーブル**を選択します。
3. **ルートタブ**を選択し、**静的ルートの作成**をクリックします。

AWS Transit Gateway ルートテーブル(/en-us/vpx/media/aws-transit-gateway-route-table.png)

4. CIDR がプライベート VIPS サブネットを指し、アタッチメントが NetScaler VPX を持つ VPC を指す静的ルートを作成します。

静的ルートの作成(/en-us/vpx/media/tg-create-static-route.png)

5. **静的ルートの作成**をクリックし、**閉じる**を選択します。

トラブルシューティング

マルチゾーン HA で HA プライベート IP ソリューションを設定する際に問題が発生した場合は、トラブルシューティングのために次の重要な点を確認してください。

- プライマリノードとセカンダリノードの両方に同じ IAM 権限セットがあること。
- 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に同じ数のインターフェイスがあること。
- インスタンスを作成する際は、デバイスインデックス番号に基づいて、プライマリノードとセカンダリノードの両方でインターフェイスをアタッチする同じ順序に従ってください。たとえば、プライマリノードでクライアントインターフェイスが最初にアタッチされ、サーバーインターフェイスが2番目にアタッチされたとします。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、インターフェイ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。
- インターフェイスのシーケンスは、次のナビゲーションパスで確認できます: **AWS console > Network & Security > ENI > Device Index number**。デフォルトでは、次のデバイスインデックス番号がこれらのインターフェイスに割り当てられます:
 - 管理インターフェイス – 0
 - クライアントインターフェイス – 1
 - サーバーインターフェイス – 2
- プライマリENIのデバイスインデックス番号のシーケンスが0、1、2の場合、セカンダリENIも同じデバイスインデックス番号のシーケンス0、1、2に従う必要があります。

デバイスインデックス番号のシーケンスに不一致がある場合、ルートの損失を避けるために、不一致のすべてのルートはインデックス0（管理インターフェイス）に転送されます。しか

し、トラフィックの輻輳を引き起こす可能性があるため、ルートが管理インターフェースに移動するのを避けるために、インターフェースをデタッチして正しいシーケンスで再度アタッチする必要があります。

- 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`) がシェル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` で利用可能な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、一貫した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 アクセスを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利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ースを追加で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。

同じアベイラビリティゾーン内での高可用性展開については、[AWSに高可用性ペアを展開する](#) を参照してください。

AWS 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の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/アクティブおよびアクティブ/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わらず、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とのハイブリッドクラウドにおけるNetScalerアプライアンスによるアプリケーション配信を示しています。

ハイブリッドクラウド AWS Outpost(/en-us/vpx/media/hybrid-cloud-aws-outpost.png)

アクティブ-アクティブ展開では、NetScalerは分散環境全体でトラフィックをグローバルに誘導します。環境内のすべてのサイトは、メトリクス交換プロトコル (MEP) を介して、リソースの可用性と健全性に関するメトリクスを交換します。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ロードバランスし、GSLB構成で指定された定義済みメソッド (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リクエストを送信します。

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。

- すべてのノードがアクティブな状態でリソース利用率を最適化します。
- 各個々のユーザーに最も近いサイトにリクエストを誘導することで、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。

- ディザスタリカバリ
- 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(/ja-jp/vpx/current-release/deploy-aws.html)
- 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(/ja-jp/vpx/current-release/deploy-aws/deploy-vpx-on-aws-outposts.html#deploy-a-citrix-adc-vpx-instance-on-aws-outposts-by-using-the-aws-web-console)
- NetScaler VPXインスタンスでGSLBを構成する(/ja-jp/vpx/current-release/deploy-vpx-on-azure/configuring-gslb-on-citrix-adc-vpx-appliance.html#configure-gslb-sites-and-other-settings)

NetScaler Web App Firewall を使用して AWS API Gateway を保護する

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強制できます。NetScaler Web App Firewall は展開が容易で、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。

- 設定の簡素化
- 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、可用性ゾーン内またはゾーン間で ADC を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、AWS API Gateway の前に NetScaler アプライアンスが展開されています。



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 ゲートウェイに転送します。
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
```

前述の例のステップ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オンします。
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/current-release/appexpert/rewrite/rewrite-action-policy-examples/example-12-rewrite-hostname-url-client.html](https://ja-jp.citrix-adc/current-release/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、デプロイメントを保護するだけでなく、ユーザーの要件に基づいてリクエスト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分散を行い、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ゲートウェイ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/current-release/load-balancing/load-balancing-customizing-algorithms.html](https://ja-jp.citrix-adc/current-release/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リクエストをバックアップ仮想サーバーに送信できます。
 - 他にも多くの負荷分散機能が利用可能です。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/current-release/getting-started-with-citrix-adc/load-balancing.html](https://ja-jp.citrix-adc/current-release/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:** クラウド本番環境での作業自動化に使用できるREST APIをサポートします。
- **データの監視:** 参照用にデータを監視し、ログに記録します。

NetScalerアプライアンスは、AWS API Gatewayと統合できるさらに多くの機能を提供します。詳細については、[NetScalerドキュメント](#)を参照してください。

バックエンドのAWSオートスケーリングサービスを追加する

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。任意の時点で必要な数のインスタンスのみをデプロイすることで、アプリケーションの実行コストを最小限に抑えることができます。これを実現するには、トラフィック、メモリ、CPU使用率などを常に監視する必要があります。しかし、トラフィックを手動で監視するのは面倒です。アプリケーション環境が動的にスケールアップまたはスケールダウンするためには、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。

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ピアリングを使用して、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
- **グレースフルな接続終了:** 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。
- **スタンバイサーバーの接続ドレイン:** スタンバイ状態のサーバーに新しいクライアント接続が送信されるのを防ぎます。ただし、スタンバイサーバーは引き続きオートスケーリンググループ

プの一部であり、既存のクライアント接続が閉じられるまで処理を続行します。サーバーが InService 状態に戻ると、新しい接続の処理を再開します。スタンバイ状態は、サーバーの更新、変更、トラブルシューティング、または要件に基づいたスケールダウンに使用できます。詳細については、[AWSドキュメント](#)を参照してください。

図: NetScaler VPXインスタンスを使用したAWSオートスケーリングサービス



この図は、AWSオートスケーリングサービスがNetScaler VPXインスタンス（負荷分散仮想サーバー）とどのように互換性があるかを示しています。詳細については、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](#)
- [シンプル通知サービス \(SNS\)](#)
- [シンプルキューサービス \(Amazon SQS\)](#)

始める前に

NetScaler VPXインスタンスでAutoscaling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。

- 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](#)
- 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](#)」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。
 - 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](#)」を参照してください。

注:

以下をお勧めします。

- AWSでNetScaler VPXインスタンスを作成するには、CloudFormationテンプレートを使用します。
- 3つの個別のインターフェイスを作成します。1つは管理用 (NSIP)、1つはクライアント向けLB仮想サーバー用 (VIP)、もう1つはサブネットIP用 (NSIP) です。
- AWS Autoscale グループを作成します。既存のオートスケーリング設定がない場合は、次の操作を行う必要があります。
 - a. 起動設定を作成する
 - 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/GettingStartedTutorial1.html>を参照してください。

- NetScaler リリース 14.1-12.x 以降では、AWS Autoscale グループで Graceful オプションを有効にしている場合にのみ、スケールダウンポリシーを指定する必要があります。14.1-12.x より前の NetScaler リリースでは、Graceful オプションが有効になっているかどうかにかかわらず、少なくとも1つのスケールダウンポリシーを指定する必要がありました。

NetScaler VPX インスタンスは、ステップスケーリングポリシーのみをサポートしています。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 アカウントに次の IAM 権限があることを確認してください。

```
{  
  
  "Version": "2012-10-17",  
  
  "Statement": \[  
  
    {  
  
      "Action": \[  
  
        "ec2:DescribeInstances",  
  
        "ec2:DescribeNetworkInterfaces",  
  
        "ec2:DetachNetworkInterface",
```

```

        "ec2:AttachNetworkInterface",

        "ec2:StartInstances",

        "ec2:StopInstances",

        "ec2:RebootInstances",

        "autoscaling:*",

        "sns:*",

        "sqs:*"

        "iam: SimulatePrincipalPolicy"

        "iam: GetRole"

    ],

    "Resource": "*",

    "Effect": "Allow"

}

\]

}

```

NetScaler VPX インスタンスに AWS Autoscaling サービスを追加する

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インします。
2. **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作成時の注意点:

- 仮想サーバーのIPアドレスは、VPXインスタンスで利用可能な空きIPアドレスから自動的に入力されます。詳細については、「[複数のIPアドレスを管理する](#)」を参照してください。
- AWSアカウントで設定したオートスケールグループの正確な名前を入力します。詳細については、「[AWS Auto Scaling グループ](#)」を参照してください。
- オートスケーリンググループのプロトコルとポートを選択する際は、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、ロードバランシング仮想サーバーまたはサービスグループがダウンしているように見えます。証明書は、仮想サーバ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** を選択し、**Delay** フィールドにタイムアウト値を指定します。このオプションはスケールダウンイベントを開始します。VPXインスタンスはサーバーをすぐに削除せず、サーバーの1つを正常な削除のためにマークします。この期間中、VPX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ます。タイムアウト後、VPXインスタンスはサーバーを削除します。

Graceful オプションを選択しない場合、オートスケールグループ内のサーバーは、負荷が低下した直後に削除されます。これにより、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。

クラウドプロファイルを作成すると、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とするサービスグループが作成されます。バックエンドサーバーは、VPXインスタンスで構成されたSNIPを介して到達可能である必要があります。



注:

- AWSコンソールでオートスケール関連情報を表示するには、**EC2 > Dashboard > Auto Scaling > Auto Scaling Group** に移動します。
- 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) 用に異なるクラウドプロファイルを作成できます。したがって、NetScaler VPXインスタンスは、パブリッククラウド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。

AWS で NetScaler GSLB を展開する

AWS で NetScaler の GSLB を設定することは、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の別の VPC 内、またはオンプレミスのデータセンター内など）にトラフィックをロードバランスするように NetScaler を構成することから成ります。



DBS の概要

クラウドロードバランサー向けのドメイン名ベースサービス (DBS) を使用した NetScaler GSLB のサポートは、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 はアクティブ/アクティブ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 ディスカバリから AWS 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、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 と 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 に向けて、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 対応サービスディスカバリを使用して、Autoscale グループで識別された DBS 名前空間のメンバーサービスリソースを決定します。

クラウドロードバランサーを使用した NetScaler GSLB DBS Autoscale コンポーネント:



AWS コンポーネントを構成する

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、それぞれ異なるルールセットが必要であるため、異なるセキュリティグループを作成することをお勧めします。この例では、簡潔にするために統合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、[VPC のセキュリティグループ](#)を参照してください。

1. ユーザーの**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. **Create Security Group**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:

詳細な強化のためには、送信元 IP アクセスの制限が推奨されます。詳細については、[Web Server Rules](#)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 > Instances**に移動します。
 - b. 以下の詳細を使用して**Launch Instance**をクリックし、**Amazon Linux**インスタンスを設定します。

このインスタンスで Web サーバーまたはバックエンドサービスを設定するための詳細を入力します。

5. NetScaler の構成

- a. ユーザーの**AWS リソースグループ**にログインし、**EC2 > Instances**に移動します。
- b. **Launch Instance**をクリックし、以下の詳細を使用して**Amazon AMI**インスタンスを設定します。

6. Elastic IP の構成

注:

コスト削減のため、NSIP にパブリック IP を持たせず、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 > NETWORK & SECURITY > Elastic IPs**に移動します。
- b. **Allocate new address**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 > LOAD BALANCING > Load Balancers**に移動します。
- b. **Create Load Balancer**をクリックして、クラシックロードバランサーを設定します。

ユーザーの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ドメイン名ベースのサービスの設定

トラフィック管理の設定については、「[NetScaler GSLBドメインベースのサービスを設定する](#)」を参照してください。

展開タイプ

3-NIC展開

- 典型的な展開
 - GSLB StyleBook
 - ADMを使用
 - GSLB (Route53とドメイン登録) を使用
 - ライセンス - プール/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の規模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や、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用の3-NIC展開
2. ライセンス
3. 展開オプション

GSLB用の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 インストールにおける標準的な 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。

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のリクエストフォームをこちらから提出することで、より高いVPC制限をリクエストできます: [Amazon VPCリクエスト](#)。

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 上で実行されている 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 Marketplace 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、「[Deploy a NetScaler VPX standalone instance on AWS](#)」を参照してください。

ハイブリッドおよびマルチクラウド展開向けのNetScalerグローバルサーバー負荷分散

NetScalerの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）ソリューションにより、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間でアプリケーショントラフィックを分散できます。NetScalerのハイブリッドおよびマルチクラウドGSLBソリューションは、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設定を管理するのに役立ちます。また、オンプレミス設定を持つユーザーは、クラウドに完全に移行する前に、NetScalerのハイブリッドおよびマルチクラウドGSLBソリューションを使用して、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のハイブリッドおよびマルチクラウドGSLBソリューションは、単一の統合コンソールから地理的な場所をまたがる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。マルチ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リングできます。さらに、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 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リクエストを受信できます。DNSリクエストを受信したGSLBノードは、設定された負荷分散方法によって選択されたロードバランサー仮想サーバーIPアドレスを返します。メトリクス（サイト、ネットワーク、および永続性メトリクス）は、NetScaler独自のプロトコルであるメトリクス交換プロトコル（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、「[Configure Metrics Exchange Protocol](#)」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ノード間のメトリクスは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、「[Getting Started](#)」を参照してください。

注：

このドキュメントでは、以下の前提条件があります。

- ユーザーが既存の負荷分散設定を持っている場合、それが稼働していること。
- SNIPアドレスまたはGSLBサイトIPアドレスが各NetScaler GSLBノードに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クス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。
- DNSトラフィックを受信するために、ADNSまたはADNS-TCPサービスが各NetScaler GSLBインスタンスに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構成

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、ADNSサービス/DNSサーバーIPアドレス用にポート53を、MEPトラフィック交換用のGSLBサイトIPアドレス用にポート3009をユーザーが開く必要があります。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ユーザーが開く必要があります。たとえば、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をユーザーが開く必要があります。NetScaler エージェントとNetScaler ADM間のNITRO通信用にポート443を開きます。

動的ラウンドトリップタイムGSLB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を開く必要があります。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ソリューションのいくつかの機能について、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、親子トポロジが設定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、次のGSLBメソッドをサポートしています。

- メトリックベースのGSLBメソッド。メトリックベースのGSLBメソッドは、メトリック交換プロトコルを介して他のNetScalerノードからメトリックを収集します。
 - 最小接続: クライアント要求は、アクティブな接続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、現在処理しているトラフィック量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。
 - 最小パケット: クライアント要求は、過去14秒間に受信したパケット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。
- 非メトリックベースのGSLBメソッド
 - ラウンドロビン: クライアントからのリクエストは、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: この方式では、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方式
 - 静的近接性: クライアントからのリクエストは、クライアントIPアドレスに最も近いロードバランサーにルーティングされます。

- 。ラウンドトリップタイム（RTT）：この方式では、RTT値（クライアントのローカルDNSサーバーとデータセンター間の接続における時間遅延）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方式の詳細については、[load balancingAlgorithms](#)を参照してください。

GSLBトポロジ

NetScalerのハイブリッドおよびマルチクラウドGSLBソリューションは、アクティブ-パッシブトポロジ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、[Configure GSLB for Disaster Recovery](#)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方式を使用してGSLBおよびロードバランシングノードを構成し、ロードバランシングノードが別の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が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、[Parent-Child Topology deployment using the MEP Protocol](#)を参照してください。

IPv6のサポート

NetScalerのハイブリッドおよびマルチクラウドGSLBソリューションは、IPv6もサポートしています。

監視

NetScalerのハイブリッドおよびマルチクラウドGSLBソリューションは、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、モニターの設定はオプションです。

永続性

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、以下をサポートします。

- ・送信元 IP アドレスベースの永続性セッション。これにより、設定されたタイムアウト期間内に同じクライアントからの複数のリクエストが到着した場合、それらが同じサービスに転送されます。クライアントが次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。

- スピルオーバー永続性。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が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、「[スピルオーバーの構成](#)」を参照してください。
- サイト永続性。これにより、GSLB ノードがクライアントリクエストを処理するデータセンターを選択し、その選択されたデータセンターの IP アドレスを以降のすべての DNS リクエストに転送します。設定された永続性がダウンしているサイトに適用される場合、GSLB ノードは GSLB メソッドを使用して新しいサイトを選択し、その新しいサイトがクライアントからの以降のリクエストに対して永続的になります。

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、負荷分散ノード 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負荷分散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 に付属のマルチクラウド 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。

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。ワークフロー図の各ステップについては、図の後に詳しく説明します。



クラウド管理者として、次のタスクを実行します。

1. NetScaler 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 企業アカウントを作成するか、会社内の誰かが作成した既存のアカウントに参加します。

2. ユーザーがNetScaler Cloudにログオンした後、**NetScaler Application Delivery Management** タイルで**Manage**をクリックし、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との間の通信を可能にするために、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/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を構成する](#)。

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、および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の両方のインスタンス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。

- マルチクラウドGSLB StyleBookを使用して、選択したGSLB NetScalerインスタンスでGSLB構成を実行します。
- 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構成がある場合は、この手順をスキップできます。) ユーザーは、NetScalerインスタンスでロードバランサーを次の2つの方法のいずれかで構成できます。
- 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、[Set up basic load balancing](#)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負荷分散 StyleBookまたはHTTP/SSL負荷分散 (モニター付き) StyleBook) のいずれかを使用して、選択したNetScalerインスタンスに負荷分散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、「[StyleBooks](#)」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ために、LBノード用のマルチクラウド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および負荷分散ノードを構成しており、負荷分散ノードが別のNetScalerインスタンスに展開されている場合。
- サイトの永続性が必要な場合。

StyleBookを使用してNetScaler負荷分散ノードでGSLBを構成する

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および負荷分散ノードを構成しており、負荷分散ノードが別のNetScalerインスタンスに展開されている場合、**LBノード用のマルチクラウド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、マルチクラウドGSLB StyleBookを使用します。

このStyleBookは、次の前提条件に基づいています。

- 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、このStyleBookで定義されているすべてのパラメータの値をユーザーが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:

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ます。

3. 次のパラメータを設定します。
 - **アプリケーション名**。子サイトを作成するGSLBサイトに展開されているGSLBアプリケーションの名前を入力します。
 - **プロトコル**。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ドロップダウンリストボックスから選択します。
 - **LBヘルスチェック (オプション)**
 - **ヘルスチェックタイプ**。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アドレスのヘルスチェックに使用されるプローブのタイプを選択します。
 - **セキュアモード**。(オプション) SSLベースのヘルスチェックが必要な場合は、このパラメータを有効にするためには**はい**を選択します。
 - **HTTPリクエスト**。(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アドレスのプローブに使用される完全なHTTPリクエストを入力します。
 - **HTTPステータス応答コードのリスト**。(オプション) ユーザーが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見なされます。
 - **サイトのパブリック IP アドレス。** (オプション) 親サイトの 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。
 - **サイトの 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) そのサイトの 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. 選択した NetScaler インスタンス (LB ノード) に LB 構成を作成するには、**作成**をクリックします。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で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、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 マシンイメージ (AMI) として利用できます。AWS で NetScaler VPX をプロビジョニングするために CloudFormation テンプレートを使用する前に、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 の推奨デプロイアーキテクチャをコード化するか、ユーザーに NetScaler を紹介するか、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することを目的としています。ユーザーは、特定の運用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/エディションの NetScaler VPX を使用するには、ユーザーがテンプレートを編集し、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 デプロイ

このテンプレートは、2つのアベイラビリティゾーンに3つのサブネット（管理、クライアント、サーバー）を持つVPCをデプロイします。パブリックサブネットにデフォルトルートを持つインターネットゲートウェイをデプロイします。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です。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でプレフィックスされます。

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、購読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でのNetScaler VPXインスタンスのプロビジョニングに関する詳細については、[NetScaler VPXインスタンスをAWSにプロビジョニングする](#)を参照してください。

StyleBooksを使用してGSLBを構成する方法については、[StyleBooksを使用してGSLBを構成する](#)をご覧ください。

災害復旧 (DR)

災害とは、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。災害はデータセンターの運用に影響を与え、災害現場で失われたリソースとデータは完全に再構築され、復元される必要があります。データセンターにおけるデータの損失や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わらず、一貫性とパフォーマンスを求めています。

災害復旧のためにGSLBを展開するには、[AWSにNetScaler VPXスタンドアロンインスタンスを展開する](#)を参照してください。

その他のリソース

[ハイブリッドおよびマルチクラウド展開向けのNetScaler ADM GSLB。](#)

AWS で NetScaler Web App Firewall を展開する

NetScaler Web App Firewall は、レイヤー 3 ネットワークデバイスとして、または顧客サーバーと顧客ユーザー間のレイヤー 2 ネットワークブリッジとしてインストールできます。通常、顧客企業のルーターまたはファイアウォールの背後に配置されます。NetScaler Web App Firewall は、Web サーバーと、ユーザーがそれらの Web サーバーにアクセスするハブまたはスイッチとの間のトラフィックを傍受できる場所にインストールする必要があります。その後、ユーザーは、Web サーバーに直接ではなく Web Application Firewall に要求を送信し、ユーザーに直接ではなく Web

Application Firewall に応答を送信するようにネットワークを設定します。Web Application Firewall は、内部ルールセットとユーザーの追加および変更の両方を使用して、最終的な宛先に転送する前にそのトラフィックをフィルタリングします。有害であると検出されたアクティビティをブロックまたは無害化し、残りのトラフィックを Web サーバーに転送します。上記の画像は、フィルタリングプロセスの概要を示しています。

詳細については、「[NetScaler Web App Firewall の仕組み](#)」を参照してください。

本番環境展開のための AWS 上の NetScaler Web App Firewall のアーキテクチャ

この画像は、AWS クラウドに NetScaler Web App Firewall 環境を構築する、デフォルトパラメータを持つ仮想プライベートクラウド (VPC) を示しています。



本番環境展開では、NetScaler Web App Firewall 環境に対して以下のパラメータが設定されます。

- このアーキテクチャは、AWS CloudFormation テンプレートの使用を前提としています。
- AWS のベストプラクティスに従って、2 つのパブリックサブネットと 4 つのプライベートサブネットで構成され、2 つのアベイラビリティゾーンにまたがる VPC。これにより、/16 クラスレスインタードメインルーティング (CIDR) ブロック (65,536 のプライベート IP アドレスを持つネットワーク) を持つ独自の仮想ネットワークが AWS 上に提供されます。
- NetScaler Web App Firewall の 2 つのインスタンス (プライマリとセカンダリ) が、それぞれのアベイラビリティゾーンに 1 つずつ配置されます。
- **3 つのセキュリティグループ**。各ネットワークインターフェース (管理、クライアント、サーバー) に 1 つずつあり、関連するインスタンスのトラフィックを制御する仮想ファイアウォールとして機能します。
- 各インスタンスに**3 つのサブネット**。管理用、クライアント用、バックエンドサーバー用がそれぞれ 1 つずつです。
- VPC にアタッチされた**インターネットゲートウェイ**と、インターネットへのアクセスを許可するためにパブリックサブネットに関連付けられたパブリックサブネットルートテーブル。このゲートウェイは、Web App Firewall ホストがトラフィックを送受信するために使用されます。インターネットゲートウェイの詳細については、「[インターネットゲートウェイ](#)」を参照してください。
- **5 つのルートテーブル** - プライマリおよびセカンダリ Web App Firewall の両方のクライアントサブネットに関連付けられた 1 つのパブリックルートテーブル。残りの 4 つのルートテーブルは、4 つのプライベートサブネット (プライマリおよびセカンダリ Web App Firewall の管理側およびサーバー側サブネット) のそれぞれにリンクしています。

- Web App Firewall の AWS Lambda は、以下の処理を行います。
 - HAモードの各アベイラビリティゾーンに2つのWeb App Firewallを設定する
 - サンプルのWeb App Firewallプロファイルを作成し、Web App Firewallに関するこの設定をプッシュする
- 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)。デフォルトでは、CloudFormation Template (CFT) が必要なIAM ロールを作成します。ただし、ユーザーはNetScaler ADCインスタンスに独自のIAMロールを提供できます。
- パブリックサブネットでは、パブリックサブネット内のリソースのアウトバウンドインターネットアクセスを許可するために、2つのマネージドネットワークアドレス変換 (NAT) ゲートウェイを使用します。

注:

既存のVPCにNetScaler Web App Firewallを展開するCFT Web App Firewallテンプレートは、アスタリスクでマークされたコンポーネントをスキップし、ユーザーに既存のVPC構成を求めます。

バックエンドサーバーはCFTによってデプロイされません。

コストとライセンス

ユーザーは、AWSデプロイメントの実行中に使用されるAWSサービスの費用に責任を負います。このデプロイメントに使用できるAWS CloudFormationテンプレートには、ユーザーが必要に応じてカスタマイズできる構成パラメータが含まれています。インスタンスタイプなどの一部の設定は、デプロイメントのコストに影響します。費用の見積もりについては、ユーザーは使用している各AWSサービスの料金ページを参照してください。価格は変更される場合があります。

AWS上のNetScaler Web App Firewallにはライセンスが必要です。NetScaler Web App Firewallをライセンスするには、ユーザーはライセンスキーをS3バケットに配置し、デプロイメントを起動するときにその場所を指定する必要があります。

注:

ユーザーがBring your own license (BYOL) ライセンスモデルを選択する場合、AppFlow機能が有効になっていることを確認する必要があります。BYOLライセンスの詳細については、[AWS Marketplace/Citrix VPX - Customer Licensed](#)を参照してください。

AWSで実行されているCitrix® ADC Web App Firewallには、以下のライセンスオプションがあります。ユーザーは、スループットなどの単一の要素に基づいてAMI (Amazon Machine Image) を選択できます。

- **ライセンスモデル:** 従量課金制 (PAYG、本番ライセンスの場合) またはBring Your Own License (BYOL、顧客ライセンスAMI - Citrix ADC Pooled Capacityの場合)。Citrix ADC Pooled Capacityの詳細については、[Citrix ADC Pooled Capacity](#)を参照してください。

- BYOLには、3つのライセンスモードがあります。

- NetScaler プール容量の構成: [Citrix ADC プール容量の構成](#)
- ネットスケイラー VPX チェックイン/チェックアウトライセンス (CICO): [シトリックス ADC VPX チェックイン/チェックアウトライセンス](#)

ヒント:

ユーザーがVPX-200、VPX-1000、VPX-3000、VPX-5000、またはVPX-8000アプリケーションプラットフォームタイプでCICOライセンスを選択する場合、NetScaler Consoleライセンスサーバーに同じスループットライセンスが存在することを確認する必要があります。

- NetScaler 仮想CPUライセンス: [NetScaler 仮想CPUライセンス](#)

注:

ユーザーがVPXインスタンスの帯域幅を動的に変更したい場合は、BYOLオプションを選択する必要があります。例えば、**NetScaler プール容量**では、NetScaler Consoleからライセンスを割り当てたり、インスタンスの最小および最大容量に応じて、オンデマンドで再起動なしにNetScalerからライセンスをチェックアウトしたりできます。ライセンスエディションを変更する場合にのみ、再起動が必要です。

- **スループット:** 200 Mbps または 1 Gbps
- **バンドル:** Premium

展開オプション

この展開ガイドでは、2つの展開オプションを提供します。

- 最初のオプションは、クイックスタートガイド形式と以下のオプションを使用して展開することです。

- **NetScaler Web App Firewallを新しいVPCに展開する（エンドツーエンド展開）**。このオプションは、VPC、サブネット、セキュリティグループ、その他のインフラストラクチャコンポーネントで構成される新しいAWS環境を構築し、この新しいVPCにNetScaler Web App Firewallを展開します。
 - **NetScaler Web App Firewallを既存のVPCに展開する**。このオプションは、ユーザーの既存のAWSインフラストラクチャにNetScaler Web App Firewallをプロビジョニングします。
- 2番目のオプションは、NetScaler Consoleを使用してWeb App Firewall StyleBookを展開することです。

AWS クイックスタート

ステップ1: ユーザーのAWSアカウントにサインインする

- AWSのユーザーアカウントにサインインします: [AWS](#)。Amazonアカウントを作成する (必要に応じて) またはAmazonアカウントにサインインするために必要な権限を持つIAM (Identity and Access Management) ユーザーロールを使用します。
- ナビゲーションバーのリージョンセレクターを使用して、ユーザーがAWSアベイラビリティゾーン全体に高可用性をデプロイしたいAWSリージョンを選択します。
- ユーザーのAWSアカウントが正しく設定されていることを確認してください。詳細については、このドキュメントの「技術要件」セクションを参照してください。

ステップ2: NetScaler Web App Firewall AMIをサブスクライブする

- このデプロイには、AWS MarketplaceでのNetScaler Web App Firewall 用のAMIのサブスクリプションが必要です。
- ユーザーのAWSアカウントにサインインします。
- 次の表のいずれかのリンクを選択して、NetScaler Web App Firewall の提供ページを開きます。
 - ユーザーが以下のステップ3でNetScaler Web App Firewall をデプロイするためにクイックスタートガイドを起動する際、彼らは**NetScaler Web App Firewall Image** パラメーターを使用して、彼らのAMIサブスクリプションに一致するバンドルとスループットオプションを選択します。以下にAMIオプションと対応するパラメーター設定を示します。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。

注:

AMI IDを取得するには、GitHubのNetScaler Products on AWS Marketplaceページ: [Citrix Products on AWS Marketplace](#) を参照してください。

- AWS マーケットプレイス AMI
 - NetScaler ウェブアプリケーションファイアウォール (Web App Firewall) - 200 Mbps : [Citrix ウェブアプリケーションファイアウォール \(Web App Firewall\) - 200 Mbps](#)
 - NetScaler ウェブアプリケーションファイアウォール (Web App Firewall) - 1000 Mbps : [Citrix ウェブアプリケーションファイアウォール \(Web App Firewall\) - 1000 Mbps](#)

- AMIページで、**購読を続ける**を選択します。

NetScaler ウェブアプリケーションファイアウォール (Web App Firewall) の AWS Marketplace ページ(/en-us/vpx/media/image-vpx-aws-appsecurity-deployment-03.png)

- ソフトウェアの使用条件を確認し、**条件に同意する**を選択します。

NetScaler Web App Firewallユーザーライセンス契約の条件に同意する(/en-us/vpx/media/image-vpx-aws-appsecurity-deployment-04.png)

注:

ユーザーは確認ページを受け取り、アカウント所有者にはメール確認が送信されます。詳細なサブスクリプション手順については、AWS Marketplaceドキュメントの「Getting Started」を参照してください: [Getting Started](#)。

- サブスクリプションプロセスが完了したら、それ以上の操作を行わずにAWS Marketplaceを終了します。AWS Marketplaceからソフトウェアをプロビジョニングしないでください。ユーザーはクイックスタートガイドを使用してAMIをデプロイします。

ステップ3: AWSクイックスタートを起動する

- ユーザーのAWSアカウントにサインインし、次のいずれかのオプションを選択してAWS CloudFormationテンプレートを起動します。オプションの選択については、このガイドの前のデプロイオプションを参照してください。
 - ここにあるAWS CloudFormationテンプレートのいずれかを使用して、AWS上の新しいVPCにNetScaler VPXをデプロイします。

- [Citrix/Citrix-ADC-AWS-CloudFormation/Templates/High-Availability/Across-Availability-Zone](#)
- [Citrix/Citrix-ADC-AWS-CloudFormation/Templates/High-Availability/Same-Availability-Zone](#)

重要:

ユーザーが既存のVPCにNetScaler Web App Firewallをデプロイする場合、VPCが2つのアベイラビリティゾーンにまたがり、各アベイラビリティゾーンにワークロードインスタンス用のパブリックサブネット1つとプライベートサブネット2つがあり、サブネットが共有されていないことを確認する必要があります。このデプロイガイドは共有サブネットをサポートしていません。「共有VPCの操作」を参照してください: [Working with Shared VPCs](#)。これらのサブネットには、インスタンスがインターネットに公開されることなくパッケージやソフトウェアをダウンロードできるように、ルートテーブルにNATゲートウェイが必要です。NATゲートウェイの詳細については、[NAT Gateways](#)を参照してください。サブネットが重複しないように設定してください。

また、ユーザーは、Amazon VPCドキュメントの「DHCPオプションセット」で説明されているように、DHCPオプションのドメイン名オプションが設定されていることを確認する必要があります: DHCP Options Sets [DHCP Options Sets](#)。クイックスタートガイドを起動すると、VPC設定の入力を求められます。

- 各デプロイには約15分かかります。
- ナビゲーションバーの右上隅に表示されているAWSリージョンを確認し、必要に応じて変更します。ここにCitrix Web App Firewallのネットワークインフラストラクチャが構築されます。テンプレートはデフォルトで米国東部 (オハイオ) リージョンで起動されます。

注:

このデプロイにはNetScaler Web App Firewallが含まれていますが、これは現在すべてのAWSリージョンでサポートされているわけではありません。サポートされているリージョンの最新リストについては、AWSサービスエンドポイント: [AWSサービスエンドポイント](#)を参照してください。

- **テンプレートの選択**ページで、テンプレートURLのデフォルト設定を維持し、次へを選択します。

- **詳細の指定**ページで、ユーザーの都合に合わせてスタック名を指定します。テンプレートのパラメータを確認します。入力が必要なパラメータには値を指定します。その他のすべてのパラメータについては、デフォルト設定を確認し、必要に応じてカスタマイズします。
- 次の表では、パラメータがカテゴリ別にリストされ、デプロイオプションごとに個別に説明されています。
- 新規または既存のVPCにNetScaler Web App Firewallをデプロイするためのパラメータ (デプロイオプション1)
- ユーザーがパラメータの確認とカスタマイズを終えたら、次へを選択します。

新しいVPCにNetScaler Web App Firewallをデプロイするためのパラメータ

VPCネットワーク構成

パラメータラベル (名前)	デフォルト	説明
プライマリ可用性ゾーン (PrimaryAvailabilityZone)	入力が必要です	プライマリNetScaler Web App Firewallデプロイメントの可用性ゾーン
セカンダリ可用性ゾーン (SecondaryAvailabilityZone)	入力が必要です	セカンダリNetScaler Web App Firewallデプロイメントの可用性ゾーン
VPC CIDR (VPCCIDR)	10.0.0.0/16	VPCのCIDRブロック。 x.x.x.x/xの形式の有効なIP CIDR範囲である必要があります。
リモートSSH CIDR IP (管理) (RestrictedSSHCIDR)	入力が必要です	EC2インスタンスにSSH接続できるIPアドレス範囲 (ポート: 22)。
	例えば、0.0.0.0/0を使用すると、すべてのIPアドレスがSSHまたはRDPを使用してユーザーインスタンスにアクセスできるようになります。 注意: 本番環境での使用は安全ではないため、特定のIPアドレスまたはアドレス範囲のみがユーザーインスタンスに	

パラメータラベル (名前)	デフォルト	説明
	アクセスできるように承認してください。	
リモートHTTP CIDR IP (クライアント) (RestrictedWebAppCIDR)	0.0.0.0/0	EC2インスタンスにHTTP接続できるIPアドレス範囲 (ポート: 80)
リモートHTTP CIDR IP(クライアント) (RestrictedWebAppCIDR)	0.0.0.0/0	EC2インスタンスにHTTP接続できるIPアドレス範囲 (ポート: 80)
プライマリ管理プライベートサブネットCIDR (PrimaryManagementPrivateSubnetCIDR)	10.0.1.0/24	アベイラビリティゾーン1に配置されているプライマリ管理サブネットのCIDRブロック。
プライマリ管理プライベートIP (PrimaryManagementPrivateIP)	—	プライマリ管理サブネットCIDRから、プライマリ管理ENIに割り当てられたプライベートIP (最後のオクテットは5から254の間である必要があります)。
プライマリクライアントパブリックサブネットCIDR (PrimaryClientPublicSubnetCIDR)	10.0.2.0/24	アベイラビリティゾーン1に配置されているプライマリクライアントサブネットのCIDRブロック。
プライマリクライアントプライベートIP (PrimaryClientPrivateIP)	—	プライマリクライアントサブネットCIDRから、プライマリクライアントENIに割り当てられたプライベートIP (最後のオクテットは5から254の間である必要があります)。
プライマリサーバープライベートサブネットCIDR (PrimaryServerPrivateSubnetCIDR)	10.0.3.0/24	アベイラビリティゾーン1にあるプライマリサーバーのCIDRブロック。

パラメータラベル (名前)	デフォルト	説明
プライマリサーバープライベートIP (PrimaryServerPrivateIP)	—	プライマリサーバーサブネットCIDRから、プライマリサーバーENIに割り当てられたプライベートIP（最後のオクテットは5から254の間である必要があります）。
セカンダリ管理プライベートサブネットCIDR (SecondaryManagementPrivateSubnetCIDR)	10.0.4.0/24	アベイラビリティゾーン2にあるセカンダリ管理サブネットのCIDRブロック。
セカンダリ管理プライベートIP (SecondaryManagementPrivateIP)	—	セカンダリ管理ENIに割り当てられたプライベートIP（最後のオクテットは5から254の間である必要があります）。セカンダリ管理サブネットCIDRからセカンダリ管理IPが割り当てられます。
セカンダリクライアントパブリックサブネットCIDR (SecondaryClientPublicSubnetCIDR)	10.0.5.0/24	アベイラビリティゾーン2にあるセカンダリクライアントサブネットのCIDRブロック。
セカンダリクライアントプライベートIP (SecondaryClientPrivateIP)	—	セカンダリクライアントENIに割り当てられたプライベートIP（最後のオクテットは5から254の間である必要があります）。セカンダリクライアントサブネットCIDRからセカンダリクライアントIPを割り当てます。
セカンダリサーバープライベートサブネットCIDR (SecondaryServerPrivateSubnetCIDR)	10.0.6.0/24	アベイラビリティゾーン2に配置されるセカンダリサーバーサブネットのCIDRブロック。

パラメータラベル (名前)	デフォルト	説明
セカンダリサーバープライベートIP (SecondaryServerPrivateIP)	—	セカンダリサーバーENIに割り当てられたプライベートIP (最後のオクテットは5から254の間である必要があります)。セカンダリサーバーサブネットCIDRからセカンダリサーバーIPを割り当てます。
VPCテナンシー属性 (VPCTenancy)	デフォルト	VPCに起動されるインスタンスの許可されるテナンシー。単一の顧客専用のEC2インスタンスを起動するには、Dedicatedテナンシーを選択します。

Bastionホストの設定

パラメータラベル (名前)	デフォルト	説明
踏み台ホストが必要 (LinuxBastionHostEIP)	いいえ	デフォルトでは、踏み台ホストは設定されません。ただし、ユーザーがサンドボックスデプロイメントを選択した場合は、メニューから「はい」を選択してください。これにより、パブリックサブネットにEIPを持つLinux踏み台ホストがデプロイされ、ユーザーはプライベートおよびパブリックサブネット内のコンポーネントにアクセスできるようになります。

NetScaler Web App Firewall の設定

パラメータラベル (名前)	デフォルト	説明
キーペア名 (KeyPairName)	入力が必要です	起動後にユーザーがユーザーインスタンスに安全に接続できるようにする公開/秘密キー

パラメータラベル (名前)	デフォルト	説明
		ペア。これは、ユーザーが希望するAWSリージョンで作成したキーペアです。技術要件のセクションを参照してください。
NetScaler インスタンスタイプ (CitrixADCInstanceType)	m4.xlarge	ADCインスタンスに使用するEC2インスタンスタイプ。選択したインスタンスタイプがAWSマーケットプレイスで利用可能なインスタンスタイプと一致していることを確認してください。そうでない場合、CFTが失敗する可能性があります。
NetScaler ADC AMI ID (CitrixADCImageID)	—	NetScaler Web App Firewallのデプロイに使用するAWS Marketplace AMI。これは、ステップ2でユーザーがサブスクライブしたAMIと一致する必要があります。
NetScaler ADC VPX IAM ロール (<code>iam:GetRole</code>)	—	このテンプレート: AWS-Quickstart/Quickstart-Citrix-ADC-VPX/Templates は、NetScaler VPXに必要なIAMロールとインスタンスプロフィールを作成します。空のままにした場合、CFTが必要なIAMロールを作成します。
クライアントパブリックIP (EIP) (ClientPublicEIP)	いいえ	ユーザーがパブリックEIPをユーザーのクライアントネットワークインターフェースに割り当てたい場合は、「はい」を選択します。そうでない場合、デプロイ後でも、必要に応じて後で割り当てるオプションがあります。

プールライセンス構成

パラメータラベル (名前)	デフォルト	説明
NetScaler Console プールライセンス	いいえ	ライセンスにBYOLオプションを選択する場合は、リストから「はい」を選択します。これにより、ユーザーはすでに購入済みのライセンスをアップロードできます。開始する前に、NetScaler Consoleのプールライセンスが利用可能であることを確認するために、NetScaler ADCのプール容量を構成する必要があります。詳細については、 NetScalerプール容量の構成 を参照してください。
到達可能な NetScalerコンソール / NetScalerコンソールエージェント IP	入力が必要です	顧客ライセンスオプションの場合、ユーザーがNetScaler Consoleをオンプレミスに展開するか、クラウドにエージェントを展開するかにかかわらず、到達可能なNetScaler Console IPがあることを確認してください。これは入力パラメータとして使用されます。

パラメータラベル (名前)	デフォルト	説明
ライセンスモード	オプション	ユーザーは次の3つのライセンスモードから選択できます: * NetScaler プール容量を構成します。詳細については、Citrix ADC プール容量の構成を参照してください。 * NetScaler VPX チェックイン/チェックアウトライセンス (CICO)。詳細については、Citrix ADC 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。 * NetScaler 仮想 CPU ライセンス。詳細については、Citrix ADC 仮想 CPU ライセンスを参照してください。
ライセンス帯域幅 (Mbps)	ゼロ Mbps	ライセンスモードが Pooled-Licensing の場合にのみ、このフィールドが関係します。BYOL ADC が作成された後、割り当てられるライセンスの初期帯域幅 (Mbps) を割り当て

パラメータラベル (名前)	デフォルト	説明
		ます。10 Mbpsの倍数である必要があります。
ライセンスエディション	プレミアム	Pooled Capacity Licensing Modeのライセンスエディションは Premium です。
アプライアンスプラットフォームタイプ	オプション	ユーザーがCICOライセンスモードを選択した場合に のみ 、必要なアプライアンスプラットフォームタイプを選択してください。ユーザーには、VPX-200、VPX-1000、VPX-3000、VPX-5000、VPX-8000のオプションが提示されます。
ライセンスエディション	プレミアム	vCPUベースのライセンスのライセンスエディションは Premium です。

AWS Quick Start の設定

注:

ユーザーが自身のデプロイプロジェクトのためにQuick Start Guideテンプレートをカスタマイズする場合を除き、以下の2つのパラメータのデフォルト設定を維持することをお勧めします。これらのパラメータの設定を変更すると、新しいQuick Start Guideの場所を指すようにコード参照が自動的に更新されます。詳細については、こちらにあるAWS Quick Start Guide Contributor's Guideを参照してください: [AWS Quick Starts/Option 1 - Adopt a Quick Start](#)。

パラメータラベル (名前)	デフォルト	説明
クイックスタートガイド S3 バケット名 (QSS3BucketName)	aws-quickstart	ユーザーがQuick Start Guide アセットのコピー用に作成し たS3バケット。ユーザーが Quick Start Guideをカスタ マイズまたは拡張して使用す る場合に適用されます。バケッ ト名には、数字、小文字、大 文字、ハイフンを含めること ができますが、ハイフンで開 始または終了することはでき ません。
Quick Start Guide S3 キープレフィックス (QSS3KeyPrefix)	quickstart-citrix-adc-vpx/	オブジェクトキーとメタデー タ: オブジェクトキーとメタ データ からのS3キー名プレ フィックスは、ユーザーが Quick Start Guideをカスタ マイズまたは拡張して使用す る場合に、Quick Start Guideア セットのユーザーコピー用の フォルダをシミュレートする ために使用されます。このプ レフィックスには、数字、小 文字、大文字、ハイフン、ス ラッシュを含めることができ ます。

- **オプション**ページで、ユーザーはスタック内のリソースのResource Tagまたはキーと値のペアを指定し、詳細オプションを設定できます。Resource Tagの詳細については、[リソースタグ](#)を参照してください。AWS CloudFormation Stack Optionsの設定の詳細については、[AWS CloudFormationスタックオプションの設定](#)を参照してください。完了したら、**次へ**を選択します。
- **レビュー**ページで、テンプレート設定を確認します。**機能**の下で、テンプレートがIAMリソースを作成すること、およびマクロを自動展開する機能が必要になる可能性があることを承認するために、2つのチェックボックスを選択します。
- スタックをデプロイするには、**作成**を選択します。
- スタックのステータスを監視します。ステータスが**CREATE_COMPLETE**になったら、NetScaler Web App Firewallインスタンスの準備が完了です。

- スタックの出力タブに表示されるURLを使用して、作成されたリソースを表示します。



ステップ4：デプロイのテスト

このデプロイのインスタンスを**プライマリ**と**セカンダリ**と呼びます。各インスタンスには異なるIPアドレスが関連付けられています。Quick Startが正常にデプロイされると、トラフィックはアベイラビリティゾーン1に設定されたプライマリのNetScaler Web App Firewallインスタンスを経由します。フェイルオーバー状態では、プライマリインスタンスがクライアント要求に応答しない場合、セカンダリのWeb App Firewallインスタンスが引き継ぎます。

プライマリインスタンスの仮想IPアドレスのElastic IPアドレスは、セカンダリインスタンスに移行し、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として引き継ぎます。

フェイルオーバープロセスでは、NetScaler Web App Firewallは次のことを行います。

- NetScaler Web App Firewallは、IPセットがアタッチされている仮想サーバーをチェックします。
- NetScaler Web App Firewallは、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アドレスを持つIPアドレスを見つけます。
- NetScaler Web App Firewallは、パブリックElastic IPアドレスを、新しいプライマリ仮想IPアドレスに属するプライベートIPアドレスに再関連付けします。

デプロイを検証するには、次の手順を実行します。

- プライマリインスタンスに接続する

たとえば、プロキシサーバー、ジャンプホスト（AWSで実行されているLinux/Windows/FWインスタンス、または踏み台ホスト）、またはそのVPCに到達可能な別のデバイス、あるいはオンプレミス接続を扱う場合はDirect Connectを使用します。

- フェイルオーバーを強制するトリガーアクションを実行し、セカンダリインスタンスが引き継ぐかどうかを確認します。

ヒント:

NetScaler Web App Firewallに関する設定をさらに検証するには、**プライマリNetScaler Web App Firewall**インスタンスに接続した後、次のコマンドを実行します。

```
Sh appfw profile QS-Profile
```

踏み台ホストを使用してNetScaler Web App Firewall HAペアに接続する

ユーザーがサンドボックスデプロイメント（例えば、CFTの一部としてバスティオンホストの構成を選択する場合）を選択している場合、パブリックサブネットにデプロイされたLinuxバスティオンホストがWeb App Firewallインターフェースにアクセスするように構成されます。

AWS CloudFormationコンソール（ここでサインインしてアクセスします: [サインイン](#)）で、マスタスタックを選択し、**Outputs**タブで**LinuxBastionHostEIP1**の値を見つけます。



- **PrivateManagementPrivateNSIP**と**PrimaryADCInstanceID**のキーの値は、後続のステップでADCにSSH接続するために使用されます。
- **Services**を選択します。
- **Compute**タブで、**EC2**を選択します。
 - **Resources**の下で、**Running Instances**を選択します。
 - プライマリWeb App Firewallインスタンスの**Description**タブで、**IPv4パブリックIPアドレス**をメモします。ユーザーはSSHコマンドを構築するためにそのIPアドレスが必要です。



- キーをユーザーのキーチェーンに保存するには、コマンド `ssh-add -K [your-key-pair].pem` を実行します。

Linuxでは、ユーザーは-Kフラグを省略する必要があるかもしれません。

- ステップ1でユーザーがメモした**LinuxBastionHostEIP1**の値を使用して、次のコマンドでバスティオンホストにログインします。

```
ssh -A ubuntu@[LinuxBastionHostEIP1]
```

- バスティオンホストから、ユーザーはSSHを使用してプライマリWeb App Firewallインスタンスに接続できます。

```
ssh nsroot@[Primary Management Private NSIP]
```

パスワード:

プライマリNetScaler Web App Firewallインスタンスへの接続中(/en-us/vpx/media/image-vpx-aws-appsecurity-deployment-08.png)

現在、ユーザーはプライマリNetScaler Web App Firewallインスタンスに接続されています。利用可能なコマンドを確認するには、helpコマンドを実行します。現在のHA構成を表示するには、show HA nodeコマンドを実行します。

NetScaler コンソール

NetScaler Application Delivery Management Serviceは、オンプレミスまたはクラウドに展開されるNetScaler MPX、NetScaler VPX、NetScaler Gateway、NetScaler Secure Web Gateway、NetScaler SDX、NetScaler ADC CPX、およびNetScaler SD-WANアプライアンスを含むNetScalerの展開を管理するための簡単でスケーラブルなソリューションを提供します。

NetScaler Console Serviceのドキュメントには、サービスの利用開始方法、サービスでサポートされている機能のリスト、およびこのサービスソリューションに固有の構成に関する情報が含まれています。

詳細については、[NetScaler Console Overview](#)を参照してください。

NetScaler Consoleを使用してAWS上にNetScaler VPXインスタンスを展開する

顧客がアプリケーションをクラウドに移行すると、アプリケーションの一部であるコンポーネントが増加し、より分散され、動的に管理する必要が生じます。

詳細については、[Provisioning NetScaler VPX Instances on AWS](#)を参照してください。

NetScaler WebアプリケーションファイアウォールとOWASP Top 10 – 2017

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OWASP Top 10 for 2017を公開しました。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。ここでは、これらの欠陥を軽減するためにNetScaler Web App Firewall (Web App Firewall) を構成する方法を詳しく説明します。Web App Firewallは、NetScaler (Premium Edition) の統合モジュールとして、またあらゆる種類のアプライアンスとして利用できます。

OWASP Top 10の完全なドキュメントは、[OWASP Top Ten](#)で入手できます。

シグネチャは、ユーザーがアプリケーションの保護を最適化するのに役立つ以下の展開オプションを提供します。

- 否定セキュリティモデル: 否定セキュリティモデルでは、ユーザーは豊富な事前設定済みシグネチャルールを利用して、パターンマッチングの力で攻撃を検出し、アプリケーションの脆弱

性から保護します。ユーザーは不要なものだけをブロックし、それ以外は許可します。ユーザーは、ユーザーアプリケーションの特定のセキュリティニーズに基づいて独自のシグネチャルールを追加し、独自のカスタマイズされたセキュリティソリューションを設計できます。

- **ハイブリッドセキュリティモデル:** シグネチャを使用することに加えて、ユーザーはポジティブセキュリティチェックを使用して、ユーザーアプリケーションに最適な構成を作成できます。ユーザーが望まないものをブロックするにはシグネチャを使用し、許可されているものを強制するにはポジティブセキュリティチェックを使用します。

シグネチャを使用してユーザーアプリケーションを保護するには、ユーザーはシグネチャオブジェクトを使用するように1つ以上のプロファイルを構成する必要があります。ハイブリッドセキュリティ構成では、ユーザーシグネチャオブジェクト内のSQLインジェクションおよびクロスサイトスクリプティングパターンとSQL変換ルールは、シグネチャルールだけでなく、シグネチャオブジェクトを使用しているWebアプリケーションファイアウォールプロファイルで構成されたポジティブセキュリティチェックによっても使用されます。

Webアプリケーションファイアウォールは、ユーザーが保護するWebサイトおよびWebサービスへのトラフィックを検査し、シグネチャに一致するトラフィックを検出します。一致は、ルールのすべてのパターンがトラフィックに一致する場合にのみトリガーされます。一致が発生すると、ルールに指定されたアクションが呼び出されます。リクエストがブロックされた場合、ユーザーはエラーページまたはエラーオブジェクトを表示できます。ログメッセージは、ユーザーアプリケーションに対して開始されている攻撃を特定するのに役立ちます。ユーザーが統計を有効にすると、Webアプリケーションファイアウォールは、Webアプリケーションファイアウォールシグネチャまたはセキュリティチェックに一致するリクエストに関するデータを維持します。

トラフィックがシグネチャとポジティブセキュリティチェックの両方に一致する場合、2つのアクションのうちより制限の厳しい方が適用されます。たとえば、ブロックアクションが無効になっているシグネチャルールにリクエストが一致しても、そのリクエストがブロックアクションが有効になっているSQLインジェクションポジティブセキュリティチェックにも一致する場合、リクエストはブロックされます。この場合、リクエストはSQLインジェクションチェックによってブロックされますが、シグネチャ違反はとしてログに記録される可能性があります。

カスタマイズ: 必要に応じて、ユーザーは独自のルールをシグネチャオブジェクトに追加できます。ユーザーはSQL/XSSパターンをカスタマイズすることもできます。ユーザーアプリケーションの特定のセキュリティニーズに基づいて独自のシグネチャルールを追加するオプションは、ユーザーに独自のカスタマイズされたセキュリティソリューションを設計する柔軟性を提供します。ユーザーは望まないものだけをブロックし、残りは許可します。指定された場所の特定の高速一致パターンは、パフォーマンスを最適化するために処理オーバーヘッドを大幅に削減できます。ユーザーはSQLインジェクションおよびクロスサイトスクリプティングパターンを追加、変更、

または削除できます。組み込みの正規表現および式エディターは、ユーザーがユーザーパターンを構成し、その正確性を検証するのに役立ちます。

NetScaler Webアプリケーションファイアウォール

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。NetScaler Web App Firewallは、Webサイト、Web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。NetScaler Web App Firewallには、IPレピュテーションベースのフィルタリング、ボット軽減、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。認証の強制、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のオプションも含まれています。基本的なWeb App Firewall保護と高度なWeb App Firewall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、NetScaler Web App Firewallは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がどのように機能するかを自動的に学習することで、Web App Firewallは開発者がアプリケーションを展開および変更しても、アプリケーションに適応します。NetScaler Web App Firewallは、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、これまでになく簡単に迅速に稼働を開始できます。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安心して確認できます。

Web App Firewallの展開戦略

Webアプリケーションファイアウォールを展開する最初のステップは、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、どれが脆弱性が低く、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コンテンツの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、動的コンテンツの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、同じアプリケーションの異なるコンテンツを保護するために複数のポリシーとプロファイルを使用できます。

次のステップは、展開のベースラインを設定することです。まず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確認します。

次に、Web App Firewallを展開します。NetScalerコンソールとWeb App Firewall StyleBookを使用して、Web App Firewall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。

Web App Firewallが展開され、Web App Firewall StyleBookで構成された後、次の有用なステップは、NetScaler ADC Web App FirewallとOWASP Top 10を実装することです。

最後に、Web App Firewallの保護機能のうち3つは、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。

ネットスケーラー コンソール

NetScaler Consoleは、オンプレミスまたはクラウドに展開されているNetScaler ADC MPX、NetScaler ADC VPX、NetScaler Gateway、NetScaler Secure Web Gateway、NetScaler ADC SDX、NetScaler ADC CPX、およびNetScaler SD-WANアプライアンスを含むNetScaler ADC展開を管理するためのスケーラブルなソリューションを提供します。

NetScaler Consoleアプリケーションの分析および管理機能

NetScaler Consoleでサポートされている機能は、App SecurityにおけるNetScaler Consoleの役割にとって重要です。

機能の詳細については、「[機能とソリューション](#)」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、前提条件が満たされていることを確認してください。詳細については、「[前提条件](#)」を参照してください。

制限事項と使用ガイドライン

AWSにCitrix ADC VPXインスタンスを展開する場合、「[制限事項と使用ガイドライン](#)」に記載されている制限事項と使用ガイドラインが適用されます。

技術要件

展開を開始するためにクイックスタートガイドを起動する前に、ユーザーアカウントが以下のリソース表に指定されているとおりに構成されている必要があります。そうしないと、展開が失敗する可能性があります。

リソース

必要に応じて、ユーザーのAmazonアカウントにサインインし、以下のリソースのサービス制限の引き上げをここでリクエストしてください: [AWS/サインイン](#)。これらのリソースを使用する既存の展開があり、この展開でデフォルトの制限を超える可能性があると思われる場合は、これを行

う必要があるかもしれません。デフォルトの制限については、AWSドキュメントのAWS Service Quotasを参照してください: [AWS Service Quotas](#)。

ここで確認できるAWS Trusted Advisor: [AWS/サインイン](#)は、一部のサービスの利用状況と制限を表示するサービス制限チェックを提供します。

リソース	この展開で使用されるもの
VPC	1
Elastic IP アドレス	0/1 (Bastion ホスト用)
IAM セキュリティグループ	3
IAM ロール	1
サブネット	6 (3/アベイラビリティゾーン)
インターネットゲートウェイ	1
ルートテーブル	5
Web App Firewall VPX インスタンス	2
Bastionホスト	0/1
NATゲートウェイ	2

リージョン

AWS上のNetScaler Web App Firewallは現在、すべてのAWSリージョンでサポートされているわけではありません。サポートされているリージョンの最新リストについては、AWSドキュメントのAWSサービスエンドポイントを参照してください: [AWS Service Endpoints](#)。

AWSリージョンとクラウドインフラストラクチャが重要である理由の詳細については、以下を参照してください: [Global Infrastructure](#)。

キーペア

ユーザーがクイックスタートガイドを使用してデプロイを計画しているリージョンのユーザーAWSアカウントに、少なくとも1つのAmazon EC2キーペアが存在することを確認してください。キーペア名をメモしてください。デプロイ中にこの情報の入力を求められます。キーペアを作成す

るには、AWSドキュメントのAmazon EC2キーペアとLinuxインスタンスの手順に従ってください:
[Amazon EC2 Key Pairs and Linux Instances](#)。

ユーザーがテストまたは概念実証の目的でクイックスタートガイドをデプロイする場合、本番インスタンスですでに使用されているキーペアを指定するのではなく、新しいキーペアを作成することをお勧めします。

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てHTMLクロスサイトスクリプティングチェックを設定する](#)
- [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てバッファオーバーフローセキュリティチェックを構成する](#)
- [シグネチャオブジェクトの追加または削除](#)
- [シグネチャオブジェクトの構成または変更](#)
- [シグネチャオブジェクトの更新](#)
- [Snort ルール統合](#)
- [ボット検出](#)
- [Microsoft Azure 上で NetScaler VPX インスタンスを展開する](#)

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

注:

高可用性セットアップにおける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。

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ースを使用するように構成できます。

NetScaler VPX AWS Marketplace Edition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ー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。

- インターフェースタイプを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

インターフェースタイプをSR-IOVに変更する

ネットワークインターフェースのデフォルト構成を確認するには、`show interface summary` コマンドを実行します。

例1: 次のCLIスクリーンキャプチャは、NetScaler VPX AWS Marketplace Edition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ー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ー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> 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を確認するには、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が有効なネットワークインターフェースを示しています。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。



これらの手順で、SR-IOVネットワークインターフェースを使用するようにVPXインスタンスを設定する手順は完了です。

高可用性セットアップでSR-IOVを設定する

高可用性は、NetScalerリリース12.0ビルド57.19以降のSR-IOVインターフェースで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たIAMロールには、次の権限が必要です。

- ec2:DescribeInstances
- ec2:DescribeNetworkInterfaces
- ec2:DetachNetworkInterface
- ec2:AttachNetworkInterface
- ec2:StartInstances
- ec2:StopInstances
- ec2:RebootInstances
- オートスケーリング:*
- SNS:*
- 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した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。

NetScaler VPXインスタンスをAWS ENAで拡張ネットワーキングを使用するように構成する

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[拡張ネットワーキング](#)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 (PPS) パフォーマンス、および一貫して低いインスタンス間遅延を提供します。

構成を開始する前に、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](#)

ENA対応インスタンスでは、次のHA構成がサポートされています:

- プライベートIPアドレスは、同じアベイラビリティ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 で実行されている 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法を使用することを推奨しています。

注記:

- NetScaler VPX AMI 用の NetScaler ソフトウェアリリース 10.1.e-124.1308.e 以降 (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 インスタンスのサポートの変更により、10.1.e-124 以降のリリースから 10.1.123.x 以前のリリースへのダウングレードはサポートされていません。
- ほとんどのアップグレードでは新しい 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、高可用性構成方法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 以降を実行している場合、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。

上記のプロシージャを使用して、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こともできます。ただし、インスタンスタイプを M3 に変更したい場合を除きます。その場合、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 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 (たとえば、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 (たとえば、200 Mbps から 1000 Mbps 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の使用 (Bring-Your-Own-License)

顧客ライセンスを使用している場合、Citrix® ウェブ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 ウェブ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 Licensing Guide」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ように、NetScaler の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](#)に記載されている標準の 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 に変更します。手順は次のとおりです。

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。

NetScaler の高可用性構成を使用して新しい NetScaler AMI インスタンスにアップグレードする

新しい NetScaler AMI インスタンスへのアップグレードに高可用性方式を使用するには、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 フェー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、AWS から再構成または削除します。

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では、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の1クリック起動では、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つだけが必要です。

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。したがって、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は異なります。
- 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）を介して流れます。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)?
```

警告メッセージが表示されるのは、2つのVPXインスタンスのシステムソフトウェアがHA互換ではないためです。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で設定を保存します。

- `save config`

- f. いずれかのノードのコマンドプロンプトで、フェイルオーバーを強制実行するために次のコマンドを入力し、その後、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認します。

- `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構成を削除します。これにより、2つのインスタンスはHA構成ではなくなります。まず、セカンダリノードからHA構成を削除し、その後、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。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、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上でAWS MarketplaceライセンスのNetScaler VPXインスタンス（時間単位料金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にログオンし、「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。

サポートPIN(/en-us/vpx/media/nsvpx-aws-citrix-techsupprt-feebasedlic2.png)

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>](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適な方法は何ですか？**

NetScaler VPXインスタンスは、以下のいずれかの方法でAWSに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
- ギットハブ上のCitrix® AWS CloudFormationテンプレート
- GitHubのCitrix Terraformスクリプト
- ギットハブ上のCitrix Ansible 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を教えてください。**

AWS で NetScaler VPX インスタンスをアップグレードするには、システムソフトウェアをアップグレードするか、「[AWS で NetScaler VPX インスタンスをアップグレードする](#)」の手順に従って新しい 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、「[ジョブを使用して NetScaler インスタンスをアップグレードする](#)」の手順に従って ADM サービスを使用することです。

- **AWS での NetScaler VPX の HA フェイルオーバー時間はどのくらいですか？**

- 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。

- **[Elastic IP を使用した異なるゾーン間での高可用性](#)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 が複数の EIP にマッピングされている複数のアプリケーションがある場合、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。

- **異なるゾーン展開間での高可用性で 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、別の利用可能な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。

- Elastic IP を使用した異なるゾーン間での高可用性 と プライベート IP を使用した異なるゾーン間での高可用性 の展開は連携できますか？

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デプロイメントにおいて、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フェイルオーバー中にHAペアのセカンダリノードはどのルートテーブルを確認すべきか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上の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` 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以外の任意のvCPU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におけるADCの展開では、3つのNICと3つのサブネットの構成が必須ですか？**

3つのNICと3つのサブネットの構成は推奨される展開であり、それぞれが管理、クライアント、サーバーネットワーク用です。この展開により、より優れたトラフィック分離とVPXパフォーマンスが得られます。2つのNICと2つのサブネット、および1つのNICと1つのサブネットは、利用可能な他のオプションです。AWSで複数のNICが1つのサブネットを共有する構成（例えば、2つのNICと1つのサブネットの展開など）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上の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するものではないためです。従来のネットワークインターフェースとは異なり、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きく異なる可能性があります。

Microsoft Azure に NetScaler VPX インスタンスを展開する

Microsoft Azure Resource Manager (ARM) に NetScaler VPX インスタンスを展開すると、以下の両方の機能セットを使用してビジネスニーズを達成できます。

- 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

ARM に NetScaler VPX インスタンスを、スタンドアロンインスタンスとして、またはアクティブ/スタンバイモードの高可用性ペアとして展開できます。

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 を介して。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する。詳細については、[NetScaler ソリューションテンプレートの GitHub リポジトリ](#) を参照してください。

注:

Azure は、Azure の外部から発信されるトラフィックへのアクセスを制限し、それらをブロックします。アクセスを提供するには、パブリック IP アドレスがアタッチされている VM の NIC に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 ネットワークに関する知識。 [ネットワーク](#) の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、「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」 (</ja-jp/citrix-adc/current-release/deploying-vpx/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を超える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イス (NIC) を接続できます。任意のNICには、静的または動的なパブリックおよびプライベートIPアドレスの1つ以上のIP構成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、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ています。詳細については、[CTX224626](#)の記事を参照してください。

シングルNICマルチIPアーキテクチャ

この展開タイプでは、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アドレスを含む複数の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アドレスがNSIP、SNIP、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できる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使用 (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 チェックインおよび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することができます。詳細および詳しい手順については、「[NetScaler VPX チェックインおよび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 インスタンスで構成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 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:

Microsoft Azure Network Adapter (MANA) は、NetScaler バージョン 14.1-51.x 以降でサポートされています。詳細については、[Microsoft ドキュメント](#)を参照してください。

注意点

- 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。
- 1 Gbps を超えるスループットを持つ 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 は仮想マシンが再起動するたびに異なる 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。ICAOnly VPN 仮想サーバーパラメーターが ON に設定されています。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。ICAOnly VPN 仮想サーバーパラメーターが OFF に設定されています。SmartAccess モードは、ライセンスのない NetScaler VPX インスタンスで、NetScaler AAA セッションユーザー5人に対してのみ機能します。

注：

SmartControl 機能を構成するには、NetScaler VPX インスタンスにプレミアム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

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ています。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、次の Azure ドキュメントを参照してください。

- [Azure 仮想ネットワークの IPv6 とは？](#)

- [Azure 仮想ネットワークの IPv4 アプリケーションに IPv6 を追加する - Azure CLI](#)
- [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 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、次の制限が課されます。

- Azure アーキテクチャは、次の 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。
- Azureが仮想マシンのプロビジョニング中に生成するデプロイIDは、ARMではユーザーに表示されません。このデプロイIDを使用して、ARMにNetScaler VPXアプライアンスをデプロイ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、内部である場合もあります。
2. Azure Resource Manager (ARM) – ARM は、Azure のサービス向け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 - バイナリラージオブジェクト – 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常駐するクラウドサービスではなく、仮想マシンまたはロールインスタンスに直接割り当てることができるパブリック IP アドレスです。これは、クラウドサービスに割り当てられている VIP (仮想 IP) に取って代わるものではありません。むしろ、仮想マシンまたはロールインスタンスに直接接続するために使用できる追加の IP アドレスです。

注：

以前は、ILPIP はパブリック IP を意味する PIP と呼ばれていました。

7. 受信 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用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 には複数の IP-Config を関連付けることができ、最大 255 個まで可能です。
9. 負荷分散ルール – 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。ロードバランサーリソースの単一の定義で、複数の負荷分散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に拡張する場合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チェック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です。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です。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ながら、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アドレスでデプロイされた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上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ている場合、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はネットワークアドレス変換を実行して、PIPをポート10080の内部IPアドレス10.x.x.xにマッピングし、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バックエンドプール、NATルール、ヘルスプローブが定義されている構成済みの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ています。インターネットからの要求にパブリックIPアドレスを使用する場合、これらをプライベートポートとして定義することはできません。

ポート 21, 22, 80, 443, 8080, 67, 161, 179, 500, 520, 3003, 3008, 3009, 3010, 3011, 4001, 5061, 9000, 7000。

ただし、VIPなどのインターネットに面したサービスで標準ポート（たとえばポート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サービスのためにNetScalerで構成された別のポートにマッピングされます。

たとえば、VIPサービスがVPXインスタンス上のポート8443で実行されているが、パブリックポート443にマッピングされている場合があります。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、要求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ロードバランサーで構成されたPIPではなく、VPXインスタンスに関連付けられたパブリックIPアドレス（PIP）を使用するトラフィックでは機能しません。

注:

Azure Resource Managerでは、NetScaler VPXインスタンスはパブリックIPアドレス（PIP）と内部IPアドレスの2つのIPアドレス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を構成するには、内部IPアドレスと利用可能な空きポートを使用してください。VIPの構成にPIPを使用しないでください。

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、その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 Azure SDK
- マイクロソフト Azure PowerShell

[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。



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:

正しい場所を選択し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。

セキュリティグループの構成(/en-us/vpx/media/config-security-group.png)

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。

これらの手順で仮想ネットワークとサブネットを作成します。

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。

VNet サブネットの構成(/en-us/vpx/media/config-vnet-subnet-1.png)

3. **仮想ネットワークの作成** ペインで、次の値を入力し、**作成** をクリックします。

- 仮想ネットワークの名前
- 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

サブネットの構成(/en-us/vpx/media/config-vnet-subnet-5a.png)

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。

VNetサブネット(/en-us/vpx/media/config-vnet-subnet-3.png)

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。

VNetサブネット構成(/en-us/vpx/media/config-vnet-subnet-4.png)

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。これらのストレージデータ形式を使用して、ARMで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にストレージアカウントを作成します。

1. 「**+新規**」 > 「**データ + ストレージ**」 > 「**ストレージアカウント**」をクリックします。
2. 「**ストレージアカウントの作成**」ペインで、次の詳細を入力します。

- アカウント名
- デプロイモード - 「**Resource Manager**」が選択されていることを確認してください。
- アカウントの種類 - ドロップダウンリストから「**汎用**」を選択します。
- レプリケーション - ドロップダウンリストから「**ローカル冗長ストレージ**」を選択します。
- 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. 「**作成**」をクリックします。

ストレージアカウントの構成(/en-us/vpx/media/config-storage-account.png)

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. 「**+新規**」をクリックします。
2. MARKETPLACEペインで「**すべて表示**」をクリックし、「**仮想マシン**」をクリックします。
3. 可用性セットを検索し、表示されたリストから「**可用性セット**」エンティティを選択します。

可用性セットの構成(/en-us/vpx/media/config-avail-set-1.png)

4. 「**作成**」をクリックし、「**可用性セットの作成**」ペインで、以下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。

利用可能な構成セット(/en-us/vpx/media/config-avail-set-2.png)

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、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ポータルでエンティティをすばやく見つけるには、Azure Marketplaceの検索ボックスにエンティティの名前を入力し、\<Enter\>を押します。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.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、リリース番号が名前に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. **仮想マシンの作成** ペインで、各セクションに必要な値を指定して仮想マシンを作成します。各セクションで **OK** をクリックして、構成を保存します。

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® では、Azure Resource Manager のリソースグループからリソースグループを作成し、その後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 スタック環境では、基本パラメーターに加えて、次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（SSD 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インバウンドおよびアウトバウンド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。

高可用性デプロイメントの場合、アクティブ/スタンバイ構成でデプロイするために、同じ可用性セットおよび同じリソースグループ内に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で使用されます。

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
アイピーコンフィグ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:**

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は、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。

開始する前に

開始する前に、このリンクに記載されている手順に従って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を構成する

このユースケースでは、NSDoc0330VM VPXインスタンスが作成されます。

スタンドアロンモードでNetScaler VPXインスタンスに複数の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アドレスを構成するには：

1. VMにIPアドレスを追加する
2. NetScalerが所有するIPアドレスを構成する

ステップ1: VMにIPアドレスを追加する

1. ポータルで、フィルターボックスに「その他のサービス > 仮想マシンと入力」を入力し、**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**ネットワーク設定** 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に表示されるブレードで、**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** をクリックします。



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** に指定し、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アドレスの可用性を確認し、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る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用にIP構成をもう1つ追加する手順に従います。パブリックIPは必須ではありません。

IP構成の表示(/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のデプロイでは、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可能性のある合計フェイルオーバー時間は最大13秒です。

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) セットアップでデプロイできます。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を持つ高可用性デプロイメントには、以下のオプションがあります。

- 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)
- Direct Server Return (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クラウドでNetScaler VPXの高可用性デプロイメントを機能させるには、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可用性セットを使用した高可用性展開アーキテクチャの例

Azure 可用性セット(/en-us/vpx/media/azure-ha.png)

アクティブ-パッシブ展開では、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展開するには、次の2つの方法があります。

- **NetScaler VPX標準高可用性テンプレート:** このオプションを使用して、3つのサブネットと6つのNICのデフォルトオプションでHAペアを構成します。
- **Windows PowerShellコマンド:** このオプションを使用して、サブネットとNICの要件に応じてHAペアを構成します。

このトピックでは、Citrixテンプレートを使用してアクティブ-パッシブHAセットアップでVPXペアを展開する方法について説明します。PowerShellコマンドを使用する場合は、「[PowerShellコマンドを使用して複数のIPアドレスとNICを持つ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展開できます。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、および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。

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、次の手順を完了します。

1. Azure Marketplaceから「**NetScaler**」を検索します。

アジュール マーケットプレイス(/en-us/vpx/media/azure-marketplace-get-it-now.png)

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。

「ライセンス付きAzure HA」 (/en-us/vpx/media/adsc-ha-license.png)

4. 「**基本**」ページが表示されます。リソースグループを作成し、「**OK**」を選択します。

「テンプレートの基本」 (/en-us/vpx/media/template-basics.png)

5. 「**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。

「一般設定」 (/en-us/vpx/media/monitoring-metrics.png)

注:

デフォルトでは、「**監視メトリックの公開**」オプションは「**false**」に設定されています。このオプションを有効にする場合は、「**true**」を選択します。リソースにアクセスできる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. 「**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。

「ネットワーク設定」 (/en-us/vpx/media/template-networksettings.png)

7. 「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。
8. 「購入」ページが表示されます。「購入」を選択してデプロイを完了します。

必要な構成でAzureリソースグループが作成されるまで、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を見つけるには、ALB \> 「フロントエンドIP構成」を選択します。



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:

HA展開と仮想サーバー構成に関連する追加情報は、以下のリンクで提供されています。

- [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でのアクティブ/スタンバイHA展開におけるGSLBの構成](#)

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。特定の一部のAzureリージョンのみがアベイラビリティゾーンをサポートしています。アベイラビリティゾーンをサポートするリージョンの

詳細については、Azureドキュメント「[Azureのアベイラビリティゾーンとは](#)」を参照してください。

図: Azureアベイラビリティゾーンを使用した高可用性展開アーキテクチャの例



Azure Marketplaceで入手できる「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。

以下の手順を実行して、テンプレートを起動し、Azureアベイラビリティゾーンを使用して高可用性VPXペアを展開します。

1. Azure Marketplaceから、Citrix®ソリューションテンプレートを選択して開始します。



2. デプロイタイプが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。



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、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。



セキュリティルールやポートの追加など、HAセットアップに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上で実行されている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。

注意点

- Azure Marketplace オファーを使用してAzureにNetScaler VPXインスタンスを展開する場合、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. **監視**セクションで、**メトリック**をクリックします。
4. **メトリック名前空間**ドロップダウンメニューから、**NetScaler**をクリックします。
5. **メトリック**ドロップダウンメニューの**すべてのメトリック**で、表示したいメトリックをクリックします。
6. **メトリックの追加**をクリックして、同じグラフで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) の構成
- 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フェールオーバー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。そのため、HAセットアップでは、プライマリノードがヘルスプローブに応答し、セカンダリノードは応答しません。プライマリインスタンスが2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ヘルスプローブに応答を開始し、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ペアを展開する方法について説明します。NetScaler VPX Standard HAテンプレートを使用する場合は、[複数のIPアドレスとNICを使用したHAセットアップの構成](#)を参照してください。

PowerShellコマンドを使用してHA-INCノードを構成する

シナリオ: HA-INC PowerShell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ペアを展開します。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展開されます。各NICにはIP構成が割り当てられます。

ALB	VPX1	VPX2
ALBはパブリックIP 3 (pip3) に関連付けられています	管理IPはIPConfig1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	サーバー側IPはIPConfig8で構成されており、これには1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します
2. ネットワークセキュリティグループを作成し、ルールを追加します
3. 仮想ネットワークと3つのサブネットを作成します
4. パブリックIPアドレスを作成します
5. VPX1のIP構成を作成します

6. VPX2のIP構成を作成します
7. VPX1のNICを作成します
8. VPX2のNICを作成します
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
- OSディスクを指定してVMを作成

```
$suffixNumber = 1
```

```
$vmName=$vmNamePrefix + $suffixNumber
```

```
$vmConfig=New-AzureRMVMConfig -VMName $vmName -VMSize $vmSize  
-AvailabilitySetId $avSet.Id
```



```
$cred=Get-Credential -Message "Type the name and password for VPX login."
```

```
$vmConfig=Set-AzureRMVMOperatingSystem -VM $vmConfig -Linux  
-ComputerName $vmName -Credential $cred
```

```
$vmConfig=Set-AzureRMVMSourceImage -VM $vmConfig -PublisherName  
$publisher -Offer $offer -Skus $sku -Version $version
```

```
$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic1.Id  
-Primary
```

```
$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic2.Id
```

```
$vmConfig=Add-AzureRMVMNetworkInterface -VM $vmConfig -Id $nic3.Id
```

```
$osDiskName=$vmName + "-" + $osDiskSuffix1
```

```
$osVhdUri=$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。

```
...
```

```
$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に割り当てられているプライベート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。

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
- ヘルスプローブを作成
- バックエンドアドレスプールを作成
- ロードバランシングルール (HTTPおよびSSL) を作成
- 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
- 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ペアのデプロイが正常に完了したら、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) を持つ負荷分散仮想サーバーを追加します。

```
add lb virtual server v1 HTTP FrontEndIPofALB 80
```

関連リソース:

Azureでのアクティブ/スタンバイHA展開におけるGSLBの構成(/ja-jp/vpx/current-release/deploy-vpx-on-azure/configuring-gslb-active-standby-ha-deployment-azure.html)

ALBがフローティングIP無効モードの場合に、NetScaler高可用性ペアをAzureにデプロイする

複数のNICを持つNetScaler VPXインスタンスのペアを、Azure上でアクティブ-パッシブの高可用性 (HA) セットアップとしてデプロイできます。各NICには多数のIPアドレスを含めることができます。

アクティブ-パッシブデプロイメント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、以下の設定:

- 。フローティングIP有効モードまたはダイレクトサーバーリターン（DSR）モード
- 。フローティングIP無効モード

ALB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が有効な状態で、Azure上にアクティブ-パッシブHAセットアップでVPXペアをデプロイする場合は、[PowerShellコマンド](#)を使用して複数のIPアドレスとNICを持つ高可用性セットアップを構成するを参照してください。

ALBがフローティングIP無効モードの場合のHAデプロイメントアーキテクチャ

アクティブ-パッシブデプロイメントでは、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となります。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フェイルオーバーするまでスタンバイモードになります。

図: アクティブ-パッシブデプロイメントアーキテクチャの例



前提条件

AzureにNetScaler VPXインスタンスをデプロイする前に、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 Networking](#)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。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 が無効な状態で Azure に VPX HA ペアを展開する方法

HA および ALB の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 のバージョンを選択します (この例では、NetScaler リリース 13.1 を使用しています)。

NetScaler のバージョンを選択(/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。



「**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）で必要な詳細を完了します。



「**ネットワーク**」タブで、管理、クライアント、サーバーの各NIC用にそれぞれ1つずつ、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の作成後に作成され、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。



4. 「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、ネットワークと追加設定を確認し、「作成」をクリックします。



5. 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。



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ついては、**詳細**を選択して既存のネットワークセキュリティグループを使用するか、**基本**を選択して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）すべてが接続されています。

セカンダリインスタンスにNICを追加するには、前述の手順を繰り返します。

両方のインスタンスに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:

NetScaler VPXインスタンスの構成時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る、クライアントNICの受信ルール内の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。

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 をデプロイします。

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デプロイします。



注:

NetScaler VM に接続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 を有効にする**チェックボックスをオフにします。



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 に 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されたユーザー名 `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、「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. 「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されたユーザー名 `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、「追加」をクリックします。
3. 「リモートノード IP アドレス」フィールドに、プライマリ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. 「セルフノードで INC (独立ネットワーク構成) モードをオンにする」チェックボックスをオンにします。
5. 「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)

さらに進む前に、「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)

「**手順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. 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NICのプライベートIPアドレスと、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アドレスを次の手順で追加します。
 - 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。
4. セカンダリVIPアドレスを次の手順で追加します。
 - a. セカンダリインスタンスのクライアントNICの内部IPアドレスと、VMインスタンスのクライアントサブネット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。

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. 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. セカンダリVIPアドレスを次の手順で追加します。
 - a. セカンダリインスタンスのクライアントNICの内部IPアドレスと、VMインスタンスのクライアントサブネット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。
3. セカンダリSNIPアドレスを次の手順で追加します。
 - 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を追加する(/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セットを作成する(/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セットを作成する(/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. 「**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する(/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 > 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. 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. 手順4で設定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で設定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. 設定を保存します。

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、すべての設定が失われます。

手順8. 設定を確認します。

フェイルオーバー後も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 DNS プライベートゾーン向けNetScaler®の展開

Azure DNS は、DNS ドメインをホストし、名前解決を提供する Microsoft Azure インフラストラクチャ上のサービスです。

Azure DNS プライベートゾーンは、プライベートネットワーク内のドメイン名を解決することに特化したサービスです。プライベートゾーンを使用すると、顧客は現在利用可能な Azure 提供の名前ではなく、独自のカスタムドメイン名を使用できます。

主要なアプリケーションデリバリーソリューションである NetScaler は、Azure DNS プライベートゾーンのロードバランシングおよび GSLB 機能を提供するのに最適です。Azure DNS プライベートゾーンを購読することで、企業は NetScaler Global Server Load Balancing (GSLB) の能力とインテリジェンスに頼り、安全な VPN トンネルを介して接続された、複数の地理的地域およびデータセンターにわたるワークロードにイントラネットトラフィックを分散させることができます。この連携により、企業は Azure パブリッククラウドに移行したいワークロードの一部にシームレスにアクセスできるようになります。

Azure DNS の概要

ドメインネームシステム (DNS) は、サービス名をその IP アドレスに変換または解決する役割を担っています。DNS ドメインのホスティングサービスである Azure DNS は、Microsoft Azure インフラストラクチャを使用して名前解決を提供します。インターネットに接続する DNS ドメインのサポートに加えて、Azure DNS は現在、プライベート DNS ドメインもサポートしています。

Azure DNS は、カスタム DNS ソリューションを必要とせずに、仮想ネットワーク内のドメイン名を管理および解決するための信頼性の高い安全な DNS サービスを提供します。プライベート DNS ゾーンを使用することで、Azure 提供の名前ではなく、独自のカスタムドメイン名を使用できます。カスタムドメイン名を使用すると、組織のニーズに最適な仮想ネットワークアーキテクチャを調整できます。仮想ネットワーク内および仮想ネットワーク間の仮想マシン (VM) の名前解決を提供します。また、顧客はスプリットホライズンビューでゾーン名を構成でき、これによりプライベート DNS ゾーンとパブリック DNS ゾーンが名前を共有できます。

Azure DNS プライベートゾーンにNetScaler GSLB を使用する理由？

今日のビジネスでは、ワークロードをオンプレミスから Azure クラウドに移行したいと考えています。クラウドへの移行により、市場投入までの時間、設備投資/価格、展開の容易さ、セキュリティを適用できます。Azure DNS プライベートゾーンサービスは、ワークロードの一部を Azure クラウドに移行している企業にとって独自の提案を提供します。これらの企業は、プライベートゾーンサービスを使用する際に、オンプレミス展開で長年使用していたプライベート DNS 名を作成できます。オンプレミスと Azure クラウドにイントラネットアプリケーションサーバーが存在し、安全な VPN トンネルを介して接続されているこのハイブリッドモデルでは、これらのイントラネットアプリケーションへのシームレスなアクセスが課題となります。NetScaler は、そのグローバルロードバランシング機能により、この独自のユースケースを解決します。この機能は、アプリケーショントラフィックをオンプレミスまたは Azure クラウドの最も最適な分散ワークロード/サーバーにルーティングし、アプリケーションサーバーのヘルスステータスを提供します。

ユースケース

オンプレミスネットワークおよび異なる Azure VNet のユーザーは、必要なコンテンツにアクセスするために、内部ネットワーク内の最適なサーバーに接続できます。これにより、アプリケーションは常に利用可能であり、コストが最適化され、ユーザーエクスペリエンスが良好であることが保証されます。ここでは、Azure プライベートトラフィック管理 (PTM) が主要な要件です。Azure PTM は、ユーザーの DNS クエリがアプリケーションサーバーの適切なプライベート IP アドレスに解決されることを保証します。

ユースケースソリューション

NetScaler には、Azure PTM の要件を満たすためのグローバルサーバーロードバランシング (GSLB) 機能が含まれています。GSLB は DNS サーバーのように機能し、DNS リクエストを取得し、その DNS リクエストを適切な IP アドレスに解決して、以下を提供します。

- シームレスな DNS ベースのフェイルオーバー。
- オンプレミスからクラウドへの段階的な移行。
- 新機能のA/Bテスト。

サポートされている多くのロードバランシング方法の中で、このソリューションでは以下の方法が役立ちます。

1. ラウンドロビン
2. 静的近接性（ロケーションベースのサーバー選択）。これは2つの方法で展開できます。
 - a. NetScaler上のEDNSクライアントサブネット（ECS）ベースのGSLB。
 - b. 各仮想ネットワークにDNSフォワーダーを展開します。

トポロジ

次図は、AzureプライベートDNSゾーンのNetScaler GSLB展開を示しています。



ユーザーは、AzureプライベートDNSゾーンのNetScaler GSLBメソッドに基づいて、Azureまたはオンプレミスのいずれかのアプリケーションサーバーにアクセスできます。オンプレミスとAzure仮想ネットワーク間のすべてのトラフィックは、セキュアなVPNトンネルのみを介して行われます。アプリケーショントラフィック、DNSトラフィック、および監視トラフィックは、上記のトポロジに示されています。必要な冗長性に応じて、NetScalerとDNSフォワーダーは仮想ネットワークとデータセンターに展開できます。簡素化のため、ここでは1つのNetScalerのみが示されていますが、Azureリージョンには少なくとも1セットのNetScalerとDNSフォワーダーを推奨します。すべてのユーザーDNSクエリは、まず、適切なDNSサーバーにクエリを転送するためのルールが定義されているDNSフォワーダーに送信されます。

Azure DNSプライベートゾーンのNetScalerを構成する

テスト済みの製品とバージョン：

製品	バージョン
アジュール	クラウドサブスクリプション
NetScaler VPX	BYOL (ライセンス持ち込み)

注:

この展開はテスト済みであり、NetScalerバージョン12.0以降で同じです。

前提条件

以下は一般的な前提条件です。

- 有効なサブスクリプションを持つMicrosoft Azureポータルアカウント。
- オンプレミスとAzureクラウド間の接続（セキュアVPNトンネル）を確保します。AzureでセキュアVPNトンネルを設定するには、「[Step-By-Step: Configuring a site-to-site VPN Gateway between Azure and on-premises](#)」を参照してください。

ソリューションの説明

HTTPsで動作し、ラウンドロビンGSLBロードバランシング方式に基づいてイントラネットアクセスを備えたAzureとオンプレミスに展開されている1つのアプリケーションAzure DNSプライベートゾーン (rr.ptm.mysite.net) をホストしたい場合。この展開を実現するには、以下の構成からなるNetScalerを使用してAzureプライベートDNSゾーンのGSLBを有効にします。

1. Azureとオンプレミスのセットアップを構成します。
2. Azure仮想ネットワーク上のNetScalerアプライアンス。

Azureとオンプレミスのセットアップを構成する

トポロジに示すように、Azure仮想ネットワーク（この場合はVNet A、VNet B）とオンプレミスセットアップを設定します。

1. ドメイン名 (mysite.net) を使用して Azure プライベート DNS ゾーンを作成します。
2. Azure リージョンに、ハブアンドスポークモデルで2つの仮想ネットワーク (VNet A、VNet B) を作成します。
3. VNet A に App Server、DNS フォワーダー、Windows 10 Pro クライアント、NetScaler をデプロイします。
4. VNet B にクライアントがいる場合は、App Server と DNS フォワーダーをデプロイします。
5. オンプレミスに App Server、DNS フォワーダー、Windows 10 Pro クライアントをデプロイします。

Azure プライベート DNS ゾーン

ドメイン名を使用して Azure プライベート DNS ゾーンを作成します。

1. Azure ポータルにログインし、ダッシュボードを選択または作成します。
2. リソースの作成をクリックし、DNS ゾーンを検索して、ドメイン名 (mysite.net) を持つ Azure プライベート DNS ゾーン (この場合は mysite.net) を作成します。

Azure ポータル DNS ゾーンの例(/en-us/vpx/media/image-02.png)

ハブアンドスポークモデルの Azure 仮想ネットワーク (VNet A、VNet B)

Azure リージョンに、ハブアンドスポークモデルで2つの仮想ネットワーク (VNet A、VNet B) を作成します。

1. 2つの仮想ネットワークを作成します。

2. 同じダッシュボードを選択し、**リソースの作成**をクリックして仮想ネットワークを検索し、VNet A と VNet B という2つの仮想ネットワークを同じリージョンに作成し、以下の画像に示すようにピアリングしてハブアンドスポークモデルを形成します。ハブアンドスポークトポロジの設定方法の詳細については、「[Azure でハブアンドスポークネットワークトポロジを実装する](#)」を参照してください。

仮想ネットワーク A (VNet A)(/en-us/vpx/media/image-03.png)



VNet A から VNet B へのピアリング

VNet A と VNet B をピアリングするには:

1. VNet A の メニューから をクリックし、VNet B をピアリングします。
2. 次の図に示すように、 と を有効にします。



次の図は、VNet A から VNet B へのピアリングが成功したことを示しています。



VNet B から VNet A へのピアリング

VNet B と VNet A をピアリングするには:

1. VNet B の メニューから をクリックし、VNet A をピアリングします。
2. 次の図に示すように、 を有効にし、リモートゲートウェイを使用します。



次の図は、VNet B から VNet A へのピアリングが成功したことを示しています。



VNet A に App サーバー、DNS フォワーダー、Windows 10 Pro クライアント、NetScaler を展開する

VNet A 上の App サーバー、DNS フォワーダー、Windows 10 Pro クライアント、および NetScaler について簡単に説明します。

1. 同じダッシュボードを選択し、**リソースの作成**をクリックします。
2. 該当するインスタンスを検索し、VNet A サブネットから IP を割り当てます。

アプリケーションサーバー

アプリケーションサーバーとは、Azure またはオンプレミスの VM に Ubuntu Server 16.04 がインスタンスとしてデプロイされた Web サーバー (HTTP サーバー) のことです。Web サーバーとして機能させるには、コマンドプロンプトで次のように入力します。

```
sudo apt install apache2
```

Windows 10 Pro クライアント

VNet A およびオンプレミスで、Windows 10 Pro インスタンスをクライアントマシンとして起動します。

NetScaler

NetScaler は、NetScaler MAS からのヘルスチェックと分析によって Azure DNS プライベートゾーンを補完します。要件に基づいて Azure Marketplace から NetScaler を起動します。ここでは、このデプロイに NetScaler (BYOL) を使用しました。

Microsoft Azure に NetScaler をデプロイする方法の詳細な手順については、「[Microsoft Azure に NetScaler VPX インスタンスをデプロイする](#)」を参照してください。

デプロイ後、NetScaler IP を使用して NetScaler GSLB を構成します。

DNS フォワーダー

これは、NetScaler GSLB (ADNS IP) にバインドされたホスト型ドメインのクライアント要求を転送するために使用されます。Ubuntu Server 16.04 を Linux インスタンス (Ubuntu Server 16.04) として起動し、DNS フォワーダーとして設定する方法については、以下の URL を参照してください。

注:

ラウンドロビン GSLB ロードバランシング方式の場合、Azure リージョンごとに 1 つの DNS フォワーダーで十分ですが、静的近接性の場合、仮想ネットワークごとに 1 つの DNS フォワーダーが必要です。

1. フォワーダーをデプロイした後、仮想ネットワークAのDNSサーバー設定を、以下の画像に示すように、VNet A DNSフォワーダーIPを使用してデフォルトからカスタムに変更します。
2. VNet A DNSフォワーダーの `named.conf.options` ファイルを変更して、ドメイン (mysite.net) とサブドメイン (ptm.mysite.net) の転送ルールをNetScaler GSLBのADNS IPに追加します。

3. ファイル `named.conf.options` で行われた変更を反映するために、DNSフォワーダーを再起動します。

VNet A DNSフォワーダーの設定

```
zone "mysite.net" {  
  
    type forward;  
  
    forwarders { 168.63.129.16; };  
  
};  
  
zone "ptm.mysite.net" {  
  
    type forward;  
  
    forwarders { 10.8.0.5; };  
  
};
```

注:

ドメイン（「mysite.net」）ゾーンのIPアドレスには、AzureリージョンのDNS IPアドレスを使用します。サブドメイン（「ptm.mysite.net」）ゾーンのIPアドレスには、GSLBインスタンスのすべてのADNS IPアドレスを使用します。

クライアントがVNet Bにある場合、AppサーバーとDNSフォワーダーをデプロイする

1. 仮想ネットワークBの場合、同じダッシュボードを選択し、**リソースの作成**をクリックします。
2. 該当するインスタンスを検索し、VNet BサブネットからIPを割り当てます。
3. VNet Aと同様に静的近接GSLBロードバランシングがある場合、AppサーバーとDNSフォワーダーを起動します。
4. VNet B DNSフォワーダーの設定を、以下の設定に示すように `named.conf.options` で編集します。

VNet B DNSフォワーダーの設定:

```
zone "ptm.mysite.net" {  
  
    type forward;  
  
    forwarders { 10.8.0.5; };  
  
};
```

以下の画像は、VNet B DNSフォワーダーの設定を示しています:



オンプレミスにAppサーバー、DNSフォワーダー、Windows 10 Proクライアントをデプロイする

1. オンプレミスの場合、ベアメタルでVMを起動し、VNet Aと同様にAppサーバー、DNSフォワーダー、Windows 10 Proクライアントを導入します。
2. 次の例に示すように、`named.conf.options` でオンプレミスDNSフォワーダー設定を編集します。

オンプレミスDNSフォワーダー設定

```
zone "mysite.net" {  
  
    type forward;  
  
    forwarders { 10.8.0.6; };  
  
};  
  
zone "ptm.mysite.net" {  
  
    type forward;  
  
    forwarders { 10.8.0.5; };  
  
};
```

`mysite.net` の場合、AzureプライベートDNSゾーンサーバーIPの代わりにVNet AのDNSフォワーダーIPを指定しました。これはオンプレミスから到達できない特別なIPアドレスであるためです。したがって、オンプレミスのDNSフォワーダー設定でこの変更が必要です。

Azure仮想ネットワークでNetScalerを構成する

トポロジに示すように、Azure仮想ネットワーク（この場合はVNet A）にNetScalerを展開し、NetScaler GUIを介してアクセスします。

NetScaler GSLBの構成

1. ADNSサービスを作成します。
2. ローカルサイトとリモートサイトを作成します。
3. ローカル仮想サーバーのサービスを作成します。
4. GSLBサービスの仮想サーバーを作成します。

ADNSサービスの追加

1. NetScaler GUIにログインします。
2. **Configuration**タブで、**Traffic Management > Load Balancing > Services**に移動します。
3. サービスを追加します。次の画像に示すように、ADNSサービスをTCPとUDPの両方で構成することをお勧めします。



GSLBサイトの追加

1. GSLBが構成されるローカルサイトとリモートサイトを追加します。
2. 「**構成**」タブで、「**トラフィック管理 > GSLB > GSLBサイト**」に移動します。次の例に示すようにサイトを追加し、他のサイトについても同じ手順を繰り返します。



GSLBサービスの追加

1. アプリサーバーの負荷分散を行うローカルおよびリモートの仮想サーバーにGSLBサービスを追加します。
2. 「**構成**」タブで、「**トラフィック管理 > GSLB > GSLBサービス**」に移動します。
3. 次の例に示すようにサービスを追加します。
4. サーバーの状態を確認するためにHTTPモニターをバインドします。



5. サービスを作成したら、GSLBサービス内の**Advanced settings**タブに移動します。
6. **Add Monitor**をクリックして、GSLBサービスをHTTPモニターにバインドし、サービスの状態をUPにします。



7. HTTPモニターにバインドすると、次の図に示すように、サービスの状態がUPとしてマークされます。



GSLB仮想サーバーを追加する

アプリサーバーのエイリアスGSLBサービスにアクセスできるGSLB仮想サーバーを追加します。

1. **Configuration**タブで、**Traffic Management > GSLB > GSLB Virtual Servers**に移動します。
2. 次の例に示すように、仮想サーバーを追加します。
3. GSLBサービスとドメイン名をそれにバインドします。



4. GSLB仮想サーバーを作成し、適切な負荷分散方法（この場合はラウンドロビン）を選択したら、GSLBサービスとドメインをバインドしてこの手順を完了します。



5. 仮想サーバー内の**Advanced settings**タブに移動し、**Add Domains**タブをクリックしてドメインをバインドします。
6. **Advanced > Services**に移動し、矢印をクリックしてGSLBサービスをバインドし、3つのサービスすべて（VNet A、VNet B、オンプレミス）を仮想サーバーにバインドします。



GSLBサービスとドメインを仮想サーバーにバインドすると、次の図のように表示されます。

GSLB仮想サーバーの設定(/en-us/vpx/media/image-23.png)

GSLB仮想サーバーが稼働しており、100%正常であることを確認します。モニターがサーバーが稼働しており正常であることを示している場合、サイトが同期しており、バックエンドサービスが利用可能であることを意味します。

GSLB仮想サーバーの負荷分散(/en-us/vpx/media/image-24.png)

展開をテストするには、クラウドクライアントマシンまたはオンプレミスクライアントマシンのいずれかからドメインURL `rr.ptm.mysite.net` にアクセスします。クラウドWindowsクライアントマシンからアクセスする場合は、サードパーティまたはカスタムDNSソリューションを必要

とせずに、オンプレミスAppサーバーがプライベートDNSゾーンでアクセスされ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
- より高いパケット/秒 (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と一致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。

VM にインターフェイスを接続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 可用性セットまたは可用性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、管理インターフェイス、クライアントインターフェイス、およびサーバーインターフェイスごとに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。

検証が成功したら、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に時間がかかる場合があります。



8. デプロイが完了したら、**リソースグループ**を選択し、構成の詳細を確認します。

仮想マシンの作成(/en-us/vpx/media/create-vm.png)

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。各NICのアクセラレーテッドネットワークの状態が**有効**または**無効**として表示されます。

アクセラレーテッドネットワークが有効(/en-us/vpx/media/an-enabled.png)

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。

次の手順を実行して、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します。

1. **Azure portal**に移動し、右上隅にある**PowerShell**アイコンをクリックします。

注:

Bashモードの場合は、PowerShellモードに切り替えてください。

アジュール PowerShell(/en-us/vpx/media/azure-powershell.png)

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ーは、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キングを有効にします。
- **False:** 指定されたNICでアクセラレーテッドネットワーキングを無効にします。

特定のNICでアクセラレーテッドネットワーキングを有効にするには：

```
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キングを無効にするには：

```
az network nic update --name citrix-adc-vpx-nic01-0 --accelerated-networking false --resource-group rsgp1-aan
```

3. デプロイ完了後にアクセラレーテッドネットワーキングのステータスを確認するには、**VM > Networking**に移動します。

以下の例では、アクセラレーテッドネットワーキングが**有効**になっていることがわかります。



以下の例では、アクセラレーテッドネットワーキングが**無効**になっていることがわかります。



NetScaler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キングを確認するには

NetScalerのFreeBSDシェルにログインし、以下のコマンドを実行してアクセラレーテッドネットワーキングのステータスを確認できます。

ConnectX3 NICの例：

以下の例は、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：

以下の例は、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上のアクセラレーテッドネットワーキングを確認するには

ConnectX3 NICの例:

以下のshow interfaceコマンドの出力は、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の両方のNICのMACアドレスは同じです。アクセラレーテッドネットワーキングが有効になると、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:

以下のshow interfaceコマンドの出力は、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の両方のNICのMACアドレスは同じです。アクセラレーテッドネットワーキングが有効になると、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における注意点

- PVインターフェースは、すべての必要な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
 - 統計情報をクリア

注:

Citrix®は、VFインターフェースに対して操作を行わ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キングにおける動的な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キングのSR-IOV VF NICを削除できます。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「Link Down」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インターフェイスにVLANを構成する

PVインターフェイスがVLANにバインドされると、関連付けられたアクセラレーテッドVFインターフェイスも、PVインターフェイスと同じVLANにバインドされます。この例では、PVインターフェイス (1/1) はVLAN (20) にバインドされています。PVインターフェイス (1/1) とバンドルされているVFインターフェイス (100/1) もVLAN 20にバインドされます。

例:

1. VLANを作成します。

```
add vlan 20
```

2. PVインターフェイスにVLANをバインドします。

```
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を使用したNetScaler®高可用性テンプレートで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して、HA-INCモードの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。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。各サブネットは、各デバイス上の異なるNICに属し、管理、クライアント、およびサーバー側のトラフィックに使用されます。

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、このデプロイを使用できます。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

テンプレートを起動し、Azure Availability Setsを使用して高可用性VPXペアをデプロイするには、次の手順を完了します。

1. Azure portalから、**カスタムデプロイページ**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 portal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、追加のセキュリティルールやポートの作成などのさらなる変更が必要な場合は、Azure portalから実行できます。

必要な構成が完了すると、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が、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と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展開をサポートするには、Mac-Based Forwarding (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、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。
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**」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。



- 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を使用し、変更を保存します。

注:

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ています。



ADCにさらに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 > フロントエンドIP構成**に移動し、**追加**をクリックして新しい内部ロードバランサーIPアドレスを作成します。



2. **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。

新しいロードバランシング規則を作成する(/en-us/vpx/media/ilb-front-end-ip2.png)

4. **Azure Load Balancer** ページで、**ロードバランシング規則**を選択し、**追加**をクリックします。

LB規則を追加する(/en-us/vpx/media/add-lb-rules.png)

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規則を作成します。**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。

フローティングIPが有効(/en-us/vpx/media/floating-ip-enabled.png)

6. これで、**フロントエンドIP構成**に適用されたLB規則が表示されます。

LB規則を適用する(/en-us/vpx/media/rules-count.png)

構成例

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 vsrver vpn_ssl SSL 10.11.1.4 443

add ssl certKey ckp -cert wild-cgwsanity.cer -key wild-cgwsanity.key

bind ssl vsrver vpn_ssl -certkeyName ckp
```

ロードバランシング構成例

```
enable feature LB SSL

enable ns mode MBF

add lb vsrver lb_vs1 SSL 10.11.1.7 443

bind ssl vsrver 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、ロードバランシングまたは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:

HA展開と仮想サーバー構成に関連する追加情報は、次のリンクを参照してください。

- 異なるサブネットで高可用性ノードを構成する([/ja-jp/citrix-adc/current-release/system/high-availability-introduction/configuring-high-availability-nodes-different-subnets.html](#))
- 基本的な負荷分散を設定する([/ja-jp/citrix-adc/current-release/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/current-release/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/citrix-adc/current-release/deploying-vpx/deploy-vpx-on-azure/configuring-gslb-active-standby-ha-deployment-azure.html](#))

インターネットに接続するアプリケーション向けの NetScaler®高可用性テンプレートを使用してHA-INCノードを構成する

インターネットに接続す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ことで、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。

インターネットに接続す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、次の手順を完了します。

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。



5. ****次へ: VM構成 >****をクリックします。



6. **VM構成**ページで、次の操作を実行します。

- パブリックIPドメイン名サフィックスを構成する
- **Azure Monitoring Metrics** を有効または無効にする
- **Backend Autoscale™** を有効または無効にする

7. **次へ: ネットワークと追加設定 >** をクリックします。



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。



9. **次へ: レビュー + 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 **citrix-adc-vpx-0** と **citrix-adc-vpx-1** として表示されます。

セキュリティルールやポートの追加作成など、HAセットアップに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。

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。



11. 以下の構成を検証するには、**citrix-adc-vpx-0** および **citrix-adc-vpx-1** ノードにログオン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) ノードには、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プロンプト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、従来のオンプレミスADC高可用性展開で両方とも同じであるのとは異なり、citrix-adc-vpx-0および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

ALBのプライマリノードでIP CLIを表示(/en-us/vpx/media/primary-node1-alb.png)

ALBのプライマリノードでHAノードCLIを表示(/en-us/vpx/media/primary-node2-alb.png)

セカンダリノード (citrix-adc-vpx-1) で

ALBのセカンダリノードでIP CLIを表示(/en-us/vpx/media/secondary-node1-alb.png)

ALBのセカンダリノードでHAノードCLIを表示(/en-us/vpx/media/secondary-node2-alb.png)

12. プライマリノードとセカンダリノードがUPになり、同期ステータスが**SUCCESS**になったら、ALB仮想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。

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。

ALBフロントエンドIP構成(/en-us/vpx/media/frontend-ip-config-alb.png)

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の受信セキュリティルールを追加します。

受信セキュリティルール1(/en-us/vpx/media/inbound-security-rule1.png)

受信セキュリティルール2(/en-us/vpx/media/inbound-security-rule2.png)

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受信セキュリティルールを作成します。バックエンドポートは、負荷分散仮想サーバーポートまたはVPN仮想サーバーポートです。



16. これで、ALBパブリックIPアドレスに関連付けられた完全修飾ドメイン名 (FQDN) を使用して、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。



サンプル構成

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。

ゲートウェイのサンプル構成

```
enable feature aaa LB SSL SSLVPN

add ip 52.172.55.197 255.255.255.0 -type VIP

add vpn vserver vpn_ssl SSL 52.172.55.197 443

add ssl certKey ckp -cert cgwsanity.cer -key cgwsanity.key

bind ssl vserver vpn_ssl -certkeyName ckp
```

ロードバランシングのサンプル構成

```
enable feature LB SSL

enable ns mode MBF

add lb vserver lb_vs1 SSL 52.172.55.197 443

bind ssl vserver lb_vs1 -certkeyName ckp
```

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、外部および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。

手順1と2ではAzureポータルを使用します。手順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「内部」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アドレスを指定することもできます。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「パブリック」を選択し、ここでパブリックIPアドレスを作成します。



1. Azure Load Balancer を作成したら、**Frontend IP configuration** 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 ロードバランシング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)

2. 「**Azure Load Balancer configuration**」 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。
3. 高可用性ペアのクライアント NIC を ILB のバックエンドプールに追加します。
4. ヘルスプローブを作成します (TCP、ポート 9000)。
5. 2 つのロードバランシング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 用のロードバランシング仮想サーバーを作成します。

1. ロードバランシング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. サービスをロードバランシング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、「基本ロードバランシングのセットアップ」([/ja-jp/citrix-adc/current-release/load-balancing/load-balancing-setup.html](#))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用のロードバランシング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」([/ja-jp/citrix-adc/current-release/load-balancing/load-balancing-customizing/multi-ip-virtual-servers.html#configure-a-multi-ip-virtual-server](https://docs.citrix.com/en-us/citrix-adc/current-release/load-balancing/load-balancing-customizing/multi-ip-virtual-servers.html#configure-a-multi-ip-virtual-server))を参照してください。

セカンダリIP構成を使用してAzureにNetScaler VPX HAペアを展開する

NetScaler VPXの高可用性ペアをAzureに展開する場合、両方のNetScaler VPXインスタンスを同じ仮想ネットワーク (VNet) 上に配置し、ネットワークインターフェイス (NIC) のセカンダリIP構成を使用できます。このHA展開により、Azure Load Balancer (ALB) は不要になります。フェイルオーバー中、プライマリノードのクライアント側およびサーバー側NICに割り当てられたセカンダリプライベートIPアドレスはセカンダリノードに移行されます。これらの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パブリックIPアドレスも、それに応じて移動されます。

重要:

展開を開始する前に、セットアップ中の問題を回避するために、Azure環境が[前提条件](#)を満たしていることを確認してください。

セカンダリIP構成を使用する利点は次のとおりです。

- **簡素化されたフェイルオーバー:** セカンダリIPアドレスにより、仮想IPアドレス (VIP) とサブネットIPアドレス (SNIP) がHAノード間で自動的に移行され、外部ロードバランシングを必要とせずにシームレスなフェイルオーバーが保証されます。
- **スケーラビリティ:** 複数のIPアドレスを単一のNICに割り当てることができ、同じVM内で柔軟なトラフィックルーティングとサービスのスケーリングを可能にします。
- **回復性:** セカンダリプライベートIPアドレスをノード間で移動できるため、ネットワークの可用性が向上し、ノード障害発生時の迅速な回復が可能です。
- **コスト効率:** ALBが不要になることで、インフラストラクチャコストを削減しながら、同じサブネット内で高可用性とトラフィック管理を維持できます。

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。



AzureセカンダリIP構成を使用してNetScaler VPX™ HAペアを展開するための前提条件

- NetScalerバージョン14.1–34.42以降を使用してください。
- サブスクリプションで新しいAzureコントロールプレーンが有効になっていることを確認してください。

注:

Azureの新しいコントロールプレーンがサブスクリプションで有効になっていない場合、高可用性デプロイメントではフェイルオーバー期間が長くなる可能性があります。そのような場合は、新しいコントロールプレーンを有効にするための支援についてAzureサポートにお問い合わせください。

AzureのセカンダリIP構成を使用してNetScaler VPX HAペアをデプロイする方法

AzureのセカンダリIP構成を使用してNetScaler VPX HAペアをデプロイするには、次の手順に従います。

1. Azureに2つのNetScaler VPXインスタンスをデプロイします。各インスタンスは、同じリソースグループとVNet内に3つのネットワークインターフェイスを持ちます。
2. 両方のNetScaler VPXインスタンスにマネージドIDを割り当てます。

3. プライマリノードのクライアントおよびサーバーネットワークインターフェイスにAzureセカンダリIP構成を割り当てます。
4. セカンダリIP構成からのAzureプライベートIPアドレスを使用して、プライマリノードでVIPとSNIPを構成します。
5. プライマリノードで、プライマリおよびセカンダリの両方のNetScalerインスタンスのサーバーネットワークインターフェイスのプライマリプライベートIPアドレスをVIPとして構成します。
6. 両方のノードでHAを構成します。

ステップ1. 同じリソースグループとVNetに2つのNetScaler VPXインスタンス（プライマリノードとセカンダリノード）をデプロイします。各NetScaler VPXインスタンスに3つのNIC（Ethernet 0、Ethernet 1、Ethernet 2）があることを確認してください。

詳細な手順については、「[Azureに2つのVPXインスタンスをデプロイする](#)」を参照してください。

ステップ2. システム割り当てまたはユーザー割り当てのいずれかのマネージドIDを両方のNetScaler VPXインスタンスに適用します。

注:

この機能ではAzureサービスプリンシパルはサポートされていません。

詳細については、「[AzureリソースのマネージドID](#)」を参照してください。

NetScaler VPX でマネージド ID を構成する方法については、「[仮想マシンでマネージド ID を構成する](#)」を参照してください。

ステップ 3. NetScaler VPX インスタンスに関連付けられているマネージド ID に、次のいずれかのAzure ロール割り当てを追加します。

- **閲覧者とネットワーク共同作成者:** Azure リソースへの読み取り専用アクセスと、ネットワークリソースを管理するために必要なアクセス許可を付与します。
- **共同作成者:** Azure リソースへのフルアクセスを提供します。

ステップ 4. プライマリノードで、イーサネット 1 (クライアント IP または VIP) とイーサネット 2 (バックエンドサーバー IP または SNIP) にプライベート IP アドレスを割り当てます。

Azure ポータルは、構成された 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 および SNIP ネットワークインターフェイスにさらにプライベート IP アドレスを割り当てるには、セカンダリ IP 構成を使用します。

Azure のネットワークインターフェイスにセカンダリ IP 構成を割り当てるには、次の手順に従います。

1. **Azure Portal** にログインし、**Virtual Machines** に移動します。
2. ネットワークインターフェイスに関連付けられている仮想マシンを選択します。
3. 仮想マシンの設定で、**Network settings** に移動し、セカンダリプライベート IP アドレスを追加する NIC を選択します。
4. ネットワークインターフェイスの設定で、**IP configurations** タブをクリックします。
5. **Add** をクリックして、タイプが **Secondary** の IP 構成を割り当てます。
 - 静的割り当て方法を選択した場合は、インスタンスのサブネット範囲内で特定の IPv4 アドレスを入力します。
 - 動的割り当て方法を選択した場合、Azure は IP アドレスを自動的に割り当てます。
6. **Add** または **Save** をクリックして変更を適用します。

ステップ 5. セカンダリ IP 構成のプライベート IP アドレスを使用して、プライマリ ノードで VIP と SNIP を構成します。

1. SSH を使用してプライマリ ノードにアクセスします。SSH クライアントを開き、次のコマンドを実行します。

```
ssh -i <location of your private key> nsroot@<public DNS of the instance>
```

2. VIP および SNIP アドレスを構成します。

VIP アドレスを構成するには、次のコマンドを実行します。

```
add ns ip <IPAddress> <netmask> -type VIP
```

例:

```
add ns ip 192.0.2.11 255.255.255.0 -type VIP
```

SNIP アドレスを構成するには、次のコマンドを実行します。

```
add ns ip <IPAddress> <netmask> -type SNIP
```

例:

```
add ns ip 192.0.3.11 255.255.255.0 -type SNIP
```

3. `save config` コマンドを実行して構成を保存します。
4. 次のコマンドを使用して、構成された IP アドレスを確認します。

```
show ns ip
```

ステップ 6. プライマリ ノードで、プライマリとセカンダリの両方の NetScaler インスタンスのサーバーネットワークインターフェイスのプライマリプライベート IP アドレスを VIP に構成します。

1. SSH を使用してプライマリ ノードにアクセスします。SSH クライアントを開き、次のコマンドを実行します。

```
ssh -i <location of your private key> nsroot@<public DNS of the instance>
```

例:

```
ssh -i ~/.ssh/mykey.pem nsroot@primary-instance.eastus.cloudapp.azure.com
```

2. プライマリ ノードで、プライマリとセカンダリの両方の NetScaler インスタンスのイーサネット 2 (バックエンドサーバー NIC) のプライマリ プライベート IP アドレスを VIP に構成します。

```
add ns ip <private IPAddress of primary instance> <netmask> -type VIP
```

```
add ns ip <private IPAddress of secondary instance> <netmask> -type VIP
```

例:

```
add ns ip 192.0.3.10 255.255.255.0 -type VIP
```

```
add ns ip 192.0.3.20 255.255.255.0 -type VIP
```

ステップ 7. 両方のノードで HA を構成します。

1. プライマリノードで、シェルクライアントを開き、次のコマンドを実行します。

```
add ha node <peer node id> <private IP address of the management NIC of  
the secondary node>
```

例:

```
add ha node 1 192.0.1.20
```

2. セカンダリノードで、次のコマンドを実行します。

```
add ha node <peer node id> <private IP address of the management NIC of  
the primary node>
```

例:

```
add ha node 1 192.0.1.10
```

3. `save config` コマンドを実行して、構成を保存します。
4. `show ha node` コマンドを実行して、構成されたHAノードを確認します。

フェイルオーバー中、VIPおよびSNIPネットワークインターフェースのセカンダリIP構成は、以前のプライマリノードから新しいプライマリノードに移動されます。

ソリューションテンプレートを使用してAzureにNetScaler VPX HAペアを展開する

Azureソリューションテンプレートは、NetScaler VPXの高可用性 (HA) 展開を自動化し、展開時間を短縮し、AzureとNetScalerの両方での手動構成を不要にします。顧客はこれらのソリューションテンプレートを使用して、最小限の労力で本番環境に対応したHAソリューションを展開できます。

このソリューションは、高可用性とフォールトトレランスのために、異なる可用性ゾーンまたは可用性セットに2つのNetScaler VPXインスタンスを展開します。

ソリューションテンプレートの詳細については、[Microsoft ドキュメント](#) を参照してください。

NetScaler VPX Azureソリューション展開の主な機能

- **高可用性:** 自動HA構成とピアノード検出により、2つのNetScaler VPXインスタンスを別々のAzure可用性ゾーンに展開します。
- **ネットワーク:**
 - **3-NIC展開:** ネットワークセグメンテーションのために、専用の管理、クライアント、およびサーバーサブネットを利用します。
 - **仮想ネットワークのサポート:** 新規および既存の仮想ネットワークの両方で動作します。
 - **パブリック IP アドレスの構成:** 管理トラフィックとクライアントトラフィックの両方で構成可能なパブリック IP アドレスを許可します。
 - **アクセラレーテッド ネットワーク:** アクセラレーテッド ネットワークを有効または無効にするオプション。
- **セキュリティとコンプライアンス:**
 - **ネットワーク セキュリティ グループ (NSG):** セキュリティを強化するためのカスタマイズ可能な受信ルールが含まれています。
 - **RBAC 統合:** システム割り当てマネージド ID を介した自動ロール割り当て。

前提条件

開始する前に、以下があることを確認してください。

- 管理者権限を持つアクティブな Azure サブスクリプション。
- 可用性ゾーンをサポートする Azure リージョン。
- NetScaler ライセンス ファイル (Bring Your Own License)。

マーケットプレイス オファー情報 (Bring Your Own License - BYOL)

ソリューション テンプレートを使用して Azure に NetScaler VPX をデプロイする場合は、Bring Your Own License (BYOL) モデルを使用します。

- オファー ID: **netscalervpx-141**
- SKU/プランID: **netcalerbyol**
- 発行元: Citrix

ソリューションテンプレートを使用したAzureへのNetScaler VPX HAペアの展開

このセクションでは、Azure Marketplaceポータルを使用してNetScaler VPXを展開する方法について説明します。

ステップ1. Azure Marketplaceから展開します。

1. [Azureポータル](#)にサインインします。
2. Azure Marketplaceで**NetScaler VPX**を検索し、展開オプションを見つけます。



ステップ2. 展開プランを選択します。

可用性要件に基づいて、以下のいずれかのプランを選択します。

- NetScaler VPX HA (可用性ゾーン) セカンダリIP経由 - BYOL
- NetScaler VPX HA (可用性セット) セカンダリIP経由 - BYOL



ステップ3. 基本設定と管理者資格情報を構成します。

- サブスクリプション - Azureサブスクリプションを選択します。
- リソースグループ - 新しいリソースグループを作成するか、既存のものを選択します。
- リージョン - 可用性ゾーンまたは可用性セットをサポートするAzureリージョンを選択します。
- VM名プレフィックス - NetScaler VPX仮想マシンインスタンスのプレフィックスを入力します。
- リリースバージョン - 希望するNetScaler VPXバージョンを指定します (例: 14.1)。
- 管理者ユーザー名 - 管理者ユーザー名を入力します。
- 認証タイプ - パスワードまたはSSH公開キーを選択します。
- 資格情報 - 選択した認証タイプに基づいて、パスワードまたはSSHキーを入力します。



ステップ4. 仮想マシン設定を構成します。

- **VMサイズ** - パフォーマンス要件を満たすサイズを選択します。
- **OSディスクタイプ** - ストレージディスクタイプを選択します（例: Standard_LRSまたはPremium_LRS）。
- **監視メトリックの公開** - このオプションを有効にすると、NetScaler ADCのパフォーマンスメトリックがAzure Monitorに送信されます。



ステップ5. ネットワーク設定を構成します。

- **仮想ネットワーク** - 新しい仮想ネットワークを作成するか、既存のものを選択します。
- **アドレス空間** - 新しい仮想ネットワークのCIDRブロックを指定します。
- **サブネット** - 管理、クライアント、およびサーバーのサブネットを構成します。
- **アクセラレーテッド ネットワーク** - 主要なインターフェースでネットワークパフォーマンスを向上させるために有効にします。
- **管理パブリック IP アドレス** - VM 1 と VM 2 にパブリック IP アドレスを割り当てます (推奨)。
- **クライアントパブリック IP アドレス** - 必要に応じて、クライアントアクセス用のパブリック IP アドレスを割り当てます。
- **DNS ラベル** - DNS 解決が必要な場合は、一意の DNS ラベルを追加します。
- **パブリックインバウンドポート** - 必要なポートを許可します (例: SSH-22、HTTP-80、HTTPS-443)。



ステップ6. デプロイを確認して完了します。

1. 選択したすべての構成パラメータを確認し、要件を満たしていることを確認します。
2. 利用規約に同意します。
3. 「作成」を選択し、デプロイを開始します。



ステップ7. デプロイを検証します。

デプロイが完了したら、NetScaler HA ペアが正しく構成され、期待どおりに動作していることを確認します。

1. HAノードのステータスを確認します。

- a. SSHを使用してNetScalerインスタンスのいずれかに接続します。
- b. HAノードのステータスを確認するには、`show ha node` コマンドを実行します。
- c. 以下を確認します。
 - 両方のノードが出力に表示されていること
 - 同期 (Sync) ステータスが正常であること
 - INCモード (Independent Network Configuration) が正しく構成されていること



2. IPアドレスの構成を検証します。

NetScalerインスタンスで `show ns ip` コマンドを実行します。出力結果を確認し、すべての仮想IPアドレス (VIP) とサブネットIPアドレス (SNIP) が存在し、正しく構成されていることを確認します。



3. 高可用性フェイルオーバーテストを実行します。

HAセットアップの回復性を検証するために、手動フェイルオーバーテストを実行することをお勧めします。

- a. フェイルオーバーを開始するには、`force ha failover -force` コマンドを実行します。
- b. 以下を確認します。
 - フェイルオーバーが正常に完了します
 - すべてのセカンダリIP構成が新しいプライマリノードに転送されます

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) およびサーバー 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 アーキテクチャの図 (/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 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 Jumpbox 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 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、[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。

- 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されている必要があります。

VMware クラウド 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。
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は、すべての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インストールのシステム要件

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（推奨1 GHz以上）
RAM	1 GB以上（推奨2 GB）
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 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。

ESX サーバーと同じネットワーク上にあるワークステーションに、次のファイルをコピーします。3 つのファイルすべてを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-[\<リリース番号>\<ビルド番号>.ovf](#) (例えば、NSVPX-ESX-13.0-79.64.ovfのような形式です)
- 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 を展開する

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 台の 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）やゲートウェイサブネットと重複しないことを確認してください。
- 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エクスプレスルート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. 「**確認 + 作成**」をクリックします。

5. 設定を確認します。設定を変更する必要がある場合は、「**前へ**」をクリックします。

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:

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上の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の動作を確認するには、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 ポータルへのアクセス

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 の資格情報をメモしておきます。

vCenter の資格情報(/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 portal 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 portalでNSX-Tセグメントを作成する

Azure portal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およびvCenterに表示されます。

1. Azure VMware Solutionプライベートクラウドで、**Workload Networking**の下にある **Segments > Add**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、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、**Manage**の下にある**Identity**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クライアントのURLを入力して、VMware NSX-T Managerを起動します。



4. NSX-T Managerで、**Networking > Segments**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。

ネットワークセグメント(/en-us/vpx/media/networking-segments.png)

詳細については、[Create an NSX-T segment in the Azure portal](#)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**フィールドと**パスワード**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するようにします。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。

ネットワークインターフェイスの追加(/en-us/vpx/media/avs-add-network-interfaces.png)

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

インターネットに公開されるアプリケーション向けに、Azure VMware Solution (AVS) 上で 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 NetScaler VPX インスタンスの VIP にアクセスするようにします。

AVS 上の VPX スタンドアロン展開(/en-us/vpx/media/vpx-standalone-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 documentation](#) を参照してください。
- Azure VMware Solution の展開の詳細については、[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 Jump box 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 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、[Add a network segment in Azure VMware Solution](#) を参照してください。
- VMware クラウドに NetScaler VPX インスタンスをインストールする方法の詳細については、[Install a NetScaler VPX instance on VMware cloud](#) を参照してください。

NetScaler ロードバランサーを使用して、AVS 上で 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、インターネットに公開されるアプリケーション向けに AVS 上で NetScaler VPX スタンドアロンインスタンスを構成するには、以下の手順に従います。

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、[Configure a NetScaler VPX standalone instance](#) 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 上で 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 上で NetScaler VPX スタンドアロンインスタンスを構成するには、以下の手順に従ってください。

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 上でのNetScaler VPX高可用性セットアップの構成

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内部のプライマリ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上でのVPX HA展開(/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、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 documentation](#)を参照してください。
- Azure VMware Solutionの展開の詳細については、[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 Jump Box 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 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適切なNSX-TネットワークセグメントをVMware SDDC上に作成します。詳細については、[Add a network segment in Azure VMware Solution](#)を参照してください。

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、次の手順に従います。

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、[Install a NetScaler VPX instance on VMware cloud](#)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、[Configuring high availability](#)を参照してください。
3. インターネットに公開されるアプリケーションが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。

Azure ルートサーバーをNetScaler VPX HAペアで構成する

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ルートを交換するために、Azure ルートサーバーをNetScaler VPXインスタンスで構成できます。NetScalerはスタンドアロンモードまたはHA-INCモードで展開し、BGPを構成できます。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。

AzureルートサーバーとのVPX HAトポロジ(/en-us/vpx/media/ars-ha-topology.jpg)

トポロジ図では、次のIPアドレスを使用します。

プライマリADCインスタンスのIP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ネットワークング](#)を参照してください。

AzureルートサーバーをNetScaler VPX HAペアで構成する方法

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 とのピアリングを設定する](#)」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。

次の手順を完了します。

- a. Azure に 2 つの VPX インスタンス (プライマリおよび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およびサーバー 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 構成の確認](#)」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仮想サーバーを構成します。

構成例:

```
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。

バックエンドの Azure 自動スケーリングサービスを追加する

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 使用率などを常に監視する必要があります。しかし、トラフィックの手動監視は面倒です。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、必要に応じてトラフィックの監視とリソースのスケールアップ/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) とともに 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、異なる仮想ネットワーク (VNet) 上のバックエンドサーバーをサポートします。



詳細については、次の Azure トピックを参照してください。

- [仮想マシン スケールセットのドキュメント](#)

- [Microsoft Azure Virtual Machines、Cloud Services、および Web Apps におけるオートスケールの概要](#)

開始する前に

- 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。
- 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ースを持つ 1 つ以上の NetScaler VPX インスタンスを作成します。
- VPX インスタンスの 0/1 インターフェー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。
- 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、以下のタスクを完了してください。

- a) VMSS を作成する
- b) VMSS でオートスケールを有効にする
- c) VMSS オートスケール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する

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。

- NetScaler VPX は、Uniform orchestration を使用する VMSS のみをサポートします。Flexible orchestration を使用する VMSS はサポートされていません。詳細については、「[Azure の仮想マシン スケール セットのオーケストレーション モード](#)」を参照してください。
- NetScaler リリース 14.1-12.x 以降、NetScaler VPX は Azure クラウドでのマネージド ID をサポートしています。マネージド ID は、サービスプリンシパルを仮想マシンなどの Azure リソースにリンクします。マネージド ID を使用すると、クラウド資格情報（アプリケーション ID、アプリケーションシークレット、テナント ID）を管理する必要がなくなり、セキュリティリスクを回避できます。現在、NetScaler VPX はシステム割り当ておよび単一ユーザー割り当

てのマネージド ID のみをサポートしています。複数ユーザー割り当てのマネージド ID はサポートされていません。

14.1-12.x より前の NetScaler リリースでは、Azure Active Directory (AAD) を介して NetScaler VPX のクラウド資格情報を手動で管理する必要があります。新しく作成された 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。クラウド資格情報は、有効期限が切れた後に定期的に再作成する必要があります。詳細については、「[Azure Active Directory アプリケーションとサービスプリンシパルを作成する](#)」を参照してください。

Azure コンソールでマネージド ID を構成し、NetScaler® でクラウド資格情報を構成する場合、マネージド ID がクラウド資格情報よりも優先されます。

仮想マシンでマネージド ID を構成する

1. Azure ポータルにサインインします。
2. 仮想マシンに移動し、 を選択します。
3. 要件に基づいて、システム割り当てまたはユーザー割り当てのいずれかを選択します。
4. ステータスでオンを選択し、保存をクリックします。

マネージドIDをオンにする(/en-us/vpx/media/azure-mi-on.png)

ステータスが保存されると、サービスプリンシパルオブジェクトが作成され、VMに割り当てられていることがわかります。

5. **Azure ロールの割り当て**をクリックします。

Azure ロールの割り当て(/en-us/vpx/media/azure-mi-on-service-principal.png)

6. **ロールの割り当ての追加**ウィンドウでスコープを選択します。次のオプションから選択できます。

- サブスクリプション

VMSSとVMが異なるリソースグループにある場合、スコープとしてサブスクリプションを使用します。

- リソースグループ

VMSSがVMと同じリソースグループにある場合、スコープとしてリソースグループを使用します。

- キー コンテナー
- ストレージ

- SQL

スコープの選択に基づいて、他のフィールドの詳細を入力します。**共同作成者**ロールを割り当て、構成を**保存**します。



Azure ロールの割り当てページに、作成したマネージドIDが表示されます。



7. ユーザー割り当てマネージドIDを作成するには、サブスクリプションを選択し、ユーザー割り当てマネージドIDを選択して、**追加**をクリックします。



VMSSをNetScaler VPXインスタンスに追加する

VPXインスタンスにオートスケール設定を追加するには、次の手順を実行します。

1. VPXインスタンスにログオンします。
2. **構成 > Azure > 資格情報の設定**に移動します。オートスケール機能が動作するために必要なAzure資格情報を追加します。



注記:

Azure マネージドIDを使用している場合、資格情報を設定する必要はありません。

3. **システム > Azure > クラウドプロファイル**に移動し、**追加**をクリックしてクラウドプロファイルを作成します。



クラウドプロファイルの作成構成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（サーバー）をオートスケーリンググループの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。

クラウドプロファイル作成時に留意すべき点

- 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 アドレスから自動的に入力されます。詳細については、「[Azure portal 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。
- オートスケール設定プロトコルとオートスケール設定ポートを選択する際は、サーバーがプロトコルとポートをリッスンしていることを確認し、サービスグループに正しいモニターをバインドしてください。デフォルトでは、TCP モニターが使用されます。
- SSL プロトコルタイプの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シング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。

注:

NetScaler リリース 13.1-42.x 以降、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。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。

Azure portal でオートスケール関連情報を表示するには、**仮想マシン スケール セット**に移動し、**仮想マシン スケール セット > スケーリング**を選択します。

Azure でのアンマネージドディスクからマネージドディスクへの移行

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展開されている場合、Azure クラウドの変更に対応するために、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](#)
- [よくある質問](#)

参照

NetScaler Application Delivery and Management を使用した Microsoft Azure での NetScaler VPX のオート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 オートスケーリング](#) を参照してください。

NetScaler VPX デプロイメントの Azure タグ

Azure クラウドポータルでは、リソースグループ間、およびポータル内でサブスクリプション間でリソースを分類および表示するために、リソースに名前:値のペア（例：Dept: Finance）でタグ付けできます。タグ付けは、課金、管理、または自動化のためにリソースを整理する必要がある場合に役立ちます。

VPX デプロイメントにおける Azure タグの仕組み

Azure Cloud にデプロイされた NetScaler VPX のスタンドアロンインスタンス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ロードバランシング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、対応するタグを持つ Azure 仮想マシン（バックエンドサーバー）およびネットワークインターフェース（NIC）、またはその両方を常に監視し、それに応じてサービスグループを更新します。

VPX インスタンスは、タグを使用してバックエンドサーバーの負荷分散を行うサービスグループを作成します。インスタンスは、特定のタグ名とタグ値でタグ付けされたすべてのリソースについて Azure API にクエリを実行します。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、サービスグループから VM または NIC の IP アドレスを自動的に追加または削除します。



始める前に

NetScaler ロードバランシング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分散を行うことができます。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. クラウドプロファイルを作成するには、「追加」をクリックします。クラウドプロファイルウィンドウが開きます。

Azureタグの追加(/en-us/vpx/media/add-azure-tag-settings.png)

1. 次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。

ロードバランサー仮想サーバーとサービスグループは、タグ付けされたVMまたはNIC用に作成されます。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  
-serviceName `<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:

すべての設定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設定が失われます。 `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トラフィック用のクラウドプロファイルのサンプルコマンドを次に示します。

```
add cloud profile MyTagCloudProfile -type azuretags -vServerName  
MyTagVServer -serviceType HTTP -IPAddress 40.115.116.57 -port 80  
-serviceName MyTagsServiceGroup -boundServiceGroupSvcType HTTP  
-vsrvbindsvcpport 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表示します。

```
show cloudprofile
```

```
1)   Name:  MyTagCloudProfile Type:  azuretags           VServerName:
      MyTagVServer ServiceType:  HTTP           IPAddress:  52.178.209.133 Port:
      80           ServiceGroupName:  MyTagsServiceGroup
      BoundServiceGroupSvcType:  HTTP

      Vsvrbindsvcport:  80      AzureTagName:  myTagName AzureTagValue:
      myTagValue AzurePollPeriod:  60      GraceFul:  NO           Delay:  60
```

クラウドプロファイルを削除するには、「rm cloud profile `<cloud profile name>`」と入力します。

例3: 次のコマンドは、例1で作成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場合があります。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終了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もrain_tagsプロセスがHAペアノードで実行され続けます。

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終了します。

```
#> shell
```

以下のコマンドを使用して、rain_tagsプロセスを強制終了します。

```
# 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。一方、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、あるいは障害が発生した場合は稼働中のデータセンターに誘導することで、データセンター間の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つのステップがあります。

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、「[PowerShellコマンドを使用してスタンダードアロンモードの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。

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が構成された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）に実装されたGSLBセットアップ。



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。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

以下は、このドキュメントにおけるこのシナリオのパラメータ設定例です。

```
$location="West Central US"
```

```
$vnetName="NSVPX-vnet"
```

```
$RGName="multiIP-RG"
```

```
$prmStorageAccountName="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に従ってください。

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を作成する(/ja-jp/vpx/current-release/deploy-vpx-on-azure/configuring-gslb-on-citrix-adc-vpx-appliance.html#create-nics)
7. VM構成オブジェクトを作成する(/ja-jp/vpx/current-release/deploy-vpx-on-azure/configuring-gslb-on-citrix-adc-vpx-appliance.html#create-vm-config-object)
8. VMの資格情報を取得し、OSプロパティを設定する(/ja-jp/vpx/current-release/deploy-vpx-on-azure/configuring-gslb-on-citrix-adc-vpx-appliance.html#get-credentials-and-set-os-properties)
9. NICを追加する(/ja-jp/vpx/current-release/deploy-vpx-on-azure/configuring-gslb-on-citrix-adc-vpx-appliance.html#add-nics)
10. OSディスクを指定してVMを作成する(/ja-jp/vpx/current-release/deploy-vpx-on-azure/configuring-gslb-on-citrix-adc-vpx-appliance.html#specify-os-disk-and-create-vm)

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ら、これらの手順を繰り返して、VM2に固有のパラメータでVM2を作成します。

リソースグループを作成する

```
New-AzureRMResourceGroup -Name $RGName -Location $location
```

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 $location
```

可用性セットを作成する

```
$avSet=New-AzureRMAvailabilitySet -Name $avSetName -ResourceGroupName  
$RGName -Location $location
```

仮想ネットワークを作成する

1. サブネットを追加する。

```
$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する。

```
$vnet=New-AzureRmVirtualNetwork -Name $vnetName -ResourceGroupName  
$RGName -Location $location -AddressPrefix 10.0.0.0/16 -Subnet  
$subnet1, $subnet2, $subnet3
```

3. サブネットを取得する。

```
$frontendSubnet=$vnet.Subnets|?{$_.Name -eq $frontendSubnetName}  
  
$backendSubnet1=$vnet.Subnets|?{$_.Name -eq $backendSubnetName1}
```

```
$backendSubnet2=$vnet.Subnets|?{$_.Name -eq $backendSubnetName2}
```

パブリックIPアドレスを作成する

```
$pip1=New-AzureRmPublicIpAddress -Name $pubIPName1 -ResourceGroupName
$RGName -Location $location -AllocationMethod Dynamic

$pip2=New-AzureRmPublicIpAddress -Name $pubIPName2 -ResourceGroupName
$RGName -Location $location -AllocationMethod Dynamic
```

NICを作成する

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

```
$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ディスクを指定してVMを作成

```
$osDiskName=$vmName + "-" + $osDiskSuffix

$osVhdUri=$prmStorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds/" +
$osDiskName + ".vhd"

$vmConfig=Set-AzureRMVMOSDisk -VM $vmConfig -Name $osDiskName -VhdUri
$osVhdUri -CreateOption fromImage

Set-AzureRmVMPlan -VM $vmConfig -Publisher $publisher -Product $offer
-Name $sku

New-AzureRMVM -VM $vmConfig -ResourceGroupName $RGName -Location $location
```

注記:

「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、VM2に固有のパラメータを使用してVM2を作成します。

IP構成の詳細

次のIPアドレスが使用されます。

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
1/1	10.0.1.10	PIP2	SNIP/GSLBサイトIPとして構成
-	10.0.1.11	-	LBサーバーIPとして構成。パブリックIPは必須ではありません

NIC	プライベートIP	パブリックIP (PIP)	説明
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で使用する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) として構成されます
1/1	20.0.1.10	PIP5	SNIP/GSLBサイトIPとして構成されます
-	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以下に示します。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の構成例を以下に示します。

```
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の構成例を以下に示します。

```
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構成例を以下に示します。

```
enable ns feature LB GSLB

add gslb site site1 10.0.1.10 -publicIP PIP2

add gslb site site2 20.0.1.10 -publicIP PIP5

add gslb service site1_gslb_http_svc1 10.0.1.11 HTTP 80 -publicIP PIP3
-publicPort 80 -siteName site1
```

```
add gslb service site2_gslb_http_svc1 20.0.1.11 HTTP 80 -publicIP PIP6  
-publicPort 80 -siteName site2  
  
add gslb vserver gslb_http_vip1 HTTP  
  
bind gslb vserver gslb_http_vip1 -serviceName site2_gslb_http_svc1  
  
bind gslb vserver gslb_http_vip1 -serviceName site1_gslb_http_svc1  
  
bind gslb vserver gslb_http_vip1 -domainName www.gslbindia.com -TTL 5
```

Azureで稼働する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事業機能の突然の停止のことです。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なければなりません。データセンターにおけるデータの損失や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、お客様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わらず、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所有する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にかかる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、より迅速な復旧が可能になります。
- 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、さまざまな種類のコンプライアンスおよびセキュリティ制御を顧客が必要とするユースケースをサポートします。これにより、自社で構築するよりも、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はお客様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Multi-NIC Multi-IP (3-NIC) 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、お客様は3-NIC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Single-NIC Multi-IP (1-NIC) 展開

お客様は、以下の理由により、非本番環境に展開する場合に1-NIC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、迅速かつ効率的にデプロイする。
- 単一サブネット構成のシンプルさを追求しながら。

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サイトでVPX HAペアを作成します。HAペアの作成方法については、「[複数のIPアドレスとNICを使用した高可用性セットアップの構成](#)」を参照してください。
2. 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、Azure Load Balancer (ALB) を構成します。

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サイト用のフロントエンド `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て、TCP 3008、TCP 3009、およびUDP 53ポートのトラフィックを許可します。

3. 各HAペアでGSLBを有効にします。

シナリオ

このシナリオには、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="vpLgs1bdns"
```

```
$lbName="MultiIPALB"
```

```
$frontEndConfigName2="FrontEndIP2"
```

```
$backendPoolName1="BackendPoolHttp"
```

```
$lbRuleName2="LBRuleGSLB1"
```

```
$lbRuleName3="LBRuleGSLB2"
```

```
$lbRuleName4="LBRuleDNS"
```

```
$healthProbeName="HealthProbe"
```

GSLBおよびDNSトラフィックを許可するためのフロントエンドIPアドレスとルールで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

各ALB（ALB 1とALB 2）には2つのフロントエンドIPアドレスがあります。1つは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、以下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 に NetScaler GSLB を展開する

需要の増加に伴い、地域顧客にサービスを提供するオンプレミスデータセンターを運用している企業は、Azure クラウドを使用してグローバルに規模を拡大し、展開したいと考えています。ネットワーク管理者は NetScaler を使用して、GSLB StyleBook を利用し、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。同じ構成を NetScaler ADM を使用して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アクセスできます。これにより、世界のどこにいてもシームレスなエクスペリエンスが得られます。

DBS の概要

NetScaler GSLB は、クラウドロードバランサー向けにドメインベースサービス (DBS) の使用をサポートしています。これにより、クラウドロードバランサーソリューションを使用して動的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 を使用すると、DNS ディスカバリから Microsoft Azure 環境のバックエンドリソースをスケーリングできます。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、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るように GSLB サービスグループを動的に更新します。この機能を構成するには、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。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はありません。GSLB サービスグループの Citrix ADC DBS 機能は、DNS 対応サービス ディスカバリを使用して、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、NetScaler GSLB DBS Autoscale コンポーネントとクラウドロードバランサーを示しています。



Azure GSLB の前提条件

The prerequisites for the NetScaler GSLB service groups include a functioning Microsoft Azure environment, along with the knowledge and ability to configure Linux Web Servers, NetScaler appliances within Azure, public IP addresses, and Azure load balancers (ALB).

- 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に 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 動的ディスカバリを使用したオート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-
xxxxxxxxxx.australiaeast.cloudapp.azure.com

add gslb serviceGroup sydney_sg HTTP -autoscale DNS -siteName sydney

bind gslb serviceGroup sydney_sg sydney_server 80
```

Azure コンポーネントを設定する

1. ユーザー Azure Portal にログインし、NetScaler テンプレートから新しい仮想マシンを作成します。
2. Azure 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: 手順 1 で作成された NetScaler
- ヘルスプローブ: 手順 4 で作成済み
- セッション永続性: なし



NetScaler GSLB ドメインベースサービスを構成する

GSLBが有効な環境で、オートスケーリング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. 手順2を繰り返して、Azureの2番目のリソースから2番目のALBを追加します。

GSLB設定

1. **Add** 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は、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. 「トラフィック管理 > GSLB > サービスグループ」に移動します。
5. サービスグループを追加するには、「追加」をクリックします。
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。トラフィックがポート80を経由するように設定します。「**作成**」をクリックします。



サービスグループメンバーのバインディングには、ALBから受け取った2つのインスタンスが入力されます。



8. 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するには、手順5と6を繰り返します。(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、「**トラフィック管理 > GSLB > 仮想サーバー**」に移動します。
10. 仮想サーバーを作成するには、「**追加**」をクリックします。
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、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、**No GSLB Virtual Server ServiceGroup Binding**をクリックします。



13. **ServiceGroup Binding**の下で、**Select Service Group Name**を使用して、前のステップで作成されたサービスグループを選択して追加します。



14. **No GSLB Virtual Server Domain Binding**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構成します。FQDNを構成してバインドします。その他のパラメーターはデフォルト設定のままにします。



15. **No Service**をクリックしてADNSサービスを構成します。
16. ロードバランシングサービスを構成するための詳細を指定します。

サービス名を追加し、**New Server**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がすでに構成されている場合は、**Existing Server**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経由で流れるように構成されていることを確認してください。



17. **Method**を**Least Connection**として、バックアップメソッドを**Round Robin**として構成します。
18. **Done**をクリックし、ユーザーGSLB仮想サーバーがUpとして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のためのNetScaler グローバル負荷分散([/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))

Azure Local に NetScaler VPX インスタンスを展開する

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 インスタンスは、スタンドアロンの Azure Local ホストに展開することも、Azure Local インスタンス (旧 Azure Stack HCI クラスター) で高可用性 (HA) ペアとして設定することもできます。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をオンプレミスで提供し、Azure クラウドサービスとのシームレスな統合を保証します。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。

Azure Local に NetScaler VPX インスタンスを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: 14.1-60.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ます。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 VM ファイルパッケージを使用して Azure Local に NetScaler VPX インスタンスを展開する

この方法を使用して、VM ファイルパッケージから Azure Local に NetScaler VPX を展開します。

VM ファイルパッケージを使用して Azure Local に NetScaler VPX を展開するには、次の 5 つの手順に従います。

1. NetScaler Hyper-V パッケージをダウンロードし、ダウンロードしたパッケージを Azure Local ノードにアップロードします。
2. 抽出したパッケージを使用して新しい仮想マシンを作成します。
3. VM 設定を構成し、仮想マシンを起動します。
4. コンソールアクセス経由で NetScaler の IP アドレスを割り当てます。
5. さらなる設定のために、NetScaler VPX にリモート接続します。

ステップ1. NetScaler Hyper-V パッケージをダウンロードします。

1. Web ブラウザを開き、<https://www.citrix.com/> に移動します。
2. 割り当てられた資格情報を使用してサインインします。
3. **ダウンロードセクション**を見つけてクリックします。
4. **製品の選択**ドロップダウンメニューから、****NetScaler (NetScaler ADC)****を選択します。
5. **NetScaler Release X.X > 仮想アプライアンスセクション**の下にある **NetScaler VPX Release X.X**を選択します。
6. 圧縮されたインストールパッケージ（例: **NSVPX-HyperV-14.1-60.57_nc_64.zip**）をサーバーに直接ダウンロードします。

7. ダウンロードしたパッケージを、任意の場所にAzure Localノードにアップロードします。パッケージの内容を抽出します。



ステップ2. 抽出したVMパッケージを使用して新しい仮想マシンを作成します。

仮想マシンを作成および構成するには、次の手順を完了します。

1. **Windows Admin Center**を開き、**仮想マシンセクション**に移動します。
2. **追加**ボタンをクリックし、**インポート**を選択します。



3. 抽出したNetScaler VMイメージを含むフォルダを参照して選択します。
4. 利用可能な仮想マシンのドロップダウンメニューから、**NetScaler仮想アプライアンス**を選択します。



5. **インポート**をクリックします。
6. **レビュー + 作成**オプションに進み、新しい仮想マシンを確定して作成します。

ステップ3. VM設定を構成し、仮想マシンを起動します。

NetScaler仮想アプライアンスをインポートした後、最高のパフォーマンスと接続性を得るためにVMを構成します。

1. **リソースの確認:** VM名、メモリ、CPU、ストレージの割り当てを確認し、必要に応じて調整します。
2. **ネットワークアダプターの割り当て:** 適切なネットワークアダプターを接続して、接続を有効にします。



3. **VMの起動:** NetScaler VMの電源を入れて初期化を開始します。



ステップ4. コンソールアクセスを通じてNetScaler IPアドレスを割り当てます。

1. **Hyper-Vマネージャー**を通じてNetScalerインスタンスコンソールに直接アクセスします。
2. Azure Local ノードの Hyper-V コンソールを介して NetScaler にログインします。



3. IP アドレスを割り当てます。

4. IP 設定を保存します。



ステップ 5. NetScaler VPX にリモート接続します。

1. NetScaler 管理インターフェイスへの接続を確認します。
2. ログインし、その後のセットアップおよび管理タスクのために NetScaler を構成します。

Windows Admin Center から VHD を使用して Azure Local に NetScaler VPX インスタンスを展開する

Windows Admin Center から VHD を使用して Azure Local に NetScaler VPX インスタンスを展開するには、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. SSH を使用して NetScaler VPX インスタンスにリモート接続し、その後のセットアップと管理を行います。
6. NetScaler 管理インターフェイスへの接続を確認し、必要な構成を完了することで、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する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。



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、**第 1 世代** を選択します。
4. NetScaler の要件とデプロイ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. Hyper-Vマネージャーを使用してネットワークを構成し、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ノード上の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 Localの検証(/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、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 上で 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、Webサイト、Web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、ボット軽減、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。基本的なWAF保護と高度なWAF保護の両方を使用することで、本ソリューション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、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、これまでになく迅速に稼働を開始できます。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、通常、顧客企業のルーターまたはファイアウォールの背後で、顧客サーバーと顧客ユーザー間のレイヤー 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、どれが脆弱性が低く、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 Application Firewall 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 を実装します。

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:

構成に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 Application Firewall 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構成

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。Web Application Firewall StyleBookを使用してNetScaler Web App Firewallを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이스の両方は、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経験豊富なユーザーを対象としています。

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、インジェクションされたSQLコードについて、リクエストペイロードの3つの場所（1）POSTボディ、2）ヘッダー、3）クッキー）を検査します。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、攻撃を無害化するよう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、その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またはネイティブルールセットは、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状態を全体的に把握できます。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:

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をさらに詳しく調べることができます。

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、ユーザーは脅威や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。報告された攻撃が誤検知である場合があります、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に対して例外（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。

- 次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、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に主に関心があります。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、ユーザーがアプリケーションのセキュリティ状態を評価し、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:

ユーザーが以下を確実に行うようにします。

- 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、「**ログ式ベースのセキュリティインサイト設定**」セクションの下にある「**セキュリティインサイトを有効にする**」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、ログ式の戻り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 > Total Violations**に移動します。アプリケーションサマリーテーブルでURLをクリックすると、ログ式の名前、コメント、およびADCインスタンスがアクションに対して返した値を含む、**Violation Information**ページで違反の完全な詳細が表示されます。



設定を展開する前に、安全指数を決定します。セキュリティ侵害は、ユーザーがADCインスタンスにセキュリティ設定を展開した後に発生しますが、ユーザーは展開する前にセキュリティ設定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設定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**Devices**の下で、ユーザーが設定した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の両方が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 > Safety Index > 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。次の目的のために、**Security > Security Violations** に移動して、単一ペインソリューションを利用します。

- **Network、Bot、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する
- アプリケーションを保護するための是正措置を講じる

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。

- ユーザーがNetScalerのプレミアムライセンスを持っていること（WAFおよびBOT違反の場合）。
- ユーザーがロードバランシングまたはコンテンツスウィッチング仮想サーバーにライセンスを適用していること（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の管理](#)」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「[Setting up](#)」セクションにある手順を参照してください。 [Setting up](#)。

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。

セットアップ

違反については、**Metrics Collector** 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**Metrics Collector** 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

- **Networks > Instances > 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、**Select Action** リストから **Configure Analytics** 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** の下で、**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とアプリケーションで発生した合計違反数。合計違反数は、選択された期間に基づいて表示されます。



- 各カテゴリごとの合計違反数。



- 影響を受けたADCの合計数、影響を受けたアプリケーションの合計数、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いた上位の違反。



違反の詳細については、「すべての違反」(</ja-jp/citrix-application-delivery-management-software/current-release/analytics/security/all-violations.html#security-violations-dashboard.html>)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」(</ja-jp/citrix-application-delivery-management-service/analytics/bot-insight#configure-bot-security-violations-in-netscaler-adm.html>)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. 次のようなアプリケーションサマリーの詳細を提供します。

- **平均RPS** – 仮想サーバーで受信された1秒あたりの平均ボットトランザクションリクエスト (RPS) を示します。
- **重大度別のボット** – 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**に基づいて分類されます。

たとえば、仮想サーバーに重大度「高」のボットが11770個、重大度「緊急」のボットが1550個ある場合、NetScaler ADMは**重大度別のボット**の下に**緊急 1.55 K**と表示します。

- **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。

たとえば、仮想サーバーにブロックリストされたボットが8000個、許可リストされたボットが5000個、レート制限超過ボットが10000個ある場合、NetScaler ADMは**最大のボットカテゴリ**の下に**レート制限超過 10 K**と表示します。

- **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。

たとえば、仮想サーバーにサンタクララで5000件、ロンドンで7000件、バンガロールで9000件のボット攻撃がある場合、NetScaler ADMは**最大の地理的ソース**の下に**バンガロール 9 K**と表示します。

- **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た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. 対応する構成済みアクションとともに、ボット攻撃の合計数を表示します。たとえば、次のように構成した場合：

- 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は次のように表示します。

- 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チャレンジで不正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、**ゼロピクセルリクエスト**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、**ボットカテゴリ**の下にある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総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総数を示します。

- **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。
- **プロフィール名** – ユーザーが設定時に入力したプロフィール名を示します。

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、検索ボックスには次の検索候補の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** – ボット攻撃後に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、検索の焦点を絞り込むために、ユーザー検索クエリで演算子を使用することもできます。たとえば、すべての悪意のある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
- 検索ボックスをもう一度クリックし、演算子 **=** を選択します
- 検索ボックスをもう一度クリックし、「**Bad**」を選択します。
- 「**Search**」をクリックして結果を表示します。



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。



「**Event Details**」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構成

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 ボット管理を構成できます。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構成

この構成は、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アプリケーションファイアウォール 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 壊れたアクセス制御	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）、署名自動更新機能
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フォルト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、およびカスタム署名
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

OWASP TOP 10 – 2021	NETSCALER WEB APP FIREWALL の機能
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、カスタム署名、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

認証されたユーザーに許可されている操作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による保護

- 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、および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非表示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、これらの保護によって検証できます。
- 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、ブラウザと交換する際には特別な予防措置が必要です。

NetScaler Web App Firewallによる保護

- 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、Safe Commerce保護で安全なオブジェクトとして構成することで、漏洩を防ぐことができます。
- 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の欠陥は、アプリケーションが信頼できないデータを適切な検証やエスケープなしに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。カスタムインジェクションパターンをアップロードして、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は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桁の整数のみを強制することができます。
- 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と照合して検証し、すべてのフォーム要素の有効性を確認します。
- 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の挿入試行をブロックします。
- XSS保護は、一般的なXSS攻撃から保護します。カスタムXSSパターンをアップロードして、許可されるタグと属性のデフォルトリストを変更できます。ADC WAFは、許可される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、攻撃者が不適切なウェブ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、最もよく見られる問題です。これは通常、安全でないデフォルト設定、不完全または即席の設定、オープンなクラウドストレージ、設定ミスのある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、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を使用して内部ファイルを公開するために使用できます。

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に対処するために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の脅威（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）を検出およびブロックすることに加えて。
- 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、および多数の属性や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の保護機能

- 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、しばしば不適切に実装されており、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他のユーザーのIDを一時的または永続的に乗っ取ったりすることを可能にします。

NetScaler Web App Firewall の保護機能

- 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オフロードとURL変換機能を使用することで、ファイアウォールはサイトがセキュアなトランスポート層プロトコルを使用し、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のにも役立ちます。

- クッキープロキシとクッキー暗号化を導入することで、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、しばしば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 の保護機能

- カスタムシグネチャによる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。
- フィールドフォーマットチェック、クッキーの一貫性、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

不十分なログ記録と監視、およびインシデント対応との統合の欠如または非効率性は、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 の保護機能

- セキュリティチェックまたはシグネチャ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が発信されている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できるため便利です。
- デフォルト形式 (PI) 式は、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とともに、ログに含まれる情報をカスタマイズする柔軟性を提供します。
- 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てHTMLクロスサイトスクリプティングチェックを設定する](#)
- [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てバッファオーバーフローセキュリティチェックを設定する](#)
- [署名オブジェクトを追加または削除する](#)
- [署名オブジェクトを設定または変更する](#)
- [署名オブジェクトを更新する](#)
- [Snortルール統合](#)
- [ボット検出](#)
- [Microsoft AzureにNetScaler VPXインスタンスを展開する](#)

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

特定の状況では、NetScaler Gateway Plug-inで接続するユーザーは、NetScaler Gatewayアプライアンスに対して一意の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る）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されるプライベートIPアドレスをAzureに登録する
- NetScaler Gatewayアプライアンスで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。VPXインスタンスのプロビジョニング中にPowerShellコマンドを使用してIPアドレスを追加する

には、「[PowerShellコマンドを使用してスタンドアロンモードの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、既存のNetScaler VPXインスタンス > **Network interfaces** に移動します。登録する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 Configurations** をクリックし、次に **Add** 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の複数の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展開にも同じスクリプトを使用できます。

注:

このドキュメントでは、IP-Configは、個々のNICに関連付けられているIPアドレスのペア（パブリックIPとプライベートIP）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	関連付けられているもの
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
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
IPコンフィグ-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以下に示します。

**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アドレスは、アプライアンスの管理NSIPアドレスとして自動的に追加されます。 `IPConfigs` に関連付けられた残りのプライベートIPアドレスは、要件に応じて、 `add ns ip` コマンドを使用してVPXインスタンスにVIPまたはSNIPとして追加する必要があります。

スタンドアロンモードでNetScaler VPX仮想アプライアンスに複数のIPアドレスを構成するために必要な手順の概要を以下に示します。

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インスタンスの作成
10. NIC構成の確認
11. VPX側の構成の確認

スクリプト

パラメータ

以下は、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です。

```
$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="multiipvp1"
```

```
$domName2="multiipvp2"
```

```
$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. リソースグループの作成

```
New-AzureRmResourceGroup -Name $rgName -Location $locName
```

2. ストレージアカウントの作成

```
$prmStorageAccount = New-AzureRMStorageAccount -Name $prmStorageAccountName  
-ResourceGroupName $rgName -Type Standard_LRS -Location $locName
```

3. 可用性セットの作成

```
$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の作成

1. ルールを追加します。トラフィックを処理するすべてのポートに対して、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、動的または静的にすることができます。

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:

NICに複数のIP構成を割り当てる場合、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 を使用する NetScaler VPX 展開では、1 つの NIC がプライマリである必要があります。そのため、その NIC を NetScaler 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 構成の確認

NetScaler VPX インスタンスの起動後、次のコマンドを使用して、NetScaler 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 側の構成の確認

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デプロイ用の追加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コマンドレットを提供します。

- 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を含めることも、グループとして管理したいリソースのみを含めることもできます。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

ストレージアカウントには、小文字と数字のみを含む一意の名前を選択してください。

`$saName="<storage account name>"`、`$saType="<storage account type>"`、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0、バックエンドサブネットは1としてインデックス付けされます。次に、以下の3つのいずれかの方法でNICを作成します:

a) パブリックIPアドレスを持つNIC

```
$nicName="<name of the NIC of the VM>"

$pip = New-AzureRmPublicIpAddress -Name $nicName -ResourceGroupName $rgName
-Location $locName -AllocationMethod Dynamic

$nic = New-AzureRmNetworkInterface -Name $nicName -ResourceGroupName $rgName
-Location $locName -SubnetId $vnet.Subnets[$subnetIndex].Id -PublicIpAddressId
$pip.Id
```


b) パブリックIP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ます:

```
Test-AzureRmDnsAvailability -DomainQualifiedName $domName -Location $locName
```

```
$nic = New-AzureRmNetworkInterface -Name $nicName -ResourceGroupName $rgName  
-Location $locName -SubnetId $vnet.Subnets[$subnetIndex].Id -PublicIpAddressId  
$pip.Id
```

例:

```
$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

```
$vmName="<VM name>"
```

```
$vmSize="<VM size string>"
```

```
$avSet=Get-AzureRmAvailabilitySet -Name $avName -ResourceGroupName $rgName
```

```
$vm=New-AzureRmVMConfig -VMName $vmName -VMSize $vmSize -AvailabilitySetId  
$avset.Id
```

7. NetScaler VPXイメージを取得

```
$pubName="<Image publisher name>"
```

```
$offerName="<Image offer name>"
```

```
$skuName="<Image SKU name>"
```

```
$cred=Get-Credential -Message "Type the name and password of the local  
administrator account."
```

VPXにログインするために使用する資格情報を入力します

```
$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®から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、発行元が提供する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VM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て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の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、以降も同様です。

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、負荷分散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. 負荷分散ルールを作成する

負荷分散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：

プロトコル-フロントエンドポート-バックエンドポートのトリプレットは、2つのNATルールで同じであってはなりません。

```
$inboundNATRule1= New-AzureRmLoadBalancerInboundNatRuleConfig -Name
SSH1 -FrontendIpConfiguration $frontendIP1 -Protocol TCP
-FrontendPort 22 -BackendPort 22

$inboundNATRule2= New-AzureRmLoadBalancerInboundNatRuleConfig -Name
SSH2 -FrontendIpConfiguration $frontendIP1 -Protocol TCP -FrontendPort
10022 -BackendPort 22
```

9. ロードバランサーエンティティを作成する

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) NIC1をVPX1に

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) NIC2をVPX2に

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、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内部ロードバランサーを使用した高可用性セットアップでNetScaler VPXペアをプロビジョニングする

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-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"
```

```
$vnetAddressPrefix="10.0.0.0/16"
```

```
$FrontendAddressPrefix="10.0.1.0/24"
```

```
$BackendAddressPrefix="10.0.2.0/24"
```

```
$vnet=New-AzureRmVirtualNetwork -Name $vnetName -ResourceGroupName
```



```
$rgName -Location $locName -AddressPrefix $vnetAddressPrefix -Subnet
$frontendSubnet,$backendSubnet\`
```

```
$frontendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
frontendSubnet -AddressPrefix $FrontendAddressPrefix
```

```
$backendSubnet=New-AzureRmVirtualNetworkSubnetConfig -Name
backendSubnet -AddressPrefix $BackendAddressPrefix
```

注:

要件に応じてAddressPrefixパラメータ値を選択してください。

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. 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し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、同じ設定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設定コマンドを実行します。

Azure上のVPXインスタンスのサポートチケットを作成する

Azure上のNetScaler VPXインスタンスで問題が発生している場合、トラブルシューティングのために、[NetScalerサポートポータル](#)でサポートチケットを作成できます。

サポートチケットを提出するには、以下を確認してください。

- ネットワークが接続されていること。
- Azureアカウント番号、Azureに展開したNetScaler®サブスクリプションベースの製品のサポートPINコード、およびAzureシリアルログを手元に用意していること。

- サポートPINコードは、VPX GUIのシステムページで確認できます。

AzureサポートPIN(/en-us/vpx/media/azure-support-pin.png)

- シリアルログは、Azureポータルで確認できます (VMのブート診断セクション)。

Azureブート診断(/en-us/vpx/media/azure-boot-diagnostics.png)

すべての情報が揃ったら、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。

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クラウド内の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**にアクセスしてください。

- 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にはどうすればよいですか？

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環境でのMAC移動とインターフェースミュートを回避するために、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VLAN（タグなし）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使用できます。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-medium, e2-standard-2, e2-highmem-2, n1-standard-2, n1-highmem-2, n2-standard-2, n2-highmem-2, n2d-standard-2, n2d-highmem-2, c3-standard-4, n4-standard-2, c4-standard-2, c4d-standard-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

注:

- N4 および C3 インスタンスタイプのサポートは、NetScaler リリース 14.1-56.x から開始されます。
- C4 および C4D インスタンスタイプのサポートは、NetScaler リリース 14.1-73.x から開始されます。

サポートされている機能

使用されるライセンスまたは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 つの vCPU と 4 GB の RAM が必要です。

注意事項

デプロイを開始する前に、以下の GCP 固有の点について考慮してください。

- インスタンスの作成後、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。
- 複数の NIC を使用するデプロイでは、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。
- 単一 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 つを超える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 つの 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 の設定時に 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 と複数の IP を持つスタンドアロンの NetScaler VPX インスタンスをデプロイする

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デプロイする方法を示します。このシナリオでは、多数の NIC を持つスタンドアロン VPX インスタンスを作成します。インスタンスはバックエンドサーバー（サーバーファーム）と通信します。



以下の目的のために、3 つの NIC を作成します。

NIC	目的	VPC ネットワークに関連付けられています
NIC 0	管理トラフィック (NetScaler IP) を処理します	管理ネットワーク

NIC	目的	VPCネットワークに関連付けられています
NIC 1	クライアント側トラフィック (VIP) を処理します	クライアントネットワーク
NIC 2	バックエンドサーバー (SNIP) と通信します	バックエンドサーバーネットワーク

次の間の必要な通信ルートを設定します。

- NetScaler VPXインスタンスとバックエンドサーバー。
- NetScaler VPXインスタンスとパブリックインターネット上の外部ホスト。

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。

GCPマーケットプレイスからNetScaler VPXインスタンスを選択します。

注:

VPCネットワークを作成したのと同じリージョンにインスタンスを作成します。

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。



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:

このシナリオでは、3つのVPCネットワークすべてが同じリージョン（asia-east1）にある必要があります。

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。

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のNetScaler VPXスタンドアロンインスタンスを展開する方法を示します。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。



次の目的のために、単一のNIC (NIC0) を作成します。

- 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。

次の間の必要な通信ルートを設定します。

- インスタンスとバックエンドサーバー。
- 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. 単一の 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. サービスまたは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。

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 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、**Create** 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. 単一 NIC のインスタンスを作成します。

単一 NIC の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 console** 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。

GCPインスタンスを作成(/en-us/vpx/media/gcp-create-instance.png)

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）。

2つのvCPUインスタンス(/en-us/vpx/media/gcp-2vcpus-instance.png)

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を設定するために、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。

GCPネットワークインターフェース(/en-us/vpx/media/gcp-network-interface.png)

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。

VMインスタンスの詳細(/en-us/vpx/media/gcp-vm-instance-details.png)

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アドレスの追加(/en-us/vpx/media/adc-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。
- 「作成」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インスタンスで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。
- 「作成」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、「追加」をクリックします。
2. 名前、プロトコル、IPアドレスタイプ (IPアドレス)、IPアドレス (クライアントエイリアスIP)、およびポートの必要な値を入力します。
3. 負荷分散仮想サーバーを作成するには、「OK」をクリックします。

負荷分散仮想サーバーを作成する(/en-us/vpx/media/create-lb-vserver-adc.png)

ステップ7. VPXインスタンスで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、「**追加**」をクリックします。
2. サービス名、IPアドレス、プロトコル、およびポートの必要な値を入力し、「**OK**」をクリックします。

ステップ8. インスタンスでサービス/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. 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、**編集** をクリックします。
3. サービスとサービスグループ ウィンドウで、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. **ステップ 7** 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。

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` とインスタンス ID をパスワードとして VPX にログオンします。プロンプトが表示されたら、パスワードを変更して設定を保存します。
- 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` ではなく、コマンド `shell /netscaler/showtech_cloud.pl` を実行します。
- GCP コンソールから NetScaler VM を削除した後、関連付けられている 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 に移動して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できるライセンスオプションは次のとおりです。

Bring your own license (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の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。

注記:

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 - 顧客ライセンス
NetScaler VPX FIPS - 顧客ライセンス

NetScaler VPXインスタンスを展開するためのGDMテンプレート

NetScaler VPX Google Deployment Manager (GDM) テンプレートを使用して、GCPにVPXインスタンスを展開できます。詳細については、[NetScaler GDM Templates](#)を参照してください。

リソース

- [複数のネットワークインターフェイ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 アドレスの使用](#)
- [プライベート IP アドレスの使用](#)
- [プライベート IP アドレスを持つシングル NIC VM の使用](#)

GCP に 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が `us-east1-b` ゾーンに、セカンダリVPXが `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

VPX高可用性展開では、バックエンドに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サポートされます。

Google Cloud Platform上で外部静的IPアドレスを持つVPX高可用性ペアをデプロイする

外部静的IPアドレスを使用して、GCP上にVPX高可用性ペアをデプロイできます。プライマリノードのクライアントIPアドレスは、外部静的IPアドレスにバインドされている必要があります。フェールオーバー時には、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アドレスとは、解放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インスタンスをデプロイする](#)」に記載されている制限事項、ハードウェア要件、注意事項をお読みください。この情報はHA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",  
  
    ]
```

- 管理インターフェイス以外のインターフェイスでエイリアスIPアドレスを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、フェールオーバー時に新しいプライマリに更新するために、「[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ます。

Google Cloud PlatformでVPX HAペアをデプロイする方法

HA展開手順の概要を次に示します。

1. 同じリージョンにVPCネットワークを作成します。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 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、([/ja-jp/vpx/current-release/deploy-vpx-google-cloud.html](#))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

([/ja-jp/vpx/current-release/deploy-vpx-google-cloud.html#scenario-deploy-a-multi-nic-multi-ip-standalone-vpx-instance](#))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イス > 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、静的外部IPアドレスは古いプライマリから移動し、新しいプライマリにアタッチされます。詳細については、Google Cloud ドキュメントの[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、([/ja-jp/citrix-adc/current-release/networking/ip-addressing/configuring-citrix-adc-owned-ip-addresses.html](http://ja-jp/citrix-adc/current-release/networking/ip-addressing/configuring-citrix-adc-owned-ip-addresses.html))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 for CLI を使用して HA を構成できます。

GUI を使用して HA を構成する

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、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。

これ以上進む前に、**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. 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インスタンスで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. 次の手順に従ってプライマリSNIPアドレスを追加します。
 - a. プライマリインスタンスのサーバ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で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. 次の手順に従って、セカンダリVIPアドレスを追加します。

- a. セカンダリインスタンスのクライアント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。



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. 次の手順に従って、セカンダリVIPアドレスを追加します。
 - a. セカンダリインスタンスのクライアント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。
3. 次の手順に従って、セカンダリSNIPアドレスを追加します。
 - a. セカンダリインスタンスのサーバ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。



ステップ 3. IPセットを追加し、両方のインスタンスでセカンダリ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(/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(/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で負荷分散仮想サーバーを追加します。

1. 「設定 > トラフィック管理 > 負荷分散 > 仮想サーバー > 追加」に移動します。
2. 名前、プロトコル、IPアドレスタイプ (IPアドレス)、IPアドレス (プライマリVIP)、ポートにそれぞれ必要な値を入力します。

プライマリインスタンス上の負荷分散仮想サーバー(/en-us/vpx/media/primary-lb-vserver.png)

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. プライマリノードでサービスまたはサービスグループを追加します。

1. 「設定 > トラフィック管理 > 負荷分散 > サービス > 追加」に移動します。
2. サービス名、IPアドレス、プロトコル、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 6. 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. **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 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. 「**Service and Service Groups**」タブを開き、「**No Load Balancing Virtual Server Service Binding**」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。

設定を保存します。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. 両方のノードで仮想 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イスの内部 IP アドレスを指します。

secondary_vip は、セカンダリインスタンスのクライアント向けインターフェイスの内部 IP アドレスを指します。

primary_snip は、プライマリインスタンスのサーバー向けインターフェイスの内部 IP アドレスを指します。

セカンダリノードで、次のコマンドを入力します。

```
add ns ip <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イスの内部 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イスの内部 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アドレスを持つ単一NIC VPX高可用性ペアをデプロイする

プライベートIPアドレスを使用して、GCPに単一NIC VPX高可用性ペアをデプロイできます。クライアントIP（VIP）アドレスは、プライマリノードでエイリアス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。

始める前に

- [Google Cloud PlatformにNetScaler VPXインスタンスをデプロイする](#)に記載されている制限事項、ハードウェア要件、注意事項をお読みください。この情報は、高可用性展開にも適用されます。
- GCPプロジェクトで**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",  
  
]
```

- 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、[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にVPX高可用性ペアをデプロイする方法

単一NICでHAペアをデプロイする手順の概要は次のとおりです。

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：アジア東部-1aおよびアジア東部-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. 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、**Create**をクリックします。

詳細については、[Deploy a NetScaler VPX instance on Google Cloud Platform](#)の**Create VPC Networks**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

[Scenario: Deploy a single-NIC, standalone VPX instance](#)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、**Edit**をクリックします。
2. **Network Interface**ウィンドウで、クライアント (NIC0) インターフェイスを編集します。
3. **Alias IP range**フィールドに、クライアントエイリアスIPアドレスを入力します。
4. **Add IP Range**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イス**ウィンドウで、クライアント (NIC0) インターフェイスを編集します。
3. **エイリアスIP範囲**フィールドに、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. **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. 「**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選択します。
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、「**追加**」をクリックします。
3. 「**リモートノードIPアドレス**」フィールドに、プライマリノードの管理NICのプライベートIPアドレスを入力します。
4. 「**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選択します。
5. 「**作成**」をクリックします。

さらに進む前に、セカンダリノードの同期状態が「**Nodes**」ページで「**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/single-nic-sync-state-success.png)

「

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、「**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。
 - 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。

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。
 - 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(/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. **設定 > トラフィック管理 > 負荷分散 > 仮想サーバー > 追加** に移動します。
2. Name、Protocol、IP Address Type (IP Address)、IP Address (プライマリクライアントエイリアスIPアドレス)、および Port 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/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。

1. **設定 > トラフィック管理 > 負荷分散 > サービス > 追加** に移動します。
2. Service Name、IP Address、Protocol、および Port 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 > 負荷分散 > 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、**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. 「**Step 4**」で設定したサービスを選択し、「**Bind**」をクリックします。

手順6. 設定を保存します。

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) 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**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。

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。

```
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アドレスを持つVPX高可用性ペアをデプロイする

GCP上でプライベートIPアドレスを使用してVPX高可用性ペアをデプロイできます。クライアントIP (VIP) は、プライマリノードでエイリアス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にNetScaler VPXインスタンスをデプロイする](#)」に記載されている制限事項、ハードウェア要件、注意事項をお読みください。この情報は高可用性デプロイメントにも適用されます。
- 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**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高可用性ペアをデプロイする方法

高可用性デプロイメント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 > ネットワーキング > VPC ネットワーク > VPC ネットワークを作成** にログオンします。
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。VPX インスタンスの内部 IP アドレスを使用して VIP を構成しないでください。

クライアントエイリアス IP アドレスを作成するには、次の手順を実行します。

1. VM インスタンスに移動し、**編集** をクリックします。

2. ネットワークインターフェイス 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。



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インスタンスを設定した後、仮想IP (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. **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. **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選択します。
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、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た後、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インスタンス内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向け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アドレスを作成する(/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。
 - 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向けインターフェイスの内部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 > 負荷分散 > 仮想サーバー > 追加** に移動します。
2. 名前、プロトコル、IPアドレスタイプ (IPアドレス) 、IPアドレス (プライマリクライアントエイリアスIPアドレス) 、およびポートに必須の値を入力し、「**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 > 負荷分散 > サービス > 追加** に移動します。
2. サービス名、IPアドレス、プロトコル、およびポートに必須の値を入力し、「**OK**」をクリックします。

ステップ5。 プライマリノードの負荷分散仮想サーバーにサービスまたはサービスグループをバインドします。

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 を使用した高可用性の構成

ステップ 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。

ステップ 2. 両方のノードで 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向けインターフェイスの内部 IP アドレスを指します。

セカンダリノードで次のコマンドを入力します。

```
add ns ip <primary_client_alias_ip> <subnet> -type VIP
```

注:

プライマリ VM インスタンス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> 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 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 ホストを持つ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 デプロイメント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 ドキュメント](#) 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、[Google Cloud VMware Engine プライベートクラウドのデプロイ](#) を参照してください。

- ポイントツーサイト VPN ゲートウェイを使用してプライベートクラウドに接続し、Google Cloud VMware Engine にアクセスして管理する方法の詳細については、[Google Cloud VMware Engine プライベートクラウドへのアクセス](#) を参照してください。
- VPN クライアントマシンで、NetScaler VPX アプライアンスのセットアップファイルをダウンロードします。
- 仮想マシンが接続する VMware SDDC 上に適切な NSX-T ネットワークセグメントを作成します。詳細については、[Google Cloud VMware Engine で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。
- VPX ライセンスファイルを取得します。NetScaler VPX インスタンスライセンスの詳細については、[ライセンスの概要](#) を参照してください。
- GCVE プライベートクラウドに作成または移行された仮想マシン (VM) は、ネットワークセグメントに接続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
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。
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 ツールをインストールするための最小システム要件を示します。

表 2. 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 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、推奨 1 GHz 以上
RAM	最小 1 GB、推奨 2 GB
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 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。「**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。

ログイン後、Citrix®のホームページから以下のパスに移動してください。

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

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、以下のファイルをコピーします。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-`<release number>-<build number>.mf` (例: NSVPX-ESX-13.0-79.64.mf)

Google Cloud VMware Engine をデプロイする

1. [GCVE portal](#)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、以下の詳細を入力します。

- 少なくとも3つのESXiホストを選択して、プライベートクラウドのデフォルトクラスターを作成します。
- 「vSphere/vSan サブネット CIDR 範囲」フィールドには、/22のアドレス空間を使用します。
- 「HCX デプロイメントネットワーク CIDR 範囲」フィールドには、/26の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。



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、「戻る」をクリックします。



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。



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 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」の下にある「表示」をクリックします。



2. vCenter の資格情報をメモします。



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。

vSphereクライアントを起動(/en-us/vpx/media/gcve-launch-vsphere-client.png)

4. この手順のステップ2でメモしたvCenterの認証情報を使用して、VMware vSphereにログインします。



5. vSphereクライアントで、GCVEポータルで作成したESXiホストを確認できます。

ESXiホストを確認(/en-us/vpx/media/gcve-verify-esxi-hosts.png)

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。

NSX-Tログイン情報(/en-us/vpx/media/gcve-nsxt-login-info.png)

2. NSX-Tの認証情報をメモします。

NSX-T認証情報(/en-us/vpx/media/gcve-nsxt-credentials.png)

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でメモした認証情報を使用して、NSX Managerにログインします。



5. 新しいセグメントまたはサブネットのDHCPサービスを設定します。
6. サブネットを作成する前に、DHCPサービスを設定します。
7. NSX-Tで、>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Tier-0および1つのTier-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、をクリックします。
9. DHCP名フィールドに、**Client-Management** 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 **を** 選択します。

11. 列に、DHCPサービスIPアドレス範囲を指定します。

12. **を** 選択します。

13. **を** クリックしてDHCPサービスを作成します。



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。



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ース用、もう1つはサーバーインターフェース用です。

16. NSX-Tで、**>** に移動します。

17. **セグメントの追加**をクリックします。



18. **セグメント名**フィールドに、「**クライアント管理**」セグメントの名前を入力します。

19. **接続済みゲートウェイ**リストで、**Tier1**を選択し、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。最後のオクテットとして **.1** を使用してサブネット範囲を指定します。例: 10.12.2.1/24。



22. **DHCP構成の設定**をクリックし、**DHCP範囲**フィールドに値を入力します。



23. **適用**をクリックして、DHCP構成を保存します。

24. **保存**をクリックします。



25. 手順17から24を繰り返して、サーバーセグメントも作成します。

26. これで、VMの作成時にvCenterでこれらのネットワークセグメントを選択できるようになります。

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 Cloud 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ら、vCenter を使用して VMware Engine に仮想アプライアンスをインストールできます。インストールできる仮想アプライアンスの数は、プライベートクラウドで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 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. **完了**をクリックして、VMware Cloud 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. 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. 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。

「GCVE の地域設定」 (/en-us/vpx/media/gcve-regional-settings.png)

2. お住まいの地域を選択し、**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。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 ポータルで次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 をマッピングするインターフェイスのプライベート IP を指定します。これは、**クライアントインターフェ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 は数分で使用できるようになります。
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必要があります。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、利用されていないリソースの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、必要に応じてリソースをスケールアップおよびスケールダウンするプロセスを自動化する必要があります。

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、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。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機能するかを示しています。

GCPオートスケールトポロジ(/en-us/vpx/media/gcp-backend-autoscaling-topology.png)

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、次のタスクを完了する必要があります。

- 要件に応じて、GCP上にNetScaler 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.instances.get",  
  
  "compute.instanceGroupManagers.get",  
  
  "compute.instanceGroupManagers.list",  
  
  "compute.zones.list",
```

```
"logging.sinks.create",  
"logging.sinks.delete",  
"logging.sinks.get",  
"logging.sinks.list",  
"logging.sinks.update",  
"pubsub.subscriptions.consume",  
"pubsub.subscriptions.create",  
"pubsub.subscriptions.delete",  
"pubsub.subscriptions.get",  
"pubsub.topics.attachSubscription",  
"pubsub.topics.create",  
"pubsub.topics.delete",  
"pubsub.topics.get",  
"pubsub.topics.getIamPolicy",  
"pubsub.topics.setIamPolicy",  
]
```

- 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

GCPオートスケーリングサービスをNetScaler VPXインスタンスに追加する

GUIを使用して、シングルクリックでオートスケーリングサービスをVPXインスタンスに追加できます。オートスケーリングサービスを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GCPマネージドインスタンスグループを選択し、**作成**をクリックしてクラウドプロファイルを作成します。

(/en-us/vpx/media/create-gcp-cloud-profile.png)

- **仮想サーバーIPアドレスフィールド**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™グループ**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する場合は、GUIで **System > Google Cloud Platform > Cloud Profile** に移動し、**Add** をクリックします。

GCPクラウドプロファイルの追加(/en-us/vpx/media/add-gcp-cloud-profile.png)

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の作成(/en-us/vpx/media/create-gcp-cloud-profile.png)

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、GCPで同じマネージドインスタンスグループを使用し、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。

クラウドプロファイル作成後(/en-us/vpx/media/after-gcp-profile.png)

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。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は異なる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の数には1対1のマッピングがあります。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。
- 高容量インスタンスタイプに接続できるNICは最大8つまでです。

たとえば、n1-standard-2インスタンスはNICを2つしか持てず、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ルを実現するには、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したのと同じ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ルに関する制限を軽減できます。

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。

始める前に

- 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アドレスは予約されている必要があります。詳細については、[静的外部IPアドレスの予約](#)を参照してください。
- GCPサービスアカウントに次のIAM権限があることを確認してください。

```
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。
- スタンドアロンの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次のIAM権限があることを確認してください。

```
REQUIRED_IAM_PERMS = [  
  
  "compute.addresses.list",  
  
  "compute.addresses.get",  
  
  "compute.addresses.use",  
  
  "compute.forwardingRules.create",  
  
  "compute.forwardingRules.delete",  
  
  "compute.forwardingRules.get",  
  
  "compute.forwardingRules.list",  
  
  "compute.instances.use",  
  
  "compute.subnetworks.use",  
  
  "compute.targetInstances.create",  
  
  "compute.targetInstances.list",  
  
  "compute.targetInstances.use",  
  
]
```

- 高可用性モードでVIPスケーリングを使用する場合、GCPサービスアカウントに次のIAM権限があることを確認してください。

```
REQUIRED_IAM_PERMS = [  
  
    "compute.addresses.get",  
  
    "compute.addresses.list",  
  
    "compute.addresses.use",  
  
    "compute.forwardingRules.create",  
  
    "compute.forwardingRules.delete",  
  
    "compute.forwardingRules.get",  
  
    "compute.forwardingRules.list",  
  
    "compute.forwardingRules.setTarget",  
  
    "compute.instances.use",  
  
    "compute.instances.get",  
  
    "compute.instances.list",  
  
    "compute.instances.setMetadata",  
  
    "compute.subnetworks.use",  
  
    "compute.targetInstances.create",  
  
    "compute.targetInstances.list",  
  
    "compute.targetInstances.use",  
  
    "compute.zones.list",  
  
]
```

注:

高可用性モードでは、サービスアカウントに所有者または編集者のロールがない場合、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. 次の情報を入力して、**カスタムメタデータ**を更新します。

- キー = vips
- 値 = 次の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

カスタムメタデータ (/en-us/vpx/media/custom-metadata.png)

詳細については、「[カスタムメタデータ](#)」を参照してください。

カスタムメタデータの例:

```
{ "外部IP1名": , "外部IP2名": }
```

この例では、NetScaler VPX インスタンスは、IP とプロトコルのペアごとに転送ルールを内部的に 1 つ作成します。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。

4 つの転送ルールは次のように作成されます。

- a. 外部IP1名 と TCP
- b. external-ip1-name および UDP の設定
- c. external-ip2-name および ICMP の設定
- d. external-ip2-name および AH の設定

注記:

HAモードでは、プライマリインスタンスにのみカスタムメタデータを追加する必要があります。フェ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インスタンス上で外部IPアドレスを持つ負荷分散仮想サーバーの設定

ステップ1. 負荷分散仮想サーバーを追加します。

1. **設定 > トラフィック管理 > 負荷分散 > 仮想サーバー > 追加** に移動します。



2. 名前、プロトコル、IPアドレスタイプ (IPアドレス)、IPアドレス (ADCでVIPとして追加される転送ルールの外部IPアドレス)、およびポートに必要な値を入力し、**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. **設定 > トラフィック管理 > 負荷分散 > サービス > 追加** に移動します。
2. サービス名、IPアドレス、プロトコル、およびポートに必要な値を入力し、**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. **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1**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。



4. ステップ3で構成したサービスを選択し、バインドをクリックします。



5. 構成を保存します。

GCP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、システムページに移動してください。

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。

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受信と送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動作する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で動作するVPXインスタンスのジャンボフレームを構成する

VMware ESXサーバーで動作するNetScaler VPXアプライアンスでジャンボフレームを構成するには、次のタスクを実行します。

1. VPXアプライアンスのインターフェイスまたはチャネルのMTUを1501～9000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受信と送信のみをサポートします。
2. VMware ESXサーバーの対応する物理インターフェイスに、その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イスでのMTUサイズの設定の詳細については、<http://vmware.com/>を参照してください。

Linux-KVMサーバーで実行されているVPXインスタンスのジャンボフレームを構成する

Linux-KVMサーバーで実行されているNetScaler VPXアプライアンスでジャンボフレームを構成するには、次のタスクを実行します。

1. VPXアプライアンスのインターフェイスまたはチャネルのMTUを1501～9216の範囲の値に設定します。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イスに、その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イスでの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 (ネットワーク 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、([/ja-jp/citrix-adc/current-release/networking/interfaces/jumbo-frames/configuring-jumbo-frames-support-on-citrix-adc-appliance.html](#))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、([/ja-jp/citrix-adc/current-release/networking/interfaces/jumbo-frames/configuring-jumbo-frames-support-on-citrix-adc-appliance.html](#))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 に基づくよくある質問を分類するのに役立ちます。

- [機能と性能](#)
- [暗号化](#)
- [価格とパッケージ](#)
- [NetScaler VPX™ Expressと90日間無料トライアル](#)
- [ハイパーバイザー](#)
- [キャパシティプランニングまたはサイジング](#)
- [システム要件](#)
- [その他の技術的なよくある質問](#)

機能と性能

NetScaler VPXとは何ですか？

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ていますか？

はい。NetScaler VPXには、すべての負荷分散、トラフィック管理、アプリケーション高速化、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の機能と性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を使用する場合の制限事項？

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の違い？

機能的には同じです。NetScaler VPX上のNetScaler Gatewayは、NetScalerソフトウェアリリース14.1で利用可能なすべてのNetScaler Gateway機能をサポートしています。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。

NetScaler VPXがハイパーバイザー上で動作するという明白な違いを除いて、NetScaler物理アプライアンスとは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アプライアンスを導入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により、お客様はデータセンター内でネットワーキングと仮想化をさらに統合できます。NetScaler VPXは、仮想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の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開拓します。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処理を扱うために、それぞれのアプリケーションの個別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を柔軟にホストできる能力が大幅に低下します。NetScaler VPXが提供する以上の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関する情報は、[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可能です。NetScaler VPXは、10 Mbps (VPX 10) から100 Gbps (VPX 100G) までの帯域幅を提供します。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上の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（例えば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に依存しません。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でも、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 Premiumの全機能が含まれています。NetScalerリリース14.1–29.65以降、NetScalerはVPX Expressの動作を変更しました。

NetScaler VPX Expressにはライセンスが必要ですか？

最新のNetScaler VPX Expressリリース（14.1–29.65以降）では、VPX Expressは無料で利用でき、インストールや使用にライセンスファイルは必要ありません。コミットメントは不要です。すでにVPX Expressライセンスをお持ちの場合、以前のライセンス動作が引き続き適用されます。ただし、既存のVPX Expressライセンスファイルを削除し、バージョン14.1–29.65以降を使用する場合、更新されたVPX Expressの動作が適用されます。

NetScaler VPX Expressライセンスは期限切れになりますか？

新しいVPX Expressでは、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。NetScaler VPX Expressユーザーは、NetScaler VPX Knowledge Centerを自由に利用でき、ディスカッションフォーラムを使用してコミュニティからヘルプを求めることができます。

NetScaler VPX Expressは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におけるVPXへの仮想ネットワークインターフェースの割り当て数？

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からのNetScaler VPXコマンドラインへのアクセス方法？

VMware vSphereクライアントは、コンソールタブを通じて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することもできます。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へのアクセス方法？

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インスタンスの高可用性セットアップでの構成？

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。NetScaler VPXではリンクアグリゲーションとLACPはサポートされていません。リンクアグリゲーションはVMwareレベルで設定する必要があります。

NetScaler VPXドキュメントにはどのようにアクセスしますか？

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性能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る可能性があります。

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に適用されますか、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に適用されますか？

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にのみ適用されます。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。

複数のNetScaler VPXインスタンスを同じサーバーで実行できますか？

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のハードディスク容量、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。推奨されるのは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へのアクセス時の遅延、ピットボスデーモンのクラッシュ、パケットドロップ、低スループットなどの問題が発生します。

すべてのVPXインスタンスが事前定義された条件を満たしていることを確認してください。

NetScaler VPXと他のアプリケーションを同じサーバーでホストできますか？

はい。例えば、NetScaler VPX、Citrix Virtual Apps ウェブインターフェース、Citrix Virtual Apps XMLブローカーはすべて仮想化でき、同じサーバー上で実行できます。最高のパフォーマンスを得るには、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容量を持っ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のライセンスを持っている場合、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を増やすことで、特に高性能なシナリオにおいて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ー（例えば、PCIパススルーやSR-IOV）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はアイドル状態でもCPUの90%以上を消費しているように見えるのはなぜですか？

これは正常な動作であり、NetScalerアプライアンス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の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は）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について説明しています。

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 でサポートされているプロセッサ 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。

種類	要件
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。ただし、1 Gbps を推奨します。
NIC	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でサポートされている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につなが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上で動作するため、BSD Unixがインストールされている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イ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。

MACベースの転送（MBF）はVPXでサポートされていますか。 NetScalerアプライアンスの実装からの変更点？

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、トレースファイルの保存スペースを含むサービス性ニーズに対応するため、20 GBにサイズ設定されています。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。互換性の目的で、これらは別々のファイルシステムとして保持されています。

次の値は、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スペースを表します。

- `/flash` = 965M
- `/var` = 14G

詳細なメモリ割り当ての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インスタンスのスペース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、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スペースを増やすオプションがあります。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NetScaler VPXインスタンスにセカンダリディスクを接続するには、Azure VMサイズにローカル一時ディスクがあることを確認してください。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：

NetScaler VPXに新しいHDDを追加した後、新しいHDDに移動されたファイルに対して動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した場合。

そのようなコマンドはすべて「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増やすことはできますか？

NetScalerリリース14.1ビルド21.x以降、管理者はNetScaler VPXのプライマリディスクサイズを20 GBから最大1 TBまで一度に動的に増やすことができます。その後も、再度最大1 TBまで増やすことができます。ディスク容量を増やすには、それぞれのクラウドまたはハイパーバイザーのUIでプライマリディスクサイズを最低1 GBまで拡張してください。

注記：

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。

NetScaler VPX のプライマリディスクサイズを手動で増やすにはどうすればよいですか？

ハイパーバイザーまたはクラウドから VPX プライマリディスクサイズを手動で増やすには、次の手順に従います。

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 20 GB をより大きな値に拡張します。たとえば、20 GB を 30 GB または 40 GB にします。Azure の場合は、デフォルトのディスクサイズ 32 GB を 64 GB に拡張します。
3. VM の電源を入れ、ブートプロンプトに入ります。
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. ディスクスペースを確認します。「gpart show」 コマンドを使用して、新しく割り当てられたディスクスペースを確認できます。
6. パーティション名をメモします。たとえば、VM パーティションは da0 です。
7. 「gpart resize」 コマンドを使用してディスクパーティションのサイズを変更します。

例: 次のコマンドを実行して、da0 MBR パーティションのサイズを変更し、10 GB の空き領域を含めます。

```
gpart resize -i 1 da0
```

8. 空き領域を最後のパーティションにマージします。

例:

```
gpart resize -i 5 da0s1
```

9. 「growfs」 コマンドを使用して、新しく割り当てられた空き領域を含めるようにファイルシステムを拡張します。

例:

```
growfs /dev/ada0s1e
```

10. VMを再起動し、シェルプロンプトで「df -h」 コマンドを使用して、増加したディスクスペースを確認します。

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採用しています。例えば、9.1_97.3.vpx、9.1_97.3.nc、9.1_97.3.cl などです。

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できますか？

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) インターフェースは、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ですか。これは最大スループットが1 Gbpsに制限されることを意味しますか？

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ますか？

NetScaler VPXはHyper-Vライブマイグレーションをサポートしていません。vMotionはNetScalerリリース13.0以降でサポートされています。ライブマイグレーション（旧称XenMotion）はNetScalerリリース14.1ビルド17.38以降でサポートされています。

fix

This is the content for the sample heading topic. It is hidden from the main table of contents but remains accessible via links.