

Amazon Web Services（AWS）での仮想 Nexus Dashboard（vND）の展開

- AWS パブリッククラウドでの vND のホスティングについて （1 ページ）
- Amazon Web Services で Nexus Dashboard クラスタを展開するための前提条件と注意事項（3 ページ）
- Nexus Dashboard クラスタ向け Amazon Web サービスの準備 （5 ページ）
- Amazon Web Services（AWS）に仮想 Nexus Dashboard（vND）を展開する （6 ページ）
- ノードの交換（RMA） （14 ページ）

AWSパブリッククラウドでのvNDのホスティングについて

この機能を使用すると、AWS パブリッククラウドで仮想 Nexus Dashboard（vND）を実行できます。このソリューションのコンポーネントは次のとおりです。

- Virtual Nexus Dashboard
- N9000 スイッチ
- 2 つの Catalyst 8000 シリーズ ルータ、または Nexus ダッシュボードが vND からオンプレミス データセンターへの VXLAN トンネルを終端して永続 IP アドレス（PIP）トラフィックに使用できる別のタイプのデバイス（N9000 スイッチなど）
- AWS パブリック クラウド アカウント

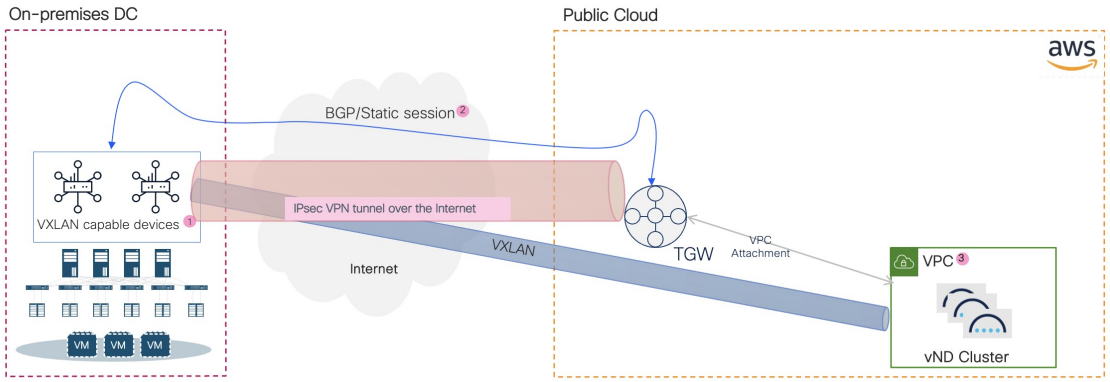
AWS パブリック クラウドでの vND の展開方法を理解する

AWS パブリッククラウドに vND を展開する場合、手動でのブートストラップは不要です。代わりに、Nexus Dashboard のブートストラップが自動的にブートストラップを実行します。AWS パブリッククラウドで vND デプロイメントプロセスを完了すると、仮想プライベートクラウド（VPC）の 3 つの可用性ゾーン（AZ）にわたる、可用性の高い 3 ノードクラスターが自動的に作成されます。Nexus Dashboard GUIで、**[管理（Administration）] [** > **システムステータ**

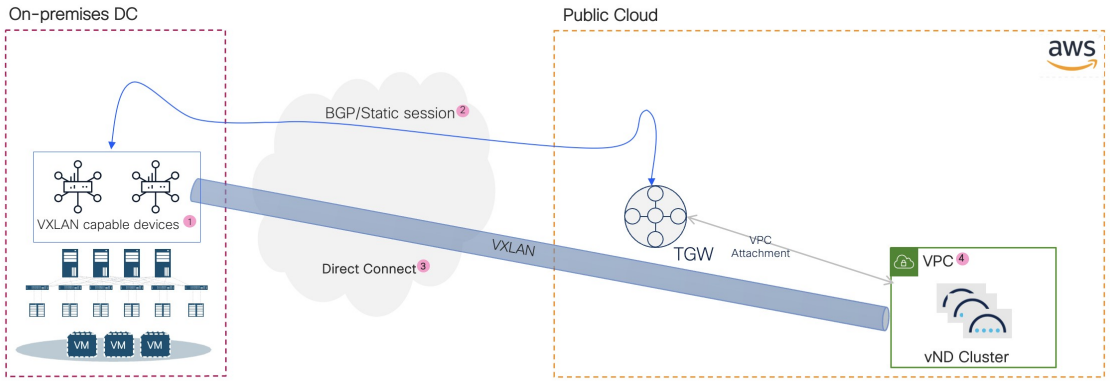
ス（System Status）]>[概要（Overview）]に移動し、[クラスタノード（Cluster nodes）]領域での 3 ノードクラスターに関する情報を表示します。

トポロジの例

次の図は、トポロジの例を示しています。



1 On-premises device with VXLAN support for example, Catalyst 8000 series router, Nexus 9000
2 BGP / static session for ND data IP reachability
3 VPC in AWS terminology stands for virtual private cloud and is an equivalent of VRF



1 On-premises device with VXLAN support for example, Catalyst 8000 series router, Nexus 9000
2 BGP / static session for ND data IP reachability
3 Direct connect takes the role of a private transport for carrying data
4 VPC in AWS terminology stands for virtual private cloud and is an equivalent of VRF

ここで、

- オンプレミスのデータセンターと AWS パブリッククラウドの間の接続は、直接接続（実稼働環境に推奨）または 2 つの間の IPSec トンネル（追加のオーバーヘッドとして PoC またはラボ用）のいずれかを使用して実現されます。
- 2 つのオンプレミスルータは、次のいずれかになります。
  - 直接接続を使用する場合は、N9000 スイッチなどの 2 つの物理スイッチを使用できます。

- IPSec トンネルを使用している場合、この場合は VXLAN トンネルを終端するため、VXLAN サポートが有効な Cisco 8000 ファミリのネットワーク アプライアンスを使用できます。

- トランジット ゲートウェイは、Nexus Dashboard ノードをホストする VPC（アプリ VPC）に接続するためのトランジット ゲートウェイ アタッチメントと、トランジット ゲートウェイで終端する別のトランジット ゲートウェイ添付ファイル、VPN 添付ファイルまたはルータを作成するために使用されます。

（注） Catalyst 8000V（C8000V）は、シスコ クラウド サービス ルータ（CSR）1000V の進化型としてリリースされました。このドキュメントでは、C8000V を VXLAN 対応エッジ デバイスの例として使用します。詳細については、*Cisco Catalyst 8000V Edge ソフトウェアのリリース ノート* を参照してください。

Amazon Web Services で Nexus Dashboard クラスタを展開するための前提条件と注意事項

Amazon Web Services（AWS）で仮想 Nexus Dashboard クラスタを展開する前に、次の手順を実行する必要があります。

（注） AWS でサポートされる vND タイプは、32vCPU、128G RAM、3TB SSD （GP3）、および 10G のネットワーク スループットのみの vND データです。

- この機能は、単一の製品としてNexus Dashboard を備えた AWS の 3ノード仮想クラスタ（データ）でサポートされています（IPv4 のみ）。
- この機能では、LAN（NX- OS）および ACI ファブリックがサポートされます。また、次の機能が有効になっている場合のみサポートされます。
  - コントローラ
  - オーケストレーション
  - テレメトリ、次の制限あり：
    - アウトオブバンドのみ
    - トラフィック分析、ただしフロー テレメトリなし
- この機能は、LAN および ACI ファブリックでサポートされています。メディア用の IP ファブリック（IPFM）または SAN ファブリックではサポートされません。

（注） AI ファブリックは LAN ファブリックと見なされます。AWS の vND でのこのタイプのファブリックの展開は、ソリューションの観点からは制限されていません。AI ファブリックの基本的な要件は、デバイスと vND 間の遅延が50 ミリ秒を超えないようにすることです。

- セカンダリ/ワーカーノードは、この機能ではサポートされていません。3 つのプライマリ ノードと 1 つのスタンバイ ノードのみがサポートされます。
- クラスターあたりのスケール：単一の 3 ノード vNDクラスター上で 100 個のスイッチ
- AWS に vND を展開した後は、トンネルエンドポイントに割り当てられた IP アドレスまたは VNI を変更することはできません。
- Nexus Dashboard vND を展開する前に、オンプレミス サイトを準備して、Catalyst 8000 シリーズ ルータまたは N9000 スイッチのペアを展開しておくことがベストです。これにより、Nexus Dashboard vND のデプロイメントの間に、必要な BDI と TEP IP アドレスを提供できます。必要に応じて、 Nexus Dashboard vND を展開した後にオンプレミス ネットワーク アプライアンスを展開できますが、その場合、 Nexus Dashboard vND のデプロイメント時に提供したのと同じ情報を使用してオンプレミスデバイスを設定する必要があります。両方の場所で同じ設定情報を提供しない場合は、 Nexus Dashboard vNDを再インストールする必要があります。
- オンプレミスの VXLAN 対応デバイスが正しく構成されていることを確認します。
  - データプレーン：入力レプリケーション
  - コントロールプレーン：フラッディングと学習
- ファクターから AWS が拡張性とサービス要件をサポートしていることを確認します。

  クラスタ フォーム ファクタに基づいて、拡張性とサービス サポートおよび共同ホストは異なります。Nexus ダッシュボード キャパシティ プラン ツールを使用して、仮想フォーム ファクタが展開要件を満たすことを確認できます。
- 全般的な前提条件とガイドラインに記載されている一般的な前提条件を確認して完了します。
- 展開する予定のサービスのリリース ノートに記載されている追加の前提条件を確認して完了します。
- AWS アカウントに適切なアクセス権限があること。

  Nexus ダッシュボード クラスタをホストするには、複数の Elastic Compute Cloud（`m5.8xlarge`）のインスタンスを起動できる必要があります。
- Nexus ダッシュボード VM に使用される CPU ファミリが AVX 命令セットをサポートしていることを確認します。

- Nexus Dashboard クラスタ向け Amazon Web サービスの準備 （5 ページ）の手順を実行します。

Nexus Dashboard クラスタ向け Amazon Web サービスの準備

Nexus Dashboard vND をAmazon Web Services（AWS）に展開する前に、次の前提条件に従ってAWS を展開できるように準備します。

- AWS とその仕組みについて理解してください。
- （オプション）AWS とオンプレミスのデータセンター間の接続（理想的には、直接接続）を確立します。
- Nexus ダッシュボード ノードのデプロイメントに使用するリージョンを特定します。
- このデプロイメントに使用する既存の VPC を選択するか、新しい VPC を作成します。
- 外部アクセスを有効化します、

  これは、Elastic IP アドレスを vND 管理インターフェイスにマッピングし、 GUI と SSH に外部からアクセスするために必要です。選択した接続方法に応じて、外部アクセスを有効にするためにインターネット ゲートウェイを作成して VPC に接続する必要がある場合と、不要な場合があります。

  - **Option 1**：イーサネットインターフェイスプロセッサ（EIP）を使用して管理インターフェイスを接続します。この場合は、インターネット ゲートウェイが必要です。
  - **Option 2**：プライベート IP アドレスを使用します。この場合、インターネットゲートウェイは必要ありません。

- セキュリティ グループを更新して、次のような必要なサービスにパブリック IP アドレスまたは範囲からアクセスできるようにします。
  - HTTPS（TCP ポート 443）：Nexus Dashboard GUI にアクセスする場合
  - SSH（TCPポート 22）：vND ノードへセキュアなリモート ログインを行う場合

  これは、 GUI と SSH にアクセスして Nexus Dashboard ノードにアクセスできるようにするために必要です。

- 6 つのサブネットを作成します：
  - ノードごとに管理用のサブネット 1 セット（3）：最小 /28
  - ノードごとにデータ用のサブネット 1 セット（3）：最小 /28

  特定のノードの管理とデータのサブネットは、同じ可用性ゾーンに存在する必要があります。

- vND のデプロイメントに使用可能な十分な AWS Elastic IPアドレス があることを確認します。

  このインストールでは、ノードごとに 1 つずつ、合計 3 つの AWS Elastic IP アドレスが必要です。各 AWS Elastic IP アドレスは、vND UI やSSH へのアクセスなど、管理サービスにアクセスするために使用されます。

- 管理サブネットの IP アドレスは、デプロイメントの一部として AWS Elastic IP アドレスにマッピングされるため、これらの管理サブネットには外部アクセスが必要です。データサブネットは、vNDからVXLANトンネル（永続的なポッドによって使用される）の終端に使用されるオンプレミスのデバイスおよびオンプレミスの Catalyst 8000 シリーズ ルータ（または N9000 スイッチなどの他のデバイス）に到達できる必要があります。

- AWS によって所有されておらず、PIP（永続的な IP アドレス）によって使用されるオンプレミスのデータセンター（ただしまだ使用されていない）から取得される /28（100.100.100.0/28 など）のサブネット 1 つ。ここでそのサブネットの 100.100.100.1 と 100.100.100.2 は、オンプレミス データセンター デバイス（Catalyst 8000 または N9000 スイッチ）の BDI IP アドレスである必要があり、残りの IP アドレスは vND 永続ポッド（トラップ、テレメトリコレクタなど）によって使用されます。

- 適切に機能させるために、オンプレミスデバイスはこれらの永続的 IP アドレスと Nexus ダッシュボード データ IP アドレスに到達できる必要があります。

- vND ノードが相互に通信してクラスターを形成し、外部およびオンプレミスのデバイスと通信できるように、必要なすべての IP アドレスとポートを使用してセキュリティ グループを作成します。

- デプロイメント用に 1 つの EC2 キーペアを設定します。

（注） 前提条件の一部としてこの設定を完了します。ただし、EC2 キーペアは現在使用されておらず、現時点でサポートされているオプションはユーザー/パスワードのみです。

Amazon Web Services（AWS）に仮想 Nexus Dashboard（vND）を展開する

ここでは、Amazon Web Services (AWS) で仮想 Nexus Dashboard（vND）を展開する方法について説明します。

始める前に

- Amazon Web Services で Nexus Dashboard クラスタを展開するための前提条件と注意事項（3 ページ） に記載されている要件とガイドラインを満たしていることを確認します。

手順

ステップ 1 AWS Marketplace で Cisco Nexus ダッシュボード製品に登録します。

a) AWS アカウントにログインし、AWS Management Console に移動します。

管理コンソールは https://console.aws.amazon.com/ で入手できます。

b) **[サービス]** > **[AWS マーケットプレイス サブスクリプション（Services AWS Marketplace Subscriptions）]**に移動します。

c) **[サブスクリプションの管理（Manage Subscriptions）]** をクリックします。

d) **[製品の検出（Discover products）]** をクリックします。

e) **Cisco Nexus Dashboard - Cloud**を検索し、結果をクリックします。

f) **[購入の表示（View Purchase）]** オプションをクリックし、 **[登録（Subscribe）]**まで下にスクロールしてクリックします。

g) 製品ページで、 **[サブスクリプションの表示（View subscription）]**をクリックします。

h) **[サブスクリプションの管理（Manage subscriptions）]** ページで、 **[Cisco Nexus Dashboard-Cloud（Cisco Nexus Dashboard - Cloud）]**の行を見つけ、その行で **[起動（Launch）]** をクリックします。

ステップ 2 ソフトウェア オプションと地域を選択します。

a) **[Cisco Nexus Dashboard - Cloud]** の **[このソフトウェアを構成する（Configure this software）]**ページで、次を選択します。

- **履行オプション（Fulfillment option）** ：デフォルトの [**Nexus Dashboard - Cloud Deployment）]** の 選択をそのままにします。
- **ソフトウェア バージョン**：ドロップダウンリストから使用可能な最新の 4.2.1 オプションを選択します。
- **リージョン**：テンプレートを展開する適切なリージョンを選択します。

  これは、VPC を作成したのと同じリージョンである必要があります。

b) **[続行して起動する（Continue to Launch）]** をクリックします

c) **[Cisco Nexus Dashboard - Cloud]** の [**このソフトウェアの起動（Launch this software）]**ページで、**[アクションの選択（Choose Action）]** フィールドを見つけ、ドロップダウン リストから **[CloudFormation の起動（Launch CloudFormation）]** を選択して、 **[起動（Launch）]**をクリックします。

[Create Stack（スタックの作成）] ページが表示されます。

ステップ 3 スタック設定を完了します。

a) **[スタックの作成（Create stack）]** ページのオプションはそのままにします。

（注） 提供されているテンプレートに変更を加えないでください。スマート デフォルト テンプレート構成を使用することによってのみ、クラスタを正常に形成できます。

- **前提条件 -テンプレートを準備**：「既存のテンプレートを選択（Choose an existing template）」オプションを「そのまま」にします。
- **[テンプレートの指定（Specify Template）]**：[Amazon S3 URL] オプションはそのままにします。
- **Amazon S3 URL**：事前設定された URL エントリをそのままにします。

b) [**次へ（Next）**] をクリックして続行します。

[スタック詳細の指定（Specify stack details）] ページが表示されます。

ステップ 4 スタックの詳細を指定します。

a) **スタック名**を入力します。

b) **[パラメータ（Parameters）]** 領域に表示される情報を確認し、必要に応じて変更を加えます。

ほとんどの場合、この vND CFT の一部である構成に基づいて、事前に入力されたフィールドをそのままにすることができます。

- **[Nexus Dashboard クラスタ名（Nexus Dashboard Cluster Name）]** フィールドに、クラスタ名を入力します。
- **[ファブリック展開モード（Fabric Deployment Mode）]** フィールドで、デフォルトの**[LAN]** オプションは、 Nexus Dashboard 4.2.1 リリースでサポートされている唯一のオプションです。
- **[VPC 識別子（VPC identifier）]**フィールドに、VPC 識別子を入力します。

  アプリケーションの VPC がこのフィールドに自動的に入力されます。このフィールドの VPC を変更する場合は、次で別の VPC を選択します： **VPC dashboard** > **[仮想プライベート クラウド（Virtual private cloud）]** > **[使用中の VPC（Your VPCs）]**
- **[セキュリティ グループ ID（Security Group Identifier）]** フィールドに、セキュリティグループ IDを入力します。

  これは事前に作成されたセキュリティ グループで、ポート 22 および 443 への入力アクセスを許可する必要があります。
- **[インスタンス タイプ（Instance Type）]**フィールドで、ノード インスタンスの EC2 インスタンス タイプを指定します。
- **[AMI ID（AMI Identifier）]**フィールドで、Nexus Dashboard の AWS AMI を指定します。
- **[パスワード（Password）]** フィールドに、 Nexus Dashboard ノードの管理者パスワードを入力します。

  Nexus Dashboard ノードの管理者パスワードには、少なくとも 1 つの文字、数字、特殊文字（@$!%*#?&）を含める必要があり、長さは 8～64 文字にする必要があります。

- （オプション）**キーペア名（Key Pair Name）** フィールドで、 Nexus Dashboard への SSH アクセスを有効にする既存の SSH キーペアの名前を指定します。

c) **[DNS 構成（DNS Configuration）]** エリアで、必要に応じて設定情報を入力します。

- **[プライマリ DNS サーバー IP'（Primary DNS Server IP）]**フィールドに、プライマリ DNS サーバーの IP アドレスを入力します。
- **[セカンダリ DNS サーバー IP（Secondary DNS Server IP）]** フィールドに、セカンダリ DNS サーバーの IP アドレスを入力します。
- **[検索ドメイン名（Search Domain Name）]** フィールドに、検索ドメイン名を入力します。

d) (オプション) **[プロキシの設定（Proxy Configuration）]** エリアに必要な情報を入力します。

- **[プロキシ タイプ（Proxy Type）]**フィールドで、プロキシタイプ（ HTTP や HTTPS など）を指定します。
- **[プロキシ URL（Proxy URL）]** フィールドに、プロトコルとポートを含む完全なプロキシ URL を指定します。例：`http://proxy.example.com:8080`。
- **[プロキシ ユーザー名（Proxy Username）]** フィールドで、認証が必要な場合はプロキシのユーザー名を指定します。
- **[プロキシ パスワード（Proxy Password）]** 認証が必要な場合は、 フィールドでプロキシ パスワードを指定します。
- **[プロキシ無視ホスト IP（Proxy Ignore Hosts IP）]**フィールドで、プロキシが無視するホスト IP アドレスを指定します。

  このフィールドには 1 つのエントリのみが許可されます（たとえば、192.168.10.101）。

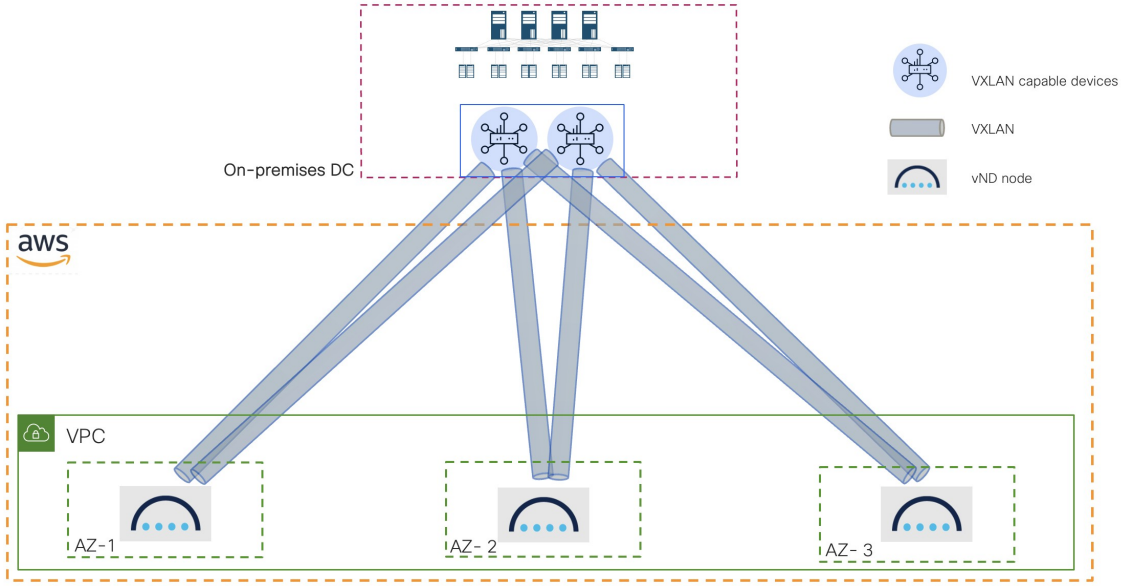
e) **[NTP 構成（NTP Configuration）]**エリアで、必要に応じて設定情報を入力します。

- **[NTP サーバー ホスト（NTP Server Host）]**フィールドで、NTP サーバー ホストを指定します。
- **[NTP サーバー キー識別子（NTP Server Key Identifier）]** フィールドで、 NTPサーバーキー識別子を指定します。
- **[NTPサーバー優先（NTP Server Preferred）]** フィールドで、サーバーが優先される場合は `true` を選択します。
- **[NTP キー識別子（NTP Key Identifier）]** フィールドで、NTP キーの識別子を指定します。
- **[NTP キー（NTP Key）]**フィールドで、 NTP キーのキーを指定します。
- **[NTP キー認証（NTP Key Authentication Type）]**フィールドで、 NTP キーの認証タイプ（MDS や SHA1 など）を指定します。
- **[信頼されている NTP キー（NTP Key Trusted）]** フィールドで、 NTP キーが信頼できるかどうかを `true` または `false` で指定します。

f) **[Cisco VXLAN 対応デバイス（Cisco VXLAN Capable Device）]** 領域に必要な情報を入力します。

- **[デバイス VXLAN ID（Device VXLAN Identifier） (VNI)]** フィールドに、Cisco VXLAN 対応デバイス（IPSec トンネルを使用している場合）または N9000 スイッチ（直接接続を使用している場合）と Nexus Dashboard ノード間の VXLAN トンネルに使用する VNI 値を入力します。

  この図に示すように、Cisco VXLAN 対応デバイスと Nexus Dashboard ノード（vND）間のすべての VXLAN トンネルに単一の VNI 値が使用されます。



- **[デバイス 1 ブリッジ ドメイン IP（Device 1 Bridge Domain IP）]** および **[デバイス 2 ブリッジ ドメイン IP（Device 2 Bridge Domain IP）]** フィールドに、両方の Cisco VXLAN 対応デバイスのブリッジ ドメイン IP アドレスを入力します。

  デバイスのブリッジドメイン IP アドレスは、**[Nexus Dashboard ポッドで使用されるプライベート IP サブネット（Private IP Subnet for Nexus Dashboard Pods）]** フィールドで指定したサブネットからのものである必要があります：たとえば、**[Nexus Dashboard ポッドのプライベート IP サブネット（Private IP Subnet for Nexus Dashboard Pods）]** フィールドに `100.100.100.0/28` と入力した場合、デバイスのブリッジドメインIPアドレスとして `100.100.100.1` および `100.100.100.2` を入力できます。

- **[デバイス 1 トンネル エンドポイント IP（Device 1 Tunnel Endpoint IP）]** および **[デバイス 2 トンネル エンドポイント IP（Device 2 Tunnel Endpoint IP）]** フィールドに、両方の Cisco VXLAN 対応デバイスのトンネル エンドポイント IP アドレス（データ IP アドレス）を入力します。

- **[Nexus Dashboard ポッドのプライベート IP サブネット（Private IP Subnet for Nexus Dashboard Pods）]** フィールドに、 Nexus Dashboard ポッドで使用されるプライベート IP サブネットを入力します。

  IPサブネットのサイズは、`100.100.100.0/28` である必要があります。

（注） この項のこの手順では、Cisco VXLAN 対応デバイスは展開されません。ルールは、 Nexus Dashboard にエッジ デバイスとの接続を確立するために必要なすべての変数があることを確認するだけです。

g) **[Nexus Dashboard ノード 1 構成（Nexus Dashboard Node 1 Configuration）]**、**[Nexus Dashboard ノード 2 構成（Nexus Dashboard Node 2 Configuration）]**、および **[Nexus Dashboardノード 3 構成（Nexus Dashboard Node 3 Configuration）]** 領域で、クラスタ内の各 vND ノードに必要な情報を入力します。

- **ND ノード *x* ホスト名**：各 Nexus Dashboard ノードのホスト名を入力します。
- **ND ノード *x* 管理サブネット**：各 Nexus Dashboard ノードの最初の管理サブネットを入力します。
- **ND ノード *x* 静的管理 IP**：上記で入力した管理サブネットの静的管理 IP アドレスを、 Nexus Dashboardノードごとに入力します。

  このフィールドに入力した IP アドレスがまだ使用されていないことを確認します。
- **ND ノード *x* 管理サブネット ネットマスク**：各 Nexus Dashboard ノードの最初の管理サブネット ネットマスクを CIDR 形式（16～28）で入力します。
- **ND ノード *x* 管理サブネット ゲートウェイ**：上記で入力した管理サブネット上の最初の管理デフォルト ゲートウェイを、Nexus Dashboard ノードに対して入力します。

  これは通常、サブネットの最初のアドレスです。
- **ND ノード *x* データ サブネット**：各 Nexus Dashboard ノードの最初のデータ サブネットを入力します。
- **ND ノード *x* 静的データ IP**:：上記で入力したデータ サブネットの静的データ IP アドレスを、Nexus Dashboard ノードごとに入力します。

  このフィールドに入力した IP アドレスがまだ使用されていないことを確認します。
- **ND ノード *x* データ サブネット ネットマスク**：各 Nexus Dashboard ノードの最初のデータ サブネット ネットマスクを CIDR 形式（16～28）で入力します。
- **ND ノード *x* データ サブネット ゲートウェイ**：上記で入力したデータ サブネット上の最初のデータ デフォルトゲートウェイを、Nexus Dashboard ノードに対して入力します。

  これは通常、サブネットの最初のアドレスです。

h) **[Kubernetes網構成（オプション）（Kubernetes Network Configuration (Optional)）]** 領域で、必要に応じて構成情報を入力します。

- **[Kubernetesサービスネットワーク（Kubernetes Service Network）]** フィールドで、Kubernetes サービス ネットワークのネットワーク アドレスを指定します。

  CIDR 範囲は「/16」固定です。

- **[Kubernetes アプリ ネットワーク（Kubernetes App Network）]** フィールドに、kubernetes アプリネットワークのネットワーク アドレスを入力します。

  CIDR 範囲は「/16」固定です。

i) [**次へ（Next）**] をクリックして続行します。

ステップ 5 **[スタック構成オプション（Configure stack options）]** ページで、必要に応じて、このページに記載されている情報を確認および変更します。

a) **[スタック障害オプション（Stack failure options）]** で、**[プロビジョニングの失敗時の動作（Behavior on provisioning failure）]** の選択を [正常にプロビジョニングされたリソースを保持する （**Preserve successfully provisioned resources**）] に変更することをお勧めします。

b) **[スタック オプションの構成（Configure stack opushon ）]** ページで情報の確認または変更が完了したら、**[次へ（Next）]** をクリックします。

ステップ 6 **[確認して作成（Review and create）]** ページで、テンプレート設定情報を確認してから、**[送信（Submit）]** をクリックします。

ステップ 7 展開が完了するのを待ってから、VM を起動します。

[CloudFormation] > **[スタック（Stacks）]** ページでインスタンスの展開のステータス（`CREATE_IN_PROGRESS` など）を表示できます。ページの右上隅にある更新ボタンをクリックすると、ステータスを更新できます。

スタックのステータスが `CREATE_COMPLETE` に変わったら、次の手順に進むことができます。

ステップ 8 **[CloudFormation]** **[** > **[スタック（Stacks）]** の下のスタックで、**[出力（Outputs）]** タブをクリックして、クラスタ内の 3 つの vND のパブリック IP アドレスを表示します。

（注） CloudFormation テンプレートは、クラスター内の接続を処理します。すべての変数が正しく入力されている場合、ノードは自動的にクラスターを形成します。

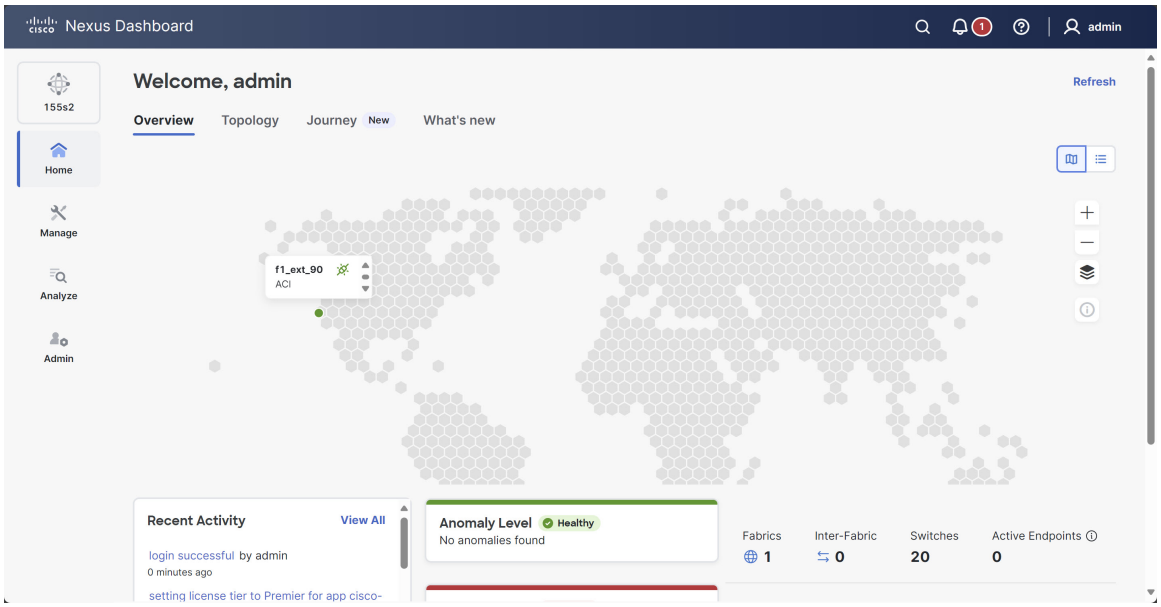
ステップ 9 前の手順にリストされているパブリック IP アドレスのいずれかを使用して、Nexus Dashboard GUI にログインします。

（注） VM を起動してから、パブリック IP アドレスの 1 つを使用して Nexus Dashboard GUI にログインできるまでに約 40 分かかる場合があります。

ステップ 10 クラスタが健全であることを検証します。

クラスタが使用可能になったら、ノードの管理 IP アドレスのいずれかを参照してアクセスできます。`admin` ユーザーのデフォルト パスワードは、最初のノードに選択した `rescue-user` のパスワードと同じです。この間、UI は上部に「サービスのインストールが進行中です。Nexus Dashboard の設定タスクは現在無効になっています」という意味のバナーを表示します。

すべてのクラスタが展開され、すべてのサービスが開始されたら**[ホーム（Home）]**>**[概要（Overview）]**ページの **異常レベル（Anomaly Level）** でクラスタが正常であることを確認できます。



または、SSH を使用し、`rescue-user` として、ノード展開中に入力したパスワードを使っていずれかのノードにログインし、`acs health` コマンドを実行してクラスタの状態を確認できます。

- クラスタが収束している間、次の出力が表示されることがあります：

```
$ acs health
k8s install is in-progress
```

```
$ acs health
k8s services not in desired state - [...]
```

```
$ acs health
k8s: Etcd cluster is not ready
```

- クラスタが稼働している場合は、次の出力が表示されます。

```
$ acs health
All components are healthy
```

（注） 場合によっては、ノードの電源を再投入（電源をオフにしてから再度オン）すると、この段階でスタックが停止することがある可能性があります。

```
deploy base system services
```

これは、pND（物理 Nexus Dashboard）クラスタの再起動後のノードの `etcd` の問題が原因です。

この問題を解決するには、影響を受けるノードで `acs reboot clean` コマンドを入力します。

ステップ 11 （オプション） Cisco Nexus DashboardクラスターをCisco Intersightに接続、可視性と利点を強化します。詳細な手順については、 「*Cisco Intersight*の操作」 を参照してください。

ステップ 12 Nexus Dashboardを展開した後、設定情報については、このリリースのコレクションページを参照してください。

次のタスク

次のタスクは、ファブリックとファブリック グループを作成することです。 Cisco Nexus Dashboardのコレクション ページ にある、このリリースの 「ファブリックとファブリック グループの作成」の記事を参照してください。

ノードの交換（RMA）

AWS ベースの Nexus Dashboard の展開では、ノードが何らかの理由で RMA または交換が必要な場合、新しい交換用ノードをAWS に展開する必要があります。

オンプレミスの物理 Nexus Dashboard（pND）およびリモート対応 Nexus Dashboard（vND）の展開で使用可能な工場出荷時の状態へのリセットとノードの復元オプションは、AWS ベースのノードではサポートされていません。したがって、影響を受けたノードをその場所で回復することはできず、新しいノードとして再展開する必要があります。

元の 3 ノード CloudFormation（CFT）テンプレートを使用して個々のノードを置き換えないでください。単一ノードの交換の場合は、以下に示す専用の単一ノード CFT テンプレートを使用して新しいノードを展開します。

```
{
    "AWSTemplateFormatVersion": "2010-09-09",
    "Description": "This cloud formation template automates the creation and launching
of a Cisco Nexus Dashboard cluster (vND data 32vCPU/128GMem/10G Network/3T Disk) in
multiple Availability Zones (AZs) of AWS. Deployment requires a Application VPC to exist
 in the account from which the cluster will be is launched. For detailed, description
of this template and usage instructions please refer to ND deployment guide.",
    "Metadata": {
        "AWS::CloudFormation::Interface": {
            "ParameterGroups": [
                {
                    "Label": {
                        "default": "Nexus Dashboard Basic Configuration"
                    },
                    "Parameters": [
                        "NDClusterName",
                        "FabricPersona",
                        "VpcId",
                        "SecurityGroupId",
                        "InstanceType",
                        "AMIId",
                        "AdminPassword",
                        "KeyPairName",
                        "ExistingEIPAllocationId1",
                        "ExistingEIPAddress1"
                    ]
                },
```

```
                {
                    "Label": {
                        "default": "Nexus Dashboard Node Configuration"
                    },
                    "Parameters": [
                        "hostNameND1",
                        "SubnetNDMgmt1",
                        "StaticNDMgmt1IP",
                        "SubnetNDMgmt1Netmask",
                        "SubnetNDMgmt1Gateway",
                        "SubnetNDData1",
                        "StaticNDData1IP",
                        "SubnetNDData1Netmask",
                        "SubnetNDData1Gateway"
                    ]
                }
            ],
            "ParameterLabels": {
                "NDClusterName": { "default": "ND Cluster Name" },
                "VpcId": { "default": "VPC identifier" },
                "hostNameND1": { "default": "ND Node Hostname" },
                "SubnetNDMgmt1": { "default": "ND Node Management Subnet" },
                "StaticNDMgmt1IP": { "default": "ND Node Static Management IP" },
               "SubnetNDMgmt1Netmask": { "default": "ND Node Management Subnet Netmask"
},
               "SubnetNDMgmt1Gateway": { "default": "ND Node Management Subnet Gateway"
},
                "SubnetNDData1": { "default": "ND Node Data Subnet" },
                "StaticNDData1IP": { "default": "ND Node Static Data IP" },
                "SubnetNDData1Netmask": { "default": "ND Node Data Subnet Netmask" },
                "SubnetNDData1Gateway": { "default": "ND Node Data Subnet Gateway" },
                "SecurityGroupId": { "default": "Security Group identifier" },
                "InstanceType": { "default": "Instance Type" },
                "AMIId": { "default": "AMI identifier" },
                "AdminPassword": { "default": "Password" },
                "KeyPairName": { "default": "Key Pair Name" },
                "ExistingEIPAllocationId1": { "default": "ND Node 1 EIP Allocation ID."
},
                "ExistingEIPAddress1": { "default": "ND Node 1 Elastic IP." }
            }
        }
    },
    "Parameters": {
        "NDClusterName": {
            "Type": "String",
            "Description": "Nexus Dashboard Cluster name.",
            "MinLength": 1
        },
        "VpcId": {
            "Type": "String",
           "Description": "VpcId of your existing Virtual Private Cloud (VPC) to launch
ND cluster.",
            "ConstraintDescription": "Must be the VPC Id of an existing Virtual Private
Cloud."
        },
        "hostNameND1": {
            "Type": "String",
            "Description": "Specify the hostname for ND Node 1.",
            "Default": "AWSND1"
        },
        "SubnetNDMgmt1": {
            "Type": "AWS::EC2::Subnet::Id",
            "Description": "Select the 1st management subnet for the ND Node 1."
        },
```

```
        "StaticNDMgmt1IP": {
            "Type": "String",
            "Description": "Specify a static management IP address from the selected
subnet. Ensure not in-use already.",
            "AllowedPattern": "^(\\d{1,3})\\.(\\d{1,3})\\.(\\d{1,3})\\.(\\d{1,3})$",
            "Default": "0.0.0.0"
        },
        "SubnetNDMgmt1Netmask": {
            "Type": "String",
            "Description": "Specify the 1st management subnet netmask in CIDR notation
(16-28).",
            "Default": "24",
            "AllowedPattern": "^(1[6-9]|2[0-8])$"
        },
        "SubnetNDMgmt1Gateway": {
            "Type": "String",
            "Description": "Specify the 1st management default gateway on the selected
management subnet. This is typically the first address on the subnet.",
            "AllowedPattern": "^(\\d{1,3})\\.(\\d{1,3})\\.(\\d{1,3})\\.(\\d{1,3})$",
            "Default": "0.0.0.0"
        },
        "SubnetNDData1": {
            "Type": "AWS::EC2::Subnet::Id",
            "Description": "Select the 1st data subnet for the ND Node 1."
        },
        "StaticNDData1IP": {
            "Type": "String",
            "Description": "Specify a static data IP address from the selected subnet.
Ensure not in-use already.",
            "AllowedPattern": "^(\\d{1,3})\\.(\\d{1,3})\\.(\\d{1,3})\\.(\\d{1,3})$",
            "Default": "0.0.0.0"
        },
        "SubnetNDData1Netmask": {
            "Type": "String",
          "Description": "Specify the 1st data subnet netmask in CIDR notation (16-28).",

            "Default": "24",
            "AllowedPattern": "^(1[6-9]|2[0-8])$"
        },
        "SubnetNDData1Gateway": {
            "Type": "String",
          "Description": "Specify the 1st data default gateway on the selected management
 subnet. This is typically the first address on the subnet.",
            "AllowedPattern": "^(\\d{1,3})\\.(\\d{1,3})\\.(\\d{1,3})\\.(\\d{1,3})$",
            "Default": "0.0.0.0"
        },
        "SecurityGroupId": {
            "Type": "AWS::EC2::SecurityGroup::Id",
           "Description": "Pre-created security group to be applied. Must allow ingress
 access for ports 22, 443. Check docs for other needed ports."
        },
        "InstanceType": {
            "Type": "String",
            "Description": "Specify EC2 instance type for the node instances.",
            "AllowedValues": ["m5.8xlarge"],
            "Default": "m5.8xlarge"
        },
        "AMIId": {
            "Type": "AWS::EC2::Image::Id",
            "Description": "Specify the AWS AMI for Nexus Dashboard."
        },
        "AdminPassword": {
            "Type": "String",
            "Description": "Admin user password for ND node (must contain atleast 1
```

```
letter, number and special char @$!%*#?& length: 8-64 chars).",
            "NoEcho": true,
            "MinLength": 8,
            "MaxLength": 64,
            "AllowedPattern":
"^(?=.*[A-Za-z])(?=.*\\d)(?=.*[@$!%*#?&])[A-Za-z\\d@$!%*#?&]{8,64}$"
        },
        "KeyPairName": {
            "Type": "AWS::EC2::KeyPair::KeyName",
           "Description": "Required - Specify Name of an existing SSH KeyPair to enable
 SSH access to ND."
        },
        "ExistingEIPAllocationId1": {
            "Type": "String",
            "Description": "Allocation ID of an existing Elastic IP for ND Node 1."
        },
        "ExistingEIPAddress1": {
            "Type": "String",
            "Description": "Public IP of the existing Elastic IP (e.g., 203.0.113.42)
for ND Node 1.",
            "Default": "0.0.0.0"
        }
    },
    "Resources": {
        "ND1": {
            "Type": "AWS::EC2::Instance",
            "Properties": {
                "InstanceType": { "Ref": "InstanceType" },
                "ImageId": { "Ref": "AMIId" },
                "KeyName": { "Ref": "KeyPairName" },
                "NetworkInterfaces": [
                    {
                        "DeviceIndex": 0,
                        "NetworkInterfaceId": { "Ref": "ND1MgmtENI" }
                    },
                    {
                        "DeviceIndex": 1,
                        "NetworkInterfaceId": { "Ref": "ND1DataENI" }
                    }
                ],
                "EbsOptimized": true,
                "BlockDeviceMappings": [
                    {
                        "DeviceName": "/dev/xvda",
                        "Ebs": { "VolumeType": "gp3", "Iops": 5000 }
                    },
                    {
                        "DeviceName": "/dev/xvdh",
                        "Ebs": {
                            "VolumeSize": 3072,
                            "VolumeType": "gp3",
                            "Iops": 5000
                        }
                    }
                ],
                "Tags": [{ "Key": "Name", "Value": "ND1-Master" }],
                "UserData": {
                    "Fn::Base64": {
                        "Fn::Sub": "{\n \"password\": \"${AdminPassword}\",\n
\"clusterName\": \"${NDClusterName}\",\n \"oobNetwork\": {\n    \"ipSubnet\":
\"${StaticNDMgmt1IP}/${SubnetNDMgmt1Netmask}\",\n    \"gateway\":
\"${SubnetNDMgmt1Gateway}\"\n  }\n}\n"
                    }
                }
```

```
                }
            },
            "ND1MgmtENI": {
                "Type": "AWS::EC2::NetworkInterface",
                "Properties": {
                    "SubnetId": { "Ref": "SubnetNDMgmt1" },
                    "PrivateIpAddresses": [
                        {
                            "PrivateIpAddress": { "Ref": "StaticNDMgmt1IP" },
                            "Primary": true
                        }
                    ],
                    "GroupSet": [{ "Ref": "SecurityGroupId" }]
                }
            },
            "EIPAssoc1": {
                "Type": "AWS::EC2::EIPAssociation",
                "Properties": {
                    "NetworkInterfaceId": { "Ref": "ND1MgmtENI" },
                    "AllocationId": { "Ref": "ExistingEIPAllocationId1" }
                }
            },
            "ND1DataENI": {
                "Type": "AWS::EC2::NetworkInterface",
                "Properties": {
                    "SubnetId": { "Ref": "SubnetNDData1" },
                    "PrivateIpAddresses": [
                        {
                            "PrivateIpAddress": { "Ref": "StaticNDData1IP" },
                            "Primary": true
                        }
                    ],
                    "GroupSet": [{ "Ref": "SecurityGroupId" }]
                }
            }
        },
        "Outputs": {
            "ND1PublicIP": {
                "Description": "Public IP of ND Node 1.",
                "Value": { "Ref": "ExistingEIPAddress1" }
            }
        }
    }
```

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。