



# OpenNebula 5.10 導入ガイド

リリース 5.10.5

OpenNebula Systems

2020年9月21日

本ドキュメントは、クリエイティブ・コモンズ 表示-非営利-継承ライセンスに基づき、OpenNebula Systems より提供されております。

本ドキュメントは「現状のまま」提供され、明示的または默示的を問わず、商品性、特定目的への適合性、および権利非侵害に関する保証を含むがこれらに限定されない、いかなる種類の保証も付随しません。いかなる場合においても、著者または著作権者は、契約、不法行為、その他を問わず、本文書に起因、関連、または関連して生じる請求、損害、その他の責任について、一切の責任を負いません。

## 目次

<b>1 クラウド設計</b>	<b>1</b>
1.1 概要 .....	1
1.2 オープンクラウドアーキテクチャ .....	2
1.3 VMware Cloud アーキテクチャ .....	7
1.4 OpenNebula プロビジョニングモデル .....	13
<b>2 OpenNebula のインストール</b>	<b>18</b>
2.1 概要 .....	18
2.2 フロントエンドのインストール .....	18
2.3 MySQLの設定 .....	26
<b>3 Node のインストール</b>	<b>29</b>
3.1 概要 .....	29
3.2 KVMノードのインストール .....	30
3.3 LXDノードのインストール .....	37
3.4 vCenter ノードのインストール .....	39
3.5 インストール内容の確認 .....	47
<b>4 認証の設定</b>	<b>56</b>
4.1 概要 .....	56
4.2 SSH認証 .....	57
4.3 x509 認証 .....	59
4.4 LDAP認証 .....	62
<b>5 サンストーンの設定</b>	<b>68</b>
5.1 概要 .....	68
5.2 サンストーンのインストールと設定 .....	69
5.3 サンストーンのビュー .....	78
5.4 ラベル .....	82
5.5 ユーザーセキュリティと認証 .....	92
5.6 クラウドサーバー認証 .....	98

5.7 大規模導入のための Sunstone の設定	101
<b>6 VMware インフラストラクチャの設定</b>	<b>107</b>
6.1 概要	107
6.2 vCenter ドライバーの設定	107
6.3 vCenter データストア	129
6.4 vCenter ネットワークの概要	137
6.5 NSX の設定	144
<b>7 オープンクラウドホストの設定</b>	<b>153</b>
7.1 概要	153
7.2 KVM ドライバー	153
7.3 LXD ドライバー	163
7.4 監視	169
7.5 PCI/バススルー	173
<b>8 オープンクラウドストレージの設定</b>	<b>180</b>
8.1 概要	180
8.2 ファイルシステム データストア	182
8.3 Ceph データストア	187
8.4 LVM データストア	193
8.5 Raw Device Mapping (RDM) データストア	196
8.6 iSCSI - Libvirt データストア	198
8.7 カーネルヒファイル データストア	200
<b>9 オープンクラウドネットワーキングの設定</b>	<b>202</b>
9.1 概要	202
9.2 ノードの設定	203
9.3 ブリッジド・ネットワーキング	205
9.4 802.1Q VLAN ネットワーク	206
9.5 VXLANネットワーク	208
9.6 Open vSwitch ネットワーク	212
<b>10 参考文献</b>	<b>216</b>
10.1 概要	216
10.2 ONED設定	216
10.3 ログ記録とデバッグ	233
10.4 Onedbツール	236
10.5 スケーラビリティテストとチューニング	240

## クラウド設計

### 1.1 概要

信頼性が高く、有用で成功するクラウドを構築する最初のステップは、明確な設計を決定することです。この設計は、クラウドの想定される用途と整合している必要があり、また、クラウドの一部となるデータセンターコンポーネントを記述する必要があります。これには、i) ネットワーク、ストレージ、認証、仮想化バックエンドなどのすべてのインフラストラクチャコンポーネント、ii) クラウドの計画規模（ワークロードの特性、ユーザー数など）、iii) プロビジョニングワークフロー、すなわちエンドユーザーがどのように分離され、クラウドを利用するかが含まれます。

OpenNebula クラウドを最大限に活用するためには、導入において求める機能、パフォーマンス、スケーラビリティ、高可用性の特性を盛り込んだ計画を作成されることをお勧めいたします。本章では、*KVM*、*LXD*、または *vCenter* に基づく OpenNebula クラウドの計画立案に関する情報を提供いたします。この情報により、導入のアーキテクチャ設計や規模設定を容易に行えるほか、仮想化リソースの管理に関わる技術とその相互関係を理解することができます。

#### 1.1.1 本章の読み方について

本章はクラウドアーキテクチャを正しく定義するために必要な概念を紹介する最初の章となります。

本章では、まず第一段階として、クラウドの設計とその規模の概要を策定する必要があります。KVMおよびLXDクラウドの場合は「*Open Cloud Architecture*」へ、vCenterクラウドの場合は「*VMware Cloud Architecture*」をお読みください。

次に、*OpenNebula*のプロビジョニングモデルをご参照いただき、エンドユーザーにリソースをプロビジョニングするための適切なモデルを特定してください。ホスト数が少ない小規模な環境では、このプロビジョニングモデルのガイドを省略し、インフラストラクチャの分割やプロビジョニングについて深く考慮せずにOpenNebulaをご利用いただけます。ただし、中規模から大規模な展開においては、ある程度の分離性と構造を提供することが望ましいでしょう。

クラウドアーキテクチャの設計が完了しましたら、次のステップとして*OpenNebula*フロントエンドのインストール方法について学ぶことになります。

#### 1.1.2 ハイパーバイザー互換性

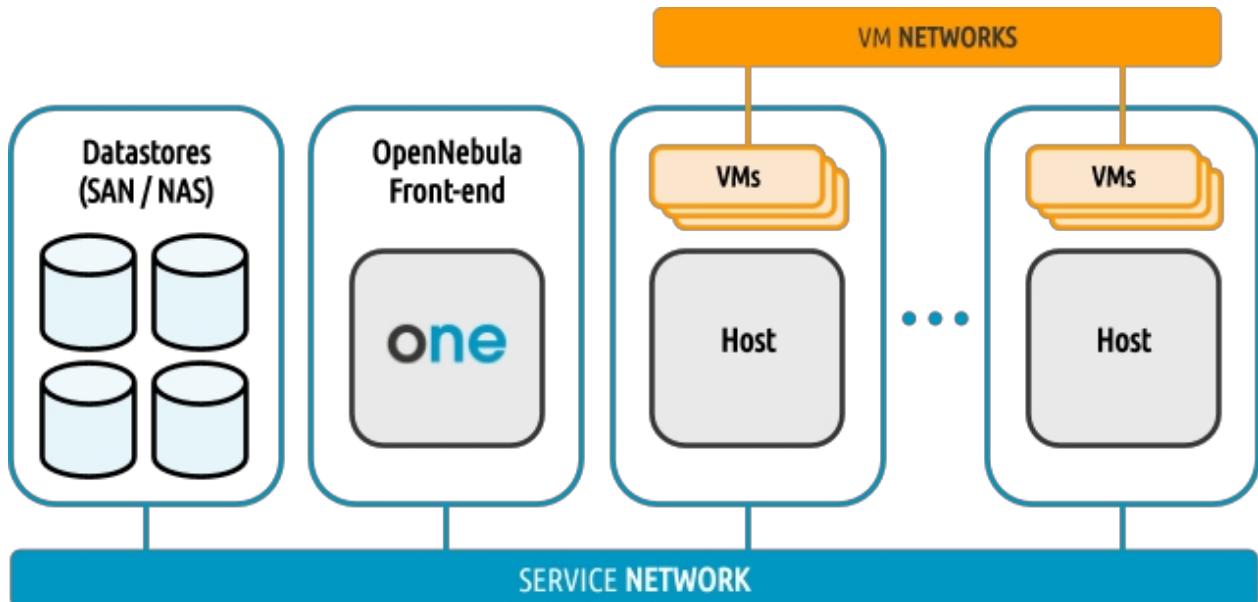
セクション	互換性
<i>オープンクラウドアーキテクチャ</i>	このセクションは、KVM および LXD に適用されます。
<i>VMware Cloud アーキテクチャ</i>	このセクションは vCenter に適用されます。
<i>OpenNebula プロビジョニングモデル</i>	このセクションは、KVM および vCenter の両方に適用されます。

## 1.2 オープンクラウドアーキテクチャ

エンタープライズ向けクラウドコンピューティングは、データセンター（DC）仮想化の進化における次の段階です。OpenNebulaは、既存の仮想化技術にマルチテナント対応、自動プロビジョニング、伸縮性といった先進機能を組み合わせた、シンプルであります。ながら機能豊富で柔軟なソリューションであり、エンタープライズクラウドや仮想化データセンターの構築・管理を実現します。OpenNebulaは、システム管理者、DevOps担当者、ユーザーの実際のニーズに基づいて推進されるボトムアップ型のアプローチを採用しています。

### 1.2.1 アーキテクチャ概要

OpenNebulaは、物理インフラストラクチャがフロントエンドと仮想マシン（VM）を実行するホスト群からなる古典的なクラスター型アーキテクチャを採用していることを前提としています。すべてのホストとフロントエンドを接続する物理ネットワークが少なくとも1つ存在します。



クラウドアーキテクチャは、ストレージ、ネットワーク、仮想化の3つのコンポーネントによって定義されます。したがって、OpenNebulaシステムの基本コンポーネントは以下の通りです：

- OpenNebulaサービスを実行するフロントエンド。
- 仮想マシンに必要なリソースを提供するハイパーバイザ対応ホスト。
- 仮想マシンのベースイメージを保持するデータストア。
- ストレージサーバー間の相互接続やOpenNebulaの制御操作といった基本サービス、および仮想マシン向けのVLANをサポートするために使用される物理ネットワーク。

OpenNebulaは、汎用ハイパーバイザからエンタープライズグレードのハイパーバイザまで幅広くサポートする、高度にモジュール化されたアーキテクチャを提供します。監視、ストレージ、ネットワーク、ユーザー管理サービスも同様にサポート対象です。本セクションでは、各サブシステムの管理において選択可能な異なるオプションについて簡潔に説明します。ご利用の特定のサービスがサポートされていない場合は、[アドオンカタログ](#)で利用可能なドライバーをご確認いただくことをお勧めいたします。また、新規ドライバーの開発方法に関する情報とサポートも提供しております。

## 1.2.2 クラウドの規模設計

クラウドインフラの規模は、そのインフラが支えなければならない仮想マシン（VM）の観点から、予想されるワークロードから直接推測できます。このワークロードの見積もりも難しい作業ではありますが、効率的なクラウドを構築するためには極めて重要な作業です。

効率的なクラウドを構築するためには、極めて重要な作業です。

OpenNebulaクラウドの規模設計時に考慮すべき主な要素は以下の通りです。

### OpenNebula フロントエンド

OpenNebulaフロントエンドの推奨最小スペックは以下の通りです：

リソース	最小推奨構成
メモリ	8ギガバイト
CPU	1 CPU (4コア)
ディスク容量	100 GB
ネットワーク	2 NIC

これらの推奨事項はあくまで目安であり、クラウドの規模やワークロードに応じて緩和または増加させる可能性があることをご留意ください。

単一の OpenNebula インスタンスで管理可能なサーバー（仮想化ホスト）の最大数は、基盤となるプラットフォームインフラストラクチャ、主にストレージサブシステムのパフォーマンスとスケーラビリティに大きく依存します。一般的な推奨事項としては、単一のインスタンスで管理するサーバーは 2,500 台以下、仮想マシンは 10,000 台以下とされています。これに関連して、[大規模な導入環境向けに OpenNebula を調整する方法についてのセクション](#)をお読みください。

### KVMノード

Linux仮想化ノードの規模について：

- **CPU**：オーバーコミットメントを行わない場合、仮想マシン（VM）に割り当てられた各CPUコアは、物理CPUコアとして存在している必要があります。例えば、2つのCPUを持つ40台のVMからなるワークロードの場合、クラウドには80個の物理CPUが必要となります。これらの80個の物理CPUは、異なるホストに分散配置することができます。例えば、8コアずつ搭載したサーバー10台、あるいは16コアずつ搭載したサーバー5台といった構成が考えられます。ただし、オーバーコミットメントを有効にした場合、CPUの規模は事前に計画することができます。その際にはCPU属性とVCPU属性を使用します。CPU属性はVMに割り当てられた物理CPUを表し、VCPU属性はゲストOSに提示される仮想CPUを表します。
- **メモリ**：メモリの計画は比較的簡単です。OpenNebulaではデフォルトでメモリのオーバーコミットメントが行われないためです。ハイパーバイザーのオーバーヘッドとして常に10%を見込むことが推奨されます。（これは絶対的な上限値ではなく、ハイパーバイザーによって異なります。ただし、コンテナのオーバーヘッドは非常に低いため、LXDノードではこの指標は無視できる程度です。）したがって、各仮想マシンに2GBのRAMを割り当てた45台の仮想マシンワークロードを維持するには、90GBの物理メモリが必要となります。ホストの数は重要です。各ホストはハイパーバイザーにより10%のオーバーヘッドが発生するためです。例えば、10GBのRAMを搭載した10台のハイパーバイザーは、それぞれ9GB（10GBの10% = 1GB）を割り当てます。したがって、これらは推定ワークロードを維持できます。経験則として、コアあたり最低1GBを確保することが推奨されますが、これは予想されるワークロードにも依存します。

### LXDノード

LXDは仮想ハードウェアの使用を回避するため、KVMと比較した場合、ハイパーバイザーに割り当てられるリソース量は大幅に少なくなります。正確な数値はございませんが、[KVMとのリソース使用量比較](#)から概ねおわかりいただけるかと思います。

### ストレージ

OpenNebulaがストレージをどのように使用するかを理解することは重要です。特に、システムデータストアとイメージデータストアの違いが重要です。

- 
- ・ **イメージデータストア**は、OpenNebula が登録したすべてのイメージを保存する場所であり、これらは VM 作成に使用されます。したがって、OpenNebula に登録されるすべてのイメージを収容できる十分なスペースを確保することが基本原則となります。
  - ・ **システムデータストア**は、現在稼働中の仮想マシンがディスクを保存する場所です。イメージデータストアに対応する保存場所がない揮発性ディスクが使用されるため、正確な見積もりはより複雑になります（揮発性ディスクはハイパーバイザー上で動的に作成されます）。

有効なアプローチの一つとして、ユーザーが要求できる最大VM数と最大揮発性ストレージ容量をクォータで定義し、利用可能なストレージを制限する方法が挙げられます。これにより、クォータで設定された制限を満たす十分なシステムおよびイメージデータストアの容量を確保できます。いずれにせよ、OpenNebulaではクラウド管理者が必要に応じてシステムおよびイメージデータストアを追加することが可能です。

いずれにせよ、OpenNebulaではクラウド管理者が必要に応じてシステムおよびイメージデータストアを追加することが可能です。

ストレージの容量設計は、通常クラウドのボトルネックとなるため、極めて重要な要素です。これは基盤となる技術に大きく依存します。例えば、中規模クラウドにおけるCephの場合、ストレージには少なくとも3台のサーバーが必要となります。各サーバーには1TBのディスク5台、16GBのRAM、4コアのCPUを2基、そして少なくとも2枚のNICが搭載されている必要があります。

#### ネットワーク

クラウドインフラストラクチャの信頼性を確保するためには、ネットワーク設計を慎重に行う必要があります。推奨される構成は、フロントエンドに2枚のNIC（パブリックネットワークとサービスネットワーク）、ストレージバックエンドによっては3枚、さらに各仮想化ノードに4枚のNIC（プライベートネットワーク、パブリックネットワーク、サービスネットワーク、ストレージネットワーク）を配置することです。ストレージとネットワークの構成によっては、より少ないNIC数で済む場合もあります。

### 1.2.3 フロントエンド

OpenNebulaをインストールするマシンはフロントエンドと呼ばれます。このマシンは、すべてのホストへのネットワーク接続が必要であり、ストレージデータストアへのアクセス（直接マウントまたはネットワーク経由）も必要となる場合があります。OpenNebulaの基本インストールに必要な容量は150MB未満です。

OpenNebulaのサービスには以下が含まれます：

- 管理デーモン (`oned`) およびスケジューラ (`mm_sched`)
- Webインターフェースサーバー (`sunstone-server`)
- 高度なコンポーネント：OneFlow、OneGate、econe など

---

**注記：**これらのコンポーネントはXML-RPCを介して通信し、セキュリティやパフォーマンス上の理由から別々のマシンにインストールされる場合があります。

---

各バージョンの OpenNebula のフロントエンドとして動作する認定プラットフォームが複数存在します。プラットフォームに関する注意事項を参照の上、ご要件に最適なものをお選びください。

OpenNebulaのデフォルトデータベースは`sqlite`を使用します。本番環境または中規模から大規模な展開を計画されている場合は、[MySQL](#)の使用をご検討ください。

OpenNebulaの高可用性クラスタの構築にご興味がおありでしたら、高可用性OpenNebulaセクションをご確認ください。

複数のデータセンターを連携させる必要があり、リソース管理を異なるOpenNebulaインスタンスで行いながらも共通の認証スキームを必要とする場合は、連携セクションをご確認ください。

### 1.2.4 監視

監視サブシステムは、ホストの状態、基本的なパフォーマンス指標、仮想マシンの状態、容量消費量など、ホストおよび仮想マシンに関する情報を収集します。この情報は、OpenNebula が提供する一連の静的プローブを実行することで収集されます。情報の送信は、以下のプロセスに従って行われます。各ホストは定期的に監視データをフロントエンドに送信し、フロントエンドはそれを収集して専用のモジュールで処理します。このモデルは拡張性が高く、その限界（1秒あたりに監視できる VM の数）は、`oned` を実行しているサーバーとデータベースサーバーのパフォーマンスに依存します。

詳細については、[モニタリングセクション](#)をご確認ください。

### 1.2.5 仮想化ホスト

ホストとは、仮想マシンを実行する物理マシンを指します。OpenNebulaの各バージョンにおいて、ノードとして動作する認定プラットフォームが複数存在します。プラットフォームに関する注意事項をご参照いただき、ご要件に最も適したプラットフォームをお選びください。仮想化サブシステムは、ホストにインストールされたハイパーバイザーとの通信を担当し、仮想マシンのライフサイクルの各段階において必要な処理を実行するコンポーネントです。

OpenNebulaは、[KVM](#)と[LXD](#)という2つのオープンソースハイパーバイザをネイティブでサポートしております。

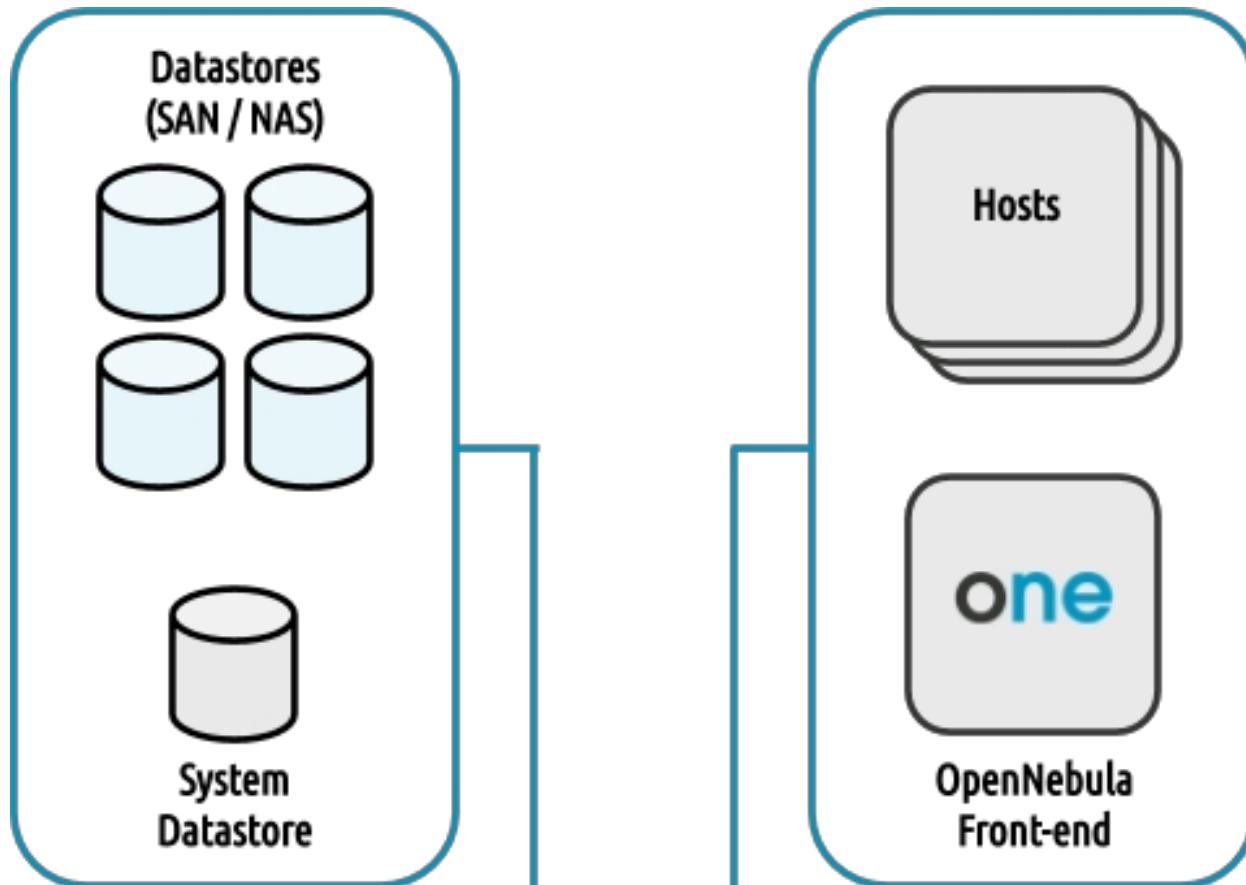
理想的には、ノードの構成は、インストールされているソフトウェアコンポーネント、onead-min管理ユーザー、アクセス可能なストレージ、ネットワーク接続性において均質であることが望ましいです。ただし、常にそうであるとは限りません。均質なホストは、OpenNebulaクラスター（例：LXDクラスター或はKVMクラスター）にグループ化することができます。

仮想化されたIT環境内におけるハードウェアおよびオペレーティングシステムの障害に対するフェイルオーバー保護にご関心がおありでしたら、仮想マシンの高可用性セクションをご確認ください。

## 1.2.6 ストレージ

OpenNebulaは、仮想マシンのディスクイメージを保存するためにデータストアを使用します。データストアとは、SAN/NASサーバーによって通常バックアップされるあらゆるストレージ媒体を指します。一般的に、各データストアは、適切な技術を用いてフロントエンドからアクセス可能である必要があります。

— NAS、SAN、または直接接続ストレージ。



仮想マシンがデプロイされると、そのイメージはデータストアからホストに転送されます。実際のストレージ技術に応じて、物理的な転送、シンボリックリンクの設定、またはLVMボリュームの構築を意味する場合があります。

OpenNebulaには3種類のデータストアクラスが同梱されています：

- **システムデータストア**：稼働中の仮想マシン用イメージを保持します。使用されるストレージ技術に応じて、これらの一時的なイメージは元のイメージの完全なコピー、qcow差分、または単純なファイルシステムリンクとなる場合があります。
- **イメージデータストア**：ディスクイメージリポジトリを保存します。ディスクイメージは、仮想マシンのデプロイ時やシャットダウン時、ディスクの接続時やスナップショット作成時に、システムデータストアとの間で移動または複製されます。
- **ファイルデータストア**：ディスクイメージではなく、通常のファイルを保存するために使用される特別なデータストアです。これらのファイルはカーネル、RAMディスク、またはコンテキストファイルとして使用できます。

イメージデータストアは、基盤となるストレージ技術に応じて、以下の異なるタイプが存在します：

- **ファイルシステム**：ディスクイメージをファイル形式で保存します。3つのタイプがあります: ssh、shared、qcow。
- **LVM**：仮想イメージを格納するために、単純なファイルではなくLVMボリュームを使用します。これによりファイルシステムを配置するオーバーヘッドが削減され、パフォーマンスが向上します。
- **Ceph**：Cephブロックデバイスを使用してディスクイメージを保存します。

**警告: デフォルト設定:** デフォルトのシステムおよびイメージデータストアは、ssh転送ドライバを使用したファイルシステムを利 用するよう設定されています。

詳細につきましては、[ストレージの章](#)をご確認ください。

### 1.2.7 ネットワーク

OpenNebulaは、既存データセンターの特定のネットワーク要件を統合するため、容易に適応・カスタマイズ可能なネットワークサブシステムを提供します。少なくとも2つの異なる物理ネットワークが必要です：

- **サービスネットワーク**：OpenNebulaのフロントエンドデーモンがホストにアクセスし、ハイパーバイザーの管理・監視、およびイメージファイルの移動を行うために使用されます。この目的専用のネットワークを設置されることを強く推奨いたします。
- **インスタンスネットワーク**：異なるホスト間で仮想マシン（VM）にネットワーク接続を提供します。VMの展開を効果的に活用するためには、1つ以上の物理ネットワークをVMが利用できるように設定する必要があるでしょう。

OpenNebula管理者は、各ホストに以下のいずれかのドライバを関連付けることができます：

- **dummy** (デフォルト)：ネットワーク操作を一切行わず、ファイアウォールルールも無視されます。
- ファイアウォールルールは適用されますが、ネットワーク分離は無視されます。
- **802.1Q**：VLANタグ付けによるネットワークアクセス制限。ハードウェアスイッチのサポートが必要です。
- **ebtables**：Ebttables ルールによるネットワークアクセスの制限。特別なハードウェア設定は不要です。
- **ovswitch**：Open vSwitch 仮想スイッチによるネットワークアクセスの制限。

- 
- *vxlan*: VXLANカプセル化プロトコルを用いて、VLANを分離されたネットワークに分割します。

OpenNebulaがサポートするネットワーク技術の詳細については、[ネットワークの章](#)をご確認ください。

## 1.2.8 認証

OpenNebulaへのアクセスには、以下の認証方法がサポートされております：

- 組み込みのユーザー名/パスワード認証
- [SSH認証](#)
- [X509認証](#)
- [LDAP認証 \(およびActive Directory\)](#)

**警告: デフォルト:** OpenNebula はデフォルトで、内部組み込みのユーザー名/パスワード認証機能を備えております。

どうぞ	ご確認ください	認証	認証	章	をご確認ください。	お探しください	探る	詳
細	詳しく	認証	認証	技術	サポートされている	OpenNebula	OpenNebula	高
度な	コンポーネント							

OpenNebulaクラウドを起動・稼働させた後、以下の高度なコンポーネントをインストールすることが可能です：

- マルチVMアプリケーションと自動スケーリング:** OneFlowは、ユーザーおよび管理者が、相互接続された仮想マシンで構成され、デプロイメント間の依存関係を持つマルチティアアプリケーションまたはサービスを定義、実行、管理することを可能にします。各仮想マシングループは単一のエンティティとしてデプロイおよび管理され、高度なOpenNebulaのユーザーおよびグループ管理と完全に統合されています。
- クラウドバースト：** クラウドバーストとは、プライベートクラウドのローカルリソースを、遠隔のクラウドプロバイダーのリソースと組み合わせるモデルです。このようなクラウドバーストのサポートにより、拡張性の高いホスティング環境を実現します。
- パブリッククラウド：** パートナーや外部ユーザーにインフラへのアクセスを提供したり、余剰容量を販売したりする場合、プライベートクラウドにクラウドインターフェースを追加できます。以下のインターフェースにより、高度な抽象化レベルでクラウド（仮想）リソースを簡便かつ遠隔管理できます：Amazon EC2 および EBS API。
- アプリケーションインサイト：** OneGateは仮想マシンのゲストが監視情報をOpenNebulaに送信することを可能にします。ユーザーや管理者はこれを利用してメトリクスを収集し、アプリケーションの問題を検知し、OneFlowの自動スケーリングルールをトリガーすることができます。

### 1.3 VMwareクラウドアーキテクチャ

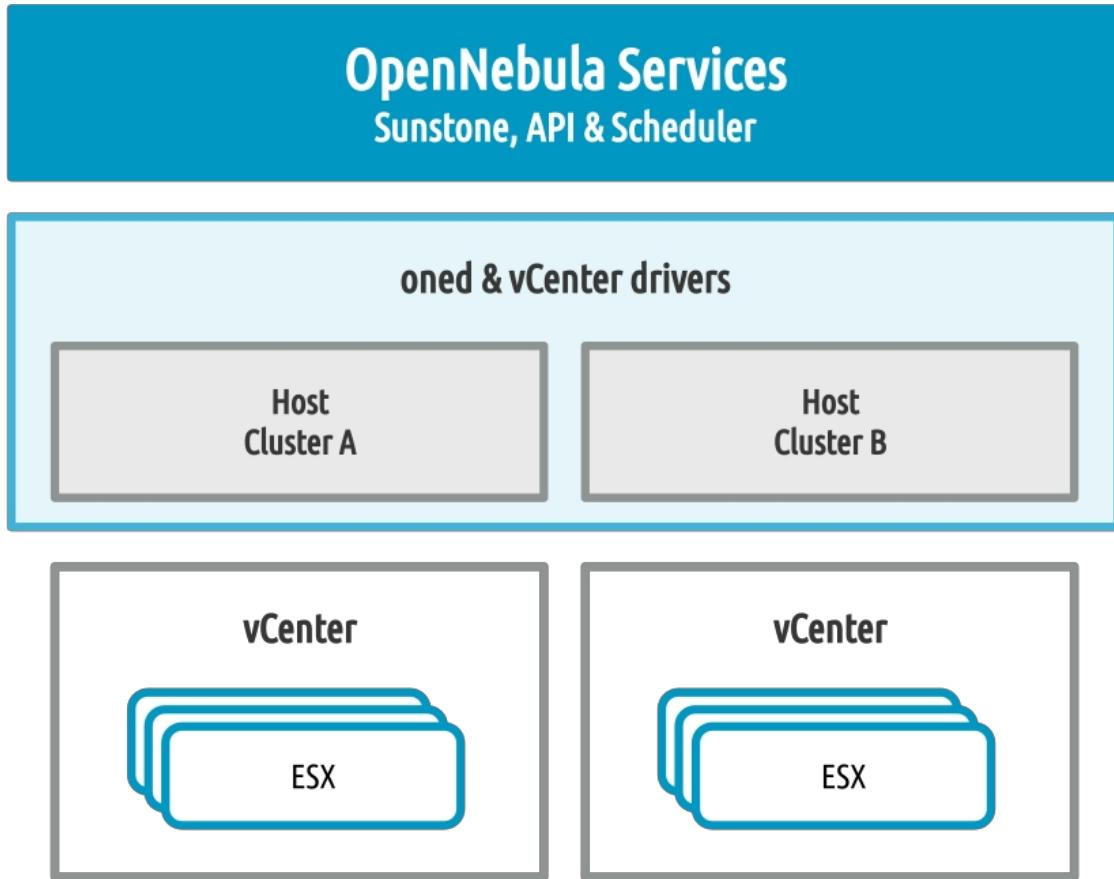
OpenNebulaは、VMwareインフラストラクチャ上にセルフサービス型クラウド環境を構築したいが、VMwareへの投資を放棄したり、スタック全体を再構築したりする必要がない企業向けに設計されています。このような環境において、OpenNebulaは既存のvCenterインフラストラクチャとシームレスに統合され、VMware vSphere製品ファミリーが提供するvMotion、HA、DRSスケジューリング、NSX-T/NSX-Vなどの高度な機能を活用します。OpenNebulaは、vCenter上にマルチテナントのクラウドのようなプロビジョニング層を提供し、仮想データセンター、データセンターフェデレーション、ハイブリッドクラウドコンピューティングなどの機能を含みます。これにより、社内のvCenterインフラストラクチャとパブリッククラウドを接続することが可能となります。

OpenNebula over vCenterは、VMwareの管理ツール、手順、ワークフローを維持したい企業様向けに設計されております。こうした企業様にとって、VMwareを廃棄しスタック全体を再構築することは解決策とはなりません。しかしながら、仮想化を超えてプライベートクラウドへの移行を検討される際には、VMwareへのさらなる投資を選択するか、あるいは戦術的には困難ではあるものの戦略的には有益なオープンインフラストラクチャの道を進むか、お選びいただけます。

### **1.3.1 アーキテクチャ概要**

OpenNebulaは、お客様の物理インフラストラクチャが、フロントエンドを備えた古典的なクラスター型アーキテクチャを採用し、仮想マシン（VM）が実行されるESXホストをグループ化するvCenterインスタンス群で構成されていることを前提としています。すべてのvCenterおよびESXホストをフロントエンドと接続する物理ネットワークが少なくとも1つ存在します。フロントエンドからvCenterへの接続は管理目的であり、フロントエンドからESXホストへの接続はVNCまたはVMRC接続をサポートするためです。

VMware vCenter ドライバーにより、OpenNebulaは1つ以上のESXクラスターを管理するvCenterサーバーにアクセスできます。各ESX クラスターはOpenNebulaにおいて集約されたハイパーバイザーとして提示されます。したがって、OpenNebulaのスケジューリング決定はESXクラスターレベルで行われ、vCenterはDRSコンポーネントを使用して仮想マシンを展開する実際のESXホストとデータストアを選択します。



クラウドアーキテクチャは、ストレージ、ネットワーク、仮想化の3つのコンポーネントによって定義されます。したがって、OpenNebulaクラウドの基本コンポーネントは以下の通りです：

- OpenNebulaサービスを実行するフロントエンド。
- 仮想マシンに必要なリソースを提供するハイパーバイザー対応ホスト。
- 仮想マシンのベースイメージを保持するデータストア。
- 仮想マシン間の相互接続など、基本サービスをサポートするために使用される物理ネットワークまたはソフトウェア定義ネットワーク。

OpenNebulaは、汎用ハイパーバイザーからエンタープライズグレードのハイパーバイザーまで幅広くサポートする、高度にモジュール化されたアーキテクチャを提供します。監視、ストレージ、ネットワーク、ユーザー管理サービスも同様にサポート対象です。本セクションでは、各サブシステムの管理において選択可能な異なるオプションについて簡潔に説明します。ご利用の特定のサービスがサポートされていない場合は、[アドオンカタログ](#)で利用可能なドライバーをご確認いただくことをお勧めいたします。また、新規ドライバーの開発方法に関する情報とサポートも提供しております。

### 1.3.2 クラウドの規模設計

クラウドインフラの規模は、そのインフラが支えなければならない仮想マシン（VM）の観点から、予想されるワークロードから直接推測できます。このワークロードの見積もりも難しい作業ではありますが、効率的なクラウドを構築するためには極めて重要な作業です。

## OpenNebula フロントエンド

OpenNebula フロントエンドの推奨最小スペックは以下の通りです：

リソース	推奨最小構成
メモリ	2 GB
CPU	1 CPU (2コア)
ディスク容量	100 GB
ネットワーク	2 NIC

上記の表に記載されている最小構成でフロントエンド上で動作する場合、OpenNebulaは以下の特性を備えたvCenterインフラストラクチャを管理することが可能です：

- 最大4つのvCenter
- 各vCenterで管理可能なESXは最大40台まで
- 合計最大1,000台の仮想マシン（各vCenterが最大250台の仮想マシンを管理）

## ESXノード

ESX仮想化ノードの仕様について：

- CPU**：オーバーコミットメントなしの場合、VMに割り当てられた各CPUコアは物理CPUコアとして存在する必要があります。例えば、2CPUのVMが40台というワークロードの場合、クラウドには80個の物理CPUが必要です。これらの80個の物理CPUは、異なるホストに分散して配置できます。例えば、8コアのサーバー10台、または16コアのサーバー5台といった構成です。ただし、オーバーコミットメントを有効にした場合、CPUの構成は事前に計画することができます。その際、CPU属性とVCPU属性を使用します。CPU属性はVMに割り当てられた物理CPUを表し、VCPU属性はゲストOSに提示される仮想CPUを表します。
- メモリ**：メモリの計画は比較的簡単です。OpenNebulaではデフォルトでメモリのオーバーコミットメントが行われないためです。ハイパーバイザーによるオーバーヘッドを10%見込むことが常に推奨されます（これは絶対的な上限ではなく、ハイパーバイザーによって異なります）。したがって、各仮想マシンに2GBのRAMを割り当てた45台の仮想マシンワークロードを維持するには、90GBの物理メモリが必要となります。ホストの数は重要です。各ホストはハイパーバイザーにより10%のオーバーヘッドが発生するためです。例えば、10GBのRAMを搭載した10台のハイパーバイザーは、それぞれ9GB（10GBの10% = 1GB）を提供するため、推定ワークロードを維持できます。経験則として、コアあたり最低1GBを確保することが推奨されますが、これも予想されるワークロードによって異なります。

## ストレージ

ストレージの設計は、通常クラウドのボトルネックとなるため、極めて重要な侧面です。OpenNebulaは、ESXにマウントされvCenterで認識可能なあらゆるデータストアを管理できます。仮想マシンが使用するデータストアは、クラウド管理者によって固定することも、クラウドユーザーに委任することも可能です。新しい仮想マシンの作成プロセスが失敗しないよう、十分な空き容量を確保することが重要です。有効なアプローチの一つとして、最大VM数によるクォータを設定し、その制限に準拠する十分なデータストア容量を確保することで、ユーザーが利用可能なストレージを制限する方法が挙げられます。いずれの場合も、OpenNebulaでは必要に応じてクラウド管理者がデータストアを追加することが可能です。

## ネットワーク

クラウドインフラストラクチャの信頼性を確保するためには、ネットワークを慎重に設計する必要があります。推奨される構成は、フロントエンドに2枚のNIC（サービスネットワークとパブリックネットワーク用）を、各ESXノードに4枚のNIC（プライベート

ートネットワーク、パブリックネットワーク、サービスネットワーク、ストレージネットワーク用) を設置することです。ストレージおよびネットワーク構成によっては、必要な NIC 枚数が少なくなる場合もあります。

### 1.3.3 フロントエンド

OpenNebulaをインストールするマシンはフロントエンドと呼ばれます。このマシンは、すべてのvCenterおよびESXホストとのネットワーク接続が必要です。OpenNebulaの基本インストールに必要な容量は150MB未満です。

OpenNebulaのサービスには以下が含まれます：

- 管理デーモン (`oned`) およびスケジューラ (`mm_sched`)
- Webインターフェースサーバー (`sunstone-server`)
- 高度なコンポーネント：OneFlow、OneGate、econe など

---

**注記：**これらのコンポーネントはXML-RPCを介して通信し、セキュリティやパフォーマンス上の理由から別々のマシンにインストールされる場合があります。

---

各バージョンの OpenNebula のフロントエンドとして動作する認定プラットフォームが複数存在します。プラットフォームに関する注意事項を参照の上、ご要件に最適なものをお選びください。

OpenNebulaのデフォルトデータベースは`sqlite`を使用します。本番環境または中規模から大規模な展開を計画されている場合は、`MySQL`の使用をご検討ください。

OpenNebula の高可用性クラスタの構築にご興味がおありでしたら、高可用性 OpenNebula セクションをご確認ください。

### 1.3.4 監視

監視サブシステムは、ホストおよび仮想マシンに関連する情報を収集します。これには、ホストの状態、基本的なパフォーマンス指標、ならびに仮想マシンの状態および容量使用状況が含まれます。この情報は、OpenNebulaが提供するフロントエンドにおいて一連のプローブを実行することで収集されます。

詳細については、[監視セクション](#)をご確認ください。

### 1.3.5 仮想化ホスト

VMware vCenter ドライバーにより、OpenNebula は 1 つ以上の ESX クラスターを管理する 1 つ以上の vCenter サーバーにアクセスできます。各 ESX クラスターは、OpenNebula において集約されたハイパーバイザーとして表示されます。仮想化サブシステムは、vCenter と通信し、VM ライフサイクルの各ステップに必要なアクションを実行する役割を担うコンポーネントです。特定の仮想マシンが稼働している ESX への VNC 接続は、フロントエンドから直接実行されますが、それ以外のすべての管理操作は、フロントエンドから vCenter に対して発行されます。

OpenNebulaはvCenterハイパーバイザーをネイティブでサポートしております。vCenterドライバーの設定はOpenNebulaフロントエンドで行います。

仮想化されたIT環境におけるハードウェアおよびオペレーティングシステムの障害に対するフェイルオーバー保護にご関心がありましたら、仮想マシンの高可用性セクションをご確認ください。

### 1.3.6 ストレージ

OpenNebulaはvCenterストレージの利用者として動作し、ESXがサポートするすべてのストレージデバイスに対応しております。仮想マシンが仮想マシンテンプレートからインスタンス化される際、OpenNebulaのスケジューラはデフォルトポリシーに基づき、利用可能なデータストアに仮想マシンを分散配置するため、データストアを選択いたします。ただし、このスケジューラポリシーは変更可能であり、SCHED\_DS\_REQUIREMENTS属性により、テンプレートからインスタンス化された仮想マシンが特定のデータストアを使用するよう強制することができます。ストレージ DRS が有効になっている場合、OpenNebula はストレージ DRS クラ

#### **OpenNebula 5.10 導入ガイド、リリース 5.10.5**

スタにストレージの推奨事項を要求し、VM インスタンス化される際にそれを適用することができます。この場合、OpenNebula はデータストアの選択を vCenter のストレージ DRS に委任することになります。

vCenter/ESXデータストアは、OpenNebula内で表現することが可能です。これにより、VMDKの作成、クローン作成、およびアップロードが行えます。OpenNebulaにおけるvCenter/ESXデータストアの表現については、[vCenter データストアのセクション](#)で説明されています。

### 1.3.7 ネットワーク

OpenNebulaにおけるネットワークは、vCenterネットワークおよび分散vSwitchの仮想ネットワーク表現を作成またはインポートすることで処理されます。これにより、定義されたネットワークインターフェースを持つ新しいVMは、OpenNebulaによってこれらのネットワークおよび/または分散vSwitchにバインドされます。OpenNebulaは、これらのvCenterネットワークおよび分散vSwitchを基に、新たな論理レイヤーを作成することができます。具体的には、vCenterネットワークリソースを表す仮想ネットワークごとに、3種類のアドレス範囲（ブレーンイーサネット、IPv4、IPv6）を定義できます。このネットワーク情報は、コンテキスト化プロセスを通じて仮想マシンに渡すことができます。

OpenNebulaは、NSX Manager APIを通じてNSX-TおよびNSX-Vの論理スイッチをサポートしております。この統合における主なポイントは以下の通りです：

- NSX ManagerはvCenter Serverから自動的に検出されます。
- トランスポートゾーンは自動的に検出されます。
- OpenNebulaで仮想ネットワークが作成されると、NSX Manager内に論理スイッチが作成されます。
- デプロイ時および実行時において、OpenNebula GUIを通じて仮想マシンのNICをこれらの論理スイッチにリンクできます。
- 既存の論理スイッチをインポートまたは削除できます。

OpenNebulaによるvCenterインフラストラクチャのネットワークサポートに関する詳細は、[vCenterネットワーク設定](#)および[NSX設定](#)をご確認ください。

### 1.3.8 認証

OpenNebulaへのアクセスには、以下の認証方法がサポートされています：

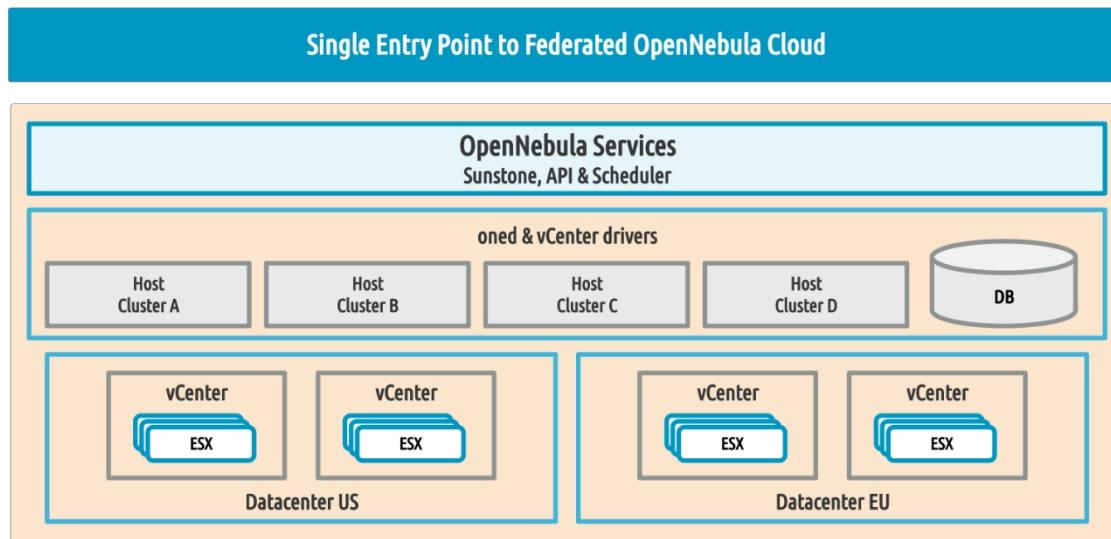
- 組み込みのユーザー名/パスワード
- [SSH認証](#)
- [X509認証](#)
- [LDAP認証](#)（およびActive Directory）

**警告: デフォルト設定:** OpenNebulaはデフォルトで、内部組み込みのユーザー名/パスワード認証機能を備えております。

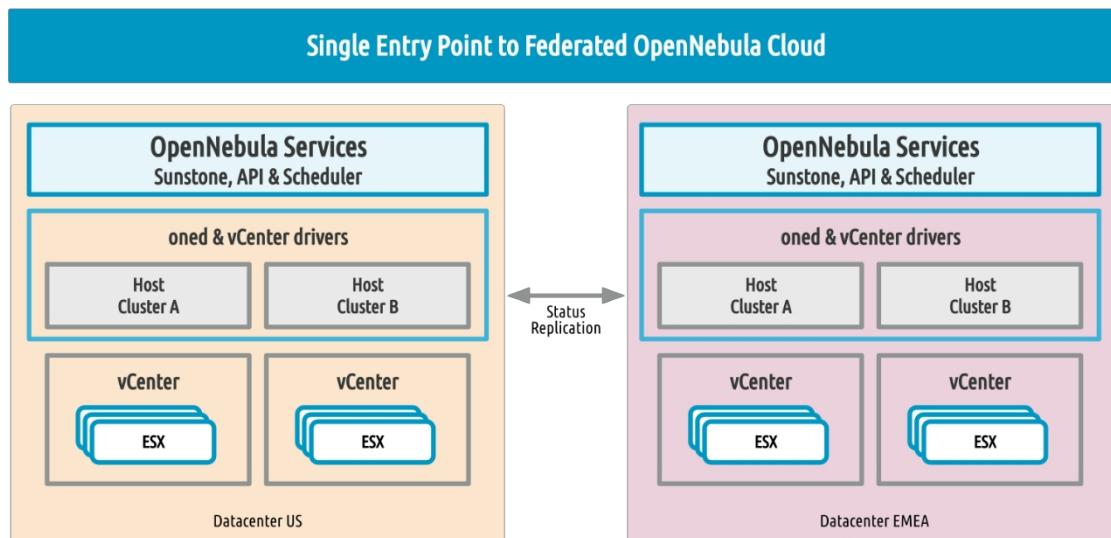
OpenNebulaがサポートする認証技術の詳細については、[認証の章](#)をご確認ください。

### 1.3.9 マルチデータセンター展開

OpenNebulaは、vCenterインスタンスとの連携をSOAP APIを介してのみ行います。この特性により、OpenNebulaインスタンスとvCenter環境が異なるデータセンターに配置されるアーキテクチャが可能となります。単一のOpenNebulaインスタンスが、異なるデータセンターに遠隔配置された複数のvCenterインスタンスをオーケストレーションできます。OpenNebulaからvCenterを確実に管理するためには、データセンター間の接続性に低遅延が求められます。



管理ドメインを分離する必要がある場合、またはデータセンター間の相互接続が单一の管理エンティティを許容しない場合、OpenNebulaはフェデレーション構成が可能です。フェデレーション内の各OpenNebulaインスタンスは「ゾーン」と呼ばれ、そのうち1つがマスターとして設定され、残りがスレーブとして設定されます。OpenNebulaフェデレーションは緊密に連携した統合であり、すべてのインスタンスが同一のユーザー アカウント、グループ、権限設定を共有します。フェデレーションにより、エンドユーザーは地理的な位置に関係なく、フェデレーション管理者によって割り当てられたリソースを利用できます。統合はシームレスであり、あるゾーンのSunstoneウェブインターフェースにログインしたユーザーは、ログアウトして別のゾーンのアドレスを入力する必要がありません。Sunstoneではいつでもアクティブなゾーンを変更でき、リクエストは自動的に対象ゾーンの適切なOpenNebulaヘリダイレクトされます。詳細については、フェデレーションセクションをご確認ください。



### 1.3.10 高度なコンポーネント

OpenNebulaクラウドを起動・稼働させた後、以下の高度なコンポーネントをインストールすることができます：

- マルチVMアプリケーションと自動スケーリング: OneFlowは、相互接続された仮想マシンで構成され、デプロイメント間の依存関係を持つサービスを、ユーザーおよび管理者が定義、実行、管理することを可能にします。各仮想マシングループは単一のエンティティとしてデプロイおよび管理され、高度なOpenNebulaのユーザーおよびグループ管理と完全に統合されて

---

います。

- クラウドバースト：クラウドバーストとは、プライベートクラウドのローカルリソースを

クラウドバーストのサポートにより、高度にスケーラブルなホスティング環境を実現します。

- ・パブリッククラウド：パートナーや外部ユーザーにインフラへのアクセスを提供したり、余剰容量を販売したりする場合、プライベートクラウドにクラウドインターフェースを追加できます。以下のインターフェースにより、高い抽象化レベルでクラウド（仮想）リソースを簡便かつ遠隔管理できます：Amazon EC2 および EBS API。
- ・アプリケーションインサイト：OneGateは、仮想マシンのゲストが監視情報をOpenNebulaに送信することを可能にします。ユーザーと管理者は、この機能を利用してメトリクスを収集し、アプリケーションの問題を検知し、OneFlowの自動スケーリングルールをトリガーすることができます。

## 1.4 OpenNebula プロビジョニングモデル

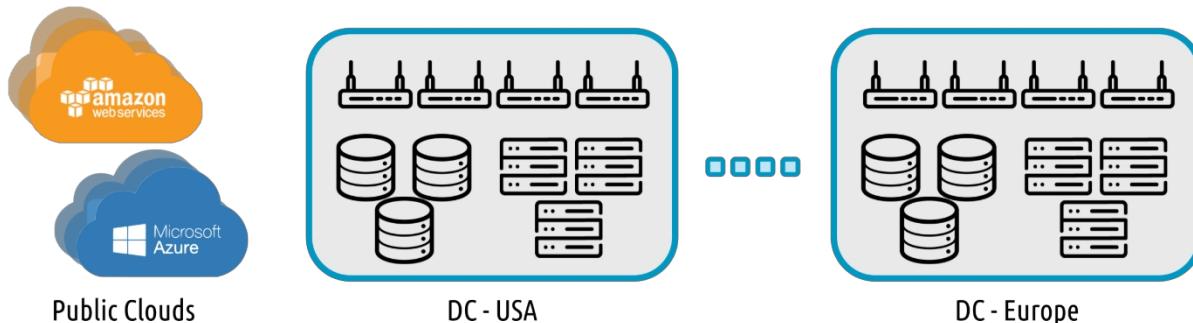
ホスト数が少ない小規模な環境では、インフラストラクチャの分割やプロビジョニングについて深く考慮することなくOpenNebulaをご利用いただけます。しかし、中規模から大規模な展開においては、ある程度の分離と構造化が必要となるでしょう。

本セクションは、クラウドアーキテクト、構築担当者、管理者の方々が、仮想リソースの管理およびプロビジョニングに関するOpenNebulaのモデルを理解するお手伝いをすることを目的としています。このモデルは、プロジェクトの全期間を通じてユーザーとコミュニティとの協力によって生まれたものです。

### 1.4.1 インフラストラクチャの観点

一般的な大規模なIT部門では、複数のデータセンター（DC）を保有しており、各データセンターは独自のOpenNebulaインスタンスを稼働させ、複数の物理的なインフラストラクチャリソース（ホスト、ネットワーク、データストア）のクラスターで構成されています。これらのクラスターは、異なるワークロードプロファイルのニーズを満たすため、異なるアーキテクチャやソフトウェア／ハードウェアの実行環境を提供することができます。さらに、多くの組織では外部パブリッククラウドへのアクセス権を有しており、Amazon AWSなどの外部クラウドリソースでデータセンターのプライベート容量を補完するハイブリッドクラウドシナリオを構築し、需要のピークに対応しています。

例えば、地理的に異なる場所（ヨーロッパと米国西海岸）に2つのデータセンターを設置し、AmazonやAzureなどのパブリッククラウドプロバイダーとクラウドバーストに関する契約を結ぶことが考えられます。各データセンターは独自のゾーンまたは完全なOpenNebulaデプロイメントを実行します。複数のOpenNebulaゾーンをフェデレーションとして構成することが可能であり、この場合、データセンター間で同じユーザーアカウント、グループ、権限を共有します。



## 1.4.2 組織的観点

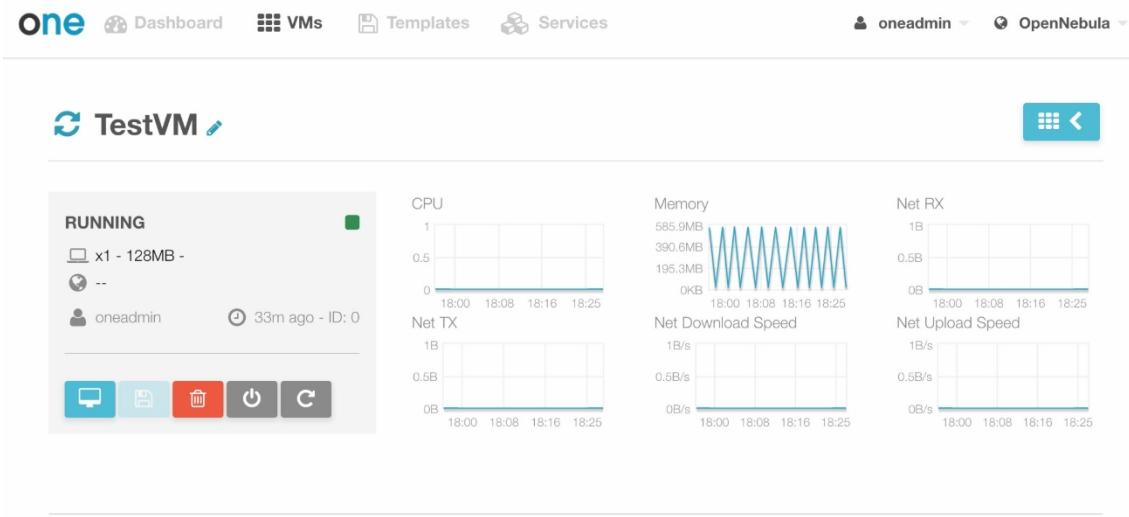
ユーザーはグループに編成されます（他の環境でプロジェクト、ドメイン、テナントなどと呼ばれるものと同様です）。グループは認可境界であり、プライベートクラウドとして考える場合は事業単位、パブリッククラウドの場合は独立した企業と見なすことができます。一方、クラスターは物理リソースを以下の基準でグループ化するために使用されます。

ネットワークトポジーや物理的な場所といった共通の特徴を持つ仮想データセンター（VDC）により、物理リソース（異なるクラスターやゾーンに属する可能性のあるもの）の「論理的な」プールを作成し、ユーザーグループに割り当てることが可能となります。

VDCは完全に分離された仮想インフラ環境であり、グループ管理者（または任意で複数のユーザーグループ）の管理下で、ユーザーグループがコンピューティングおよびストレージ容量の作成・管理を行えます。グループ内のユーザー（グループ管理者を含む）には仮想リソースのみが表示され、基盤となる物理インフラは見えません。グループに割り当てられた物理リソースは、クラウド管理者によってVDCを通じて管理されます。VDC内にグループ化されたこれらのリソースは、グループ専用に割り当てることが可能であり、物理レベルでの分離も実現します。

グループユーザーおよび管理者による、他のユーザーが作成した仮想リソースに対する操作権限は設定可能です。例えば、後述する高度なクラウドプロビジョニングのケースでは、ユーザーは仮想マシンテンプレートをインスタンス化して自身のマシンを作成できますが、グループの管理者は他のユーザーのリソースを完全に制御でき、グループ内に新規ユーザーを作成することも可能です。

ユーザーは、CLI、Sunstone Cloud View、OCA、AWS APIなど、既存のOpenNebulaインターフェースのいずれかを通じて自身のリソースにアクセスできます。グループ管理者は、CLIまたはSunstoneのグループ管理画面を通じてグループを管理できます。クラウド管理者は、CLIまたはSunstoneを通じてグループを管理できます。



VDCに基づくクラウドプロビジョニングモデルは、大規模なマルチデータセンター環境において、インフラストラクチャリソースを異なる顧客、事業部門、またはグループに動的にプロビジョニングするための統合された包括的なフレームワークを実現します。これにより、以下の利点が得られます：

- ユーザーグループ間のクラウド物理リソースの分割
- ユーザー、組織、ワークロードの完全な分離
- セキュリティレベル、パフォーマンス、または高可用性が異なるクラスターの割り当て
- ソフトウェア定義データセンターの実行用コンテナ
- グループメンバーから物理リソースを隠蔽する方法
- 単一のクラウドインスタンスやデータセンターを超えたプライベートクラウドインフラストラクチャの簡易な連携、スケーラビリティ、クラウドバースト

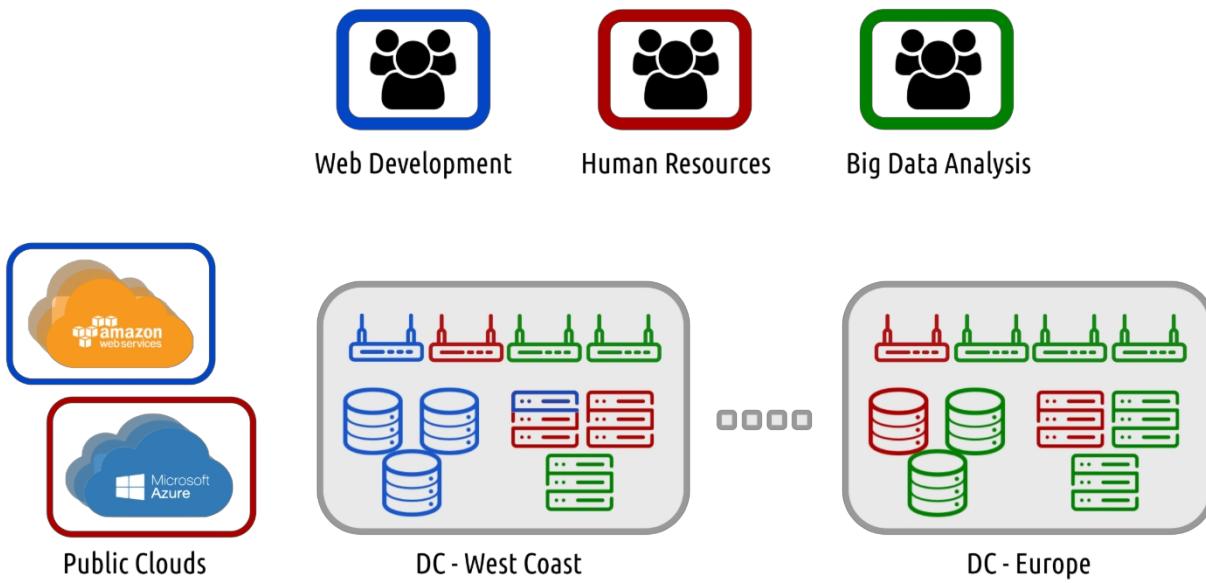
### 1.4.3 プロビジョニングのユースケース例

以下は、大規模なクラウドコンピューティング導入における一般的なエンタープライズユースケースです：

- 複数のプロジェクト、部門、ユニット、または組織にサービスを提供するオンプレミス型プライベートクラウド。大規模組織におけるオンプレミス型プライベートクラウドでは、仮想および物理インフラへのアクセス権限を管理し、利用可能なりソースを動的に割り当てるための強力かつ柔軟な仕組みが求められます。このようなシナリオでは、クラウド管理者は各部門ごとにVDC（仮想データセンター）を定義し、部門のニーズに応じてリソースを動的に割り当てるとともに、グループの内部管理を部門のIT管理者に委任します。
- 仮想プライベートクラウドコンピューティングを提供するクラウドプロバイダー。お客様に完全に構成可能で分離された環境を提供し、ユーザーとリソースを管理する完全な制御権と能力を付与するクラウドプロバイダーです。これはパブリッククラウドと、個人向けプライベートクラウドシステムで通常見られる制御機能を組み合わせたものです。

例えば、Web開発、人事、ビッグデータ分析といった事業部門を、プライベートOpenNebulaクラウド内のグループとして考え、各部門にデータセンターやパブリッククラウドからリソースを割り当て、3つの異なるVDCを作成することができます。

- VDC BLUE:** Web開発向けに（ClusterA-DC\_West\_Coast + クラウドバースト）を割り当てるVDC
- VDC RED:** 人事部門向けに（ClusterB-DC\_West\_Coast + ClusterA-DC\_Europe + クラウドバースト）を割り当てるVDC
- VDC GREEN:** ビッグデータ分析に割り当てられる VDC （ClusterC-DC\_West\_Coast + ClusterB-DC\_Europe）



#### 1.4.4 クラウドプロビジョニングシナリオ

OpenNebulaには、3つの典型的なエンタープライズクラウドシナリオを実現するための3つの事前定義されたユーザー ロールが用意されています：

- データセンターインフラストラクチャ管理
- シンプルなクラウドプロビジョニングモデル
- 高度なクラウドプロビジョニングモデル

これらの3つのシナリオにおいて、クラウド管理者は物理インフラストラクチャの管理、ユーザーおよびVDCの作成、ユーザー向けのベーステンプレートやイメージの準備などを行います。

クラウド管理者は通常、CLIまたはSunstoneの管理ビューを使用してクラウドにアクセスします。

役割	機能
<b>クラウド管理者</b>	<ul style="list-style-type: none"> <li>・ クラウドインフラストラクチャ（コンピューティングノード、ネットワークファブリック、ストレージサーバーなど）の運用を行います。</li> <li>・ OpenNebulaインフラストラクチャリソース（ホスト、仮想ネットワーク、データストア）の作成および管理</li> <li>・ マルチVMアプリケーション（サービス）の作成および管理</li> <li>・ 新規グループの作成およびVDCの作成</li> <li>・ グループおよび物理リソースをVDCに割り当て、クォータ制限を設定します</li> <li>・ ユーザーが使用する基本インスタンスタイプを定義します。これらのタイプは、仮想マシンの容量（メモリ、CPU、追加ストレージ）および接続性を定義します。</li> <li>・ ユーザーが使用するVMイメージを準備します</li> <li>・ クラウドの状態と健全性を監視します</li> <li>・ 活動レポートを生成します</li> </ul>

### データセンターインフラストラクチャ管理

このモデルは、データセンターの仮想化を管理し、異なるデータセンターに存在する既存のIT資産を統合・連携するために使用されます。この使用モデルでは、ユーザーは仮想化の概念に精通しています。インフラストラクチャリソースを除き、ウェブインターフェースではクラウド管理者と同様の操作が可能です。これらは「上級ユーザー」であり、「限定的なクラウド管理者」ともみなすことができます。

ユーザーはクラウド管理者によって事前定義されたテンプレートやイメージを利用できますが、通常は独自のテンプレートやイメージを作成することも許可されています。また、リソースのライフサイクル管理が可能であり、仮想マシンゲストに悪影響を及ぼす可能性のある高度な機能（新規ディスクのホットプラグ、仮想マシンのサイズ変更、起動パラメータの変更など）も扱えます。

クラウド管理者はグループを使用してユーザーを分離し、VDC と組み合わせてリソースを割り当てますが、オンデマンドでの提供は行われません。

これらの「上級ユーザー」は、通常、CLI または Sunstone のユーザービューを使用してクラウドにアクセスします。これは、グループ「ユーザー」にデフォルトで設定されているモデルではありません。

ロール	機能

上級ユーザー	<ul style="list-style-type: none"><li>自身のテンプレートを使用して仮想マシンをインスタンス化します</li><li>新しいテンプレートとイメージを作成します</li><li>高度なライフサイクル機能を含む、自身の仮想マシンの管理を行います</li><li>マルチVMアプリケーション（サービス）の作成および管理</li><li>ご利用状況と割り当て容量をご確認ください</li><li>仮想マシンにアクセスするためのSSHキーをアップロードしてください</li></ul>
--------	---

## シンプルなクラウドプロビジョニング

シンプルなインフラストラクチャプロビジョニングモデルでは、クラウドは個々のユーザーにインフラストラクチャ・アズ・ア・サービスを提供します。ユーザーは「クラウドユーザー」または「クラウドコンシューマー」と見なされ、操作が大幅に制限されます。これらのユーザーは、非常に直感的で簡素化されたウェブインターフェースにアクセスし、事前定義されたテンプレートから仮想マシンを起動できます。自身の仮想マシンにアクセスし、シャットダウンなどの基本操作を実行できます。仮想マシンのディスクへの変更は保存できますが、新規イメージをゼロから作成することはできません。

クラウド管理者はグループを使用してユーザーを分離し、VDCと組み合わせてリソースを割り当てますが、オンデマンドでの提供は行われません。

これらの「クラウドユーザー」は、通常、Sunstoneのクラウドビューを使用してクラウドにアクセスします。これは、グループ「Users」に対して設定されたデフォルトのモデルです。

ロール	機能
クラウドユーザー	<ul style="list-style-type: none"> <li>クラウド管理者によって定義されたテンプレート、およびクラウド管理者またはグループ管理者によって定義されたイメージを使用して仮想マシンをインスタンス化します。</li> <li>以前に実行した仮想マシンから保存した独自のイメージを使用して仮想マシンをインスタンス化します。</li> <li>自身の仮想マシンを管理します。これには以下が含まれます。 <ul style="list-style-type: none"> <li>再起動</li> <li>電源オフ/オン（短時間の電源切断）</li> <li>削除</li> <li>VM を新しいテンプレートとして保存</li> <li>基本的なモニター情報とステータス（IPアドレスを含む）の取得</li> </ul> </li> <li>既存のVMテンプレートおよびディスクスナップショットを削除する</li> <li>ユーザー アカウントの使用状況と割り当て容量の確認</li> <li>仮想マシンにアクセスするためのSSHキーをアップロードする</li> </ul>

## 高度なクラウドプロビジョニング

高度なプロビジョニングモデルは、クラウド プロバイダーがユーザーグループ（プロジェクト、企業、部門、事業部門など）に対してオンデマンドでVDCを提供する、従来のモデルの拡張版です。各グループは、1人以上のユーザーをグループ管理者として指定できます。これらの管理者は、グループ内で新規ユーザーを作成できるほか、他のユーザーのリソースを管理することも可能です。例えば、グループ管理者は他のユーザーが使用するVMをシャットダウンし、グループのクォータ使用量を解放することができます。

これらのグループ管理者は通常、Sunstoneのグループ管理者ビューを使用してクラウドにアクセスします。

### **OpenNebula 5.10 導入ガイド、リリース 5.10.5**

グループユーザーは、前述のシナリオで説明した機能を有し、通常はSunstoneのクラウドビューを使用してクラウドにアクセスします。

ロール	機能
<b>グループ管理</b>	<ul style="list-style-type: none"><li>• グループ内に新規ユーザーを作成します</li><li>• グループの仮想マシンおよびディスクイメージを操作します</li><li>• 保存済みテンプレートをグループのメンバーと共有します</li><li>• グループの使用状況と割り当て容量を確認します</li></ul>

## OPENNEBULA のインストール

### 2.1 概要

フロントエンドは、OpenNebula インストールの中核部分です。サーバーソフトウェアがインストールされ、クラウド管理のために接続するマシンです。物理ノードまたは仮想インスタンスのいずれかとなります。

#### 2.1.1 本章の読み方

本章をお読みになる前に、必ず「[クラウド設計](#)」の章をお読みいただき、内容を理解されていることをご確認ください。

本章の目的は、OpenNebula を迅速に導入するためのクイックスタートガイドを提供することです。これは可能な限りシンプルなインストール方法ですが、高度なコンポーネント（ホストおよび VM の高可用性、クラウドバーストなど）を用いたより複雑な設定の基盤ともなります。

まず「[フロントエンドのインストール](#)」セクションをお読みください。デフォルトではSQLiteデータベースを使用しますが、本番環境では推奨されません。したがって、小規模な概念実証でない場合は、インストール手順に従う際に[MySQL](#)をご利用ください。

この章をお読みになった後は、クラウドにハイパーバイザーを追加するため、次に「[ノードのインストール](#)」の章をお読みください。

#### 2.1.2 ハイパーバイザーの互換性

セクション	互換性
<a href="#">フロントエンドのインストール</a>	このセクションはすべてのハイパーバイザーに適用されます
<a href="#">MySQL の設定</a>	このセクションはすべてのハイパーバイザーに適用されます
スケジューラ	このセクションはすべてのハイパーバイザーに適用されます

### 2.2 フロントエンドのインストール

このページでは、バイナリパッケージから OpenNebula をインストールする方法をご説明します。

当サイトで提供しているパッケージのご利用をお勧めいたします。これにより最新版のインストールが保証され、異なるディスト

リビューション間でのパッケージの差異を回避できます。具体的には二つの方法がございます：お使いのシステムに**当方のパッケージリポジトリ**を追加いただくか、[ソフトウェアメニュー](#)よりお使いのLinuxディストリビューション向けの**最新版パッケージをダウンロードしてください。**

お使いのディストリビューションに対応するパッケージが存在しない場合は、ソースコードからのビルドガイドをご参照ください。

---

**ご注意：**フロントエンドをLXDコンテナ内にインストールすることは可能ですが、機能が制限されます。この状況ではLXDマーケットプレイスはご利用いただけません。フロントエンドはブロックデバイスのマウントを必要としますが、LXDコンテナではこれが不可能です。

---

## 2.2.1 ステップ1. CentOS/RHELにおけるSELinuxの設定

SELinuxは、OpenNebulaフロントエンドによって開始される一部の操作をブロックする場合があり、その結果として障害が発生することがあります。管理者がSELinuxの設定に慣れていない場合は、**予期せぬ障害を避けるため、この機能を無効化することをお勧めいたします。**インストールが正常に動作するようになった後、いつでもSELinuxを有効化することができます。

### SELinuxを無効化（推奨）

SELinuxを無効化するには、`/etc/selinux/config`内の以下の行を変更してください：

```
SELINUX=disabled
```

変更後は、マシンの再起動が必要です。

### SELinuxを有効にする

SELinuxを強制モードで有効にするには、`/etc/selinux/config` ファイル内の以下の行を変更してください：

```
SELINUX=enforcing
```

無効状態から変更する際は、次回起動時にファイルシステムの再ラベル付けを実行する必要があります。そのためには、`/.autorelabel` ファイルを作成してください。例：

```
$ touch /.autorelabel
```

変更後は、マシンを再起動してください。

---

**注記：** SELinux の設定およびトラブルシューティングに関する詳細情報は、SELinux ユーザーおよび管理者ガイドをご参照ください。

---

## 2.2.2 ステップ 2. OpenNebula リポジトリの追加

### CentOS/RHEL

OpenNebula リポジトリを追加するには、root ユーザーとして以下のコマンドを実行してください：

#### CentOS/RHEL 7

```
# cat << "EOT" > /etc/yum.repos.d/opennebula.repo [opennebula]
name=opennebula baseurl=https://downloads.opennebula.org/repo/5.10/CentOS/7/$basearch enabled=1
gpgkey=https://downloads.opennebula.org/repo/repo.key gpgcheck=1
```

```
repo_gpgcheck=1 EOT
```

#### **CentOS/RHEL 8**

```
# cat << "EOT" > /etc/yum.repos.d/opennebula.repo [opennebula]
name=opennebulabaseurl=https://downloads.opennebula.org/repo/5.10/CentOS/8/$basearchenabled=1
gpgkey=https://downloads.opennebula.org/repo/repo.key gpgcheck=1
repo_gpgcheck=1 EOT
```

#### **Debian/Ubuntu**

Debian/Ubuntu に OpenNebula リポジトリを追加するには、root 権限で以下のコマンドを実行してください：

```
# wget -q -O- https://downloads.opennebula.org/repo/repo.key | apt-key add -
```

#### **Debian 9**

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Debian/9 stable opennebula" > /etc/apt/source
```

#### **Debian 10**

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Debian/10 stable opennebula" > /etc/apt/source
```

#### **Ubuntu 16.04**

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Ubuntu/16.04 stable opennebula" > /etc/apt/source
```

#### **Ubuntu 18.04**

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Ubuntu/18.04 stable opennebula" > /etc/apt/source
```

#### **Ubuntu 19.04**

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Ubuntu/19.04 stable opennebula" > /etc/apt/source
```

#### **Ubuntu 19.10**

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Ubuntu/19.10 stable opennebula" > /etc/apt/source
```

### **2.2.3 ステップ3. ソフトウェアのインストール**

#### **CentOS/RHEL へのインストール**

OpenNebulaは、ベースリポジトリには含まれていないパッケージに依存しております。インストール前に以下のリポジトリを有効にする必要がございます：

- RHEL 7のみ: **optional** および **extras** RHEL リポジトリ
- **EPEL** (Extra Packages for Enterprise Linux)

#### **EPEL リポジトリ**

CentOS では、追加リポジトリの設定と共にパッケージをインストールするだけで EPEL を有効化できます：

```
# yum install epel-release
```

RHEL 7 では、以下のコマンドを実行することで EPEL を有効にできます：

---

**注記：RHEL 7 のオプションおよびエクストラリポジトリは、まず設定する必要があります。**

---

```
# subscription-manager repos --enable rhel-7-server-optional-rpms# subscription-
manager repos --enable rhel-7-server-extras-rpms
# rpm -ivh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

RHEL 8 では、以下のコマンドを実行して EPEL を有効にします：

```
# rpm -ivh https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

## OpenNebula のインストール

CentOS/RHEL 用の OpenNebula フロントエンドを、**当社のリポジトリ**からパッケージをインストールして導入します。以下のコマンドを root 権限で実行してください：

```
# yum install opennebula-server opennebula-sunstone opennebula-ruby opennebula-gate
←opennebula-flow
```

### CentOS/RHEL パッケージ説明

OpenNebula フロントエンドおよび仮想化ホスト用のパッケージは以下の通りです：

- **opennebula**: コマンドラインインターフェース。
- **opennebula-server**: OpenNebula のメインデーモン、スケジューラなど。
- **opennebula-sunstone**: *Sunstone* (GUI) および EC2 API。
- **opennebula-gate**: VM と OpenNebula 間の通信を可能にする OneGate サーバーです。
- **opennebula-flow**: OneFlow はサービスと弾力性を管理します。
- **opennebula-provision**: OneProvision は、リモートベアメタルクラウドプロバイダ上に新しいクラスタを展開します。
- **opennebula-node-kvm**: oneadmin ユーザー、libvirt、および kvm をインストールするメタパッケージです。
- **opennebula-common**: OpenNebula パッケージ共通のファイル群です。
- **opennebula-rubygems**: 必要なすべての Ruby gem に依存するメタパッケージです。
- **opennebula-rubygem-\$NAME**: Ruby gem \$NAME を含むパッケージ。
- **opennebula-debuginfo**: デバッグ情報を含むパッケージです。
- **opennebula-ruby**: Ruby バインディング。
- **opennebula-java**: Java バインディング。
- **python-pyone**: Python バインディング。
- **python3-pyone**: Python3 バインディング。

**注記:** 設定ファイルは /etc/one および /var/lib/one/remotes/etc に配置されています。

---

## Debian/Ubuntu へのインストール

Debian/Ubuntu フロントエンドに OpenNebula を当リポジトリのパッケージを使用してインストールするには、root 権限で以下を実行してください:

```
# apt-get update
# apt-get install opennebula opennebula-sunstone opennebula-gate opennebula-flow
```

## Debian/Ubuntu パッケージの説明

これらのディストリビューションで利用可能なパッケージは以下の通りです：

- **opennebula**: OpenNebula デーモンです。
- **opennebula-common**: ユーザーおよび共通ファイルを提供します。
- **opennebula-tools**: コマンドラインインターフェースを提供します。
- **opennebula-sunstone**: *Sunstone* (GUI)。
- **opennebula-gate**: 仮想マシンとOpenNebula間の通信を可能にするOneGateサーバーです。
- **opennebula-flow**: OneFlow はサービスと弾力性を管理します。
- **opennebula-provision**: OneProvisionは、リモート上のペアメタルクラウドプロバイダー上に新しいクラスターを展開します。
- **opennebula-node**: ノードを OpenNebula KVM ノードとして準備します。
- **opennebula-node-lxd**: ノードを OpenNebula LXD ノードとして準備します。
- **opennebula-lxd-snap**: LXD スナップをインストールします (Ubuntu 16.04 および 18.04 のみ)。
- **opennebula-rubygems**: 必要なすべての Ruby gem に依存するメタパッケージです。
- **opennebula-rubygem-\$NAME**: Ruby gem \$NAME を含むパッケージです。
- **opennebula-dbgsym**: デバッグ情報を含むパッケージです。
- **ruby-opennebula**: Ruby API。
- **libopennebula-java**: Java API。
- **libopennebula-java-doc**: Java API ドキュメント。
- **python-pyone**: Python API。
- **python3-pyone**: Python3 バインディング。

---

**注記:** 設定ファイルは `/etc/one` および `/var/lib/one/remotes/etc` に配置されています。

---

## 2.2.4 ステップ4. Rubyランタイムのインストール（任意）

**警告**：OpenNebula 5.10以降、この手順は**任意**となり、必要なRuby gemsは全てopennebula-rubygem-\$NAMEパッケージ群およびopennebula-rubygemsメタパッケージとして提供されます。Ruby gemsは専用のディレクトリ /usr/share/one/gems-dist/にインストールされますが、OpenNebulaは(シンボリックリンクされた)場所 /usr/share/one/gems/を通じてそれらを利用します。この場所は gems-dist/ ディレクトリを指しています。gems/ ディレクトリが(デフォルトで)存在する場合、OpenNebulaは検索パスから他のシステム Ruby gems の場所をすべて除外することで、このディレクトリ内の gems を**排他的に**使用します。

```
# ls -lad /usr/share/one/gems*
lrwxrwxrwx 1 root root      9 Aug 13 11:41 /usr/share/one/gems -> gems-dist
drwxr-xr-x 9 root root 4096 Aug 13 11:41 /usr/share/one/gems-dist
```

パッケージ化されたRuby gemsではなく、システム全体のRuby gemsを使用したい場合は、シンボリックリンク /usr/share/one/gems/ を削除し、下記で説明するinstall\_gemsスクリプトを使用して必要な依存関係をすべてインストールしてください。削除された

/usr/share/one/gems/ シンボリックリンクは、次の OpenNebula アップグレード時には再作成されません。OpenNebula 付属の Ruby gems はアンインストールできませんが、/usr/share/one/gems/ シンボリックリンクを削除することで使用を無効化できます。

**カスタムドライバやツールで追加のRuby gemsが必要な場合は、導入済みの専用ディレクトリにインストールする必要があります**  
GEM\_HOME=/usr/share/one/gems/  
gem名を\$GEM\_NAMEに設定し、特権ユーザとしてrootで以下を実行します：

OpenNebulaのいくつかのコンポーネントはRubyライブラリを必要とします。OpenNebulaでは、必要なgemおよび開発用ライブラリパッケージをインストールするスクリプトを提供しております。

root 権限で以下を実行してください：

```
# test -L /usr/share/one/gems && unlink /usr/share/one/gems #
/usr/share/one/install_gems
```

上記のスクリプトは、一般的な Linux ディストリビューションを検出し、必要なライブラリをインストールするよう設計されています。お使いのシステムで必要なパッケージが見つからない場合は、以下のパッケージを手動でインストールしてください：

- sqlite3 開発ライブラリ
- mysql クライアント開発ライブラリ
- curl 開発ライブラリ
- libxml2 および libxslt 開発ライブラリ
- ruby 開発ライブラリ
- gcc および g++
- make

特定のコンポーネント向けに特定のgemのみをインストールしたい場合は、「ソースコードからのビルド」をご参照ください。そちらでより詳細に説明されています。

## 2.2.5 ステップ5. MySQL/MariaDBの有効化（任意）

OpenNebulaをできるだけ迅速にデプロイしたい場合、この手順はスキップ可能です。ただし、本番環境やより厳格な環境でのデプロイをご検討の場合は、必ず「MySQL の設定」セクションをご確認ください。

SQLite から MySQL への切り替えは可能ですが、データベースの移行はより煩雑になるため、迷われる場合は最初から MySQL をご利用されることをお勧めいたします。

## 2.2.6 ステップ 6. OpenNebula の起動

**警告：**アップグレード作業中の方は、この手順と次の手順をスキップし、アップグレード文書に戻ってください。

---

oneadmin ユーザーとしてログインし、以下の手順に従ってください：

/var/lib/one/.one/one\_auth ファイルは、ランダムに生成されたパスワードで作成されています。以下のような内容が含まれているはずです：oneadmin:<パスワード>。OpenNebula を起動する前に、パスワードを変更していただいて構いません。例：

```
$ echo "oneadmin:mypassword" > ~/.one/one_auth
```

**警告：**これにより初回起動時に oneadmin のパスワードが設定されます。以降は `oneuser passwd` コマンドを使用して oneadmin のパスワードを変更する必要があります。oneadmin パスワードの変更方法の詳細はこちらをご参照ください。

OpenNebula デーモンを起動する準備が整いました。systemdを採用しているLinuxディストリビューションでは、`systemctl`を使用できます：

```
# systemctl start opennebula
# systemctl start opennebula-sunstone
```

または、古いLinuxシステムではserviceコマンドを使用してください：

```
# service opennebula start
# service opennebula-sunstone start
```

---

注: OpenNebula サービスの起動に失敗した場合は、以下の点を確認してください。

- [ステップ4. Ruby ランタイムのインストール（任意）](#) に従いました。
  - Debian および Ubuntu における既知の問題の影響を受けません。
- 

## 2.2.7 ステップ7. インストール確認

OpenNebulaを初めて起動した後、コマンドがOpenNebula デーモンに接続できることを確認してください。これはLinux CLIまたはグラフィカルユーザーインターフェースであるSunstoneで行うことができます。

### Linux CLI

フロントエンドで、oneadmin として以下のコマンドを実行してください：

```
$ oneuser show USER 0
INFORMATION

ID          : 0
NAME        : oneadmin
GROUP       : oneadmin
PASSWORD    : 3bc15c8aae3e4124dd409035f32ea2fd6835efc9
AUTH_DRIVER : コア
ENABLED     : はい

ユーザーテンプレート
TOKEN_PASSWORD="ec21d27e2fe4f9ed08a396cbd47b08b8e0a4ca3c"

リソース使用状況とクォータ
```

エラーメッセージが表示された場合、OpenNebula デーモンが正常に起動できなかった可能性があります：

```
$ oneuser show
localhost:2633 への TCP 接続を開けませんでした (接続拒否 - connect(2) for
--"localhost" port 2633)
```

OpenNebula のログは `/var/log/one` に保存されています。少なくとも `oned.log` と `sched.log`、つまりコアとスケジューラのログファイルが存在しているはずです。`oned.log` ファイルに [E] でマークされたエラーメッセージがないかご確認ください。

## サンストーン

サンストーンのウェブインターフェースにログインをお試しいただけます。 ブラウザで `http://<フロントエンドのアドレス>:9869` へアクセスしてください。問題がなければ、ログインページが表示されます。ユーザー名は `oneadmin` パスワードはフロントエンドの `/var/lib/one/.one/one_auth` ファイルに記載されているものとなります。

ページが読み込まれない場合は、`/var/log/one/sunstone.log` および `/var/log/one/sunstone.error` を必ずご確認ください。また、TCP ポート 9869 がファイアウォールで許可されていることをご確認ください。

## ディレクトリ構造

以下の表は、インストール後にフロントエンドで利用可能な主なパスの一覧です：

パス	説明
<code>/etc/one/</code>	設定ファイル
<code>/var/log/one/</code>	ログファイル（主なもの： <code>oned.log</code> 、 <code>sched.log</code> 、 <code>sunstone.log</code> 、および <code>&lt;vmid&gt;.log</code> ）
<code>/var/lib/one/</code>	<code>oneadmin</code> ホームディレクトリ
<code>/var/lib/one/datastores/&lt;dsid&gt;/</code>	データストアのストレージ
<code>/var/lib/one/vms/&lt;vmid&gt;/</code>	仮想マシン用のアクションファイル（デプロイメントファイル、転送マネージャースクリプトなど）
<code>/var/lib/one/.one/one_auth</code>	<code>oneadmin</code> の認証情報
<code>/var/lib/one/remotes/</code>	ホストに同期されるプローブおよびスクリプト
<code>/var/lib/one/remotes/hooks/</code>	フックスクリプト
<code>/var/lib/one/remotes/vmm/</code>	仮想マシンマネージャードライバスクリプト
<code>/var/lib/one/remotes/auth/</code>	認証ドライバスクリプト
<code>/var/lib/one/remotes/im/</code>	情報マネージャー（監視）ドライバー スクリプト
<code>/var/lib/one/remotes/market/</code>	マーケットプレイス ドライバスクリプト
<code>/var/lib/one/remotes/datastore/</code>	データストア ドライバスクリプト
<code>/var/lib/one/remotes/vnm/</code>	ネットワーク ドライバスクリプト
<code>/var/lib/one/remotes/tm/</code>	転送マネージャードライバスクリプト

## ファイアウォール設定

以下のリストは、OpenNebula が使用するポートを示しています。OpenNebula が正常に動作するためには、これらのポートを開く必要があります：

ポート	説明
9869	Sunstoneサーバー。
29876	VNC プロキシポート。VNC 接続をハイパーバイザーへ変換およびリダイレクトするために使用されます。
2633	OpenNebulaデーモン、主要なXML-RPC APIエンドポイントです。
2474	OneFlowサーバー。このポートは、OneFlowサーバーをご利用の場合にのみ開ける必要があります。
5030	OneGateサーバー。このポートはOneGateサーバーを使用する場合にのみ開ける必要があります。

OpenNebulaはSSH（ポート22）を介してハイパーバイザーに接続します。さらに、onedはOpenNebulaマーケットプレイス（<https://marketplace.opennebula.systems/>）およびLinuxコンテナマーケットプレイス（<https://images.linuxcontainers.org>）に接続し、利用可能なアプライアンスの一覧を取得する場合があります。これらのポートおよびプロトコルへのアウトバウンド接続を開放する必要があります。注記：これらはデフォルトのポートです。各コンポーネントは特定のポートにバインドするよう設定したり、HTTPプロキシを使用したりすることが可能です。

## 2.2.8 ステップ8. 次に実施すべき手順

OpenNebulaサービスの起動に成功しましたので、クラウドにハイパーバイザーを追加するため、次の章「ノードのインストール」に進んでください。

---

注記: oneadmin のパスワードを変更するには、以下の手順に従ってください:

```
#oneuser passwd 0 <パスワード>
#echo 'oneadmin:パスワード' > /var/lib/one/.one/one_auth
```

*oneuser show* コマンドを使用して、すべてが正常に動作することを確認してください。

---

## 2.3 MySQL の設定

MySQL/MariaDB バックエンドは、デフォルトの SQLite バックエンドの代替となります。本ガイドおよび OpenNebula のその他のドキュメントや設定ファイルでは、このデータベースを MySQL と呼称いたします。ただし、OpenNebula は MySQL または MariaDB のいずれかを使用可能です。

両方のバックエンド（SQLite と MySQL）を同時に共存させることはできません。OpenNebula のインストール計画を立てる際に、どちらを使用するか決定する必要があります。

---

ご注意：OpenNebulaをMySQLバックエンドでインストールされる場合は、初回起動前にこちらのガイドに従っていただくことで、oneadminおよびserveradminの認証情報に関する問題を回避できます。

---

### 2.3.1 インストール

まず、正常に動作する MySQL サーバーが必要です。OpenNebula のインストール用に新たにデプロイするか、フロントエンドからアクセス可能な既存の MySQL を再利用することができます。

### **MySQLの設定**

新しいユーザーを追加し、`opennebula` データベースに対する権限を付与する必要があります。この新しいデータベースは、OpenNebula が初回起動時に作成するため、事前に存在させる必要はありません。

デフォルト値を使用する場合、MySQLサーバーにログインし、以下のコマンドを実行してください：

```
$ mysql -u root -p Enter
password:
MySQL モニターへようこそ。[...]
mysql> GRANT ALL PRIVILEGES ON opennebula.* TO 'oneadmin' IDENTIFIED BY '<thepassword>
->';
クエリが正常に実行されました。影響を受けた行数: 0 行 (0.00 秒)
```

アカウントの管理方法については、[MySQLのドキュメント](#)をご参照ください

。次に、トランザクションの分離レベルを設定します：

```
mysql> SET GLOBAL TRANSACTION ISOLATION LEVEL READ COMMITTED;
```

### OpenNebulaの設定

OpenNebulaを初めて実行される前に、[oned.configuration](#) ファイルに接続の詳細情報と、権限を付与されたデータベースを設定する必要があります。

```
# MySQL のサンプル設定
DB = [ backend = "mysql", server =
       "localhost", port      =
       0,
       user      = "oneadmin", passwd =
       "<パスワード>", db_name =
       "opennebula" ]
```

フィールド：

- サーバー**: MySQLサーバーを実行しているマシンのURLです。
- ポート**: サーバーへの接続に使用するポート番号です。0に設定された場合、デフォルトのポートが使用されます。
- user**: MySQLのユーザー名です。
- passwd**: MySQL のパスワードです。
- db\_name**: OpenNebula が使用する MySQL データベースの名前です。

### 2.3.2 OpenNebula と MySQL の併用

このインストールおよび設定プロセスが完了しましたら、通常通りOpenNebulaをご利用いただけます。

### 2.3.3 MySQL データベースのメンテナンス

データベースのパフォーマンスを最適化するためには、環境の負荷に応じて定期的に実施すべき作業がいくつかあります。それらを以下に列挙します。

#### 検索インデックス

OpenNebulaのデータベースでは、様々な属性で仮想マシンを検索できるようにするため、全文検索（FTS）インデックスが設定されています。このインデックスのサイズは、クラウドの負荷状況により急速に増加する可能性があります。空き領域を確保するため、定期的に以下のメンテナンス作業を実施してください：

```
alter table vm_pool drop index ftidx;
alter table vm_pool add fulltext index ftidx (search_token);
```

### DONE 状態の VM

仮想マシンが終了されると、OpenNebulaはその状態をDONEに変更しますが、将来的に仮想マシンの情報が必要となる場合（例：会計レポートの生成）に備え、データベース内に仮想マシンを保持します。仮想マシンテーブルのサイズを削減するため、不要となったDONE状態の仮想マシンは定期的に削除することをお勧めいたします。このタスクには、`onedb purge-done`ツールをご利用いただけます。

## ノードのインストール

### 3.1 概要

OpenNebulaフロントエンドの設定が完了しましたら、次に仮想マシンを実行するホストの準備を行います。

#### 3.1.1 本章の読み方について

本章をお読みになる前に、[フロントエンドが正しくインストールされていることをご確認ください。](#)

本章では、OpenNebula のデプロイを完了するために必要な最小限のノード設定に焦点を当てています。ストレージやネットワークなどの概念は簡略化されています。そのため、他のサブシステムを設定するには、他の特定の章を自由に参照してください。

なお、他のガイドを参照せずに本章のみを進めていただいても、最終的には仮想マシンのデプロイが可能となります。将来的に他のストレージ技術やネットワーク技術への切り替えをご希望の場合も、その準備は整っています。

ノードのインストールと[検証](#)が完了しましたら、クラウドの利用を開始するか、さらにコンポーネントを設定することができます。

- [認証](#)（任意）。OpenNebulaをLDAP/ADと統合する場合、またはその他の認証技術でさらにセキュリティを強化する場合に必要です。
- [Sunstone](#)。この段階ではOpenNebulaのGUIは動作しアクセス可能であるはずですが、本ガイドをお読みいただくことで、Sunstone向けの特定の拡張設定について学ぶことができます。

#### 3.1.2 ハイパーバイザーの互換性

セクション	互換性
<a href="#">KVM ノードのインストール</a>	このセクションは KVM に適用されます。
<a href="#">LXD ノードのインストール</a>	このセクションは LXD に適用されます。
<a href="#">vCenter ノードのインストール</a>	このセクションは vCenter に適用されます。
<a href="#">インストール内容の確認</a>	このセクションは、vCenter、KVM、および LXD に適用されます。

クラウドが KVM または LXD ベースの場合は、以下の手順もご参照ください：

- [Open Cloud Host の設定](#)。
- [オープンクラウドストレージの設定](#)。
- [オープンクラウドネットワーキング](#)

グの設定を行います。ただし、VMwareベ

ースの場合は以下の手順となります：

- *VMware インフラストラクチャ設定*の章をご覧ください。

## 3.2 KVMノードのインストール

このページでは、バイナリパッケージから OpenNebula をインストールする方法をご説明します。

当サイトで提供しているパッケージを使用することをお勧めいたします。これにより、最新版のインストールが保証され、異なるディストリビューション間でパッケージの差異が生じる可能性を回避できます。ここでは2つの方法がございます：当社のパッケージリポジトリをシステムに追加するか、ソフトウェアメニューにアクセスして、お使いのLinuxディストリビューション向けの最新版パッケージをダウンロードしてください。

### 3.2.1 ステップ 1. OpenNebula リポジトリの追加

#### CentOS/RHEL

OpenNebulaリポジトリを追加するには、root権限で以下のコマンドを実行してください：

##### CentOS/RHEL 7

```
# cat << "EOT" > /etc/yum.repos.d/opennebula.repo [opennebula]
name=opennebulabaseurl=https://downloads.opennebula.org/repo/5.10/CentOS/7/$base_searchenabled=1
gpgkey=https://downloads.opennebula.org/repo/repo.key gpgcheck=1
repo_gpgcheck=1 EOT
```

##### CentOS/RHEL 8

```
# cat << "EOT" > /etc/yum.repos.d/opennebula.repo [opennebula]
name=opennebulabaseurl=https://downloads.opennebula.org/repo/5.10/CentOS/8/$base_searchenabled=1
gpgkey=https://downloads.opennebula.org/repo/repo.key gpgcheck=1
repo_gpgcheck=1 EOT
```

#### Debian/Ubuntu

Debian/Ubuntu に OpenNebula リポジトリを追加するには、root 権限で以下のコマンドを実行してください：

```
# wget -q -O- https://downloads.opennebula.org/repo/repo.key | apt-key add -
```

##### Debian 9

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Debian/9 stable opennebula" > /etc/apt/source
```

##### Debian 10

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Debian/10 stable opennebula" > /etc/apt/source
```

##### Ubuntu 16.04

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Ubuntu/16.04 stable opennebula" > /etc/apt/source
```

**Ubuntu 18.04**

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Ubuntu/18.04 stable opennebula" > /etc/apt/so
```

**Ubuntu 19.04**

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Ubuntu/19.04 stable opennebula" > /etc/apt/so
```

**Ubuntu 19.10**

```
# echo "deb https://downloads.opennebula.org/repo/5.10/Ubuntu/19.10 stable opennebula" > /etc/apt/so
```

### 3.2.2 ステップ2. ソフトウェアのインストール

#### CentOS/RHEL へのインストール

以下のコマンドを実行し、ノードパッケージをインストールし、OpenNebulaが提供する設定ファイルを使用するためにlibvirtを再起動します：

```
$ sudo yum install opennebula-node-kvm
$ sudo systemctl restart libvirdt
```

#### 新しいQEMU/KVM (CentOS/RHEL 7のみ)

より新しく機能豊富なエンタープライズ版QEMU/KVMをご利用いただくことで、メリットが得られる場合がございます。基本版（qemu-kvm）とエンタープライズ版（RHELではqemu-kvm-rhev、CentOSではqemu-kvm-ev）パッケージの違いについては、[Red Hatカスタマーポータル](#)に記載されております。

CentOS 7では、エンタープライズパッケージは別のリポジトリに含まれています。基本パッケージを置き換えるには、以下の手順に従ってください：

```
$ sudo yum install centos-release-qemu-ev
$ sudo yum install qemu-kvm-ev
```

RHEL 7では、Red Hat Virtualization (RHV) または Red Hat OpenStack (RHOS) 製品の有料サブスクリプションが必要です。Red Hat Enterprise Linux のライセンスだけでは不十分です。ライセンスを取得したバージョンに対応する RHV インストールガイドをご確認ください。通常、以下のコマンドでエンタープライズパッケージを有効化およびインストールできます：

```
$ sudo subscription-manager repos --enable rhel-7-server-rhv-4-mgmt-agent-rpms
$ sudo yum install qemu-kvm-rhev
```

詳細な設定については、[KVM](#)の具体的なガイドをご確認ください。

#### Debian/Ubuntu へのインストール

以下のコマンドを実行し、ノードパッケージをインストールし、OpenNebulaが提供する設定ファイルを使用するためにlibvirtを再起動してください：

```
$ sudo apt-get update
$ sudo apt-get install opennebula-node
$ sudo service libvirtd restart # Debian
$ sudo service libvirt-bin restart # Ubuntu
```

詳細な設定については、[KVM](#)の具体的なガイドをご確認ください。

### 3.2.3 ステップ 3. CentOS/RHEL における SELinux の設定

**警告：**アップグレード作業中の方は、この手順および次の手順をスキップし、アップグレード文書に戻ってください。

SELinuxは、OpenNebulaフロントエンドが開始する一部の操作をブロックする場合があり、その結果、すべてのノード操作が完全に失敗する（例：oneadminユーザーのSSH認証情報が信頼されていない場合）か、仮想マシンに関連する特定の操作のみが個別に失敗することがあります。管理者がSELinuxの設定に慣れていない場合は、**予期せぬ失敗を避けるため、この機能を無効化することをお勧めします。** インストールが正常に動作している状態であれば、後からいつでもSELinuxを有効化できます。

#### SELinuxを無効化（推奨）

SELinuxを無効化するには、/etc/selinux/config内の以下の行を変更してください：

```
SELINUX=disabled
```

変更後は、システムを再起動する必要があります。

#### SELinuxを有効にする

SELinuxを強制モードで有効にするには、/etc/selinux/configファイル内の以下の行を変更してください：

```
SELINUX=enforcing
```

無効状態から変更する場合、次回起動時にファイルシステムの再ラベル付けを実行する必要があります。そのためには、/.autorelabel ファイルを作成してください。例：

```
$ touch /.autorelabel
```

変更後は、マシンを再起動してください。

---

**注記：**OpenNebula のデプロイメントタイプによっては、SELinux を有効にした KVM ノードで以下の対応が必要となる場合があります：

- \* util-linux パッケージが 2.23.2-51 より新しいバージョンでインストールされていること
- SELinuxのvirt\_use\_nfsブール値を有効化（NFS上のデータストアを使用する場合）：

```
$ sudo setsebool -P virt_use_nfs on
```

SELinux の設定およびトラブルシューティングに関する詳細情報は、SELinux ユーザーおよび管理者ガイドをご参照ください。

---

### 3.2.4 ステップ 4. パスワード不要の SSH を設定する

OpenNebulaフロントエンドは、SSHを使用してハイパーバイザーホストに接続します。すべてのマシンからoneadminユーザーの公開鍵を、すべてのマシンの/var/lib/one/.ssh/authorized\_keysファイルに配布する必要があります。SSH鍵の配布方法は複数存在します。最終的には管理者が方法を選択する必要がありますが、構成管理システムの使用をお勧めします。本ガイドでは、手動でSSH鍵をscp転送する方法を紹介します。

フロントエンドにパッケージをインストールした際、SSHキーが生成され、`authorized_keys` ファイルに追加されました。フロントエンドからノードへ、`id_rsa`、`id_rsa.pub`、および `authorized_keys` を同期いたします。加えて、`known_hosts` ファイルを作成し、同様にノードへ同期する必要があります。`known_hosts` ファイルを作成するには、フロントエンド上でユーザー `oneadmin` として、以下のコマンドを実行し、すべてのノード名とフロントエンド名をパラメータとして指定いたします：

```
$ ssh-keyscan <フロントエンド> <ノード1> <ノード2> <ノード3> ... >> /var/lib/one/.ssh/known_hosts
```

次に、ディレクトリ `/var/lib/one/.ssh` を全てのノードにコピーする必要があります。最も簡単な方法は、全てのホストで `oneadmin` ユーザーの一時パスワードを設定し、フロントエンドからディレクトリをコピーすることです：

```
$ scp -rp /var/lib/one/.ssh <node1>:/var/lib/one/
$ scp -rp /var/lib/one/.ssh <node2>:/var/lib/one/
$ scp -rp /var/lib/one/.ssh <node3>:/var/lib/one/
$ ...
```

フロントエンドからユーザー `oneadmin` としてノードおよびフロントエンド自体へ接続する際、またノードからフロントエンドへ接続する際、パスワードが要求されることをご確認ください。

```
$ ssh <frontend>
$ exit

$ ssh <node1>
$ ssh <frontend>
$ exit
$ exit

$ ssh <node2>
$ ssh <frontend>
$ exit
$ exit

$ ssh <ノード3>
$ ssh <frontend>
$ exit
$ exit
```

追加のセキュリティ層が必要な場合、秘密鍵をすべてのハイパーバイザーにコピーするのではなく、フロントエンドノードのみに保持することができます。この方法により、ハイパーバイザー上の `oneadmin` ユーザーは他のハイパーバイザーにアクセスできなくなります。これを実現するには、フロントエンドの `/var/lib/one/.ssh/config` を変更し、鍵を転送するための `ForwardAgent` オプションをハイパーバイザーホストに追加します：

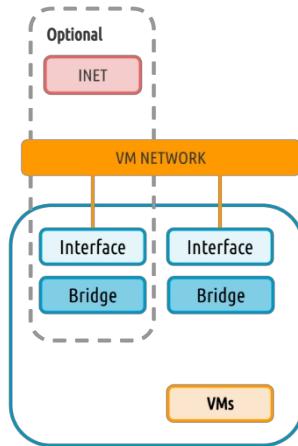
```
$cat /var/lib/one/.ssh/configHost
host1
  User
  oneadminForwardAgent
  yes
Host host2
  User oneadmin
  ForwardAgent yes
```

---

**注記：**OpenNebulaを起動する前に、対応する秘密鍵をインポートした状態でssh-agentを実行している必要があります。ssh-agentはeval "\$(ssh-agent -s)"を実行して起動し、秘密鍵はssh-add /var/lib/one/.ssh/id\_rsaを実行して追加できます。

---

### 3.2.5 ステップ 5. ネットワーク設定



OpenNebulaのフロントエンドデーモンがホストを管理・監視し、イメージファイルを転送するためには、ネットワーク接続が必要です。この目的には専用ネットワークの使用を強く推奨いたします。

ネットワークモデルには様々な種類がございます。OpenNebulaがサポートするネットワーク技術については、[ネットワーク構成](#)の章をご参照ください。

最もシンプルなネットワークモデル、すなわち [ブリッジドライバー](#)に対応するモデルをご利用になることをお勧めいたします。このドライバーをご利用になるには、Linuxブリッジを設定し、物理デバイスをそのブリッジに含める必要があります。その後、OpenNebulaでネットワークを定義する際、このブリッジの名前を指定すると、OpenNebulaは仮想マシンをこのブリッジに接続すべきことを認識し、ブリッジに接続された物理ネットワークデバイスとの接続性を提供します。例えば、2つの物理ネットワークを持つ典型的なホスト（1つはパブリックIPアドレス用、例えばeth0 NICに接続、もう1つはプライベート仮想LAN用、例えばNIC eth1）には、2つのブリッジが必要です：

```
$ ip link show type bridge4: br0:
...
5: br1: ...

$ ip link show master br0 2:
eth0: ...

$ ip link show master br1 3:
eth1: ...
```

**注記：**これはホストでのみ必要であり、フロントエンドでは不要です。また、リソースの正確な名称（br0、br1など）は重要ではありませんが、すべてのホストにおいてブリッジと NIC の名称が一致していることが重要です。

### 3.2.6 ステップ 6. ストレージ設定

OpenNebulaを試しに利用したいだけの場合、この手順は完全にスキップ可能です。デフォルト設定では、イメージの保存にはフロントエンドのローカルストレージが、稼働中の仮想マシンのストレージにはハイパーバイザーのローカルストレージが使用されるよう構成されています。

ただし、この段階でCeph、NFS、LVMなどの別のストレージ構成を設定したい場合は、「*Open Cloud Storage*」の章をお読みください。

### 3.2.7 ステップ7. OpenNebulaへのホストの追加

この手順では、OpenNebulaフロントエンドにインストールしたノードを登録し、OpenNebulaがそのノード上で仮想マシンを起動できるようにします。この操作は、コマンドラインインターフェース（CLI）またはグラフィカルユーザーインターフェース（Sunstone）のいずれかで実行できます。両方の方法で行う必要はありません、いずれか一方の方法で実行してください。

ホストサブシステムの詳細については、こちらのガイドをご参照ください。

#### Sunstone を使用したホストの追加

こちらのドキュメントに記載されている手順に従い、Sunstoneを起動してください。左側のメニューより「Infrastructure」→「Hosts」を選択し、「+」ボタンをクリックしてください。

The screenshot shows the OpenNebula Sunstone interface. On the left is a sidebar with navigation links: Dashboard, Instances, Templates, Storage, Network, Infrastructure (with Clusters and Hosts selected), Zones, System, and Settings. The Settings section includes a 'Sign in' button. The main area is titled 'Hosts' and contains a table with columns: ID, Name, Cluster, RVMs, Allocated CPU, Allocated MEM, and Status. A large 'info' icon and the text 'There is no data available' are displayed. At the top of the main area is a toolbar with buttons for 'Select cluster', 'Enable', 'Disable', 'Offline', and a search bar.

次に、ホスト名フィールドにノードの完全修飾ドメイン名（FQDN）を入力してください。

The screenshot shows the 'Create Host' page. On the left is a sidebar with navigation links. The main area has a 'Create' button, a 'Type' dropdown set to 'KVM', a 'Cluster' dropdown set to '0: default', and a 'Hostname' input field containing 'node01'. A 'Create' button is at the bottom.

最後に、ホスト一覧に戻り、ホストの状態がONに切り替わっていることをご確認ください。この処理には20秒から1分程度かかる場合があります。状態をより頻繁に確認するには、更新ボタンをクリックしてみてください。

The screenshot shows the 'Hosts' list page. The table has columns: ID, Name, Cluster, RVMs, Allocated CPU, Allocated MEM, and Status. A row for 'node01' is listed with values: ID 0, Name node01, Cluster 0, RVMs 0, Allocated CPU 0 / 800 (0%), Allocated MEM 0KB / 16GB (0%), and Status ON. The 'Status' column for 'node01' is circled in red.

ホストがONではなくerr状態になった場合は、`/var/log/one/oned.log`をご確認ください。SSHに問題がある可能性が高いです。

### CLI 経由でのホストの追加

クラウドにノードを追加するには、フロントエンドで `oneadmin` として以下のコマンドを実行してください：

```
$ onehost create <node01> -i kvm -v kvm

$ onehost list
  ID NAME          クラスター    RVM      割り当て済みCPU      割り当て済みメモリ状態
  1 localhost      デフォルト     0          -                  - 初期化

# しばらく経つて      (20秒～1分)

$ onehost list ID
  NAME          クラスター    RVM      割り当て済み CPU      割り当て済みメモリ STAT
  0 ノード01      デフォルト     0          0 / 400 (0%)   OK / 7.7G (0%) 正常
```

ホストがオン状態ではなくエラー状態になった場合は、`/var/log/one/oned.log` を確認してください。おそらくSSHに問題がある可能性があります。

### 3.2.8 ステップ8. 現在実行中の仮想マシンのインポート（任意）

この手順は、仮想マシンのインポートはいつでも行えるため、スキップすることも可能です。ただし、OpenNebula で以前にデプロイした仮想マシンを表示したい場合は、仮想マシンのインポート機能をご利用いただけます。

### 3.2.9 ステップ9. 次に取るべき手順

テスト用仮想マシンを起動するには、オプションの「[インストール内容の確認](#)」セクションに進んでください。

それ以外の場合は、クラウドの利用を開始するか、以下のコンポーネントの設定を続行できます：

- [認証](#)（任意） OpenNebula を LDAP/AD と統合する場合、またはその他の認証技術でさらにセキュリティを強化する場合に設定します。
- [Sunstone](#)。この段階ではOpenNebulaのGUIは動作しアクセス可能ですが、本ガイドをお読みいただくことでSunstoneの特定の拡張設定について学ぶことができます。
- [Open Cloud Host の設定](#)。
- [Open Cloud Storage の設定](#)。
- [Open Cloud Networking の設定](#)。

## 3.3 LXDノードのインストール

LXDノードのインストールに必要な手順の大部分は、KVMと同様です。これは、どちらもLinux OSに依存しているためです。同様の手順には、名称にリンクが記載され、「[KVMと同様](#)」と明記されます。

### 3.3.1 ステップ1. OpenNebulaリポジトリの追加

[KVMと同様](#)

### 3.3.2 ステップ 2. ソフトウェアのインストール

LXDノードは、Ubuntu 16.04、18.04、19.04、19.10 および Debian 10 でのみサポートされています。

**ご注意：LXD 3.0.x リリースのみがサポート対象となります。LXD 2.x およびそれ以前のバージョンは互換性がございません。**

## Ubuntuへのインストール

Ubuntu 16.04 および 18.04 では、可能な限りディストリビューションパッケージ（APT経由）として提供されているLXDのご利用をお勧めいたします。[LXC CLI](#)を使用する際、snapがoneadminで正常に動作しない可能性がございます。また、snapベースのインストールは、チャネルで更新が利用可能になるたびに自動的に更新されます。Ubuntu 18.10以降、LXDはsnapとしてのみ提供され、ディストリビューションの移行用パッケージによって管理されております。

**注記：**OpenNebula 5.8.2 以降、Ubuntu 16.04 および 18.04 向けに LXD スナップインストールを管理する新たなメタパッケージ **opennebula-lxd-snap** が提供されています。ユーザー様は、ディストリビューションパッケージ **lxd** として提供される LXD と、**opennebula-lxd-snap** によりインストールされる LXD スナップのいずれかを選択いただけます。ご希望のものを事前に手動でインストールすることで選択が可能です。パッケージを選択しない場合、既存のスナップインストールが存在していても、パッケージ化されたLXDが使用されます。Ubuntu 18.10では、LXDスナップのインストールは**opennebula-node-lxd**によって直接行われます。既にインストールされているLXDスナップを上書きすることはありませんので、ご注意ください。

**警告：**Ubuntu にパッケージ化された LXD のバージョンについて：

- Ubuntu 16.04 — 非互換の LXD 2.0 が含まれております。xenial-backports には LXD 3.0 が提供されております。
- Ubuntu 18.04 — LXD 3.0
- Ubuntu 19.04 以降 — lxd は LXD スナップ (3.0/安定版チャネル) をインストールするための移行用パッケージです

Ubuntu 16.04 では、サポート対象の LXD 3.0 パッケージを利用するには、**xenial-backports** を通常優先度で有効にする必要があります。

以下のデプロイメントオプションをご確認いただき、LXDが既にインストールされ使用されている環境でのインストールには十分ご注意ください。

### Ubuntu 16.04

```
$ sudo apt-get install opennebula-node-lxd
```

または、OpenNebula 5.8.2以降でスナップとしてインストールされているLXDをご利用ください：

```
$ sudo apt-get install opennebula-lxd-snap opennebula-node-lxd
```

### Ubuntu 18.04

```
$ sudo apt-get install opennebula-node-lxd
```

または、OpenNebula 5.8.2以降では、snapとしてインストールされたLXDをご利用ください：

```
$ sudo apt-get install opennebula-lxd-snap opennebula-node-lxd
```

### Ubuntu 19.04 以降および Debian 10 以降

```
$ sudo apt-get install opennebula-node-lxd
```

## すべてのオプション

Cephストレージドライバーをご利用いただくには、以下のコマンドで追加パッケージのインストールが必要になる場合がございます：

```
$ sudo apt-get install rbd-nbd
```

詳細な設定については、[LXD](#) の特定ガイドをご確認ください。

### 3.3.3 ステップ4. パスワード不要のSSHの設定

*KVMと同様です*

### 3.3.4 ステップ5. ネットワーク設定

*KVMと同様です*

### 3.3.5 ステップ6. ストレージ設定

*KVMと同様です*

### 3.3.6 ステップ7. OpenNebulaへのホストの追加

*KVMと同様です*

CLIおよびSunstoneにおいて、lxrdをKVMに置き換えてください

### 3.3.7 ステップ8. 既存コンテナのインポート（任意）

既存のコンテナを管理したい場合は、VMインポート機能をご利用いただけます。インポートするには、コンテナ名が「one-<id>」というパターンで命名されていないことが必要です。また、limits.cpu.allowance、limits.cpu、limits.memoryキーが定義されている必要があります。これらが定義されていない場合、OpenNebulaはコンテナをインポートできません。opennebula-node-lxdパッケージは、これらの値を含むデフォルトテンプレートを設定するはずです。

### 3.3.8 ステップ9. 次に取るべき手順

*KVMと同様です*

## 3.4 vCenterノードのインストール

本セクションでは、OpenNebulaによる管理を可能とするために、vCenterおよびESXインスタンスにおいて必要な設定要件を説明します。

VMware vCenter ドライバーにより、OpenNebula は 1 つ以上の ESX クラスターを管理する 1 つ以上の vCenter サーバーにアクセスすることが可能となります。各 ESX クラスターは、OpenNebula において集約されたハイパーバイザー、すなわち OpenNebula ホストとして表現されます。これは、ESX クラスターごとに 1 つの OpenNebula ホストが表現されることを意味します。

したがって、OpenNebula のスケジューリング決定は ESX クラスタレベルで行われます。vCenter はその後、DRS コンポーネントを

---

使用して、仮想マシンを展開する実際の ESX ホストとデータストアを選択します。

### 3.4.1 要件

機能する vCenter 環境を実現するには、以下の要件を満たす必要があります：

- サポート対象の vSphere バージョン（プラットフォームノートをご確認ください）と、少なくとも 1 つの ESX を集約するクラスターが 1 つ以上存在すること。
- OpenNebula 用に vCenter ユーザーを定義します。この vCenter ユーザー（仮に `oneadmin` と称します）は、OpenNebula が管理する ESX クラスターへのアクセス権限が必要です。問題発生を避けるため、最も手間のかからない方法は、**この `oneadmin` ユーザーを管理者として宣言すること** です。ただし、一部の企業環境では、ユーザーを管理者として宣言することが許可されていない場合があります。その場合、特定のタスクを実行するための権限を付与する必要があります。必要な権限の一覧は、本章の末尾に記載されています。
- OpenNebula に公開される同一 ESX クラスタに属するすべての ESX ホストは、少なくとも 1 つのデータストアを共有している必要があります。
- ESX クラスタでは DRS を有効化しておく必要があります。DRS は必須ではありませんが、推奨されます。OpenNebula は ESX ホスト単位でのスケジューリングを行わないため、クラスタ内の実際の ESX ホストを選択するには DRS が必要です。DRS が有効でない場合、仮想マシンは仮想マシンテンプレートが作成された ESX ホスト上で起動されます。
- 仮想標準スイッチを使用する場合、同じ ESX クラスタに属するすべての ESX ホストにそれらのスイッチが存在することを確認してください。これにより、ポートグループで表されるネットワークは、VM がどの ESX ホストで実行されても使用可能となります。分散仮想スイッチを使用する場合は、ESX ホストがスイッチに追加されていることを確認してください。
- VNC 機能を有効にするには、下記の「[ESX ホスト上の VNC](#)」セクションの詳細情報をご確認ください。

---

**重要：**OpenNebula は、vCenter ネットワークドライバを使用する場合の仮想スイッチおよびポートグループの作成、ならびに VMDK および/または ISO ファイル用のイメージ作成を除き、vCenter の設定を変更いたしません。

---

**ご注意：**セキュリティ上の理由から、異なる ESX クラスタへのアクセスには異なるユーザーを定義することが可能です。

OpenNebula では、OpenNebula ホストとしてカプセル化された各 ESX クラスタごとに異なるユーザーを定義できます。

---

### 3.4.2 設定

OpenNebula が vCenter と連携できるように設定するには、以下の簡単な手順が必要です：

#### ステップ1: 接続性の確認

OpenNebula フロントエンドは、管理対象となるすべての vCenter へのネットワーク接続が必要です。

さらに、起動された仮想マシンへの VNC アクセスを有効にするため、フロントエンドはすべての ESX ホストへのネットワーク接続性も必要とします。

<b>警告：</b> OpenNebula は vCenter インスタンスとの通信にポート 443 を使用します。ポート 443 は vCenter のデフォルトポートであるため、当該ポートをフィルタリングしていない場合、または vSphere Web Client からの接続を待機するポートを別々に設定している場合、vCenter との接続が不可能になります。	43
---	----

ステップ 2: フロントエンドでのドライバ設定 (`oned.conf`) [オプション]

/etc/one/oned.conf ファイル内の以下のセクションでは、vCenter に関する情報と仮想化ドライバーについて説明しています。これらはデフォルトで有効化されています：

```

#-----
# vCenter 情報ドライバ マネージャー設定 # -r ホスト監視時の再試行回数
#      -t スレッド数、すなわち同時に監視するホストの数
#-----
IM_MAD = [
    NAME          = "vcenter",
    SUNSTONE_NAME = "VMWare vCenter",
    EXECUTABLE    = "one_im_sh",
    引数          = "-c -t 15 -r 0 vcenter" ]
#-----

#-----
# vCenter 仮想化ドライバー マネージャーの設定 #      -r ホスト監視時の再試行回
数
#      -t スレッド数、すなわち同時に監視するホストの数
#      -p ホストごとに複数のアクションを並行実行する場合、ハイパー-バイザーのサポートが必要です #-s <シェル> コマンドを実
行するためのシェルを指定します。デフォルトはbashです
#      -d デフォルトのスナップショット戦略。'detach' または 'suspend' のいずれかです。デフォルトは 'suspend' で
す。      デフォルトは 'suspend' です。
#-----
VM_MAD = [
    NAME          = "vcenter",
    SUNSTONE_NAME = "VMWare vCenter",
    EXECUTABLE    = "one_vmm_sh",
    引数          = "-p -t 15 -r 0 vcenter -s sh", default =
    "vmm_exec/vmm_exec_vcenter.conf", TYPE      = "xml",
    IMPORTED_VMS_ACTIONS = "terminate, terminate-hard, hold, release, suspend, resume, delete,
                           reboot, reboot-hard, resched, unresched, poweroff, poweroff-hard, disk-attach, disk-
                           detach, nic-attach, nic-detach,
                           [スナップショット作成、スナップショット削除]
]
#-----
```

仮想化ドライバーとして、vCenter ドライバーは、その実行を制御する一連のパラメータを受け付けます。許可されるパラメータは以下の通りです：

パラメータ	説明
-r <数値>	アクション実行時の再試行回数
-t <num>	スレッド数、すなわち同時に実行されるアクションの数

これらのパラメータの詳細、およびドライバのカスタマイズや拡張方法については、仮想マシンドライバのリファレンスをご参照ください。

さらに、vCenter ドライバーの動作の一部は、/var/lib/one/remotes/etc/vmm/vcenter/vcenterrc で設定できます。ここで変更可能なパラメーターは以下の通りです：

param- eter	説明
:delete_ima	gYess : OpenNebulaを使用してイメージを削除できます。No : インポートされたイメージにはVCENTER_IMPORTED属性が設定されます 。この属性は、イメージが削除されるのを防ぎます。

/etc/one/oned.confファイルに変更を加えた後は、OpenNebulaを再起動する必要があります。以下のコマンドで実行できます：

```
$ sudo systemctl restart opennebula
```

### ステップ3: vCenterクラスタのインポート

OpenNebulaには、vCenterクラスタ、VMテンプレート、ネットワーク、および稼働中のVMをインポートするための強力なCLIツールが同梱されています。**onecenter**ツールは直感的に操作でき、vCenterホストへのアクセスに必要な認証情報とFQDN/IPを設定し、画面の指示に従うだけでご利用いただけます。

vCenterクラスタのインポート方法については、[vCenterインポートツール](#)をご参照ください。

vCenterクラスタの監視が正常に完了すると、ONがホストステータスとして表示されます。ERRORが表示された場合は、接続状態を確認し、原因を特定するために/var/log/one/oned.logファイルをご確認ください。

ESXクラスタを表すOpenNebulaホストには、以下の変数が追加されます：

操作	注記
VCENTER_HOST	vCenterホストのホスト名またはIPアドレス
VCENTER_USER	vCenter ユーザー名
VCENTER_PASSWORD	vCenter ユーザーのパスワード
VCENTER_VERSION	OpenNebulaによって検出されたvCenterのバージョン（例：5.5）
VCENTER_CCR_REF	vCenter クラスタへの管理対象オブジェクト参照
VCENTER_INSTANCE_ID	vCenterインスタンスのUUID識別子

---

**注記：**OpenNebulaは起動時に特別な鍵を生成し、/var/lib/one/.one/one\_keyに保存します。この鍵は、OpenNebulaがアクセス可能な全てのvCenterのパスワードを暗号化および復号化するための秘密鍵として使用されます。したがって、vCenterを表すOpenNebulaホストに表示されるパスワードは、この特別な鍵で暗号化された元のパスワードとなります。

---



---

**注記：**マネージドオブジェクト参照とは何か、またvCenterインスタンスのUUIDとは何かについての詳細は、vCenterドライバーセクションに記載しております。

---

### ステップ4: 次に実施すべき手順

仮想マシンを起動する、またはVMWareインフラストラクチャの設定方法について学ぶには、「[インストール内容の確認](#)」セクションへお進みください。

#### 3.4.3 権限要件

vCenter操作に使用するユーザー アカウントが管理者として宣言されていない場合、OpenNebulaがvCenterで実行するタスクに必要な権限を以下の表にまとめます。

権限 ID	権限名	備考
Datastore.AllocateSpace	スペースの割り当て	以下のVMFSデータストアすべてに対して
Datastore.Browse	データストアの参照	すべての VMFS データストアに対して

### OpenNebula 5.10 導入ガイド、リリース 5.10.5

Datastore.FileManagement	低レベルファイル操作	すべてのVMFSデータストアにおいて
Datastore.Delete	データストアの削除	以下のオブジェクトが表すすべての VMFS データストアに対して
DVPortgroup.Create	作成	OpenNebula に作成させたい場合に必要です

表 3.1 – 前ページからの続き

DVPortgroup.Delete	削除	OpenNebula に削除を行わせたい場合に必要です
DVPortgroup.Modify	変更	OpenNebula に作成させたい場合に必要です
DVSwitch.Create	作成	OpenNebula に作成させたい場合に必要です
DVSwitch.Delete	削除	OpenNebula に削除を行わせたい場合に必要です
DVSwitch.HostOp	ホスト操作	OpenNebula に作成を行っていただく場合に必要です
DVSwitch.Modify	変更	OpenNebula で作成したい場合に必要な設定です。
DVSwitch.PortSetting	ポート設定操作	OpenNebula に作成させたい場合に必要な設定です
Host.Config.Network	ネットワーク設定	すべての ESX ホストで必要です。
Network.Assign	ネットワークの割り当て	仮想マシンが接続するすべてのネットワークで必要です
リソースの推奨事項を適用	推奨事項を適用	すべてのストレージポッド (Storage DRS クラスタ) に対して
リソース仮想マシンをプールに割り当て	仮想マシンをリソースプールに割り当て	仮想マシンにリソースプールを割り当てるために必要です
Resource.ColdMigrate	電源オフ状態の仮想マシンの移行	電源オフ状態の仮想マシンを移行するために必要です
Resource.HotMigrate	電源が入った仮想マシンの移行	電源投入済みの仮想マシンを移行する必要があります
システムを読み込みます	読み取り	アップリンクポートグループの名前の変更に必要な操作
VirtualMachine.Config.AddExistingDisk	既存ディスクの追加	既存ディスクの参照および接続に必要です
VirtualMachine.Config.AddNewDisk	新規ディスクの追加	新しい仮想ディスクを作成し、接続するためには必要です
VirtualMachine.Config.AddRemoveDevice	デバイスの追加または削除	仮想デバイスの追加または削除に必要な操作です
VirtualMachine.Config.AdvancedConfig	詳細設定	詳細設定を行うために必要です
VirtualMachine.Config.Annotation	注釈の設定	仮想マシンへのアノテーション設定に必須
VirtualMachine.Config.ChangeTracking	ディスク変更追跡	変更追跡を有効化または無効化するために必要です
VirtualMachine.Config.CPUCount	CPU数の変更	仮想マシンのディスク数を変更するために必要です
VirtualMachine.Config.DiskExtend	仮想ディスクを拡張します	仮想ディスクを拡張するために必要です
VirtualMachine.Config.HostUSBDevice	ホスト USB デバイス	仮想マシンの追加、削除、編集に必要な設定
VirtualMachine.Config.Memory	メモリ	仮想マシンのメモリ容量を設定するために必要です
VirtualMachine.Config.RawDevice	Rawデバイス	仮想マシンの生デバイスに必要です
VirtualMachine.Config.RemoveDisk	ディスクの削除	データおよびオプションで削除するために必要です
VirtualMachine.Config.Rename	仮想マシンの名前を変更します	仮想マシンの名前を変更するために必要です
VirtualMachine.Config.Settings	設定	仮想マシンの設定変更が必要となります
VirtualMachine.Config.SwapPlacement	スワップファイルの配置	配置ポリシーの設定が必要です
VirtualMachine.Interact.DeviceConnection	デバイス接続	メディアの接続/切断に必要な設定です
VirtualMachine.Interact.PowerOff	電源オフ	仮想マシンの電源を切る、またはシャットダウンするために必要です
VirtualMachine.Interact.PowerOn	電源投入	仮想マシンの電源投入または再開に必要な操作です
VirtualMachine.Interact.Reset	リセット	仮想マシンのリセット (電源の再投入)
VirtualMachine.Interact.SetCDMedia	CDメディアの設定	仮想マシン用に別のメディアを設定します
VirtualMachine.Interact.SetFloppyMedia	フロッピーメディアの設定	別のメディアを設定するために必要です
VirtualMachine.Interact.Suspend	一時停止	仮想マシンの停止が必要となります
VirtualMachine.Inventory.Create	新規作成	新しい仮想マシンを作成するために必要です
VirtualMachine.Inventory.CreateFromExisting	既存から作成	仮想マシンのベースを作成するために必要です

### OpenNebula 5.10 導入ガイド、リリース 5.10.5

VirtualMachine.Inventory.Delete	削除	仮想マシンの削除に必要です
仮想マシンを移動する	移動	仮想マシンを移動するために必要です
VirtualMachine.Inventory.Register	登録	既存の仮想マシンを追加するために必要です
VirtualMachine.Inventory.Unregister	登録解除	仮想マシンの登録解除に必要な操作です
VirtualMachine.Provisioning.CloneTemplate	クローンテンプレート	テンプレートのクローン作成に必要な操作です
VirtualMachine.Provisioning.DeployTemplate	テンプレートのデプロイ	仮想マシンを展開するために必要です
VirtualMachine.Provisioning.ReadCustSpecs	カスタマイズ仕様書をお読みください	カスタマイズ仕様の読み取りが必要です
VirtualMachine.State.CreateSnapshot	スナップショットの作成	仮想マシンの新規スナップショットを作成するためには必要です
VirtualMachine.State.RemoveSnapshot	スナップショットの削除	仮想マシンからスナップショットを削除するためには必要です
VirtualMachine.State.RevertToSnapshot	スナップショットに復元	仮想マシンをスナップショットに復元するためには必要です

### 特別な許可

上記の権限は、1つを除きクラスター・レベルで設定可能です。ただし、OpenNebulaが正常に監視を行うためには、カスタマイズ仕様へのアクセス権限が必要となります。これはvCenterサーバー・レベルで適用する必要があるため、特別な権限となります。つまり、前述の権限をクラスター/データセンターおよびその継承対象に適用しようとすると、OpenNebulaは失敗し、上位レベルの権限が必要であると通知します。

推奨されるアプローチは、2つのロールを作成することです。1つはクラスター・レベルで適用可能な一般的な権限（「opennebulapermissions」）用、もう1つはこの単一の権限を扱うためのものです。これにより、すべてのOpenNebula権限を管理するロールと、次の権限のみを持つ別のロール（例えばreadcustspecと呼ぶ）を作成できます：

権限 ID	権限名	備考
VirtualMachine.Provisioning.ReadCustSpecs	カスタマイズ仕様の読み取り 仕様	カスタマイズ仕様の読み取りに必要です

適切なロールを作成した後、これらの権限を管理する方法の一つとして、2つのグループを作成する方法がございます。

- 最初のグループには**readcustspec** ロールを割り当てます。OpenNebulaユーザーをこのグループに配置し、vCenterインスタンスに対する権限をグループに付与します。
- 二つ目のグループには **opennebulapermissions** ロールを割り当てます。OpenNebula ユーザーをこのグループに配置し、対象のクラスターに対する権限をグループに付与してください。

---

**注記：**OpenNebula ユーザーがアクセスするデータストアおよびリソースに対して、適切な権限を追加することをお忘れなく。

---

### 3.4.4 ESXホスト上のVNC

VNC機能を有効にするには、ESXホスト上のVNCポートへのアクセスを許可する必要があります。デフォルトでは、これらのポートへのアクセスはファイアウォールによってフィルタリングされています。当社では、VNCルールセット（既知の予約済みポートを除くポート範囲5900-11999）を追加し、これらのポートへのアクセスを許可するインストールパッケージを提供しております。また、OpenNebulaもこの特定のVNCポート範囲を尊重するよう再構成する必要があります。このパッケージは各ESXホストにインストールする必要があり、CLIまたはWeb UI経由で実行可能です。ここでは両方の方法に必要な手順を説明いたします。

VIBインストールパッケージまたはZIPバンドルの場所：

- OpenNebula フロントエンドサーバー上の `/usr/share/one/esx-fw-vnc/` ディレクトリにあります。RHEL/CentOS 上の
  - RHEL/CentOS では **opennebula-server** パッケージの一部としてインストールされます
  - Debian および Ubuntu では **opennebula** パッケージの一部としてインストールされます。
- 公開ダウンロードサーバー上。インストールに問題が生じた場合、自己責任で非セキュアなHTTPアクセスをご利用いただけます。
  - <https://downloads.opennebula.org/packages/opennebula-5.10.0/fw-vnc-5.10.0.vib>
  - <https://downloads.opennebula.org/packages/opennebula-5.10.0/fw-vnc-5.10.0.zip>

**注記：**OpenNebulaフロントエンドからESXホストにアクセス可能であることをご確認ください。

---

ESXホストでホワイトリスト登録されたVNCポート範囲は、`/etc/one/oned.conf`にあるOpenNebula設定ファイルで指定する必要があります。`VNC_PORTS`セクションを以下の方法で変更してください：

```
VNC_PORTS = [
    START      = 5900,
    RESERVED  = "5988:5989, 6999, 8000, 8042:8045, 8080, 8100, 8182, 8200, 8300:8302,
    ↪8889, 9000, 9080, 12000:65535"
]
```

また、OpenNebulaを再起動してください：

```
$ sudo systemctl restart opennebula
```

### CLI を使用する場合

---

**注記：**プレースホルダー変数 \$ESX\_HOST (ESXホスト名)、\$ESX\_USER (アクセスユーザー名)、および \$ESX\_PSWD (アクセスユーザーのパスワード) は、お客様のインフラ構成に応じた有効なアクセスパラメータに置き換えてください。

---

### SSH 経由

ESX ホストで直接 SSH アクセスを有効にしている場合は、scp を使用して VIB インストールパッケージを ESX ホストにコピーしてください。SSH 経由で ESX ホストにログインし、コミュニティパッケージのインストールを許可してインストールを行ってください。

---

**注記：**VIBの絶対パスを指定する必要があります。

---

```
$ scp /usr/share/one/esx-fw-vnc/fw-vnc.* $ESX_HOST:/tmp/
$ ssh $ESX_HOST
$ esxcli software acceptance set --level=CommunitySupported
$ esxcli software vib install -v /tmp/fw-vnc.vib
```

これにより、任意のリモートホストに対してVNCポートが有効になります。VNCへのアクセスはOpenNebulaフロントエンドからのみ制限されるべきです。この例では、IPアドレス192.168.0.1からのアクセスのみを制限します。

```
$ esxcli network firewall ruleset set --ruleset-id VNC --allowed-all false
$ esxcli network firewall ruleset allowedip add --ruleset-id VNC --ip-address 192.168.
↪0.1/32
$ esxcli network firewall ruleset allowedip list --ruleset-id VNC
```

各 ESX ホストについて、上記の手順を繰り返します。

### VMware vSphere CLI

VMware vSphere CLIが正常に動作している場合、esxcliを使用してパッケージをリモートでインストールでき

```
$ esxcli --server $ESX_HOST --username $ESX_USER --password $ESX_PSWD system version
↪get
```

ます。まず、CLIが動作していることを確認してください：

信頼できないフィンガープリントにより接続が失敗した場合は、有効なフィンガープリントを esxcli の追加パラメータとして指定してください --thumbprint。例：

```
$ esxcli --server $ESX_HOST --username $ESX_USER --password $ESX_PSWD system version
↪取得
$ESX_HOST への接続に失敗しました。サーバーの SHA-1 フィンガープリント: 00:11:22:33:...:11:22:33 (信頼されていません)
↪信頼できません)
```

```
$ esxcli --server $ESX_HOST --username $ESX_USER --password $ESX_PSWD --thumbprint
↳ '00:11:22:33:...:11:22:33' システム バージョン取得 製品: VMware
ESXi
バージョン: 6.5.0
ビルト: リリースビルト-4887370アップデ
ート: 0
バッチ: 9
```

上記のテストで必要な接続パラメータをすべて取得しましたら、esxcliを使用してコミュニティパッケージのインストールを許可し、インストールを進めてください。

**注記：**VIBはESXホストからアクセス可能な状態である必要があります。具体的には、ESXホスト上の絶対ファイルパス、またはダウンロード可能なURLとして指定してください。

```
$ esxcli <接続オプション> software acceptance set --level=CommunitySupported
$ esxcli <接続オプション> software vib install -v 'https://downloads.opennebula.
↳org/packages/opennebula-5.10.0/fw-vnc-5.10.0.vib'
```

これにより、任意のリモートホストに対してVNCポートが有効になります。VNCへのアクセスはOpenNebulaフロントエンドからのみ制限されるべきです。この例では、IPアドレス192.168.0.1からのアクセスのみを制限します。

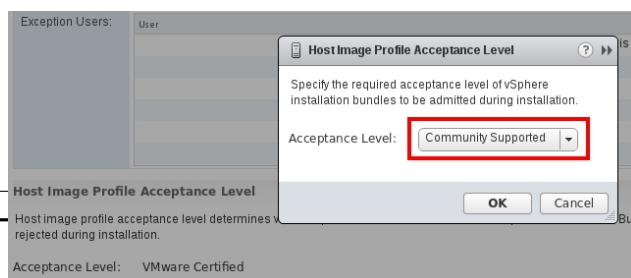
```
$ esxcli <接続オプション> network firewall ruleset set --ruleset-id VNC --allowed-
↳all false
$ esxcli <接続オプション> network firewall ruleset allowedip add --ruleset-id VNC
↳--ip-address 192.168.0.1/32
$ esxcli <接続オプション> network firewall ruleset allowedip list --ruleset-id VNC
```

各ESXホストに対して繰り返してください。

## UIを使用した方法

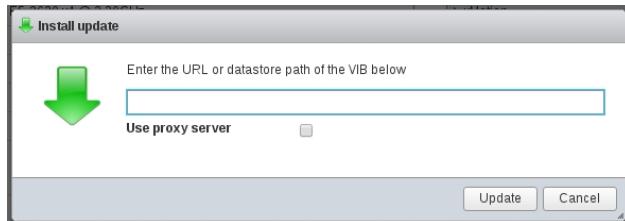
VIBパッケージは、vSphereおよびESXのWeb UIからもインストール可能です。

- カスタムVIBパッケージのインストールを許可します（vSphereクライアント内にて）
  - vSphereクライアントにログインしてください
  - ホーム->インベントリ->ホストとクラスタへ移動します
  - ESXホストを選択し、その「管理」または「構成」タブを選択します（vSphereのバージョンによって異なります）
  - システムカテゴリ内の「セキュリティプロファイル」を選択します
  - 最下部にある「ホストイメージプロファイルの受け入れレベル」で「編集」を選択してください
  - 「コミュニティサポート」に切り替え、「OK」で確定してください

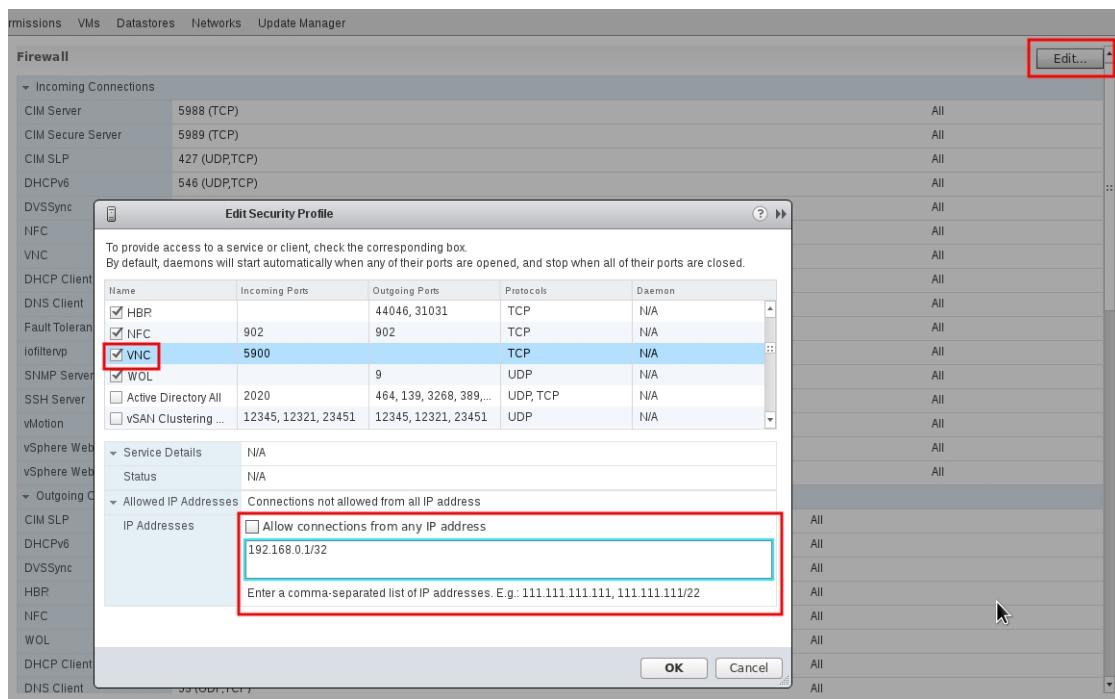


- 
- VIBパッケージをインストールしてください（ESXホストのユーザーインターフェースにて）

- ESXホストのユーザーインターフェースにログインしてください
- 右上の「ヘルプ」→「更新」を選択してください
- VIBのURLまたは絶対ローカルパスを入力し、「更新」をクリックしてください



- vSphereクライアントにて、VNCアクセスをOpenNebulaフロントエンドのみに制限してください
  - vSphereクライアントで再度ESXホストの詳細画面に戻ります
  - vSphereページを再読み込みし、最新のデータをご確認ください
  - システムカテゴリ内のセキュリティプロファイルを再度ご確認ください。ファイアウォール/着信接続において、新しいVNC項目が追加されていないかご確認ください
  - ファイアウォールの「編集」をクリックしてください
  - VNCを見つけ、必要に応じてアクセスを OpenNebula フロントエンド（例：192.168.0.1）のみに制限してください。



各ESXホストで同様の設定を繰り返してください。

### 3.5 インストール内容の確認

この章は、このオプションのセクションで終了します。ここでは、仮想マシンを起動してクラウドをテストし、すべてが正常に動くことを確認します。

---

作していることを確認することができます。

お使いのハイパーバイザに該当する特定のサブセクションのみに従ってください。

### 3.5.1 KVM/LXDベースのクラウド検証

このサブセクションの目的は、小さな仮想マシン（この場合は TTYLinux、テスト専用の非常に小さな仮想マシンです）を起動することです。

#### 要件

このセクションを完了するには、<http://marketplace.opennebula.systems> からアプライアンスをインポートするため、フロントエンドからインターネットへのネットワーク接続が必要です。

#### ステップ1. ホストが稼働中であることを確認してください

ご登録いただいたホストは、ON状態である必要があります。

ID	Name	Cluster	RVMs	Allocated CPU	Allocated MEM	Status
0	node01	0	0	0 / 800 (0%)	0KB / 16GB (0%)	ON

#### ステップ2. マーケットプレイスからアプライアンスをインポートします。

クラウドにイメージを追加する必要があります。左側のメニューから「ストレージ」→「アプリ」へ移動します（1）。アプライアンスの一覧が表示されますので、tty kvm でフィルタリング（2）し、本ガイドで使用するアプライアンスをお探し下さい。その後、該当するアプライアンスを選択（3）し、「クラウドへ」ボタン（4）をクリックしてインポートします。

Apps

ID	Name	Owner	Group	Size	State	Registration Time	Marketplace	Zone
9	ttylinux - kvm	oneadmin	oneadmin	200MB	READY	01:00:00 01/01/1970	OpenNebula Public	0

選択したアプライアンスの情報が表示されたダイアログが表示されます。「リソースを保存するデータストアを選択してください」(1)でデータストアを選択し、「ダウンロード」(2)ボタンをクリックしてください。

### Download App To OpenNebula

2

← Download

Name:

VM Template Name:

Select the Datastore to store the resource

You selected the following datastore: default

1

ID	Owner	Group	Name	Capacity	Cluster	Type	Status
1	oneadmin	oneadmin	default	158.9GB / 461.5GB (34%)	0	IMAGE	ON

次に、ストレージ → イメージタブに移動し、イメージのステータスが「準備完了」に切り替わるまで更新してください。

ID	Name	Owner	Group	Datastore	Type	Status	#VMS
0	ttylinux - kvm	oneadmin	oneadmin	default	OS	READY	0

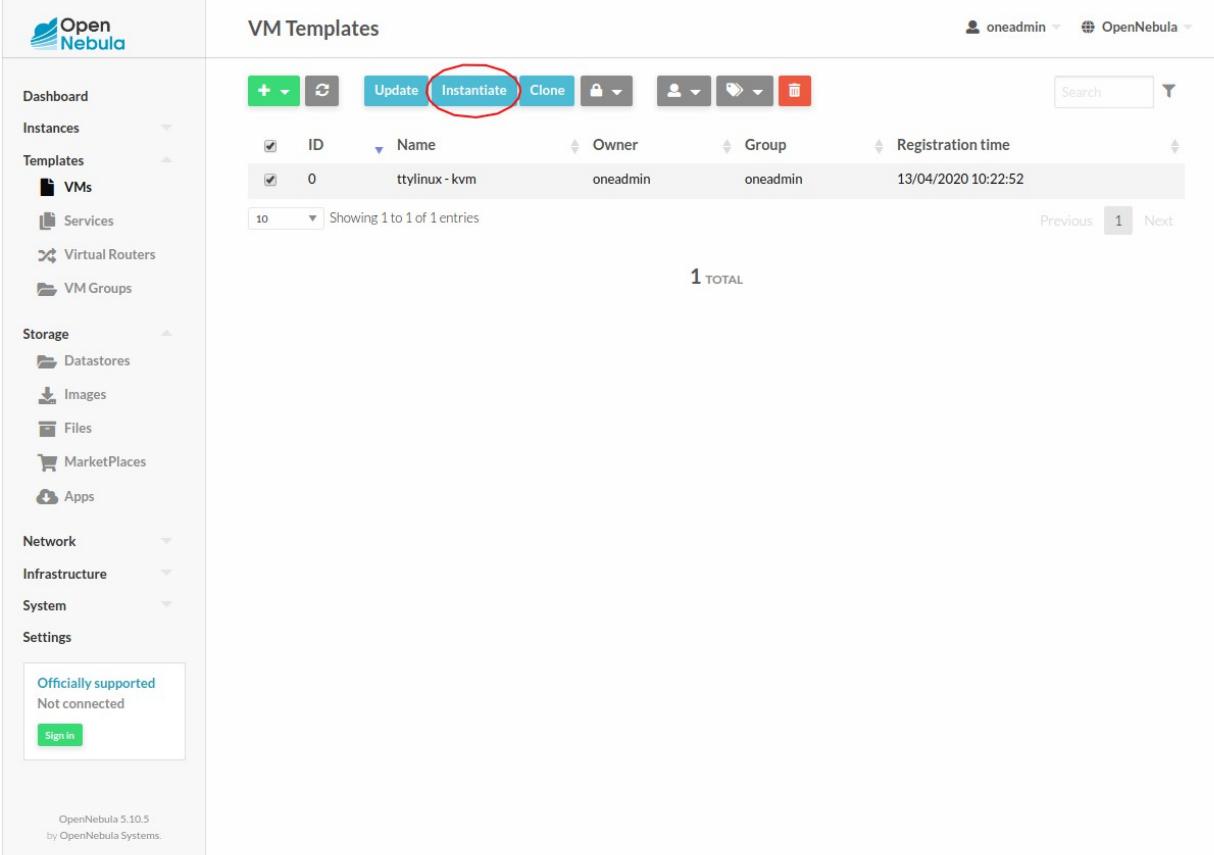
1 TOTAL 200 MB TOTAL SIZE

Officially supported  
Not connected  
[Sign in](#)

OpenNebula 5.10.5  
by OpenNebula Systems.

### ステップ3. 仮想マシンをインスタンス化します

テンプレート → VM に移動し、作成済みの `ttylinux - kvm` テンプレートを選択して「インスタンス化」をクリックしてください。



The screenshot shows the 'VM Templates' page in the OpenNebula web interface. On the left is a sidebar with navigation links for Dashboard, Instances, Templates (VMs selected), Services, Virtual Routers, VM Groups, Storage (Datastores, Images, Files, MarketPlaces, Apps), Network, Infrastructure, System, and Settings. A callout box in the Settings section indicates 'Officially supported' and 'Not connected' with a 'Sign in' button. The main area has a header with 'VM Templates' and buttons for '+', 'Update', 'Instantiate' (which is circled in red), 'Clone', 'Lock', 'Edit', and 'Delete'. Below is a table with columns: ID, Name, Owner, Group, and Registration time. One entry is listed: ID 0, Name ttylinux-kvm, Owner oneadmin, Group oneadmin, Registration time 13/04/2020 10:22:52. At the bottom, it says '1 TOTAL'.

このダイアログでは、単に「インスタンス化」ボタンをクリックしてください。

Instantiate VM Template

Instantiate 

Instantiate as persistent 

VM Name  Number of instances  Hold 

ttylinux - kvm

 Capacity 

Memory  128 MB

CPU  0.1 VCPU 

DISK 0: ttylinux - kvm  
40 MB

 Network

Add another Network Interface

#### ステップ4. VNCアクセスをテストする

インスタンス → VM に移動します。しばらくすると、VM のステータスが RUNNING に切り替わります（更新ボタンをクリックする必要があるかもしれません）。切り替わったら、VNC アイコン（下図の右側）をクリックできます。

The screenshot shows the OpenNebula 5.10 web interface. On the left is a sidebar with navigation links: Dashboard, Instances (VMs, Services, Virtual Routers), Templates (VMs, Services, Virtual Routers, VM Groups), Storage (Datastores, Images, Files, MarketPlaces, Apps), Network, Infrastructure, System, and Settings. The Settings section includes a 'Officially supported' status (Not connected) and a 'Sign in' button. At the bottom of the sidebar is the text 'OpenNebula 5.10.5 by OpenNebula Systems.'.

The main area is titled 'VMs' and displays a table of virtual machines. The table has columns: ID, Name, Owner, Group, Status, Host, and IPs. One entry is shown: ID 0, Name ttylinux-kvm-0, Owner oneadmin, Group oneadmin, Status RUNNING, Host node01, and IPs --. A red circle highlights the 'Console' icon in the toolbar above the table. Below the table, it says 'Showing 1 to 1 of 1 entries'. At the bottom right, there are buttons for 'Previous', '1', and 'Next'.

ID	Name	Owner	Group	Status	Host	IPs
0	ttylinux-kvm-0	oneadmin	oneadmin	RUNNING	node01	--

1 TOTAL 1 ACTIVE 0 OFF 0 PENDING 0 FAILED

仮想マシンが起動に失敗した場合は、該当する仮想マシンをクリックし、[ログ]タブで原因を確認してください。あるいは、ログファイル /var/log/one/<vmid>.log を参照することも可能です。

#### ステップ5. 仮想マシンへのネットワーク接続の追加

お気づきかもしれません、この仮想マシンにはまだネットワーク機能が設定されていません。これは、ネットワーク技術に大きく依存するためです。テンプレートにネットワーク機能を追加するには、以下の手順に従ってください。

1. ネットワーク技術を選択するには、「[ネットワーク](#)」の章をお読みください。ただし、[ノードのインストールセクション](#)で説明されているダミードライバーを使用することを選択し、ブリッジを適切に構成済みの場合はこの限りではありません。
2. ネットワーク -> 仮想ネットワーク ダイアログで新しいネットワークを作成し、ブリッジや物理デバイス、IP 範囲などのネットワークの詳細情報を入力してください。
3. ttylinux - kvm テンプレートを選択し、更新してください。テンプレートのネットワークセクションで、前の手順で作成したネットワークを選択します。
4. 仮想マシンをインスタンス化し、接続してください。

#### 3.5.2 vCenterベースのクラウド検証

## OpenNebula 5.10 導入ガイド、リリース 5.10.5

OpenNebulaクラウドの正しいインストールを確認するため、以下の手順に従ってください。以下の手順は、[フロントエンドインストールセクション](#)で説明されている通り、Sunstoneが起動して動作していることを前提としています。Sunstoneにアクセスするには、ブラウザを

[http://<frontend\\_address>:9869](http://<frontend_address>:9869) にアクセスしてください。

### ステップ 1. データストアのインポート

vCenter VM テンプレートは仮想ディスクを使用しており、それらの仮想ディスクはデータストアに保存されています。仮想ディスクを含む vCenter データストアを OpenNebula にインポートする必要があります。

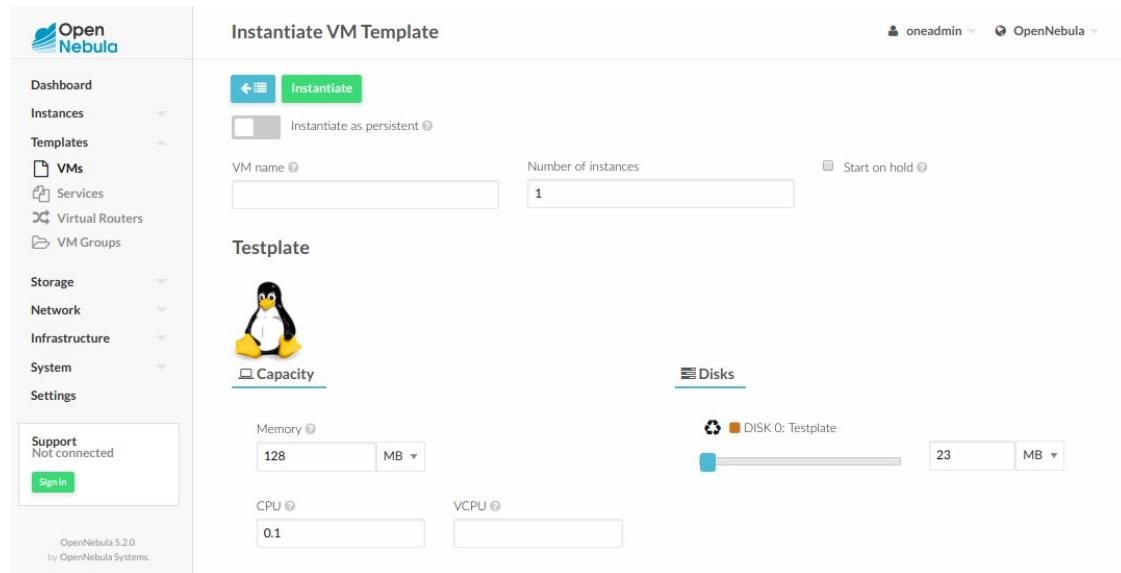
[vCenter データストアのインポートセクション](#)に記載されている手順に従って、データストアをインポートしてください。

### ステップ 2. VM テンプレートのインポート

vCenterに定義されたVMテンプレートが必要となります。その後、[vCenter テンプレートのインポートセクション](#)に記載されている手順に従い、それをインポートしてください。

### ステップ3. VMテンプレートのインスタンス化

Sunstone を使用すれば、テンプレートから簡単にインスタンスを作成し、新しい VM を生成できます。左側メニューの「テンプレート」タブ、VM セクションに進み、インポートしたテンプレートを選択し、「インスタンス化」ボタンをクリックしてください。



### ステップ 4. VM の稼働状態を確認する

スケジューラは、[vCenter ノードのインストール](#)セクションでインポートしたvCenterクラスタにVMを配置します。数分後（VMテンプレートで定義されたディスクのサイズによって異なります）、VMの状態は実行中です。Sunstoneの左側メニューにある「インスタンス」タブの「VM」セクションでプロセスをご確認いただけます。

仮想マシンが起動したら、青いVNCアイコンをクリックしてください。仮想マシンのコンソールが表示されれば、おめでとうございます！完全に機能するOpenNebulaクラウドが構築されました。

ID	Name	Owner	Group	Status	Host	IPs
1	TestVM	oneadmin	oneadmin	RUNNING	cluster	--

1 TOTAL    1 ACTIVE    0 OFF    0 PENDING    0 FAILED

次のステップとして、OpenNebulaクラウドをさらにご自身のニーズに合わせて設定することをお勧めいたします。詳細は [VMware インフラストラクチャ設定ガイド](#)

### 3.5.3 次のステップ

本章を終えれば、クラウドの利用を開始できる状態となります。あるいは、以下のコンポーネントの設定をさらに進めることも可能です：

- [認証（オプション）](#) OpenNebula を LDAP/AD と統合する場合、またはその他の認証技術でさらにセキュリティを強化する場合。
- [Sunstone](#)。この段階ではOpenNebulaのGUIは動作しアクセス可能ですが、本ガイドをお読みいただくことでSunstoneの特定の拡張設定について学ぶことができます。

クラウド環境がKVM/LXDベースの場合は、以下の手順もご参照ください：

- [Open Cloud Host の設定](#)
- [Open Cloud Storage の設定](#)
- オープンクラウドネットワーク設

定。一方、VMwareベースの場合は：

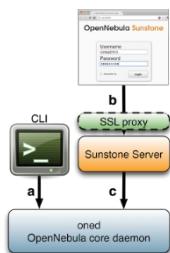
- [VMware インフラストラクチャ設定](#)の章をご覧ください。

## 認証設定

### 4.1 概要

OpenNebula はデフォルトで内部ユーザー/パスワード認証システムを搭載しております。詳細については、ユーザーとグループサブシステムガイドをご参照ください。外部認証ドライバを有効化することも可能です。

#### 4.1.1 認証



この図では、OpenNebula でカスタマイズ可能な 3 種類の認証構成をご覧いただけます。

#### a) CLI/API 認証

コマンドラインからOpenNebulaにアクセスする際には、以下の認証ドライバーからお選びいただけます：

- 組み込みのユーザー名/パスワード認証およびトークン認証
- [SSH認証](#)
- [X509認証](#)
- [LDAP認証](#)

#### b) Sunstone認証

デフォルトでは、OpenNebula で動作するように設定された認証ドライバは、Sunstone でもそのままご利用いただけます。さらに、[Sunstone のドキュメント](#)に記載されているように、Sunstone に TLS セキュリティを追加することも可能です。

### c) サーバー認証

この方法は、OpenNebulaと連携する上位ツールに認証プロセスを委譲することを目的としています。独自のサーバーを開発されている場合に、この方法が有用となるでしょう。

デフォルトでは、OpenNebulaにはSunstoneとEC2の2つのサーバーが同梱されています。ユーザーがこれらのいずれかとやり取りを行う際、サーバーはリクエストを認証した後、要求された操作をOpenNebulaデーモンに転送します。

転送されたリクエストは、デフォルトで対称鍵方式を用いて暗号化されます。以下のガイドでは、x509証明書を使用してこれらのリクエストのセキュリティを強化する方法をご説明します。これは、サーバーをフロントエンド以外のマシンで実行している場合に特に重要です。

- [クラウドサーバー認証](#)

#### 4.1.2 本章の読み方

クラウドのアーキテクチャ設計時には、ユーザー認証情報の保存場所を選択する必要があります。異なる認証方法を同時に設定し、ユーザーごとに選択することが可能です。各認証方法の主な違いは、API、CLI、および/またはSunstone（ウェブインターフェース）のみで使用できるかどうかです。

API、CLI、Sunstone で使用可能:

- 組み込みのユーザー名/パスワード認証
- LDAP

APIおよびCLIでのみ利用可能:

- SSH

サンストーンとのみご使用いただけます：

- X509

以下のセクションは独立した内容となっており、選択された認証方法の設定を説明するガイドに直接お進みいただけます。

#### 4.1.3 ハイパーバイザー互換性

セクション	互換性
組み込みのユーザー名/パスワードおよびトークン認証	このセクションは、KVM および vCenter の両方に適用されます。
<a href="#">SSH 認証</a>	このセクションは、KVM および vCenter の両方に適用されます。
<a href="#">X509 認証</a>	このセクションは、KVM および vCenter の両方に適用されます。
<a href="#">LDAP 認証</a>	このセクションは、KVM および vCenter の両方に適用されます。
<a href="#">Sunstone ドキュメント</a>	このセクションは、KVM および vCenter の両方に適用されます。

## 4.2 SSH 認証

本ガイドでは、OpenNebula CLIにおけるSSH認証の有効化と使用方法についてご説明いたします。この認証方法では、ユーザー様はご自身の秘密鍵で暗号化されたトークンを用いてOpenNebulaにログインいたします。

## 4.2.1 必要条件

追加のソフトウェアをインストールする必要はありません。

## 4.2.2 注意事項と制限事項

現在のリリースでは、この認証方法はCLIを使用したOpenNebulaとのやり取りにのみ有効です。

## 4.2.3 設定

### OpenNebulaの設定

SSH認証を使用したAuth MADはデフォルトで有効化されています。動作しない場合は、認証方法が有効な方法のリストに含まれていることをご確認ください。

```
AUTH_MAD = [
    実行可能 = "one_auth_mad",
    認証方式 = "ssh,x509,ldap,サーバー暗号化方式,サーバーx509証明書"
]
```

外部平文ユーザー名/パスワード認証ドライバーが存在し、既存のアカウントは従来通り動作を継続いたします。

## 4.2.4 使用方法

### 新規ユーザーの作成

この認証方式では、標準的なSSH RSA鍵ペアによる認証を行います。これらのファイルが存在しない場合、ユーザーは以下のコマンドで作成できます：

```
$ ssh-keygen -t rsa
```

OpenNebulaコマンドは、生成されたファイルを標準の場所 (`~/.ssh/id_rsa`) で探します。そのため、デフォルトのパスを変更しないことをお勧めいたします。また、秘密鍵をパスワードで保護することも推奨されます。

新規アカウントを申請されるユーザー様は、公開鍵を生成の上、管理者様へお送りいただく必要があります。公開鍵の抽出方法は以下の通りです：

```
$ oneuser key
PEMパスフレーズを入力してください: MIIBCAKCAQEApUO+JISjSf02rFVtDr1yar/34EoUoVETx0n+RqWNavy+5wi+gHiPp3e03AfEkXzjDYi8F
voS4a4456f1OUQ1QddfyPECn590eX8Zu4DH3gp1VUuDeeE8WJWYAzdK5hg6F+RdyP1pT26mnyunZB8Xd
b118seoIAQiOS6t1Vfa8FrftwLGmdETfttS9ukyGxw5vdTpse/fcam+r9AXBR06zjc77x+DbRFbXcgI
1XiDpvRjCFL0fdn53L0aU7kTE9VNEXRxK8sPv1Nfx+FQWpx/HtH8ICs5WREsZGmXPAO/IkrSpMVG5taS
jie9JAQOMesjFIwgTWBuH6cNXuYsQ/5wIwIBIw==
```

コンソールに表示された文字列は、管理者へ送信する必要があります。これにより、デフォルトのユーザー名/パスワード認証ユーザーと同様の方法で、新しいユーザーを作成することが可能となります。

以下のコマンドは、ユーザー名 `newuser` で新規ユーザーを作成します。なお、既存の公開鍵がテキストファイル `/tmp/pub\_key` に保存されていることを前提としています：

```
$ oneuser create newuser --ssh --read-file /tmp/pub_key
```

--read-file オプションを使用する代わりに、公開鍵を第二引数として指定することも可能です。

管理者がユーザーの秘密鍵にアクセスできる場合、以下のコマンドで新規ユーザーを作成できます：

```
$ oneuser create newuser --ssh --key /home/newuser/.ssh/id_rsa
```

#### 既存ユーザーのSSH設定を更新する

既存ユーザーの認証方法をSSHに変更するには、以下のコマンドを実行してください：

```
$ oneuser chauth <id|name> ssh
$ oneuser passwd <id|name> --ssh --read-file /tmp/pub_key
```

作成コマンドと同様に、2番目のパラメータとして公開鍵を指定するか、--key オプションを使用してユーザーの秘密鍵を利用するすることができます。

#### ユーザーログイン

ユーザー様はログイントークンを生成するため、oneuser login コマンドを実行する必要があります。生成されたトークンは \$ONE\_AUTH 環境変数に保存されます。このコマンドには OpenNebula のユーザー名と認証方法（この場合は --ssh）が必要です。

```
$ oneuser login newuser --ssh
```

デフォルトの SSH キーは ~/.ssh/id\_rsa に存在すると想定されます。それ以外の場合は --key オプションでパスを指定できます。オプションで指定できます。

生成されたトークンの有効期限はデフォルトで10時間となっております。--timeオプションを使用して変更することが可能です。

## 4.3 x509認証

本ガイドでは、OpenNebula における x509 証明書認証の有効化と使用方法についてご説明いたします。OpenNebula では、x509 証明書を以下の 2 つの異なる方法で使用することが可能です。

本ガイドで最初に説明されているオプションは、CLIで証明書を使用することを可能にします。この場合、ユーザーは自身の秘密鍵を使用してログイントークンを生成します。OpenNebulaは証明書を検証し、トークンを復号化してユーザーを認証します。

2つ目のオプションでは、SunstoneおよびOpenNebulaに含まれるパブリッククラウドサーバーで証明書を使用することが可能となります。この場合、認証処理はApacheまたは管理者が設定したTLS対応のHTTPプロキシに委譲されます。証明書が有効であると確認された場合、サーバーはサーバー証明書を用いて認証情報を暗号化し、トークンをOpenNebulaへ送信します。

### 4.3.1 要件

Sunstoneまたはいずれかのパブリッククラウドでx509証明書をご利用になる場合、証明書検証を処理するため、それらの上にTLS対応のHTTPプロキシを導入する必要があります。

### 4.3.2 考慮事項と制限事項

X509 ドライバーは、証明書 DN をユーザーパスワードと同様に扱います。証明書 DN 内のスペースはすべて削除されます。これにより、CA署名証明書のサブジェクトがスペースのみ異なるという稀な状況において、問題が発生する可能性があります。

### 4.3.3 設定

以下の表は、x509 ドライバ (/etc/one/auth/x509\_auth.conf) で使用可能なオプションをまとめたものです。

VARIABLE	値
:ca_d	信頼された CA ディレクトリへのパスです。サーバー用の信頼された CA が含まれている必要があります。各 CA 証明書は <CA_hash>.0
:check_crl	_Bcyrldefault としてください。CRL ファイルを <CA_hash>.r0 の形式で CA ディレクトリに配置した場合、OpenNebula はそれらをチェックします それをチェックします。:check_crl を定義することで CRL チェックを強制できます。つまり、CRL ファイルが見つからない場合、認証は失敗します。.r0 ファイルを移動または名前変更することで、この機能をいつでも無効にできます

oneadmin の認証方法を x509 に変更するには、以下の手順に従ってください：

**警告：**この手順が失敗した場合に元に戻せるよう、oneadmin グループに別のアカウントを用意しておくことをお勧めします。

- oneadmin のパスワードを oneadmin 証明書の DN に変更してください。

```
$ oneuser chauth 0 x509 --x509 --cert /tmp/newcert.pem
```

- 信頼された CA 証明書を 証明書ディレクトリに追加します:

```
$ openssl x509 -noout -hash -in cacert.pem 78d0bbd8
$ sudo cp cacert.pem /etc/one/auth/certificates/78d0bbd8.0
```

- x509 オプションを使用して oneadmin のログインを作成します。このトークンの有効期限はデフォルトで 1 時間に設定されています。この値は --time オプションを使用して変更できます。

```
$ oneuser login oneadmin --x509 --cert newcert.pem --key newkey.pem PEM パスフレーズを入力
してください:
export ONE_AUTH=/home/oneadmin/.one/one_x509
```

- ONE\_AUTH を x509 ログインファイルに設定します

```
$ export ONE_AUTH=/home/oneadmin/.one/one_x509
```

### 4.3.4 使用方法

#### 信頼済みCA証明書の追加と削除

信頼済みCA証明書はすべて、証明書ディレクトリにコピーし、それぞれを <CA\_hash>.0 という名前でリネームする必要があります。ハッシュは openssl コマンドで取得できます：

```
$ openssl x509 -noout -hash -in cacert.pem 78d0bbd8
$ sudo cp cacert.pem /etc/one/auth/certificates/78d0bbd8.0
```

特定の認証局（CA）の信頼を停止するには、その証明書を証明書ディレクトリから削除するだけで結構です。

この処理は OpenNebula を再起動せずに実行可能です。認証要求が行われるたびに、ドライバが証明書を検索します。

### 新規ユーザーの作成

新規アカウントを申請するユーザーは、信頼できる認証局（CA）によって署名された証明書を管理者に送信する必要があります。以下のコマンドは、ユーザーの証明書が `/tmp/newcert.pem` に保存されていることを前提に、ユーザー名 `newuser` で新規ユーザーを作成します：

```
$ oneuser create newuser --x509 --cert /tmp/newcert.pem
```

このコマンドは、パスワードに証明書のサブジェクトDNを含む新規ユーザーを作成します。したがって、管理者がサブジェクトDNを把握している場合、以下の方法でユーザーを作成できます：

```
$ oneuser create newuser --x509 "user subject DN"
```

### 既存ユーザーをx509および複数DNに更新する

既存ユーザーの認証方法をx509に変更するには、以下のコマンドを実行してください：

- ユーザー証明書を使用する場合：

```
$ oneuser chauth <id|name> x509 --x509 --cert /tmp/newcert.pem
```

- ユーザー証明書のサブジェクトDNを使用する場合：

```
$ oneuser chauth <id|name> x509 --x509 "user subject DN"
```

複数の証明書を同じOpenNebulaアカウントに割り当てるこも可能です。各証明書のDNをパスワードフィールドに「/」で区切って追加してください。

で区切ってパスワードフィールドに追加してください：

```
$ oneuser passwd <id|name> --x509 "/DC=es/O=one/CN=user|/DC=us/O=two/CN=user"
```

### ユーザーログイン

ユーザーはログイントークンを生成するため、`oneuser login` コマンドを実行する必要があります。トークンは `ONE_AUTH` 環境変数に保存されます。このコマンドには OpenNebula のユーザー名と認証方法（この場合は `--x509`）が必要です。

```
newuser@frontend $ oneuser login newuser --x509 --cert newcert.pem --key newkey.pemPEM パスフレーズを入力してください:
```

生成されたトークンの有効期限はデフォルトで10時間です。`--time`オプションで変更できます。

### 4.3.5 チューニングと拡張

x509認証方式は、AUTH\_MADで有効化されているドライバーの一つに過ぎません。全てのドライバーは/var/lib/one/remotes/authに配置されています。

OpenNebulaはデフォルトでx509認証を使用するよう設定されています。有効なドライバは`oned.conf`のAUTH\_MAD属性でカスタマイズできます。複数の認証方法を定義することも可能です:

```
AUTH_MAD = [
    実行可能 = "one_auth_mad",
    認証方式 = "ssh,x509,ldap,サーバー暗号化方式,サーバーx509証明書"
]
```

### 4.3.6 Sunstoneにおけるx509認証の有効化

`/etc/one/sunstone-server.conf` の`:auth`パラメータを更新し、x509認証を使用するように設定します:

```
:auth: x509
```

## 4.4 LDAP認証

LDAP認証アドオンにより、ユーザーはLDAPと同じ認証情報を使用できるようになり、認証を効果的に一元化できます。これを有効にすると、正しく認証されたLDAPユーザーは誰でもOpenNebulaを利用できるようになります。

### 4.4.1 前提条件

**警告:** このアドオンには、「net-ldap」gemによって提供される「net/ldap」Rubyライブラリが必要です。

このアドオンは、LDAPサーバーをインストールしたり、いかなる方法であれ設定を行ったりすることはありません。接続先のLDAPサーバーにおいて、エントリの作成、削除、変更を行うこともありません。唯一の要件は、既に稼働中のLDAPサーバーへの接続が可能であること、ldapbind操作を正常に実行できること、そしてユーザー検索を実行できるユーザーが存在することです。したがって、認証を行うユーザーのLDIFエントリにおいて、特別な属性や値は必要ありません。

### 4.4.2 設定

認証モジュールの設定ファイルは`/etc/one/auth/ldap_auth.conf`にあります。デフォルトの設定は以下の通りです：

```
server 1:
  LDAPユーザーはクエリを実行できます。設定されていない場合は匿名接続となります。Active Directory
  の場合はドメイン名を付加してください。例：
  Administrator@my.domain.com :user:
  'admin'
  #:パスワード: 'password'

  # LDAP認証方法
  :認証方式: :シンプル認証
  # LDAPサーバー
```

#### 4.4. LDAP認証

```

:host: localhost
:port: 389

# 接続および認証のタイムアウト#:timeout: 15

# TLS接続をご希望の場合は、この行のコメントを外してください#:encryption:
:simple_tls

ユーザーおよびグループを検索するベース階層
:base: 'dc=domain'

# グループを検索する代替ベース階層 (:base: の代わりに)

# ユーザーが所属する必要があるグループ。設定されていない場合、どのユーザーでも構いません #:group:
'cn=cloud,ou=groups,dc=domain'

# ユーザー名を保持するフィールド。設定されていない場合、「cn」が使用されます
:user_field: 'cn'

# Active Directory をご利用の場合は、こちらのユーザーフィールドをご利用ください #:user_field: 'sAMAccountName'

# グループメンバーシップのフィールド名。デフォルトは 'member' です #:group_field: 'member'

グループフィールド内に存在するユーザーフィールドが設定されていない場合、「dn」が使用されます。 ↴
← 使用されます
:#user_group_field: 'dn'

# グループテンプレート情報からマッピングファイルを生成します
:mapping_generate: true

# マッピングファイルが変更されずに保持される秒数（次回再生成まで）
:mapping_timeout: 300

# OpenNebulaのvarディレクトリ内のマッピングファイル名
:mapping_filename: server1.yaml

# OpenNebulaテンプレートのキーをADグループにマッピングするためのキー
:mapping_key: GROUP_DN

ADグループに属するユーザーでマッピングされていない場合に使用されるデフォルトのグループID
:mapping_default: 1

# グループには RFC2307bis を使用します
# falseの場合、LDAPサーバーの設定によっては、

# user_field および user_group_field を 'uid'、group_field を 'memberUid' に設定します
:rfc2307bis: true

# この例示サーバーは :order リストに含まれていないため呼び出されません
server 2:
:auth_method: :simple
:host: localhost
:port: 389
:base: 'dc=domain'
:#group: 'cn=cloud,ou=groups,dc=domain'

```

```
:user_field: 'cn'

サーバーの照会順序を以下に列挙します

# :order は、サーバー名および/またはネストされたリストのリストとして定義されます
# メインリスト内のサーバーは、記述順に認証が成功するまで順次照会されます。# ネストされたサーバーリスト
(可用性グループ) では、# ユーザーは最初のオンラインサーバーに対してのみ認証されます。
# グループ) では、ユーザーは最初にオンライン状態のサーバーに対してのみ認証されます。# ユーザーが可用性グル
ープ内のサーバーによって許可または拒否された場合、
# 同一グループ内の他のサーバーは照会されません。ただし、
# 認証プロセスはメインリストの次のサーバー/グループで続行されます。
#
# 例：
#
# 1) 単純なリスト #
:order:
#      - server1
#      - サーバー2
#      - サーバー3 #
# 2) 可用性グループ付きリスト # :order:

#
#      - server1
#      - ['server2', 'server3', 'server4']          # 可用性グループ
#      - server5

:順序：
- サーバー1
 #- サーバー2
```

この構造体はハッシュであり、:order 以外のキーには、クエリを実行したい LDAP サーバー 1 台の設定が含まれます。特別なキー :order には、設定済みのサーバーをクエリする順序を格納した配列が含まれます。

---

**注記：**:order の項目は、サーバー名、または可用性グループを表すサーバー名のネストされた配列です。:order の項目は、ユーザー認証が成功するか、リストの末尾に到達するまで、一つずつ処理されます。可用性グループ内では、正常に接続できる最初のサーバーのみがクエリ対象となります。:order にリストされていないサーバーはクエリされません。

---

変数	説明
:user	LDAP をクエリできるユーザーの名前です。匿名でクエリを実行できる場合は設定しないでください
:password	:user で定義されたユーザーのパスワードです。匿名アクセスが有効な場合は設定しないでください
:auth_method	:simpleのみがサポートされています
:encryption	SSL接続が必要な場合は :simple_tls に設定できます
:host	LDAPサーバーのホスト名
:port	LDAPサーバーのポート番号
:timeout	接続および認証のタイムアウト
:base	ユーザー検索を実行するベースリーフ
:group_base	:base ではなくグループ検索を実行する代替ベースリーフ
:group	設定されている場合、ユーザーはこのグループに所属している必要があります
:user_field	LDAP 内のユーザー名を保持するフィールド
:mapping_gen	自動的にマッピングファイルを生成します。手動で作成する必要がある場合は、この機能を無効にすることができます。
:mapping_tim	自動マッピングファイル生成の間隔（秒単位）
:mapping_fil	マッピングファイルの名前。サーバーごとに異なる名前を付ける必要があります
:mapping_key	マッピングファイル生成に使用されるグループテンプレートのキーです。 :mapping_key
:mapping_def	aDefault グループは、マッピングされたグループが見つからない場合に使用されます。 :
:rfc2307bis:	Active Directory を使用する場合は true、LDAP を使用する場合は false に設定してください。 ldap認証を有効化するには、上記のパラメータを設定する必要があります。また、OpenNebulaで外部認証を有効化するための設定も必要です。

LDAP認証を有効化するには、上記のパラメータを設定する必要があります。また、OpenNebulaでは外部認証を有効化する設定も必要です。/etc/one/oned.confに以下の行を追加してください

```
DEFAULT_AUTH = "ldap"
```

#### 4.4.3 ユーザー管理

LDAP認証モジュールを使用する場合、管理者はoneuserコマンドでユーザーを作成する必要はありません。これは自動的に行われます。

ユーザーは、認証情報を以下の形式で \$ONE\_AUTH ファイル（通常は \$HOME/.one/one\_auth）に保存できます：

```
<user_dn>:ldap_password
```

ここで

- <user\_dn> は LDAP サービス内のユーザーの DN（識別名）
- ldap\_password は LDAP サービスにおけるユーザーのパスワードです

あるいは、ユーザーは oneuser ログインコマンドを使用して認証トークンを生成することも可能です。これにより、LDAP パスワードを平文ファイルで保管する必要がなくなります。要求された際に LDAP パスワードを入力するだけで構築です。ログイントークンの管理および \$ONE\_AUTH ファイルに関する詳細は、『ユーザー管理ガイド』をご参照ください。

#### 特殊文字を含むDN

ユーザーDNまたはパスワードに空白が含まれる場合、LDAP ドライバーはそれらをエスケープ処理し、OpenNebulaユーザーの作成に使用できるようにします。そのため、ユーザーは\$ONE\_AUTHファイルを適切に設定する必要があります。

## **OpenNebula 5.10 導入ガイド、リリース 5.10.5**

ユーザーは、`oneuser encode <ユーザー名> [<パスワード>]` コマンドを使用して、エスケープされた \$ONE\_AUTH トークンを簡単に作成できます。例：

```
$ oneuser encode 'cn=First Name,dc=institution,dc=country' 'pass word'
cn=First%20Name,dc=institution,dc=country:pass%20word
```

このコマンドの出力は、\$ONE\_AUTHファイルに記述してください。

#### 4.4.4 Active Directory

LDAP認証ドライバーはActive Directoryへの接続が可能です。以下のものが必要となります：

- ・シンプルユーザー/パスワード認証をサポートするActive Directoryサーバー。
- ・Active Directory ユーザーツリーに対する読み取り権限を持つユーザー。

設定ファイル（/etc/one/auth/ldap\_auth.conf）において、以下の値を変更する必要があります：

- ・:user: ユーザーツリーに対する読み取り権限を持つActive Directoryユーザー名にドメインを追加した形式。例：ドメイン win.opennebula.org のユーザー Administrator の場合は Administrator@win.opennebula.org と指定します。
- ・:password: このユーザーのパスワード
- ・:host: ドメイン コントローラーのホスト名または IP アドレス
- ・:base: ユーザーを検索するためのベースDNです。 ドメイン名全体を分解し、各部分をDNコンポーネントとして使用する必要があります。例として、win.opennebula.org の場合、ベース DN は次のようにになります： DN=win, DN=opennebula, DN=org
- ・:user\_field: sAMAccountName に設定してください

#### 4.4.5 グループマッピング

新規ユーザーを特定のグループまたは複数のグループに所属させることができます。これを行うには、LDAPグループから既存の OpenNebula グループへのマッピングが生成されます。このシステムでは、:mapping\_file パラメータで指定されるマッピングファイルを使用し、そのファイルは OpenNebula の var ディレクトリ内に配置されます。マッピングファイルは、どの LDAP グループが特定のグループにマッピングされるかを定義するグループテンプレートのデータを用いて自動的に生成できます。例えば、グループテンプレートに以下の行を追加できます：

```
GROUP_DN="CN=technicians,CN=Groups,DC=example,DC=com"
```

LDAP 設定ファイルにおいて、:mapping\_key を GROUP\_DN に設定いたします。これにより、ドライバーは指定されたテンプレートパラメータ内でグループ DN を検索するようになります。 このマッピングは、mapping\_timeout で指定された秒数が経過すると失効いたします。これは、認証処理が継続的に OpenNebula に問い合わせを行うことを防ぐためです。

:mapping\_timeout で指定された秒数が経過すると失効します。これにより、認証が OpenNebula に継続的にクエリを送信する事態を防ぎます。

このファイルの自動生成を無効にし、手動でマッピングを行うことも可能です。マッピングファイルは YAML 形式で、ハッシュ構造を採用しています。キーには LDAP のグループ DN を、値には OpenNebula グループの ID を指定します。例：

```
CN=technicians,CN=Groups,DC=example,DC=com: '100'
CN=Domain Admins,CN=Users,DC=example,DC=com: '101'
```

複数のサーバーを設定する場合、衝突を避けるため、各サーバーごとに異なる :mapping\_key および :mapping\_file の値を設定する必要があります。例：

```
internal:
  :mapping_file: internal.yaml
  :mapping_key: INTERNAL_GROUP_DN
```

外部:

```
:mapping_file: external.yaml
:mapping_key: EXTERNAL_GROUP_DN
```

また、OpenNebulaのグループテンプレートでは、サーバーごとに2つのマッピングを定義できます：

```
INTERNAL_GROUP_DN="CN=technicians,CN=Groups,DC=internal,DC=com"EXTERNAL_GROUP_DN=
CN=staff,DC=other-company,DC=com"
```

---

**注記 :**マップが更新された場合（例：LDAPデータベースの変更）、ユーザー認証時にユーザーグループが更新されます。また、変更が反映されるには、ユーザーが使用しているログイントークンの有効期限が切れる必要がある点にもご注意ください。トークンの最大有効期間は、各ドライバごとにoned.confで設定可能です。OpenNebulaコアによるユーザーグループ更新（およびOpenNebulaからのグループ割り当て制御）を無効化したい場合は、ldap AUTH\_MAD\_CONF 設定属性のDRIVER\_MANAGED\_GROUPSを更新してください。

---

#### 4.4.6 SunstoneにおけるLDAP認証の有効化

/etc/one/sunstone-server.conf の :auth パラメータを更新し、opennebulaを使用するように設定してください：

```
:auth: opennebula
```

この方法を使用すると、ログイン画面で提供された認証情報は OpenNebula コアに送信され、指定されたドライバーを使用してそのユーザーの認証が OpenNebula 認証システムに委譲されます。したがって、この方法を通じて任意の OpenNebula 認証ドライバー（例：LDAP）を使用してユーザーを認証することが可能です。

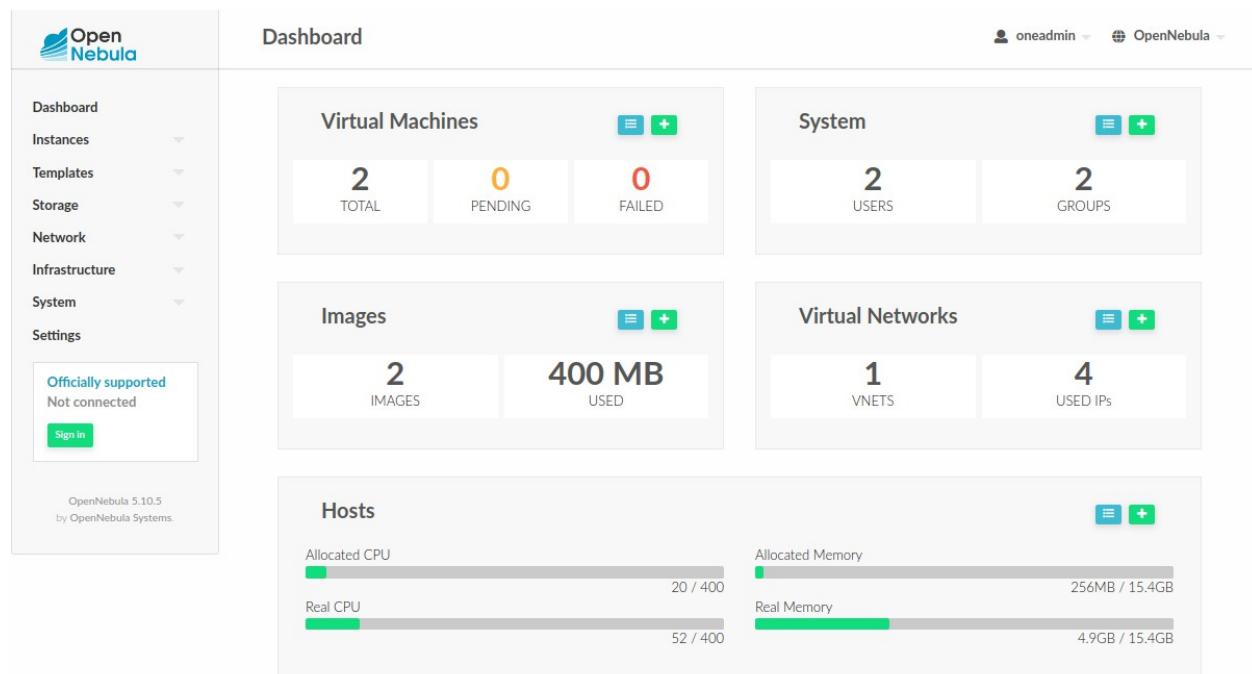
DN の特殊文字セクションで説明されているように認証情報を自動的にエンコードするには、sunstone 設定に以下のパラメータも追加してください：

```
:encode_user_password: true
```

## SUNSTONE の設定

### 5.1 概要

OpenNebula Sunstoneは、エンドユーザーと管理者の双方を対象としたグラフィカルユーザーインターフェース（GUI）であり、プライベートクラウドおよびハイブリッドクラウドインフラストラクチャにおける一般的な管理操作を簡素化します。OpenNebula Sunstoneを使用すると、すべてのOpenNebulaリソースを容易に管理し、それらに対して一般的な操作を実行することが可能です。



#### 5.1.1 本章の読み方

「[Sunstone のインストールと設定](#)」セクションでは、Sunstoneの設定およびカスタマイズオプションについて説明しています。

Sunstoneの稼働後、「[Sunstone ビュー](#)」セクションでは、各ユーザーロールごとに異なるSunstoneの動作を定義できます。Sunstoneの導入をカスタマイズおよび拡張する方法の詳細については、以下のリンクをご参照ください：

- [セキュリティと認証方法](#): x509認証とSSLによるセキュリティ強化
- [クラウドサーバー認証](#): SunstoneとOpenNebula間のセキュリティに関する詳細なリファレンス

- **高度なデプロイメント**: スケーラビリティの向上とサーバーの分離

## 5.1.2 ハイパーバイザー互換性

Sunstone は、すべてのハイパーバイザーでご利用いただけます。 vCenter をご利用の場合、クラウド管理者は admin\_vcenter、groupadmin\_vcenter、cloud\_vcenter の Sunstone ビューを有効にする必要があります。

## 5.2 Sunstone のインストールと設定

### 5.2.1 要件

OpenNebula Sunstone をご利用いただくには、適切に設定され稼働中の OpenNebula サイトが必要です。まずプライベートクラウドを構築するため、OpenNebula インストールおよび設定ガイドを必ずご確認ください。また、本セクションでは OpenNebula の設定および使用方法について既にご理解いただいていることを前提としております。

OpenNebula Sunstone は、OpenNebula のインストール時に導入されました。インストールガイドに従われた場合、既に全ての Ruby gem 要件が満たされています。そうでない場合は、root 権限で以下の install\_gem スクリプトを実行してください：

```
# /usr/share/one/install_gems sunstone
```

Sunstone Operation Center では、仮想マシンへの VNC/SPICE セッションを開始することができます。これは、クライアント側で VNC/SPICE WebSocket ベースのクライアント (noVNC) を使用し、サーバー側で接続を変換・リダイレクトする VNC プロキシによって実現されます。

**ご注意:** SPICE Web クライアントはプロトタイプであり、機能が限定されています。このコンポーネントの詳細については、以下のリンクをご参照ください。

**警告:** sunstone のログディレクトリに空き容量があることを必ずご確認ください。空き容量がない場合、sunstone はエラーを発生させずに停止します。デフォルトのログディレクトリは /var/log/one です。

要件：

- WebSockets 対応ブラウザ（任意）：Firefox および Chrome は WebSockets をサポートしております。Firefox の一部のバージョンでは手動での有効化が必要となります。WebSockets が有効化されていない場合、Flash エミュレーションが使用されます。
- VNC のパフォーマンス向上のため、python-numpy パッケージのインストールをお勧めいたします。

### 5.2.2 設定

#### sunstone-server.conf

Sunstone の設定ファイルは /etc/one/sunstone-server.conf にあります。YAML 構文を用いて以下のオプションを定義します：

利用可能なオプションは以下の通りです：

オプション	説明
:tmpdir	アップロードされた画像は、OpenNebula にコピーされる前に一時的にこのフォルダに保存されます
:one_xmlrpc	OpenNebula デーモンのホストとポート
:one_xmlrpc_timeout	sunstoneからのXMLRPC呼び出しのタイムアウト（秒単位）を設定します。シェル環境変数をご参照ください
:host	サーバーが待機するIPアドレスです。デフォルトでは0.0.0.0となります。

オプション	説明
:port	サーバーが待機するポート番号です。デフォルトは9869です。
:sessions	ユーザーセッションの保持方法。メモリまたはmemcacheのいずれかを選択できます。複数のプロセスを生成するサーバーの場合
:memcache_host	memcachedサーバーが存在するホスト
:memcache_host	memcachedサーバーのポート番号
:memcache_namespace	セッションを保存するmemcacheのネームスペースです。複数のサービスでmemcachedサーバーを使用する場合に便利です
:debug_level	ログのデバッグレベル：0 = エラー、1 = 警告、2 = 情報、3 = デバッグ
:env	Sunstoneの実行環境。dev環境では、圧縮されたJavaScriptファイルではなく、すべてのファイルが取得されます
:max_upload_file_size	アップロード可能な画像の最大サイズ（バイト単位）。無制限の場合はコメントアウトしたままにしてください
:auth	着信リクエストの認証ドライバー。可能な値はsunstone、opennebula、remoteです。
:core_auth	OpenNebulaコアと通信するための認証ドライバーです。設定可能な値はx509またはcipherです。
:encode_user_password	LDAPなどの外部認証ドライバ向けです。Oに送信される認証情報に対してURLエンコードを実行します。
:lang	Sunstoneインターフェースのデフォルト言語です。ユーザーが言語を設定していない場合に使用されるデフォルト言語となります。
:vnc_proxy_port	VNCプロキシのベースポートです。Sunstoneサーバーが稼働している間、プロキシはこのポートで動作します。デフォルトは29876です。
vnc_proxy_support_wss	はい、いいえ、のみ。有効にすると、プロキシはセキュアなWebSocketsを使用するための証明書と鍵で設定されます。
:vnc_proxy_cert	WSS接続用の証明書ファイルのフルパスです。
:vnc_proxy_key	鍵ファイルのフルパス。鍵が証明書に含まれている場合は不要です。
:vnc_proxy_ipv6	novncでIPv6を有効にします。（trueまたはfalse）
:vnc_client_port	VNC JSクライアントが接続するポート番号です。設定されていない場合、:vnc_proxy_portのポートセクションが使用されます。
:vnc_request_password	外部ウィンドウのVNCパスワードを要求します。デフォルトでは要求されません（trueまたはfalse）
:table_order	デフォルトのテーブル順序。リソースはID順に昇順または降順で並べ替えられます。
:marketplace_username	マーケットプレイスに接続するためのユーザー名認証情報。
:marketplace_password	マーケットプレイスに接続するためのパスワードです。
:marketplace_url	マーケットプレイスに接続するためのエンドポイントです。コメントアウトされている場合、503 サービス利用不可エラーが発生します。
:oneflow_server	OneFlowサーバーに接続するためのエンドポイントです。
:routes	カスタムルートを含むファイルの一覧です。これらのファイルは読み込まれる対象となります。詳細についてはサーバーブラグインをご確認ください。
:mode	デフォルトのビューディレクトリです。
:keep_me_logged	Sunstoneログイン時に「ログイン状態を維持する」オプションを表示するにはtrueを設定してください。
:get_extended_vm_info	Trueに設定すると、拡張 VM プールに対してonedを要求し、テーブルにIPを表示します。
:allow_vnc_federation	フェデレーションにVNCアイコンを表示するにはTrueを設定してください
:http_proxy	HTTP トラフィック用のプロキシサーバー。
:no_proxy	プロキシを使用すべきでないIPアドレスまたはドメイン名のパターン

**注記：**SunstoneをIPv6専用システムおよびシンクライアントHTTPサーバーで使用する場合、:hostフィールドには完全なIPv6アドレスをご入力ください。ローカルホストアドレス（::1）または未指定アドレス（::）を設定する必要がある場合は、以下の形式をご利用ください：

例：:host: 0::1, :host: 0::0

**注記：**SunstoneをIPv6専用システムおよびシンクライアントHTTPサーバーと併用する場合、:hostフィールドには完全なIPv6アドレスをご使用ください。ローカルホストアドレス（::1）または未指定アドレス（::）を設定する必要がある場合は、以下の形式をご利用ください：

---

例: :host: 0::1, :host: 0::0

---

Sunstoneの動作はユーザーテンプレート (SUNSTONE=[]) ベクトル値内、例：SUNSTONE=[TABLE\_ORDER="asc"]）からも設定可能です：

オプション	説明
DISPLAY_NAME	Sunstone に表示されるユーザー名
TABLE_ORDER	昇順 (ascending) または降順 (descending)
デフォルトビュー	名称 の デフォルトビュー デフォルト ビュー (以下 表示される /etc7on/ /etc7on/ sunstone-views に掲載されているものとして)
TABLE_DEFAULT_PAGE_LENGTH	Sunstoneデーターテーブルのページデフォルト長
LANG	サンストーン言語 (デフォルトは en_US)
DEFAULT_ZONE_ENDPOINT	Sunstoneログイン時のデフォルトゾーン。デフォルトはローカルゾーンとなります

### Sunstone の起動

Sunstone を起動するには、oneadmin として以下のコマンドを実行してください

```
# service opennebula-sunstone start
```

Sunstone サーバーのログファイルは /var/log/one/sunstone.log に保存されます。エラーは /var/log/one/sunstone.error に記録されます。

### 5.2.3 商用サポートの統合

本番環境においては、専門的で効率的なサポートへのアクセスが不可欠であることを認識しております。このため、Sunstoneに統合タブを導入し、OpenNebula Systems（OpenNebulaの開発元企業、旧称C12G）の専門サポートをご利用いただけるようにいたしました。これにより、サポートチケットの管理をSunstoneを通じて行えるため、業務の妨げを避け、生産性の向上につながります。

The screenshot shows the Sunstone interface with the 'Commercial Support Requests' tab selected. On the left, there's a sidebar with navigation links like Dashboard, Instances, Templates, Storage, Network, Infrastructure, System, Settings, and a 'Officially supported' section which is currently 'Not connected'. A 'Sign in' button is also present. The main content area has a heading 'The Support Subscription provides expert integration and production support on supported platforms and include:' followed by a bulleted list of services. Below this, a link 'For more info on support subscriptions, click here' is shown. To the right, there's a login form for 'Commercial Support' with fields for Email (sysadmin@opennebula.io) and Password, and buttons for 'Sign in' and 'Get an account'. There's also an 'or' link for creating an account.

#### Additional Help Resources



Documentation



Community

This tab can be disabled in the sunstone views configuration files

このタブは、各 ビュー の yaml ファイル で無効化することができます。

```
enabled_tabs: [ ... ]
#- support-tab
```

## 5.2.4 トラブルシューティング

### OneFlowサーバーに接続できません

サービスインスタンスおよびテンプレートのタブに、以下のメッセージが表示される場合があります：

OneFlow サーバーに接続できません

The screenshot shows the OpenNebula Sunstone web interface. On the left is a sidebar with navigation links: Dashboard, Instances (VMs, Services), Templates, Storage, Network, Infrastructure, System, and Settings. Under Settings, there's a box stating 'Officially supported' and 'Not connected' with a 'Sign in' button. The main content area is titled 'Services' and contains tabs for '+', 'Recover', and user/group actions. Below these are columns for ID, Owner, Group, Name, and State. A large cloud icon with an info sign is centered, and the text 'There is no data available' is displayed. At the bottom, there's a pagination section showing 'Showing 0 to 0 of 0 entries' and 'Previous'/'Next' buttons. A prominent red error message box at the bottom right contains the text 'Cannot connect to OneFlow server.'

このセクションに従ってOneFlowコンポーネントを起動するか、*Sunstone*ビューのyamlファイル内で「サービス」および「サービステンプレート」メニュー項目を無効にする必要があります。

### VNCのトラブルシューティング

VNC アイコンをクリックすると、セッション開始のプロセスが開始されます：

- リクエストが行われ、VNCセッションが可能であれば、SunstoneサーバーはVMホストを許可されたVNCセッション対象のリストに追加し、それに紐づくランダムなトークンを生成します。
- サーバーはセッショントークンを返答し、その後 noVNC ダイアログが表示されます。
- このダイアログに組み込まれたVNCコンソールは、WebSockets（デフォルト）を使用するか、Flashでエミュレートしてプロキシへの接続を試みます。正しいトークンを提供した接続のみが成功します。トークンには有効期限があり、再利用できません。

noVNCがマシンに正しく接続できない原因は複数考えられます。以下に一般的な問題点のチェックリストを記載します：

- noVNCのWebSocketsプロキシ機能を動作させるには、Python 2.5以上が必要です。また、python2<バージョン>-numpyな

---

どの追加モジュールが必要になる場合もございます。

- 有用な情報は `/var/log/one/novnc.log` から取得できます。
- ドキュメントに記載されている通り、VNCを有効にするにはVMテンプレートにGRAPHICSセクションが必要です。IP属性が正しく設定されていることを確認してください（`0.0.0.0`に設定すると外部からの接続を許可します）。設定が不適切な場合、外部からの接続は許可されません。
- お使いのブラウザはWebSocketをサポートし、有効化されている必要があります。現在のChromeおよびFirefoxではこれがデフォルト設定ですが、Firefoxの旧バージョン（例：3.5）では手動での有効化が必要でした。有効化されていない場合、Flashエミュレーションが使用されます。
- ファイアウォールが接続をブロックしていないことをご確認ください。プロキシは、VNCプロキシポートからVMのテンプレートに記載されているVNCポートへWebSocketデータをリダイレクトします。プロキシポートの値は`sunstone-server.conf`で定義されています。
- 通常のVNCクライアントツール（`vncviewer`など）を使用して、SunstoneフロントエンドからVMに直接接続できることを確認してください。
- セキュアWebSocketをご利用の際は、Sunstoneの設定ファイルにおいて、証明書および鍵（証明書に含まれていない場合）が正しく設定されていることをご確認ください。wss接続が正常に機能するためには、証明書が有効かつ信頼されている必要がありますのでご注意ください。ブラウザで受け入れられない証明書を使用している場合、`https://sunstone.server.address:vnc_proxy_port` にアクセスして手動でブラウザの信頼リストに追加できます。ブラウザは証明書が安全でないことを警告し、手動で信頼するよう促します。
- 接続速度が非常に遅い場合は、トークンの有効期限切れが原因である可能性があります。下記の手動プロキシ起動手順で確認してください。
- まだ動作しない場合は、Sunstoneを起動し、websockifyプロキシを終了させてください。その後、`/var/log/one/novnc.log` の先頭に記録されているコマンドを使用して、コンソールウィンドウでプロキシを手動で再起動してください。PythonプロセスのPIDを含むロックファイルを`/var/lock/one/.novnc.lock` に生成する必要があります。そのまま実行した状態で、Sunstone上の同じVMのVNCアイコンを再度クリックしてください。コンソールにプロキシからの出力が表示され、接続が機能しない原因が判明するはずです。
- 上記の提案事項をすべて試行された後、サポートフォーラムへお問い合わせください。その際は、Sunstoneの完全なログ、表示されたエラー、およびインフラストラクチャに関する関連情報（ファイアウォールなどの有無）を必ずご提供ください。
  - 「SecurityError: この操作は安全ではありません。」というメッセージは、通常、同一生成元ポリシー（Same-Origin-Policy）の問題に関連しています。SunstoneをTLSで保護している状態で、VNC用の非保護WebSocketに接続しようとすると、Firefoxはこれをブロックします。Firefoxでは、このエラーを回避するには両方の接続を保護する必要があります。また、サーバーに自己署名証明書を使用しないでください。これを使用すると再度エラーが発生します（独自の小さなCAを設定することは可能ですが、自己署名サーバー証明書は使用しないでください）。別の方法として、Firefoxの設定（`about:config`）で`network.websocket.allowInsecureFromHTTPS`を`true`に設定する方法があります。

## RDPのトラブルシューティング

仮想マシン内のネットワークインターフェイスカード（NIC）に1つのRDP接続リンクを追加するには、以下の2つの方法がございます。

- テンプレートの「ネットワーク」タブでオプションを有効にする方法：

The screenshot shows the 'Update VM Template' interface for an Ubuntu 19.04 template. The 'Network' tab is active, displaying three network interfaces (NIC 0, NIC 1, NIC 2). Under 'Interface type', there are options for 'Alias' and 'Automatic selection'. Under 'RDP connection', there is an 'Activate' button.

- または、仮想マシン内で以下を追加して定義することも可能です：

```
NIC=[  
    ...  
    RDP = "YES"  
]
```

仮想マシンがインスタンス化されると、ユーザーはRDPファイルの設定をダウンロードできるようになります。

The screenshot shows the VM details for an Ubuntu 19.04-5 instance. The 'VNC RDP' tab is selected, displaying the RDP connection settings. The IP address listed is 192.168.122.13.

**重要：**RDP接続は、単一のNICでのみ有効化されます。いずれの場合も、ファイルRDPには、このプロパティが有効化された最初のNICのIPアドレスのみが含まれます。

---

**注記：**仮想マシンテンプレートのコンテキスト設定セクションにパスワードとユーザー名が設定されている場合、これらはRDPファイルに反映されます。詳細については、仮想マシン定義ファイルのリファレンスセクションをご参照ください。

---

## 5.2.5 チューニングと拡張

### 国際化と言語

Sunstoneは複数の言語に対応しております。新しい言語の追加、修正、翻訳の完了をご希望の場合は、以下のページをご覧ください：

- Transifexプロジェクトページ

Transifexによる翻訳は簡単かつ迅速です。すべての翻訳はTransifex経由で提出してください。

ユーザーはいつでも翻訳を更新または貢献できます。通常、ベータ版リリース後の各リリース前には、フォーラムで翻訳の募集が行われます。その後、Transifexでソース文字列が更新され、すべての翻訳が最新のOpenNebulaバージョンに更新されます。一定の完成度を満たした翻訳は、最終的なOpenNebulaリリースに追加されます。

### VMロゴのカスタマイズ

VMテンプレートには、ゲストOSを識別するためのイメージロゴが用意されております。利用可能なロゴの一覧を変更したり、新しいロゴを追加したりするには、`/etc/one/sunstone-logos.yaml` を編集してください。

```
- { 'name': "Arch Linux", 'path': "images/logos/arch.png"}
- { 'name': "CentOS", 'path': "images/logos/centos.png"}
- { 'name': "Debian", 'path': "images/logos/debian.png"}
- { 'name': "Fedora", 'path': "images/logos/fedora.png"}
- { 'name': "Linux", 'path': "images/logos/linux.png"}
- { 'name': "Redhat", 'path': "images/logos/redhat.png"}
- { 'name': "Ubuntu", 'path': "images/logos/ubuntu.png"}
- { 'name': "Windows XP/2003", 'path': "images/logos/windowsxp.png"}
- { 'name': "Windows 8", 'path': "images/logos/windows8.png"}
```

The screenshot shows the 'Create VM Template' interface. On the left, there's a sidebar with various navigation links. The 'Templates' section has 'VMs' selected. The main form is titled 'Create VM Template' and includes tabs for General, Storage, Network, OS & CPU, Input/Output, Actions, Context, Scheduling, and Hybrid. Under General, there are fields for 'Name' (empty), 'Hypervisor' (radio buttons for KVM, vCenter, LXD, with KVM selected), 'Description' (empty), 'Logo' (set to 'Debian'), 'Memory' (any value), 'Cost' (any value), 'CPU' (any value), 'VCPU' (any value), and 'Memory modification' (any value). To the right of the form is a 'debain' logo.

## サンストーンポータルのブランディング

ログイン画面およびメイン画面にロゴを追加するには、以下の通り `logo:` 属性を更新するだけで簡単に行えます：

- ログイン画面は `/etc/one/sunstone-views.yaml` で定義されています。
- メイン UI 画面のロゴは、各ビューの `view.yaml` ファイルで定義されています。

### `sunstone-views.yaml`

OpenNebula Sunstoneは、異なるユーザーロールに合わせて適応させることができます。例えば、ユーザーがアクセス権を持つリソースのみを表示するように設定できます。その動作は、[ビューを通じてカスタマイズおよび拡張することができます](#)。

各グループが利用可能なビューを選択する推奨方法は、Sunstone からグループ設定を更新することです。詳細は [「Sunstone ビュー」セクションに記載されています](#)。また、`/etc/one/sunstone-views.yaml` ファイルを使用することで、ユーザーまたはグループごとにビューを設定する代替方法も定義されています。

Sunstoneは、以下の手順で各ユーザーが利用可能なビューを算出します：

- ユーザーが所属するすべてのグループから、各グループ内で定義されたビューを組み合わせてユーザーに提示します
- ユーザーの所属グループから利用可能なビューが存在しない場合、デフォルト値は `/etc/one/sunstone-views.yaml` から取得されます。ここでは、以下の対象に対してビューを定義できます：
  - 各ユーザー (`users: セクション`)：各ユーザーと、そのユーザーが利用可能なビューの一覧を記載してください。
  - 各グループ (`groups: セクション`)：グループのビューセットを一覧表示します。
  - デフォルトのビュー：ユーザーが「`users:`」セクションに記載されておらず、かつその所属グループが「`groups:`」セクションに記載されていない場合、デフォルトのビューが適用されます。
  - グループ管理者のデフォルトビュー：グループ管理者ユーザーが「`users:`」セクションに、またそのグループが「



デフォルトでは、oneadmin グループのユーザーはすべてのビューにアクセスでき、users グループのユーザーはクラウドビューをご利用いただけます。

ビューをご利用いただけます。

以下の /etc/one/sunstone-views.yaml の例では、helen に対してユーザー (user.yaml) ビューとクラウド (cloud.yaml) ビューを有効にし、cloud-users グループに対してクラウド (cloud.yaml) ビューを有効にします。特定のユーザーに対して複数のビューが利用可能な場合、最初のビューがデフォルトとなります。

```
---
logo: images/opennebula-sunstone-v4.0.png
users:
  helen:
    - cloud
    - ユーザー
グループ:
  クラウドユーザー:
    - クラウド
デフォルト:
  - ユーザーデフォルトグ
ループ管理者:
  - グループ管理者
  - クラウド
```

### 各ビューごとに異なるエンドポイント

OpenNebula *Sunstone* のビューは、ユーザーの種類ごとに異なるエンドポイントをデプロイするように適応させることができます。

例えば、管理者用とクラウドユーザー用に別々のエンドポイントを設定したい場合などが該当します。*新しいSunstone* サーバーをデプロイし、各Sunstoneインスタンスに対してデフォルトのビューを設定するだけで実現できます：

```
# 管理者用 Sunstone
cat /etc/one/sunstone-server.conf
...
:host: admin.sunstone.com
...

/etc/one/sunstone-views.yaml の内容を表示します
...
users:
groups:
default:
- admin
```

```
# ユーザー sunstone
/etc/one/sunstone-server.conf を表示してください
...
:host: user.sunstone.com
...

cat /etc/one/sunstone-views.yaml
...
users:
groups:
default:
- ユーザー
```

## 5.3 Sunstone ビュー

OpenNebula Sunstone Viewsは、2つの異なるレイアウトに分類できます。一方、従来のSunstoneレイアウトではクラウドの完全なビューを提供し、管理者や上級ユーザーがクラウドの物理リソースや仮想リソースを完全に制御できるようになります。他方、クラウドレイアウトではクラウドの簡略化されたバージョンを提供し、エンドユーザーは物理リソースの管理を気にすることなく、クラウドの仮想リソースを管理できるようになります。

OpenNebula Sunstone Viewsをご利用いただくことで、OpenNebulaクラウドのエンドユーザー向けに簡素化されたユーザーインターフェースを提供することが可能となります。 OpenNebula Sunstone Viewsは完全にカスタマイズ可能ですので、特定の情報のタブや操作ボタンを簡単に有効化または無効化できます。 [異なるユーザーグループ向けに複数のビューを定義できます](#)。各ビューはUIコンポーネントのセットを定義するため、各ユーザーは自身の役割に関連するクラウドの部分のみにアクセスし、閲覧できます。デフォルトのビュー：

- 管理者 [ビュー](#)
- グループ管理者 [ビュー](#)。
- クラウド [ビュー](#)。
- ユーザー [ビュー](#)。

これらのビューは、OpenNebula がデフォルトで提案する 3 つのディレクトリ (`kvm`、`vcenter`、`mixed`) に格納されています:

- KVM/ハイパーバイザ用: `/etc/one/sunstone-views/kvm`
- `/etc/one/sunstone-views/vcenter` は vCenter ハイパーバイザ用です。
- `/etc/one/sunstone-views/mixed`: 両方のハイパーバイザを同一ビューで管理する場合。

ビューを選択するには、`sunstone-server.conf` 内の設定パラメータ `mode` を変更するだけで結構です。

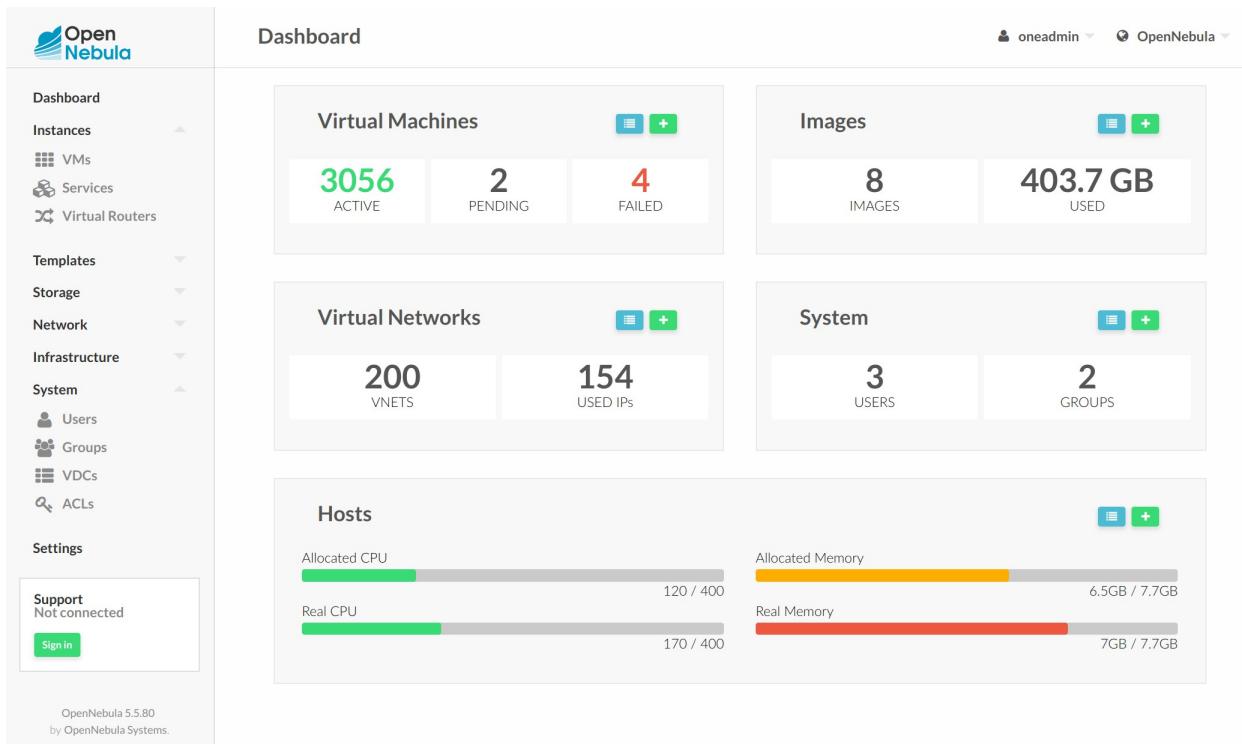
```
# デフォルトのビューディレクトリ
:mode: 'mixed'
```

混合ディレクトリには、上記で説明したビューに加え、KVMとvCenterの全機能が統合されています。

### 5.3.1 デフォルトビュー

#### 管理ビュー

このビューではクラウドを完全に制御できます。詳細は `/etc/one/sunstone-views/* / admin.yaml` ファイルで設定可能です。



## グループ管理ビュー

管理者ビューを基にしています。グループに属するすべてのリソースを管理できますが、そのグループ外のリソースにはアクセスできません。つまり、グループの物理リソースおよび仮想リソースに制限されます。このビューでは、グループ内に新規ユーザーを作成したり、ユーザークオータを設定・追跡したりする機能を備えています。このシナリオの設定方法の詳細については、こちらのセクションをご覧ください。

## クラウドビュー

これは簡略化されたビューであり、主に事前定義されたテンプレートから簡単に新しい仮想マシンをプロビジョニングできるポータルを必要とするエンドユーザー向けに設計されています。このビューに関する詳細情報は、`/etc/one/sunstone-views/*/cloud.yaml` ファイルをご確認ください。

このシナリオでは、クラウド管理者は一連のテンプレートとイメージを準備し、クラウドユーザーが利用できるようにする必要があります。これらのテンプレートはインスタンス化可能な状態で用意されている必要があります。ユーザーは使用前に、任意で仮想マシンの容量をカスタマイズしたり、新しいネットワークインターフェースを追加したり、テンプレートで要求される値を提供したりできます。これにより、ユーザーはネットワークやストレージなどのインフラストラクチャの詳細を知る必要がありません。このシナリオの設定方法の詳細については、このセクションをご覧ください。

**Virtual Machines**

Owner	RUNNING	DEPLOYING	OFF	ERROR
11	103	17	46	0

4  
Groups

**Actions:**

## ユーザービュー

管理者ビューを基にした、上級ユーザー向けのビューです。クラウドビューで利用可能な限定された操作セットよりも多くのアクションへのアクセスを必要とするユーザーを対象としています。ユーザーはクラウドのホストやクラスターを管理・取得することができません。新しいイメージや仮想マシンを作成する際に利用するため、データストアや仮想ネットワークを表示することは可能ですが、新規作成はできません。詳細設定は、`/etc/one/sunstone-views/*/user.yaml` ファイルで構成できます。

**Dashboard**

- Dashboard
- Instances
- Templates
  - VMs
  - Services
  - Virtual Routers
  - VM Groups
- Storage
- Network
- Settings

OpenNebula 5.5.80  
by OpenNebula Systems.

**Virtual Networks**

15	6
VNETS	USED IPs

**Images**

1	1 GB
IMAGES	USED

**Virtual Machines**

10	2	0
ACTIVE	PENDING	FAILED

### 5.3.2 ビューへのアクセス設定

デフォルトでは、admin ビューは `oneadmin` グループのみが利用可能です。新規ユーザーは `users` グループに追加され、デフォルトのクラウドビューを使用します。特定のグループに割り当てるビューは、グループ作成フォームで定義するか、既存のグループを更新することで設定でき、異なる OpenNebula モデルを実装できます。異なる OpenNebula モデルの詳細については、「[Understanding OpenNebula](#)」ドキュメントをご参照ください。

### 5.3.3 ご利用方法

サンストーンをご利用のお客様は、画面右上のドロップダウンメニューから現在の表示を変更できます：

また、設定タブからいくつかのオプションを構成することも可能です：

- ビュー：利用可能な異なるビュー間で切り替えます。
- 言語：UIで使用する言語を選択します。
- VNCにセキュアWebSocketsを使用する：VNCセッション開始時にセキュアWebSocketsを使用した接続を試みます。

- 表示名：ユーザーがSunstoneに表示される用户名をカスタマイズしたい場合、SUNSTONE/DISPLAY\_NAMEという特別なパラメータに希望の値を追加することで設定が可能です。なお、クラウド管理者はユーザー作成時のフックを使用してこれを自動化し、OpenNebula外部から用户名を取得することを検討されることをお勧めいたします。

これらのオプションは、ユーザーテンプレートに保存されます。同様に、ユーザーごとにデータテーブルに表示される項目数をSunstoneが記憶するための属性など、その他の非表示設定も保存されます。定義されていない場合、/etc/one/sunstone-server.confのデフォルト値が使用されます。

## 5.4 ラベル

OpenNebulaのリソースの大半は、管理画面からラベルを定義できます。各リソースはラベル情報を自身のテンプレートに保存するため、CLIやSunstoneから簡単に編集が可能です。この機能により、

特定のラベルで異なるリソースをグループ化し、管理画面やクラウドビューでそれらをフィルタリングすることが可能となります。これにより、ユーザーはインスタンス化したいテンプレートを簡単に見つけたり、特定のアクションを適用するリソース群を選択したりできるようになります。

The screenshot shows the OpenNebula Sunstone interface. On the left, there's a sidebar with navigation links like Dashboard, Instances (VMs, Services, Virtual Routers), Templates, Storage, Network, Infrastructure, System, and Settings. The main area is titled 'VMs' and displays a table with two entries:

ID	Name	Owner	Group	Status
1	Ttylinux - KVM-1	oneadmin	oneadmin	RUNNING
0	Ttylinux - KVM-0	oneadmin	oneadmin	RUNNING

Below the table, it says 'Showing 1 to 2 of 2 entries' and shows statistics: 2 TOTAL, 2 ACTIVE, 0 OFF, 0 PENDING, 0 FAILED.

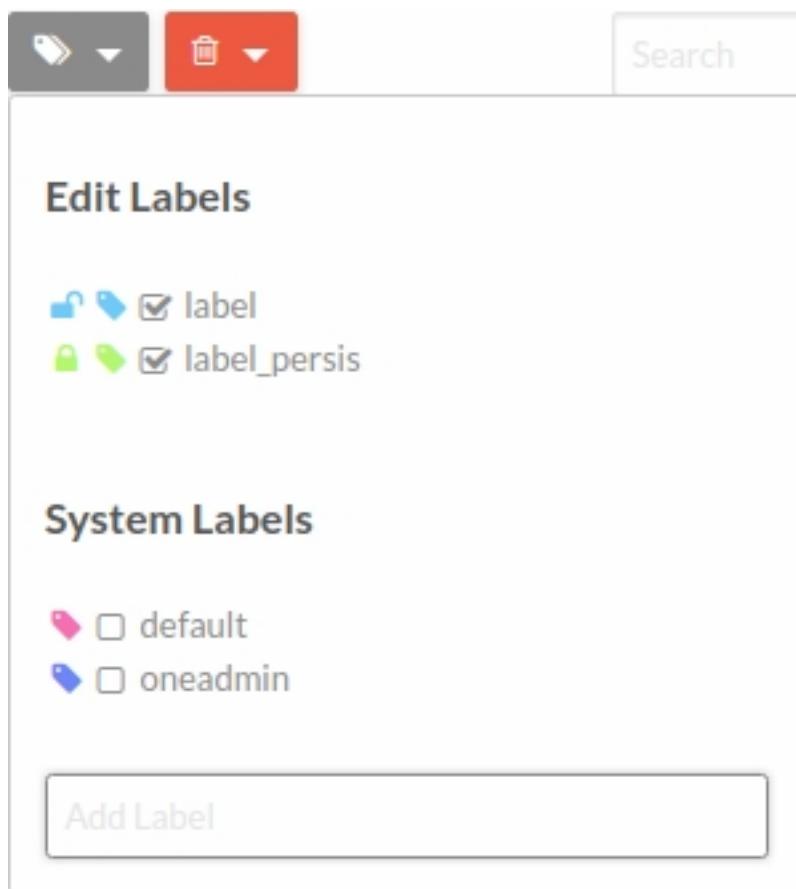
To the right of the table, there are two sections: 'Edit Labels' and 'System Labels'. Under 'Edit Labels', there are checkboxes for 'label' and 'label.persist'. Under 'System Labels', there are checkboxes for 'default' and 'oneadmin'. A text input field at the bottom is labeled 'Add Label'.

各プールに定義されたラベルの一覧は、左側のナビゲーションメニューに表示されます。これらのラベルのいずれかをクリックすると、そのラベルが付与されたリソースのみがテーブルに表示されます。このフィルター機能は、仮想マシン作成フォーム内のクラウドビューでもご利用いただけます。これにより、特定のテンプレートを簡単に選択することができます。

ラベルの階層構造を作成するには、「/」文字をご使用ください。例えば、「Linux/Ubuntu」や「Linux/CentOS」といったラベルを設定できます。なお、「Linux/Ubuntu」というラベルが付与されたリソースは、親ラベルである「Linux」に自動的に追加されるわけではありませんが、明示的に追加することは可能です。

#### 5.4.1 永続ラベル

永続ラベルの作成も可能です。このタイプのラベルは、関連するリソースが存在しなくなった場合でも削除されません。永続タグを定義するには、システムタグとして定義する方法と、ファイル `/etc/one/sunstone-views.yaml` に含める方法、またはユーザーのテンプレートに追加する方法の 2 つの選択肢がございます。後者の方法は、CLI または Sunstone インターフェース（既に作成済みのタグの鍵アイコンをクリック）を通じて実行できます。



## ユーザー ラベル

ユーザーが鍵アイコンをクリックすると、これらのラベルはユーザーのテンプレートに保存されます。これらのラベルは、CLIまたはSunstoneインターフェースから簡単に編集可能です。ユーザーのテンプレートにラベルを保存する際には、以下のテンプレートを追加してください。

```
TEMPLATE = [
    LABELS = "labels_persist,label_persist_2"
]
```

## システム ラベル

これらのラベルは `/etc/one/sunstone-views.yaml` で定義されています。ユーザーグループごとに分離することも、デフォルトセクションに導入することも可能です。

```
logo: images/opennebula-5.0.png groups:
  oneadmin:
    - admin
    - admin_vcenter
    - groupadmin
    - グループ管理者_vcenter
    - ユーザー
    - クラウド
    - クラウド_vcenter
```

デフォルト:

- クラウド デフォルト

グループ管理者:

- グループ管理者
- クラウド ラ

ペルグルーブ:

```
oneadmin:
  - oneadminデ
```

フォルト:

- デフォルト

## 5.4.2 新しいOpenNebula Sunstoneビューの定義または既存ビューのカスタマイズ

ビューの定義は、`/etc/one/sunstone-views/*`ディレクトリに配置されます。各ビューは設定ファイルによって定義され、その形式は以下の通りです：

`<ビュー名>.yaml`

ビュー名は、yaml拡張子を除いたファイル名となります。

```
/etc/one/
...
|-- sunstone-views/
|   |-- mixed/
|   |   |-- admin.yaml           <--- the admin ビュー
|   |   `-- cloud.yaml          <--- the クラウド ビュー
|   |-- kvm/
|   |   |-- admin.yaml           <--- the admin ビュー
|   |   `-- cloud.yaml          <--- the クラウド ビュー
|   |-- vcenter/
|   |   |-- admin.yaml           <--- the admin ビュー
|   |   `-- cloud.yaml          <--- the クラウド ビュー
`-- sunstone-views.yaml
...
```

---

**注記:** カスタムビューを作成する最も簡単な方法は、`admin.yaml` または `cloud.yaml` ファイルをコピーし、必要に応じて修正することです。

---

### 管理画面のカスタマイズ

ビューファイルの内容は、ロゴ、リンク、確認アクション、グループフィルター、有効な機能、表示タブ、有効なアクションを指定します。

- `small_logo`: 管理者ビューにおけるSunstoneロゴをカスタマイズするための画像パス。デフォルトではOpenNebulaロゴが使用されます。
- `provision_logo`: ユーザービューにおけるSunstoneロゴをカスタマイズするための画像パス。デフォルトではOpenNebulaロゴが表示されます。
- `link_logo`: ロゴの下にある外部URL。
- `text_link_logo`: `link_logo` のテキストリンク。

- 
- *confirm\_vms*: true の場合、ユーザーは仮想マシンに対する操作を確認する必要があります。
  - *filter\_view*: false の場合、フィルタリング対象のグループ名を非表示にします。

ダッシュボードでは、以下のウィジェットを設定できます：

```
# 以下のウィジェットは、下記の「_per_row」設定のいずれか内で使用できます。# 名前の通り、ウィジェットは1行に収まるようサイズ調整されます。
1行あたり2つ、または3つです。フッターではウィジェットをフルサイズで使用します。つまり、1行あたり1つとなります。
#
# - ストレージ #
- ユーザー
# - ネットワーク
# - ホスト
# - 仮想マシン
# - グループクオータ #
- クオータパネルタブ:

操作:
ダッシュボードの更新: いいえサンストーンの上
部表示切り替え: いいえ

ウィジェット_1行あたり1個:
- ホストウィジェット_3列表
示: ウィジェット_2列表示:
- 仮想マシン
- ユーザー
- ストレージ
- ネットワークウィジェ
ツト_1行あたり1個:
```

機能内には12の設定項目がございます：

- **showback**: この設定がfalseの場合、すべてのShowback機能が非表示となります。月次レポートの表、およびVM作成ウィザードにおける新規VMのコスト表示が対象です。
- **secgroups**: true の場合、VM 作成ウィザードで各ネットワークインターフェースにセキュリティグループを追加できるようになります。
- **instantiate\_hide\_cpu**: trueの場合、VM作成ダイアログ内のCPU設定を非表示にします。
- **instantiate\_cpu\_factor**: False に設定すると、VCPU から CPU をスケーリングしません。スケーリングする場合は [0, 1] の数値を指定します。
- **instantiate\_persistent**: インスタンスを永続化するためのオプションを表示するには true を指定します。
- **vcenter\_vm\_folder**: true を指定すると、vCenter VM をデプロイする仮想マシンおよびテンプレートのパス/フォルダを指定するための入力欄が表示されます。
- **show\_ds\_instantiate**: True に設定すると、VM をインスタンス化するデータストアのデータテーブルを表示します。
- **show\_vmgp\_instantiate**: True に設定すると、VM をインスタンス化するための vmgroup データテーブルを表示します。
- **show\_vnet\_instantiate**: True に設定すると、VM をインスタンス化する vnet データテーブルを表示します。
- **show\_host\_instantiate**: True を指定すると、VM をインスタンス化するホストデータテーブルが表示されます。
- **show\_monitoring\_info**: True に設定すると、監視情報（VM および VRouter）を表示します。
- **show\_attributes\_info**: 属性の情報（VM および VRouter）を表示する場合は True に設定します。

- 
- `resize_enforce`: `True` に設定すると、ホストの容量がチェックされます。これは `oneadmin` リクエストにのみ影響し、一般ユーザーのサイズ変更リクエストに対する制限は常に適用されます。
  - `deploy_enforce`: `True` に設定すると、ホストの容量がチェックされます。これは `oneadmin` リクエストにのみ影響し、一般ユーザーのサイズ変更リクエストに対する制限は常に適用されます。
  - `migrate_enforce`: `True` に設定すると、ホストの容量がチェックされます。これは `oneadmin` リクエストにのみ影響し、一般ユーザーのサイズ変更リクエストに対する制限は常に適用されます。

**機能:**

```
# showback の月次レポートとVMコストを表示するにはTrue
showback: true

# VM作成ダイアログ上で、各ネットワークインターフェースのセキュリティグループを変更できるようにします
secgroups: true
```

このファイルでは、ビューで利用可能なタブも定義します（注：タブはUIの主要なセクションの一つであり、左側のメニューに表示されます）。各タブは、`enabled_tabs:` 属性を更新することで有効化または無効化できます。例えば、クラスター タブを無効化するには、`clusters-tab` エントリをコメントアウトしてください：

```
enabled_tabs:
  - dashboard-tab
  - instances-top-tab
  - vms-tab
  - oneflow-services-tab
  - テンプレート-トップタブ
  - テンプレートタブ
  - ワンフロー・テンプレートタブ
  - ストレージトップタブ
  - データストアタブ
  - 画像タブ
  - ファイルタブ
  - マーケットプレイス
  - マーケットプレイスアプリタブ
  - ネットワークタブ
  - 仮想ネットワークタブ
  - 仮想ルータータブ
  - 仮想ネットワークのトポジタブ
  - セキュリティグループタブ
  - インフラストラクチャのトップタブ
#- クラスター タブ
  - ホストタブ
  - ゾーンタブ
  - システム-トップタブ
  - ユーザータブ
  - グループタブ
  - VDCSタブ
  - アクセス制御リストタブ
  - 設定タブ
  - サポートタブ
```

各タブは、以下の選択により調整が可能です：

- タブ内で利用可能な個別のリソースタブ (`panel_tabs:` 属性) です。これらはオブジェクトが選択された際にアクティブになるタブです（例：仮想マシンタブ内の情報タブや容量タブなど）。
- メイン情報テーブルに表示される列 (`table_columns:` 属性)。
- ビューで使用可能なアクションボタン (`actions:` 属性)。

### OpenNebula 5.10 導入ガイド、リリース 5.10.5

上記各項目の属性は、その名称から内容が推測できるはずです。例として、情報パネルタブとアクションボタンを省略した簡易データストアタブを以下に定義します：

```
datastores-tab: panel_tabs:  
    datastore_info_tab:  
        false  
    datastore_image_tab: true
```

```

datastore_clusters_tab: false table_columns:
- 0          # チェックボックス
- 1          # ID
- 2          # 所有者
- 3          # グループ
- 4          # 名前
- 5          # 収容人数
- 6          # クラスター
#- 7          # ベースパス
#- 8          # TM
#- 9          # DS
- 10         # タイプ
- 11         # ステータス
#- 12        # ラベル

操作:
データストアの更新: true データストアの作成
ダイアログ: false データストアのインポートダ
イアログ: false データストアのクラスタへの追
加: false データストアの名前変更: false デ
ータストアの所有者変更: false データストアの
グループ変更: false データストアの権限変更:
false データストアの削除: false データスト
アの有効化: false データストアの無効化:
false

```

## クラウドビューのカスタマイズ

クラウドのレイアウトは、対応する /etc/one/sunstone-views/ の YAML ファイルを変更することでカスタマイズすることも可能です。このファイルでは、新しいテンプレートのインスタンス化時に利用可能なオプション、ダッシュボードの設定、またはクラウドユーザーが利用可能なリソースをカスタマイズできます。

### 機能

- `showback`: この設定が`false`の場合、すべてのShowback機能が非表示となります。具体的には、月次レポートの表およびVM作成ウィザードにおける新規VMのコスト表示が対象となります。
- `secgroups`: `true` の場合、VM 作成ウィザードでは各ネットワークインターフェースにセキュリティグループを追加できるようになります。
- `instantiate_hide_cpu`: `true` の場合、VM 作成ダイアログの CPU 設定を非表示にします。
- `instantiate_cpu_factor`: `False` を指定すると、VCPU から CPU をスケーリングしません。スケーリングする場合は [0, 1] の数値を指定します。
- `instantiate_persistent`: インスタンスを永続化するためのオプションを表示するには `true` を指定します。
- `cloud_vm_create`: クラウドユーザーによるマシン作成を許可する場合は `true` となります。
- `show_monitoring_info`: 監視情報（VM および VRouter）を表示する場合は `True` です。
- `show_attributes_info`: 属性の情報を表示するには `True` を設定してください（VM および VRouter）。

- 
- `show_attach_disk_advanced`: ディスク接続ダイアログに詳細オプションを表示するには True を指定してください。
  - `show_attach_nic_advanced`: NIC 接続ダイアログで詳細オプションを表示するには True を指定します。

```

features:
  # showback の月次レポートと VM コストを表示するには True を設定してください。
  showback: true

  # VM 作成ダイアログ上で、各ネットワークインターフェースのセキュリティグループを変更可能にします
  secgroups: true

```

## リソース

仮想マシンのリストは常に表示されます。 仮想マシンテンプレートおよび OneFlow サービスのリストは、 provision\_tabs 設定で非表示にすることができます。

```

tabs:
  プロビジョニングタブ:
    provision_tabs:
      flows: true
      templates: true

```

## ダッシュボード

ダッシュボードでは、ユーザーのクオータ、グループのクオータ、ユーザー仮想マシンの概要、およびグループ仮想マシンの概要を表示するように設定できます。

```

タブ:
  ダッシュボード:
    # 接続中のユーザーのクオータ
    quotas: true
    # 接続済みユーザーの仮想マシン概要
    vms: true
    # グループのクオータ
    vdcquotas: false
    グループ内の仮想マシンの概要
    vdcmms: false

```

## 仮想マシン作成ウィザード

仮想マシン作成ウィザードは、以下のオプションで設定できます:

```

タブ:
  create_vm:
    # 容量 (CPU、メモリ、VCPU) のカスタマイズを許可する場合は true
    capacity_select: true
    # NIC のカスタマイズを許可する場合は true
    network_select: true
    # ディスクサイズのカスタマイズを許可する場合は true
    disk_resize: true

```

## アクション

特定の仮想マシンに対して利用可能なアクションは、yamlファイルを修正することでカスタマイズおよび拡張が可能です。管理画面の仮想マシンパネルをこのビューに挿入することもでき、例えばディスクスナップショットやスケジュールされたアクションを利用できます。

- 削除ボタンの非表示

```
tabs:
  プロビジョニングタブ:
    ...
    actions: &provisionactions
      ...
      VM.shutdown_hard: falseVM.delete:
        false
```

- 電源オフではなくアンデプロイを使用する

```
タブ:
  プロビジョニングタブ:
    ...
    アクション: &プロビジョニングアクション
      ...
      VM.poweroff: falseVM.poweroff_hard:
        falseVM.undeploy:
          trueVM.undeploy_hard: true
```

- 管理ビューからパネルを追加する場合、例えばディスクスナップショットタブなど

```
タブ:
  プロビジョニングタブ:
    panel_tabs:
      ...
      vm_snapshot_tab: true
      ...
    アクション: &プロビジョニングアクション
      ...
      VM.ディスクスナップショット作成: trueVM.ディス
        イスクスナップショット復元: trueVM.ディス
          クスナップショット削除: true
```

Open Nebula Dashboard VMs Templates Services oneadmin OpenNebula Officially supported

### Ttylinux - KVM-1

**RUNNING**

- x0.1 - 128MB - Ttylinux - KVM
- 192.168.122.12, 192.168.122.13
- oneadmin 1h ago - ID: 1

**Actions:** [Edit] [Delete] [Power Off] [Reboot]

**Metrics:**

- CPU: 100% (16:33 - 17:23)
- Memory: 97.7MB (146.5MB, 48.8MB, 0KB) (16:33 - 17:23)
- Net RX: 293KB (195.3KB, 97.7KB) (16:33 - 17:23)
- Net TX: 1B (16:33 - 17:23)
- Net Download Speed: 200B/s (600B/s, 400B/s, 200B/s) (16:33 - 17:23)
- Net Upload Speed: 0B/s (1B/s, 0.5B/s) (16:33 - 17:23)

**Storage:**

ID	Target	Image / Size-Format	Size	Persistent	Actions
0	vda	Ttylinux - KVM	27MB/200MB	NO	[Edit] [Delete] [Attach disk]
1	hda	Context	1MB/-	NO	

Showing 1 to 2 of 2 entries

Cost / MByte: 0

Disk read bytes: 781.3KB, 585.9KB, 390.6KB, 195.3KB (16:33 - 17:23)

Disk write bytes: 9.8KB, 4.9KB (16:33 - 17:23)

Disk read IOPS: 60, 40, 20 (16:33 - 17:23)

Disk write IOPS: 2.5, 2, 1.5, 1, 0.5 (16:33 - 17:23)

#### 自動更新

個々のVMビューにおけるVM自動更新時間（ミリ秒）。デフォルトは10秒です。以下の行を  
vms-tab に以下の行を追加する必要があります:

```
actions:
  VM.refresh: true
  ...
  VM.menu_labels:
trueautorefresh_info: 5000 # ミリ秒
```

## 5.5 ユーザーセキュリティと認証

デフォルトでは、Sunstone はコア認証方法（ユーザー名とパスワード）で動作しますが、OpenNebula がサポートするあらゆる認証メカニズムを設定することが可能です。このセクションでは、他の認証方法を有効化する方法と、SSL を通じて Sunstone 接続を保護する方法について学びます。

### 5.5.1 認証方法

認証には二つの側面があります：

- **WebクライアントとSunstoneサーバー**です。認証は、OpenNebulaデータベースに保存されているユーザーの認証情報に基づいて行われます。これらの認証情報の種類に応じて、認証方法は以下のいずれかになります：sunstone、x509、opennebula（LDAPやその他のカスタム方法をサポート）。
- **サンストーンサーバーとOpenNebulaコア**。ユーザーからのリクエストは、元のユーザー名を含めてコアデーモンに転送されます。各リクエストは、特別なサーバーユーザーの認証情報で署名されます。この認証メカニズムは、対称鍵暗号（デフォルト）またはx509証明書のいずれかに基づいています。これらの方法の設定方法の詳細については、「[クラウド認証](#)」セクションをご参照ください。

以下のセクションでは、クライアントからサンストーンサーバーへの認証方法について詳しく説明します。

#### 基本認証

基本モードでは、ログイン時にユーザーを認証するため、ユーザー名とパスワードが OpenNebula のデータベースに登録されているものと照合されます。その後、Rack のクッキーベースのセッションを使用してリクエストの認証と認可が行われます。

このログイン方法を有効にするには、`/etc/one/sunstone-server.conf` の `:auth:` オプションを `sunstone` に設定してください。

```
:auth: sunstone
```

#### OpenNebula認証

この方法を使用すると、ヘッダーに含まれる認証情報が OpenNebula コアに送信され、指定されたドライバーを使用してそのユーザーの認証が OpenNebula 認証システムに委譲されます。したがって、この方法を通じて任意の OpenNebula 認証ドライバー（例：LDAP）を使用してユーザーを認証できます。`sunstone` の設定は以下の通りです：

```
:auth: opennebula
```

#### x509認証

この方法は、x509 証明書 DN（識別名）に基づいて OpenNebula へのログインを実行します。DN は証明書から抽出され、ユーザーデータベース内のパスワード値と照合されます。

ユーザーパスワードの変更には、以下のいずれかのコマンドを実行する必要があります：

```
$ oneuser chauth new_user x509 "/C=ES/O=ONE/OU=DEV/CN=clouduser"
```

または、証明書ファイルを使用した同じコマンド：

```
$ oneuser chauth new_user --x509 --cert /tmp/my_cert.pem
```

この認証方式を使用する新規ユーザーは、以下の方法で作成してください：

```
$ oneuser create new_user "/C=ES/O=ONE/OU=DEV/CN=clouduser" --driver x509
```

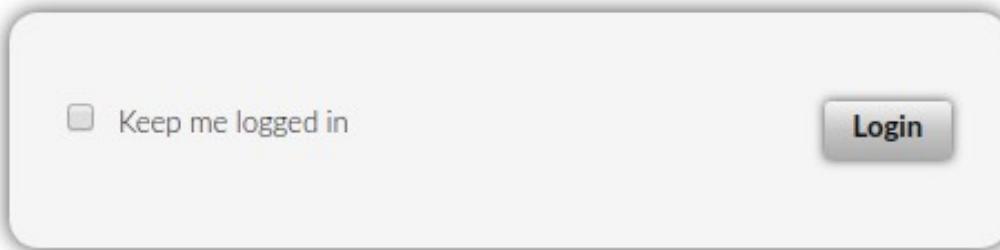
または証明書ファイルを使用する場合：

```
$ oneuser create new_user --x509 --cert /tmp/my_cert.pem
```

このログイン方法を有効にするには、/etc/one/sunstone-server.conf の :auth: オプションを x509 に設定してください:

```
:auth: x509
```

ログイン画面には、ユーザー名とパスワードの入力欄が表示されなくなります。すべての情報はユーザー証明書から取得されるためです:



OpenNebulaは、ログイン時にユーザーが有効な証明書を保持しているかどうかを検証しません。これは、Sunstoneサーバーの外部コンテナ（通常はApache）が行うべき処理です。外部コンテナの役割は、ユーザーのブラウザに対してサイトがユーザー証明書を要求していることを通知し、選択された認証局（CA）によって一貫して署名された証明書であることを確認することです。

**警告：** Sunstoneのx509認証方式は、ログイン時のユーザー認証のみを処理します。ユーザー証明書の認証は補完的な設定であり、Apacheに依存することが可能ですが。

## リモート認証

この方法はx509認証と類似しております。KerberosのREMOTE\_USERに基づいてOpenNebulaへのログインを実行します。REMOTE\_USER変数からUSER@DOMAINが抽出され、ユーザーのパスワード値と照合されます。

データベース。Kerberos認証をご利用いただくには、ユーザーをパブリックドライバーで設定する必要があります。なお、これによりXML-RPCインターフェース経由での認証はできなくなり、これらのユーザーにはSunstoneへのアクセスのみが許可されます。既存ユーザーをKerberos認証に対応させるには、ドライバーをパブリックに変更し、パスワードを以下のように更新してください：

```
$ oneuser chauth new_user public "new_user@DOMAIN"
```

この認証方式を使用する新規ユーザーは、以下の方法で作成してください：

```
$ oneuser create new_user "new_user@DOMAIN" --driver public
```

このログイン方法を有効にするには、/etc/one/sunstone-server.conf の :auth: オプションを remote に設定してください。

```
:auth: remote
```

ログイン画面にはユーザー名とパスワードの入力欄が表示されなくなります。すべての情報はKerberosサーバーまたはリモート認証サービスから取得されるためです。

なお、OpenNebulaはログイン時にユーザーが有効な Kerberos チケットを保持しているかどうかを検証しません。これは Sunstone サーバーの外部コンテナ（通常は Apache）の役割であり、ユーザーのブラウザに対してログインには有効なチケットが必要であることを通知するものです。

**警告:** Sunstone リモート認証方式は、ログイン時のユーザー認証のみを処理します。リモートチケットの認証は補完的な設定であり、Apache に依存することが可能です。

## 二要素認証

この方法では、ユーザー名とパスワードの入力が求められるだけでなく、Google Authenticator、Authy、またはMicrosoft Authenticatorのいずれかのアプリケーションで生成されたトークンも必要となります。

この機能を有効にするには、以下の手順に従ってください：

- Sunstoneにログイン後、**設定**メニューを選択してください。設定画面内で「認証」タブを探し、選択してください。

ID	0
Name	oneadmin
Table Order	-
Language	-
View	-

- 「二段階認証」タブ内にある「二段階認証を管理」ボタンを選択してください

The screenshot shows the OpenNebula Settings interface. On the left is a sidebar with navigation links like Dashboard, Instances, Templates, Storage, Network, Infrastructure, System, and Users. The main area has tabs for Info, Quotas, Group Quotas, Accounting, Showback, and Auth. The Auth tab is active. Under Authentication, there's a section for Two factor authentication. It shows a QR code for Google Authenticator, a 'Manage two factor authentication' button, a 'View' button, and a 'Manage login tokens' button.

- QRコードが表示されたウィンドウが開きます。このQRコードは、Google Authenticator、Authy、またはMicrosoft Authenticatorのいずれかのアプリでスキャンする必要があります。これにより6桁のコードが生成されますので、そのコードをコード入力欄に入力してください。

## Two Factor Authentication

Two factor authentication can be enabled for logging into Sunstone UI.



- Get authenticator app: [Authy](#), [Google Authenticator](#), [Microsoft Authenticator](#).
- Scan this Barcode.
- Enter Verification Code.

**Enable**

内部的には、sunstoneがTWO\_FACTOR\_AUTH\_SECRETフィールドを追加します。

```
SUNSTONE=[  
    TWO_FACTOR_AUTH_SECRET="2LJUUXIEEYAK6NQR"  
    TOKEN_PASSWORD="5dcff75786fdb552f31e3c6af105a7de0b5cd0dfcca42a77349c7c9ff9038cc3"]
```

- 2段階認証を無効にするには、設定の「認証」に移動し、「無効化」ボタンを選択してください。

The screenshot shows the OpenNebula Settings interface. The sidebar and tabs are identical to the previous screenshot. Under Authentication, there's a section for Two factor authentication. It shows a 'Disable' button, a 'View' button, and a 'Manage login tokens' button.

## 5.5.2 SSLプロキシの設定

OpenNebula Sunstoneは、通常のHTTP接続のみでネイティブに動作します。SSLによる追加のセキュリティが必要な場合、SSL接続を処理するプロキシを設定できます。このプロキシは、Sunstoneサーバーにリクエストを転送し、クライアントに回答を返します。

この設定には以下のものが必要です：

- SSL接続用のサーバー証明書
- SSLを理解するHTTPプロキシ
- プロキシからのリクエストを受け入れるためのOpenNebula Sunstoneの設定

SSL設定を簡単に試してみたい場合、以下の手順は、正しく設定されたOpenNebula SunstoneへのHTTPプロキシとして動作するよう構成されたWebサーバーで使用する自己署名証明書を設定する例を示しています。

プロキシを起動するサーバーを `cloudserver.org` と仮定します。したがって、手順は以下の通りです：

### ステップ1: サーバー証明書 (Snakeoil)

スネークオイル証明書を生成します。Ubuntuシステムをご利用の場合は、以下の手順に従ってください（他のシステムの場合、手順が若干異なる場合がありますが、大きな違いはありません）。

- 1. `ssl-cert` パッケージをインストールします

```
# apt-get install ssl-cert
```

- 証明書の生成

```
# /usr/sbin/make-ssl-cert generate-default-snakeoil
```

- lighttpdをご利用のため、lighttpdで有効なサーバー証明書を取得するには、証明書に秘密鍵を追加する必要があります。

```
# cat /etc/ssl/private/ssl-cert-snakeoil.key /etc/ssl/certs/ssl-cert-snakeoil.pem > /etc/lighttpd/server.pem
```

### ステップ 2: SSL HTTP プロキシ

#### lighttpd

`/etc/lighttpd/lighttpd.conf` 設定ファイルを編集し、

- 以下のモジュールを追加してください（既に存在しない場合）
  - mod\_access
  - mod\_alias
  - mod\_proxy
  - mod\_accesslog
  - mod\_compress
- root ユーザーとして lighttpd を実行する場合はサーバーポートを 443 に変更してください。それ以外の場合は 1024 以上の任意の番号を設定して

ください:

```
server.port = 8443
```

- プロキシモジュールセクションを追加してください:

```
#### proxy モジュール
## 詳細については proxy.txt をご参照ください
proxy.server =
  = ( "" =>
    ( "" =>
      "host" => "127.0.0.1",
      "port" => 9869
    )
  )

#### SSL エンジン ssl.engine
ssl.engine = "有効"
ssl.pemfile = "/etc/lighttpd/server.pem"
```

ホストは、Sunstoneサーバーを実行しているコンピュータのサーバーホスト名である必要があります、ポートはSunstoneサーバーが実行されているポートである必要があります。

## nginx

nginxで新しい仮想ホストを設定する必要があります。 オペレーティングシステムやインストール方法によって異なりますが、nginxは仮想ホストの設定を/etc/nginx/conf.dまたは/etc/nginx/sites-enabledから読み込みます。

- 以下に、cloudserver.org の仮想ホスト設定例を示します：

```
#### OpenNebula Sunstone upstream
sunstone {
  server 127.0.0.1:9869;
}

#### cloudserver.org HTTP 仮想ホスト
server {
  listen 80;
  server_name cloudserver.org;

  #### HTTPSへの恒久的なリダイレクト（オプション）return 301
  https://$server_name:8443;
}

#### cloudserver.org HTTPS 仮想ホスト サーバー
server {
  listen 8443;
  server_name cloudserver.org;

  #### SSL パラメータ
  ssl_certificate /etc/ssl/certs/ssl-cert-snakeoil.pem;ssl_certificate_key
  /etc/ssl/private/ssl-cert-snakeoil.key;

  #### アップストリームロケーションへのリクエストをプロキシします
  / {
```

```

    プロキシパス プロキシヘッ
    ダー設定 プロキシヘッダー
    設定 プロキシヘッダー設定
}

}

```

アップストリームで使用されるIPアドレスとポート番号は、Sunstoneサーバーが稼働しているサーバーのものと一致している必要があります。一般的なインストール環境では、nginxのマスター・プロセスはrootユーザーとして実行されるため、HTTPSポートを変更する必要はありません。

### ステップ 3: Sunstone の設定

/etc/one/sunstone-server.conf を編集し、localhost:9869 でリスンするように設定してください。

```
:host: 127.0.0.1
:port: 9869
```

プロキシサーバーが起動されると、OpenNebula SunstoneによるHTTPS URIを使用したリクエストは、`https://cloudserver.org:8443`へ転送されます。その後、通信は復号化され、ローカルホストのポート 9869 へ渡されます。そこで処理が行われ（正常に処理されることを願います）、再度暗号化されてクライアントへ返されます。

---

**注記:** サーバー管理者のパスワードを変更するには、以下の手順に従ってください:

```
#oneuser passwd 1 --sha256 <PASSWORD>
#echo 'serveradmin:PASSWORD' > /var/lib/one/.one/oneflow_auth #echo
'serveradmin:PASSWORD' > /var/lib/one/.one/ec2_auth #echo
'serveradmin:PASSWORD' > /var/lib/one/.one/onegate_auth #echo
'serveradmin:PASSWORD' > /var/lib/one/.one/occi_auth #echo
'serveradmin:PASSWORD' > /var/lib/one/.one/sunstone_auth
```

パスワード変更後は、Sunstone を再起動してください。

---

## 5.6 クラウドサーバー認証

OpenNebulaには、SunstoneとEC2の2つのサーバーが同梱されています。ユーザーがこれらのいずれかとやり取りを行う際、サーバーはリクエストを認証した後、要求された操作をOpenNebulaデーモンに転送します。

サーバー間およびコアデーモン間で転送されるリクエストには、元のユーザー名が含まれており、特別なサーバーユーザーの認証情報で署名されます。

本ガイドでは、このリクエスト転送の仕組みと、対称鍵アルゴリズムまたは x509 証明書によるセキュリティ保護の方法について説明します。

### 5.6.1 サーバーユーザー

*Sunstone* および EC2 サービスは、サーバーユーザーを使用してコアと通信します。OpenNebula は、起動時に認証ドライバ `server_cipher` (対称鍵) を使用して `serveradmin` アカウントを作成します。

このサーバーユーザーは、サーバーが別のユーザーに代わって操作を実行できるようにする特別な認証メカニズムを使用します。

サーバーからコアデーモンへのリクエストのセキュリティを強化するには、serveruser のドライバを `server_x509` に変更してください。これは、サーバーをフロントエンド以外のマシンで実行している場合に特に重要です。

サーバーユーザー用の `server_*` ドライバーは、必要に応じていくつでも設定可能です。例えば、Sunstone には `server_x509` ドライバーを、EC2 には `server_cipher` ドライバーを設定するといった構成が可能です。

## 5.6.2 対称鍵

### 有効化

この機能はデフォルトで有効になっております。`serveradmin` というユーザー名が割り当てられ、`server_cipher` ドライバーが使用されます。

ご利用いただくには、`server_cipher` ドライバーを持つユーザーが必要です。/etc/one 内の関連設定ファイルで有効化してください：

- Sunstone: /etc/one/sunstone-server.conf
- EC2: /etc/one/econe.conf

```
:core auth: cipher
```

### 設定

サーバー管理者のパスワードを変更する場合、または `server_cipher` ドライバーを使用して別のユーザーを作成する場合は、`/var/lib/one/.one` 内の設定ファイルを更新する必要があります。

```
$ ls -l /var/lib/one/.one ec2_auth
sunstone_auth

$ cat /var/lib/one/.one/sunstone_auth serveradmin:1612b78a4843647a4b541346f678f9e1b43bbcf9
```

**警告:** `serveradmin` のパスワードはデータベース内でハッシュ化されています。このユーザーに対して `oneuser passwd` コマンドを実行する際には、`--sha256` フラグをご利用いただけます。

**警告:** Sunstone が oned とは異なるマシンで実行されている場合、SSL 接続を使用する必要があります。これは stunnel や Apache/nginx などの SSL プロキシをプロキシとして動作させることで実現できます。OpenNebula XML-RPC 接続を保護した後、Sunstone をプロキシポート経由で https を使用するように設定してください:

```
:one_xmlrpc: https://frontend:2634/RPC2
```

## 5.6.3 x509 暗号化

### 有効化

有効化するには、`serveradmin` ユーザーの認証ドライバを変更するか、ドライバを指定して新規ユーザーを作成してください。`server_x509`:

```
$ oneuser chauth serveradmin server_x509
$ oneuser passwd serveradmin --x509 --cert usercert.pem
```

serveradmin アカウントは次のようになります:

```
$ oneuser list

 IDグループ      お名前      認証
 ↳パスワード
 0 oneadmin oneadmin      コア
 ↳c24783ba96a35464632a624d9f829136edc0175e
 1 oneadmin serveradmin    server_x
 ↳CN=サーバー
                                         /C=ES/O=ONE/OU=DEV/
```

/etc/one/auth/server\_x509\_auth.conf を編集し、すべてのフィールドのコメントを外す必要があります。デフォルト設定で動作するはずです：

```
# x509サーバー認証に使用するユーザー
:srv_user: serveradmin

OpenNebulaサービスで使用される証明書のパス証明書はPEM形式である必要があります
:one_cert: "/etc/one/auth/cert.pem"
:one_key: "/etc/one/auth/pk.pem"
```

証明書と秘密鍵を :one\_cert: および :one\_key: で設定されたパスにコピーするか、単にパスを更新してください。

その後、/etc/one 内の関連する設定ファイルを編集してください:

- Sunstone: /etc/one/sunstone-server.conf
- EC2: /etc/one/econe.conf

```
:core_auth: x509
```

## 設定

サーバー管理者証明書（デフォルトパスを使用した場合 /etc/one/auth/cert.pem）を信頼するには、CAの証明書を /etc/one/auth/x509\_auth.conf で定義されている ca\_dir に追加する必要があります。詳細については、[x509認証ガイド](#)をご参照ください。

```
$ openssl x509 -noout -hash -in cacert.pem 78d0bbd8
$ sudo cp cacert.pem /etc/one/auth/certificates/78d0bbd8.0
```

## 5.6.4 チューニングと拡張

### ファイル

ドライバは以下のパスにございます:

- /var/lib/one/remotes/auth/server\_cipher/authenticate
- /var/lib/one/remotes/auth/server\_server/authenticate

## 認証セッション文字列

OpenNebula のユーザーで、`server_cipher` または `server_x509` ドライバーを使用している場合は、特別な認証セッション文字列（XML-RPC 呼び出しの最初のパラメータ）を使用します。通常の認証トークンは次の形式です：

```
username:secret
```

一方、`server_*` ドライバーを使用するユーザーは、以下のトークン形式を使用する必要があります：

```
username:target_username:secret
```

コアデーモンは、この認証セッショントークンを含むリクエストを「対象ユーザーに代わってこの操作を実行する」という指示として認識します。トークンの秘密部分は、前述の二つのメカニズムのいずれかで署名されます。

## 二要素認証

2FAをご利用になる場合は、以下の [リンク](#)をご参照ください。

## 5.7 大規模導入におけるSunstoneの設定

小規模から中規模のエンタープライズクラウドでは、通常、SunstoneはOpenNebulaデーモンと共に単一のマシンにデプロイされます。ただし、このシンプルなデプロイメントは以下のように改善できます：

- WebクライアントからのSunstoneサーバーへのアクセスを分離します。これは、Sunstoneサーバーを別のマシンにデプロイすることで実現できます。
- 大規模なユーザープールに対応するため、サーバーのスケーラビリティを向上させます。通常、Sunstoneは1つ以上のホスト上で、独立したアプリケーションコンテナ内にデプロイされます。

[APIのスケーラビリティガイド](#)も併せてご確認ください。これらのヒントはSunstoneのパフォーマンスにも影響を及ぼします。

### 5.7.1 Sunstone を別のマシンにデプロイする場合

デフォルトでは、Sunstoneサーバーはフロントエンドで実行されるように設定されていますが、フロントエンドとは異なるマシンにSunstoneサーバーをインストールすることも可能です。

- サーバーを実行するマシンには、Sunstoneサーバーパッケージのみをインストールする必要があります。ソースからインストールする場合は、`install.sh`スクリプトに-sオプションを指定してください。
- `sunstone-server.conf` 内の `:one_xmlprc:` 変数が、OpenNebula フロントエンドが実行されている場所を指していることをご確認ください。未定義のままにし、代わりに `ONE_XMLRPC` 環境変数をエクスポートすることも可能です。
- サーバー管理者の認証情報は、ファイル `/var/lib/one/.one/sunstone_auth` に記述してください。サーバー管理者のパスワードを変更された場合は、[クラウドサーバー認証ガイド](#)をご確認ください。
- OpenNebulaにファイルをアップロードする場合は、`sunstone`と`oned`間でアップロードディレクトリ（デフォルトは`/var/tmp`）を共有する必要があります。一部のサーバーでは`TMPDIR`環境変数が考慮されないため、その場合は設定ファイル（例：`Passenger`の`client_body_temp_path`）でこのディレクトリを定義する必要があります。

```
$ cat /var/lib/one/.one/sunstone_auth
serveradmin:1612b78a4843647a4b541346f678f9e1b43bbc19
```

この設定では、仮想マシンのログは利用できません。この情報を取得する必要がある場合は、サーバーをフロントエンドにデプロイする必要があります。

## 5.7.2 別のウェブサーバー内の Sunstone の実行

Sunstoneの自己完結型デプロイメント（sunstone-serverスクリプトを使用）は、小規模から中規模のインストールには適しています。ただし、サービスに多数の同時ユーザーが存在し、システム内のオブジェクト数が高い場合（例えば、2000台以上の同時仮想マシンなど）、この方法は適切ではなくなります。

サンストーンサーバーはラックサーバーとして動作できるよう改修されました。これにより、このプロトコルをサポートするあらゆるウェブサーバーでの運用が可能となります。Rubyの世界では、これはほとんどのウェブサーバーが標準でサポートする仕様です。これにより、unicornのように複数のプロセスを生成する機能をサポートするウェブサーバー、あるいはPassengerモジュールを使用してApacheやnginxウェブサーバー内にサービスを埋め込むことが選択可能となりました。さらに、複数のサーバーでサンストーンを稼働させ、それらの間で負荷を分散させることも可能となります。

**警告：**フェデレーション環境においてプロキシの背後でSunstoneを展開する場合、フェデレーションが要求するSunstoneヘッダーを適切に処理するために、特定の設定が必要となります。

- `nginx`: `underscores\_in\_headers` を有効にしてください；および `proxy\_pass\_request\_headers` を有効にしてください；

### memcachedの設定

これらのWebサーバーのいずれかをご利用の際には、memcachedサーバーのご利用が必須となります。Sunstoneはユーザーセッションを保存する必要があり、これにより各操作ごとにユーザー名やパスワードの入力を求められずに済みます。デフォルトでは、Sunstoneはメモリセッションを使用するよう設定されており、セッションはプロセスメモリ内に保存されます。ThinおよびWebrickウェブサーバーは、新しいプロセスではなく新しいスレッドを生成します。そして、すべてのスレッドがそのセッションプールにアクセスできます。Sunstoneサーバーに複数のプロセスを使用する場合、この情報を保存し、すべてのプロセスからアクセス可能なサービスが必要です。この場合、memcachedのインストールが必要となります。ほとんどのディストリビューションに同梱されており、デフォルト設定で問題ないはずです。また、アクセス可能なRubyライブラリのインストールも必要です。必要なRubyGemsライブラリはmemcache-clientです。お使いのディストリビューションにこのライブラリのパッケージがない場合は、RubyGemsを使用してインストールできます：

```
$ sudo gem install memcache-client
```

その後、Sunstoneの設定ファイル（/etc/one/sunstone-server.conf）において、`:sessions` の値を `memcache` に変更する必要があります。

noVNCをご利用になる場合は、そのサービスを実行している必要があります。以下のコマンドで起動できます：

```
$ novnc-server start
```

もう一点、サーバーが実行されるユーザーについても考慮が必要です。インストール時には `oneadmin` ユーザーおよびグループに権限が設定され、Sunstone の設定ファイルや認証情報などのファイルは他のユーザーからは読み取れません。Apache は通常 `www-data` ユーザーおよびグループで実行されるため、サーバーをこのユーザーで実行するには、これらのファイルのグループを変更する必要があります。例えば：

```
$ chgrp www-data /etc/one/sunstone-server.conf
$ chgrp www-data /etc/one/sunstone-plugins.yaml
$ chgrp www-data /var/lib/one/.one/sunstone_auth
$ chmod a+x /var/lib/one
$ chmod a+x /var/lib/one/.one
$ chmod a+x /var/lib/one/sunstone
$ chgrp www-data /var/log/one/sunstone*
$ chmod g+w /var/log/one/sunstone*
```

インストールにはPassengerのご利用をお勧めいたしますが、ここでは例としてUnicornウェブサーバー内でSunstoneを実行する方法をご説明いたします。

Rackをサポートするウェブサーバーの詳細や、Rackに関するより詳しい情報につきましては、[Rackのドキュメント](#)ページをご参照ください。あるいは、[Rubyウェブサーバーの一覧](#)もご確認いただけます。

### Unicorn での Sunstone の実行

このウェブサーバーの詳細については、[公式ウェブページ](#)をご参照ください。Unicornはマルチプロセス型のウェブサーバーであり、リクエスト処理のために新しいプロセスを生成します。

インストールはrubygems（または利用可能なパッケージマネージャー）を使用して行います：

```
$ sudo gem install unicorn
```

Sunstoneファイルが配置されているディレクトリ（/usr/lib/one/sunstone または /usr/share/opennebula/sunstone）には、config.ru というファイルがございます。このファイルはRackアプリケーション専用の設定ファイルであり、アプリケーションの実行方法を指定するものです。Unicornを使用して新しいサーバーを起動するには、該当ディレクトリから以下のコマンドを実行してください：

```
$ unicorn -p 9869
```

デフォルトのunicorn設定はほとんどの環境で問題ありませんが、設定ファイルを作成して調整することも可能です。例えば、unicornに4つのプロセスを生成させ、標準エラー出力を/tmp/unicorn.logに書き込むように指示するには、unicorn.confというファイルを作成し、以下のように記述します：

```
worker_processes 4
logger debug
stderr_path '/tmp/unicorn.log'
```

サーバーを起動し、デーモン化するには以下のコマンドを使用します：

```
$ unicorn -d -p 9869 -c unicorn.conf
```

設定オプションの詳細については、[Unicornのドキュメント](#)をご参照ください。

### Apache で Passenger を使用した Sunstone の実行

**警告：**OpenNebula 5.10以降、必要なRuby gemはすべてパッケージ化され、専用ディレクトリ

/usr/share/one/gems-dist/ にパッケージ化され、インストールされます。このディレクトリは /usr/share/one/gems/ へのシンボリックリンクです。[詳細はフロントエンドのインストールをご確認ください。](#)

シンボリックリンクされた場所が保持されている場合、同梱のRuby gemsが排他的に使用されます。ウェブサーバー内で動作するRubyに専用の場所を使用させるためには、GEMS\_HOMEおよびGEMS\_PATH環境変数を設定する必要があるかもしれません。

例えば、Apacheの設定ファイルに以下の設定を追加することで実現できます：

Phusion Passengerは、ApacheおよびNginxウェブサーバー向けのモジュールであり、Ruby Rackアプリケーションを実行します。これはSunstoneサーバーの実行に使用でき、そのライフサイクル全体を管理します。これらのサーバーのいずれかを既に使用している場合、またはいすれかに慣れている場合は、この方法の使用をお勧めします。この種のデプロイメントは、より優れた同時実行性を提供し、httpsエンドポイントの追加を可能にします。

Apache Webサーバー向けの設定手順をご提供いたしますが、Nginxの場合もPassengerの公式ドキュメントに従えば同様の手順となります。

まず最初に行うべきことは、Phusion Passenger のインストールです。これには、お使いのディストリビューション向けの事前構築済みパッケージをご利用いただくか、公式ウェブページに記載されているインストール手順に従ってください。インストール手順は分かりやすく説明されており、全プロセスをガイドします。指示に従って進めれば、Sunstone を実行する準備が整います。

次に、Sunstoneサーバーを稼働させる仮想ホストの設定を行います。

publicディレクトリを指すように設定します。例を以下に示します：

```
<VirtualHost *:80>ServerName
    sunstone-serverPassengerUser
    oneadmin

    OpenNebula 5.10 以降では、専用ディレクトリを設定する変数

    # 付属の Ruby gems 用の専用ディレクトリを設定する変数は、これらの gems が明示的に # 無効化されない
    # い場合（指定されたディレクトリのシンボリックリンクを削除することで）、設定する必要があります。
    SetEnv GEM_PATH /usr/share/one/gems/SetEnv
    GEM_HOME /usr/share/one/gems/

    # !!! DocumentRoot は必ず 'public' を指定してください！
    DocumentRoot /usr/lib/one/sunstone/public
    <Directory /usr/lib/one/sunstone/public>
        # これにより Apache のセキュリティ設定が緩和されます。
        AllowOverride all
        # MultiViews は無効にする必要があります。
        Options -MultiViews
        Apache 2.4 以降をご利用の場合、以下のコメントを外してください：
        #Require all granted
        OpenNebula 5.6.0未満をご利用の場合は、以下の行をコメントアウトしてください:
        Options FollowSymLinks
    </Directory>
</VirtualHost>
```

---

**注記：**仮想ホストにFollowSymLinksオプションを追加することは必須です。

---



---

**ご注意：**ログインに問題が生じた場合、Sunstoneは複数サーバーインスタンス間でのセッションをサポートしておりませんため、Passengerの設定でPassengerMaxInstancesPerAppを1に設定するか、memcachedのご利用をお試しください。

---

設定はこれで完了しているはずです。アプリケーションを起動するため、Apacheの設定を再起動（または再読み込み）し、仮想ホストにアクセスして正常に動作しているかご確認ください。

### nginx SSLプロキシ経由でのSunstoneの実行

Sunstoneと暗号化VNC用にnginx SSLプロキシを設定する方法：

```
# サーバートークンを無効化しま
す。server_tokens off;

# OpenNebula Sunstone 上流設定 upstream
sunstone {
    server 127.0.0.1:9869;
}

# HTTP 仮想ホスト、HTTPS へのリダイレクト server {
    listen 80 default_server;
    return 301 https://$server_name:443;
```

```

}

# HTTPS 仮想ホスト、Sunstone サーバーへのプロキシ {
listen 443 ssl default_server;
ssl_certificate           /etc/ssl/certs/opennebula-certchain.pem;
ssl_certificate_key       /etc/ssl/private/opennebula-key.pem; ssl_stapling on;
}

```

また、`sunstone-server.conf` には以下の変更が必要です：

#### UI設定

```

:vnc_proxy_port: 29876
:vnc_proxy_support_wss: only
:vnc_proxy_cert: /etc/one/ssl/opennebula-certchain.pem
:vnc_proxy_key: /etc/one/ssl/opennebula-key.pem
vnc_proxy_ipv6: false

```

自己署名証明書をご利用の場合、SunstoneからのVNCウィンドウへの接続は失敗いたします。正規の証明書を取得いただぐか、Sunstoneで試行する前にブラウザで手動にて自己署名証明書を承認してください。これでVNCセッションのタイトルに「暗号化済み」と表示されるはずです。OpenNebulaのIPアドレスまたは完全修飾ドメイン名（FQDN）において、443ポートと29876ポートの両方について、お使いのブラウザがその証明書を信頼するように設定する必要があります。

#### ApacheでFreeIPA/Kerberos認証を使用したPassengerによるSunstoneの実行

Sunstoneリモート認証をApacheおよびPassengerと併用することも可能です。この場合の構成はPassengerの設定と非常に似ていますが、Apacheのauthモジュール行を含める必要があります。FreeIPAサーバーとKerberosの設定方法については、本ドキュメントの範囲外となります。詳細はFreeIPAのApache設定例をご参照ください。

例えば、Kerberos認証を追加するには、`mod_auth_gssapi`または`mod_authnz_pam`のいずれかを使用し、http サービス用のキータブを生成します。以下に Passenger を使用した例を示します：

```

LoadModule auth_gssapi_module modules/mod_auth_gssapi.so

<VirtualHost * :80> ServerName
    sunstone-server PassengerUser
    oneadmin

    # OpenNebula 5.10 以降では、専用ディレクトリを設定する変数
    # これらのジェムが明示的に無効化されていない場合（指定されたディレクトリのシンボリックリンクを削除することで）、設定する必要があります。
    SetEnv GEM_PATH /usr/share/one/gems/SetEnv
    GEM_HOME /usr/share/one/gems/
    # !!! DocumentRoot は必ず 'public' を指定してください！
    DocumentRoot /usr/lib/one/sunstone/public
    <Directory /usr/lib/one/sunstone/public>
        このディレクトリには、有効なチケットを使用してのみアクセスが可能です。
        AuthType GSSAPI
        認証名 "EXAMPLE.COM ログイン"GssapiCredStore
        keytab:/etc/http.keytab有効なユーザーを要求します
        ErrorDocument 401 '<html><meta http-equiv="refresh" content="0; URL=https://
        ↳yourdomain"><body>Kerberos認証に失敗しました。</body></html>'
        AllowOverride all
        # マルチビューは必ず無効にしてください。

```

```
Options -MultiViews  
</Directory>  
</VirtualHost>
```

---

**注記：**ユーザーは、Sunstoneサービスにアクセスするために、`kinit`を実行して有効なチケットを生成する必要があります。また、認証失敗時にユーザーに警告するカスタム401ドキュメントを設定することも可能です。

---

これで Passenger と Kerberos を使用する設定が完了しました。Apache の設定を再起動または再読み込みし、有効なチケットを使用して仮想ホストにアクセスし、すべてが正常に動作しているかご確認ください。

#### 複数のサーバーでの Sunstone の実行

Sunstoneは複数のサーバーで実行し、それらに接続するロードバランサーをご利用いただけます。セッションには`memcache`を使用し、両方のSunstoneサーバーが同一のmemcachedサーバーに接続していることをご確認ください。これを行うには、設定ファイル内の`:memcache_host`パラメータを変更してください。また、両方のSunstoneインスタンスが同一のOpenNebulaサーバーに接続していることもご確認ください。

#### マーケットプレイス

MarketPlaceAppのダウンロード機能をご利用になる場合、SunstoneサーバーはMarketPlaceバックエンドへのアクセス権限が必要となります。

Phusion Passengerをご利用の場合は、以下の推奨事項をご考慮ください：

- `PassengerResponseBufferHighWatermark` を `0` に設定してください。
- `PassengerMaxPoolSize` を増やしてください。MarketPlaceApp のダウンロードごとに、これらのアプリケーションプロセスが 1 つ消費されます。
- Passenger Enterpriseが利用可能な場合、`PassengerConcurrencyModel`を`thread`に設定してください。

Passenger以外のバックエンドをご利用の場合は、これらの推奨事項を該当するバックエンドに適用してください。

## VMWARE インフラストラクチャの設定

### 6.1 概要

OpenNebula フロントエンドと vCenter ノードの設定が完了しましたら、次のステップとして、vCenter インフラストラクチャから活用できる機能を把握し、OpenNebula クラウドを微調整してそれらを活用する方法について学びます。

仮想化サブシステムは、ハイパーバイザとの通信を担当し、仮想マシンのライフサイクルの各段階に必要なアクションを実行するコンポーネントです。本章では、vCenter ドライバーの詳細、vCenter が管理するリソース、および OpenNebula を設定してさまざまな vCenter 機能を活用する方法について説明します。

#### 6.1.1 本章の読み方について

この章は、[vCenter ノードのインストール](#)を完了した後にご一読ください。

本章は、以下のセクションで構成されています：[vCenter ドライバーセクション](#)：OpenNebula の観点から vCenter 統合アプローチを紹介し、VM テンプレートのインポート・作成・使用方法、リソースプール、制限事項などについて説明します。次に、[ネットワーク設定](#)セクションでは、OpenNebula がネットワークを消費または作成する方法、およびそれらのネットワークの使用方法について概説します。[データストア設定セクション](#)では、vCenter データストアに関する OpenNebula データストアの概念と、OpenNebula による VMDK イメージ管理について紹介します。[NSX 設定セクション](#)では、OpenNebula と NSX コンポーネントの統合について説明します。

本章をお読みいただいた後は、[リファレンス章](#)にて OpenNebula のアップグレード、ロギング、スケーラビリティといった高度なトピックについてさらに深く学ぶことができます。次のステップとしては、運用ガイドに進み、クラウドユーザーが設定済みのクラウドリソースをどのように利用するかを学ぶことをお勧めいたします。

#### 6.1.2 ハイパーバイザの互換性

本章の内容はすべて、vCenter ハイパーバイザにのみ適用されます。

### 6.2 vCenter ドライバーの設定

vCenter ドライバーは、OpenNebula と VMware ベースのインフラストラクチャとのすべての統合を担当しております。OpenNebula と vSphere 間のすべての相互作用は、vCenter API を介して行われます。ただし、VNC コンソール接続に関しては例外であり、Sunstone

サーバー（より具体的にはnoVNCサーバー）がESXハイパーバイザと直接通信いたします。

本セクションでは、VMwareリソースをOpenNebulaクラウドに組み込むために必要な手順を説明します。

## 6.2.1 vCenter インポートツール

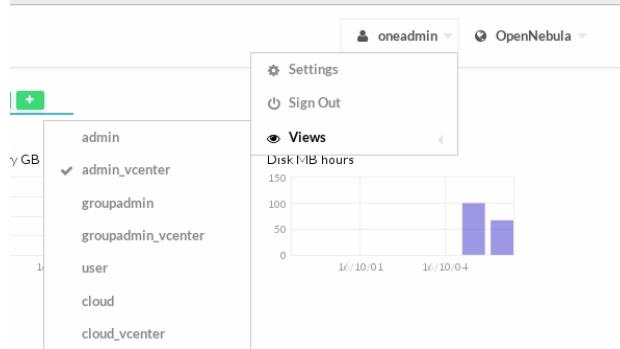
vCenter クラスタ、VM テンプレート、ネットワーク、データストア、および vCenter データストア内の VMDK ファイルは、OpenNebula に簡単にインポートできます。

- コマンドラインインターフェースから **onevcenter** ツールを使用します

```
$ onevcenter <コマンド> -o <オブジェクトタイプ> -h <opennebulaホストID> [<オプション>] [<引数>]
```

- Sunstoneのインポートボタンをご利用ください。

Sunstoneでadmin\_vcenterビューが有効になると、「インポート」ボタンが利用可能になります。有効化するには、ユーザー名（Sunstoneの右上）をクリックしてください。ドロップダウンメニューが表示されますので、「ビュー」を選択し、最後に「admin\_vcenter」をクリックしてください。



**警告：**画像のインポート操作には長時間かかる場合がございます。Sunstoneクライアントをご利用中に「サーバーに接続できません：サーバーは稼働中かつ到達可能ですか？」というメッセージが表示された場合、Sunstoneの30秒タイムアウトに達した可能性があります。この場合は、SunstoneをApache/NGINXの背後で動作するように設定するか、代わりにCLIツールをご利用ください。

以下のvCenterリソースは、OpenNebulaへ簡単にインポートできます：

- vCenterクラスター（OpenNebulaホストとしてモデル化されます）
- データストア
- ネットワーク
- VMテンプレート
- ワイルドVM（OpenNebula外部で起動された仮想マシン）
- イメージ

## 6.2.2 vCenterクラスタのインポート

vCenterクラスタは、他のすべてのvCenterリソースがこれに依存するため、vCenterインストールに最初に追加すべきものです。OpenNebulaはこれらのクラスタをOpenNebulaホストとしてインポートするため、Sunstone（インフラストラクチャ/ホスト）または

CLI (onehost) を使用して簡単に監視できます。また、認証が必要な唯一のステップであるため、プロセスが正常に完了することを確認することが重要です。

#### **onevcenter を使用したクラスタのインポート**

vCenterクラスタのインポート先を選択すると、OpenNebulaは当該vCenterクラスタを表すOpenNebulaホストを作成いたします。また、そのOpenNebulaホストをどのOpenNebulaクラスタにグループ化するかについても、OpenNebulaに指定していただく必要がございます。

ホストをどのOpenNebulaクラスタにグループ化するか指定する必要があります。既存のクラスタを選択しない場合、デフォルトではOpenNebulaが自動的にOpenNebulaクラスタを作成します。vCenterクラスタは、データセンターの異なるフォルダやサブフォルダ内で同じ名前を持つ場合があります。OpenNebulaはvCenterクラスタの場所を通知しますので、それによって識別できます。

以下にサンプルセクションを示します：

```
$ onevcenter hosts --vcenter <vcenter-host> --vuser <vcenter-username> --vpass
-<vcenter-password>
```

vCenterへの接続中: vcenter.host...完了しました！vCenterリソースの探索中

...完了しました！

データセンター Datacenter を処理しますか (y/[n])? y

\* vCenterクラスタが見つかりました:

- 名前 : Cluster2
- 場所 : /

クラスタをインポートしますか (y/[n])? y

vCenterクラスターをどのOpenNebulaクラスターに含めますか？

- ID: 100 - NAME: Cluster
- ID: 101 - 名前: Cluster3

クラスタのIDを指定するか、OpenNebulaが新しいクラスタを作成する場合Enterキーを押してください

←新しいクラスターを作成します:

OpenNebulaホストCluster2 (ID 2) の作成が正常に完了しました。

**注記:** vCenter がデフォルトポート以外のポートを使用している場合、-port コマンドをご利用いただけます。

### Sunstone からのクラスタのインポート

Sunstone からクラスタをインポートすることも可能です。インフラストラクチャメニューの「ホスト」をクリックし、プラス記号をクリックすると、新しいウィンドウが開きます。



新しいウィンドウで、[タイプ] ドロップダウンメニューから「VMWare vCenter」を選択してください。

vCenter のホスト名（デフォルト以外のポートの場合は<サーバー名>:<ポート番号> の表記を使用できます）または IP アドレス、

および vCenter インスタンスの管理に使用する認証情報を入力し、「クラスターを取得」をクリックしてください。

The screenshot shows the 'Create Host' page. At the top, there are 'Type' (VMWare vCenter) and 'Cluster' (0:default) dropdown menus. Below them is a 'vCenter' section containing 'Hostname' (redacted), 'User' (redacted), and 'Password' (redacted). At the bottom are 'Get Clusters' and 'Import' buttons.

vCenterの認証情報を入力すると、まだインポートされていないvCenterクラスタの一覧が表示されます。vCenterクラスタの名前と、vSphereの「ホストとクラスタ」ビュー内でのそのクラスタの場所が確認できます。

**注記：**vCenter クラスタは、そのクラスタの moref および vCenter インスタンスの UUID が OpenNebula のイメージプール内で見つからない場合、インポートされていないものとみなされます。

OpenNebulaが新規クラスターを検出した場合、それらは所属するデータセンターごとにグループ化されます。

The screenshot shows the 'Datacenter DataCenter' page with a table of clusters. A single cluster is selected (checked). The columns are 'Cluster', 'vCenter ref', and 'Location'. The selected cluster has 'vCenter ref: domain-c14042' and 'Location: /'. Other clusters listed are 'Cluster2' (domain-c1752, /), 'Cluster with spaces' (domain-c7, /), and another unnamed cluster (domain-c7, /). A pagination bar at the bottom shows 'Showing 1 to 3 of 3 entries'.

インポートするvCenterクラスタを1つ以上選択する前に、ドロップダウンのクラスタメニューからOpenNebulaクラスタを選択できます。デフォルトのデータストア（ID:0）を選択した場合、OpenNebulaが新しいOpenNebulaクラスタを作成いたします。

The screenshot shows a dropdown menu labeled 'Cluster' with the option '0: default' selected.

インポートしたいvCenterクラスタを選択し、最後に「インポート」ボタンをクリックしてください。インポートツールが完了すると、vCenterクラスタの表現として作成されたOpenNebulaホストのIDが表示されます。

The screenshot shows the 'Datacenter DataCenter' page after import. Two clusters are selected. The table includes a column for status messages: 'Host created successfully. ID: 4' for Cluster2 and 'Host created successfully. ID: 3' for the unnamed cluster. The status bar at the bottom indicates 'Showing 1 to 3 of 3 entries'.

vCenterクラスタを表すホストの名前にはクラスタ名が含まれており、以前にインポートされたvCenterクラスタと名前が衝突する場合は、衝突を回避するために文字列が追加されます。また、デフォルトのデータストアを選択した場合、OpenNebulaはインポートされたvCenterクラスタと同じ名前の新しいOpenNebulaクラスタを割り当てます。

The screenshot shows a table of hosts. The columns are 'ID', 'Name', 'Cluster', 'RVMs', 'Allocated CPU', 'Allocated MEM', and 'Status'. There are two entries: one for 'Cluster2' (ID 4) and one for 'Cluster' (ID 3). Both hosts have 0 RVMs, 0/1000 MHz allocated CPU, and 0/31.8GB(0%) allocated MEM. Both are in an 'ON' status.

なお、vCenterクラスタを表すOpenNebulaホストを削除した後、再度インポートを試みると、以下のようなエラーが発生する可能性があります。



その場合は、クラスタドロップダウンメニューから正しいクラスタを指定するか、OpenNebulaクラスタを削除してください。そうすることで、vCenterクラスタのインポート時にOpenNebulaが自動的にクラスタを再作成できるようになります。

---

**注記：**OpenNebulaはvCenterクラスターをOpenNebulaホストとして認識し、新しいvCenterクラスターをOpenNebulaホストとしてインポートすると、同時にOpenNebulaクラスターも作成される点にご留意ください。当該vCenterクラスターの全リソース（ネットワークおよびストレージ）は、自動的に同じOpenNebulaクラスターへインポートされます。

---

**注記：**ホストテンプレート内でVM\_PREFIX属性を定義できます。この属性は、このホストでVMをインスタンス化する際に、すべてのVM名がVM\_PREFIXで始まるこことを意味します。

---

### 6.2.3 vCenterリソースのインポート

vCenterクラスタのインポートが完了しましたら、残りのvCenterリソースをインポートいただけます。この際、認証処理はインポート済みのOpenNebulaホストに委譲されます。重要な点として、インポートしたホストが正常に動作していることを必ずご確認ください。動作確認が取れない場合、当該ホストの認証情報を使用したリソースのインポートは一切行えません。

インポートツールは、Sunstoneとコマンドラインインターフェースの両方で同様の方法で動作します。残りのリソースをインポートするためには、少なくとも1つのvCenterクラスターが既に稼働していることが必須条件となります。また、イメージなどの場合、適切なデータストアを事前にインポートしておく必要があります。ネットワークやデータストアなどのリソースは複数のクラスターに属する可能性があるため、ツールはそのような状況について警告を表示いたします。

OpenNebulaによるvCenterリソースの作成は、以下の2段階に分けてご説明いたします：

- vCenterサーバーおよびご希望のリソースの種類に関する具体的な情報を取得するには、**以下のコマンドを実行してください**：
  - [CLI] onevcenter list -o <リソースタイプ> -h <ホストID> [追加情報] を使用します。
  - [Sunstone] Sunstoneの該当セクションに移動し、インポートボタンをクリックして適切なホストを選択してください。

これにより、インポート可能なオブジェクトの一覧が表示され、いくつかの情報が提供されます。

- 最初のステップで収集した情報に基づき、選択したリソースを**インポートします**：

– [CLI] onevcenter import <対象オブジェクト> -o <リソースタイプ> -h <ホストID> [追加情報] を実行します。

この操作にはいくつかの方法がありますが、このリストでは、REF列に加えて、インポートされていないリソースを整理するID列が表示されます。両方の列を使用して特定のリソースを選択することができます  
：

コマンド (例)	注記
onevcenter import ref	これにより、リファレンス refを持つリソースがインポートされます。
onevcenter import 0	これにより、リストに表示されている最初のリソース、つまり IM_ID 0 のリソースをインポートします
onevcenter import ‘ref0, ref1’	これにより、ref0 および ref1 の両方の項目がインポートされます

onevcenter import 0.5

これにより、IM\_ID 0、1、2、3、4、5 のアイテムがインポートされます

- [Sunstone] 前のリストから必要なリソースを選択してください（任意のオプションをチェックするだけで結構です）

### デフォルト設定すべてのリソースをインポートします

特定の状況では、vCenterサーバーの全リソースをインポートする必要が生じます。ネットワークなどの一部リソースは、設定オプションが必要なため「対話型」インターフェースを備えています。この特性により、自動インポートは管理者にとって困難な作業となります。しかしながら、Sunstoneおよびonevcenterの両ツールでは、vCenterホストの全リソースをデフォルト設定で一括インポートすることができます。これにより、全リソースのインポート作業が容易になります。

- [CLI] onevcenter import\_defaults コマンドの使用方法:

```
onevcenter import_defaults -o datastores -h 0
```

これにより、ID: 0 のインポート対象 OpenNebula ホストに関連するすべてのデータストアがインポートされます。

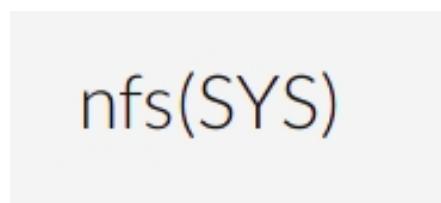
- [Sunstone] テーブルの隅にある最初のチェックボックスをクリックしてください。

### 6.2.4 vCenter データストアのインポート

vCenterの仮想マシンおよびテンプレートに接続される仮想ハードディスクは、OpenNebulaではイメージとして表現される必要があります。イメージはOpenNebulaのイメージデータストアに配置する必要があります、インポートツールにより簡単に作成できます。vCenterのデータストアは、onevcenterツールまたはSunstoneユーザーインターフェースを使用してインポートできます。

インポートツールを実行すると、OpenNebulaは各データセンター上で検出されたデータストアに関する情報を提供します。具体的には、データストア名、データストアの容量、およびvCenterデータストアを割り当て可能なOpenNebulaクラスタのIDです。OpenNebulaクラスタのIDが表示されない場合、このデータストアを使用するvCenterクラスタをまだインポートしていないことを意味します。vCenterデータストアのインポート前にvCenterクラスタをインポートすることは必須ではありませんが、後ほどOpenNebulaクラスタにデータストアを割り当てる必要が生じる可能性があります。これにより、OpenNebulaの仮想マシンやテンプレートがそのデータストアを利用できるようになります。

vCenterのデータストアはデータセンター内で一意であるため、異なるデータセンターやvCenterインスタンス間で同じ名前のデータストアが存在する可能性があります。データストアをインポートする際、OpenNebulaは衝突を回避する名前を生成します。この名前には、データストア名、括弧内にデータストアの種類、および同名のデータストアが存在した場合の接尾辞が含まれます。この名前は、データストアのインポート後に、より分かりやすい名前に変更することができます。以下にサンプル名を示します：



インポートされたデータストアに関して、重要な点がございます。vCenterデータストアをインポートする際、OpenNebulaはvCenterのホスト名またはIPアドレス、vCenterユーザー名、vCenterパスワード（暗号化済み）をデータストアテンプレート定義内に保存いたします。これはOpenNebulaがvCenter上でAPI操作を実行するために、これらの認証情報が必要となるためです。したがって、OpenNebulaからvCenter接続用のユーザー名やパスワードを変更される場合は、データストアテンプレートを編集し、該当するユーザー名および/またはパスワードを変更する必要があります（パスワードは平文で入力可能ですが、OpenNebulaは暗号化して保存します）。

VCENTER_HOST	vcenter.vcenter3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
VCENTER_PASSWORD	6TRPipwZQAA==	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
VCENTER_USER	administrator@vsphere.local	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

#### 6.2. vCenter ドライバーの設定

141

**警告：**データストアをインポートするには、事前にvCenterクラスタをインポートしておく必要があります。そうしない場合、OpenNebulaは関連するvCenterクラスタのインポートが必要であるとエラーを表示します（前の項目を参照）。



### 1つのvCenterを使用したデータストアのインポート

コマンドラインインターフェースを使用してデータストアをインポートする方法を示す例を以下に示します。まず、ID 0 の vCenter クラスタが既にインポートされている状態です。

onevcenter list -o datastores -h 0			
# vCenter: vCenter.server			
IMID	REF	名前	クラスター
0	データストア-15	データストア2	[102]
1	データストア-11	データストア1	[]
2	データストア-15341	データストア1 (1)	[100]
3	データストア-16	NFS	[102, 100]

インポートツール（リスト）は、各データセンター内のデータストアを検出します。検出されたデータストアには、データストア名、容量、およびこのデータストアが追加されるOpenNebulaクラスタIDが表示されます。

インポートしたいデータストアが確定しましたら：

```
onevcenter import datastore-16 -o datastores -h 0 ID: 100
ID: 101
```

データストアを選択すると、OpenNebula には同じデータストアの 2 つの表現が作成されます。IMAGE データストアと SYSTEM データストアです。そのため、2 つのデータストアが作成されたことが確認できます（ただし、データストアが StorageDRS の場合は、SYSTEM データストアのみが作成されます）。

### Sunstone を使用してデータストアをインポートする

Sunstoneにおいて、ストレージメニュー内の「データストア」をクリックし、続いて「インポート」ボタンをクリックしてください。新しいウィンドウが開きます。



新しいウィンドウで、このvCenterインスタンスへの認証に使用するクラスターを選択し、「データストアを取得」をクリックしてください。

ID	Name	Cluster	RVMs	Allocated CPU	Allocated MEM	Status
4	Cluster2	103	0	0 / 1000 (0%)	0KB / 31.8GB (0%)	ON
3	Cluster	102	0	0 / 1000 (0%)	0KB / 31.8GB (0%)	ON

「データストアを取得」ボタンをクリックすると、データストアの一覧が表示されます。データストア名、容量、およびデータストアが割り当てられる既存のOpenNebulaクラスタのIDが表示されます。ご注意ください。OpenNebulaクラスタIDの列が空欄の場合、インポートツールがデータストアをグループ化できるOpenNebulaクラスタを見つけられなかったことを意味します。後で手動で割り当てる必要があるか、データストアインポートツールの操作をキャンセルし、vCenterクラスタを先にインポートすることをお

---

試してください。

OpenNebula は、まだインポートされていないデータストアを検索します。

vCenter Datastores: vcenter:vcenter65-1						
Select all 3 Datastores						
	Name	vCenter ref	Datacenter	Total MB	Free MB	OpenNebula Cluster IDs
	datastore2	datastore-15	Datacenter	946176	930353	103
	datastore1 (1)	datastore-15341	Datacenter	946176	583734	102
	nfs	datastore-16	Datacenter	4691989	1675846	103,102

リストからインポートしたいデータストアを選択し、最後に「インポート」ボタンをクリックしてください。データストアを選択して「インポート」ボタンをクリックすると、作成されたデータストアのIDが表示されます：

Datacenter DataCenter						
Clear Imported Datastores						
2 Datastores selected. Select all 4 Datastores						
	Name	Total MB	Free MB	OpenNebula Cluster IDs		
	datastore2	132352	129304	128		
<input checked="" type="checkbox"/>	datastore1	230656	32433	128	Datastores created successfully! IDs: 337,340	
	DRSDatastoreCluster	132352	129304	128		
	nfs	4691989	1562932	128	Datastores created successfully! IDs: 338,339	

データストア一覧では、データストア名をご確認いただけます。また、括弧内にSYSTEMデータストアの場合は「SYS」、IMAGEデータストアの場合は「IMG」、StorageDRSクラスターの表現の場合は「StorDRS」と表示されます。データストアはインポート後に名前を変更できる点にご留意ください。最後に、OpenNebulaクラスタID列にIDが表示されている場合、データストアはOpenNebulaクラスタにも追加されます。

107	nfs(SYS)	oneadmin	oneadmin	102,103	SYSTEM	ON	
106	nfs(IMG)	oneadmin	oneadmin	OKB/-	102,103	IMAGE	ON
105	datastore1 (1)(SYS)	oneadmin	oneadmin	102	SYSTEM	ON	
104	datastore1 (1)(IMG)	oneadmin	oneadmin	OKB/-	102	IMAGE	ON

## 6.2.5 vCenter VMテンプレートのインポート

onevcenterツールおよびSunstoneインターフェースを使用することで、vCenterから既存のVMテンプレートをインポートすることが可能です。

**重要：**この手順は、前述の通りテンプレートのハードディスクファイルが配置されているデータストアをインポートした後に行なってください。

**重要：**テンプレートをインポートする前に、仮想ハードディスクをホストするデータストアが監視対象となっており、そのサイズと使用状況情報が報告されることをご確認ください。データストアは監視対象になっていないと、イメージを作成できません。

インポートツール（onevcenter ツールまたは Sunstone のいずれか）はテンプレートに関する情報を提供します。テンプレートを選択してインポートする際、OpenNebula は仮想ディスクおよび仮想ネットワークインターフェイスカードを検索するためにテンプレートを検査することに留意してください。

vCenterテンプレートをインポートする前に、そのテンプレートで使用されているvCenterデータストアを必ずインポートしてください。OpenNebulaでは、検出された仮想ディスクを表すイメージを格納するためのIMAGEデータストアが必要となります。OpenNebulaがデータストアを検出できない場合、インポート操作は失敗します。

ERROR: datastore nfs: has to be imported first as an image datastore!

### **OpenNebula 5.10 導入ガイド、リリース 5.10.5**

OpenNebulaは、検出されたディスクを表すOpenNebulaイメージと、仮想NICが使用するポートグループを表すOpenNebula仮想ネットワークを作成します。例えば、3つのディスクと1つのNICを持つテンプレートがある場合、

仮想マシンネットワークポートグループに接続されています。

▶ Hard disk 1	70,00 MB
▶ Hard disk 2	10,00 MB
▶ Hard disk 3	5,00 MB
▶ Network adapter 1	VM Network (disconnected)

実際、インポート操作が完了すると、テンプレート内に存在する各仮想ディスクを表す3つのイメージが生成されます。イメージ名はOpenNebulaによって生成され、ファイル名、保存先のデータストア、OpenNebulaのテンプレートIDが含まれています。これにより、どのイメージがどのテンプレートに関連付けられているかを容易に把握できます。なお、これらのイメージは非永続的です。イメージ名はインポート後に変更可能です。

278	corelinux7_x86_64-5-000001-rfs	oneadmin	oneadmin	rfs[vcentercenter3-Datacenter]	OS	READY	0
277	corelinux7_x86_64-3-000001-rfs	oneadmin	oneadmin	rfs[vcentercenter3-Datacenter]	OS	READY	0
276	corelinux7_x86_64-4-000001-rfs	oneadmin	oneadmin	rfs[vcentercenter3-Datacenter]	OS	READY	0

また、仮想ネットワークも作成されます。その名称はvCenterと同じになります。なお、この仮想ネットワークはvCenterクラスターを含むOpenNebulaクラスターに自動的に追加されます。例えば、以下のスクリーンショットでは、このネットワークは2つのOpenNebulaクラスター（100、101）に属しています。



vCenterテンプレートの名前はフォルダ内でのみ一意となりますため、データセンター内の異なるフォルダに同じ名前のテンプレートが2つ存在する場合があります。OpenNebulaが衝突を検出した場合、OpenNebula内の名前重複を防止するため、VMテンプレートの特性に基づくSHA1ハッシュ演算に基づく文字列を名前に付加します。OpenNebula内のVMテンプレート名は、インポート後に変更が可能です。以下のスクリーンショットに例を示します：



### onevcenterを使用したVMテンプレートのインポート

onevcenterツールを使用した手順は以下の通りです。

```
$ onevcenter list -o templates -h 0
# vCenter: vcenter.Server

IMID REF          NAME
0  vm-8720        corelinux7_x86_64 (スペース付き)
1  vm-9199        one-corelinux7_x86_64
2  vm-8663        dist_nic_test
```

この例では、vcenter.serverには3つのテンプレートがあり、それらはIM\_ID = 0から2までリストされています。インポートの準備が整いましたら、以下のコマンドを実行してください：

```
onevcenter import vm-1754 -o templates -h 0
- テンプレート: corelinux7_x86_64
```

このセクションでは、いくつかの質問が表示され、回答内容に応じて異なる処理が行われます。まず最初に、リンクドクローンを使用するかどうかを確認するプロンプトが表示されます。

このテンプレートに基づく仮想マシンでリンクドクローンを使用しますか (y/[n]) ?

前述の通り、テンプレートでリンクドクローンを使用したい場合は、元のテンプレートをそのまま残すために、テンプレートのコピーを作成することができます。

OpenNebula にテンプレートのコピーを作成させ、元のテンプレートを  
←元のテンプレートはそのまま残りますか ([y]/n) ?

テンプレートのコピーを作成したい場合は、新しい名前を付けるか、元の名前の一部に「one-」という接頭辞を付けて同じ名前を使用することができます。

新しいテンプレートは、元のテンプレートの名前に「one-」という接頭辞を付けて命名されます。

←テンプレート

別の名前をご希望の場合はご指定いただかず、Enterキーを押してデフォルトを使用してください：

←corelinux7\_linked\_x86\_64

テンプレートのコピーを使用する場合、テンプレートとそのディスクの完全なクローン作成が必要となるため、この操作には時間がかかる場合がございます。

警告！！！ ディスクのサイズによっては、複製操作に時間がかかる場合がございます。

お待ちください...

リンクドクローンを使用する場合、差分ディスクが作成されます。この処理にも時間がかかります。

差分ディスクの作成中です。少々お待ちください...

リンクドクローンを使用するか否かにかかわらず、このテンプレートに基づく仮想マシンをvSphereの「仮想マシンとテンプレート」インベントリに表示するフォルダを選択できます。

このテンプレートに基づいて展開された仮想マシンを表示するフォルダを指定されますでしょうか？

←vSphereの「仮想マシンとテンプレート」セクションに表示されるフォルダを指定しますか？

パスが設定されていない場合、仮想マシンはテンプレートが存在する場所と同じ場所に配置されます。スラッシュ（例：/Management/VMs）でフォルダを区切ってパスを指定するか、

←Enter キーを押すとデフォルトが使用されます：

OpenNebula は vCenter テンプレートを検査し、テンプレートに関連付けられた仮想ディスクおよび仮想ネットワーク用のイメージとネットワークを作成します。これらの処理には完了までしばらく時間がかかります。

テンプレート内の既存ディスクおよびネットワークのインポートが進行中です。少々お待ちください。

テンプレートはほぼ準備が整っておりますが、リソースプールを指定するか、ユーザーが使用するリソースプールを選択できるリストを提供いただくことが可能です。

デフォルトでは、vCenterクラスタを表すホストに対して特定のリソースプールが設定されていない限り、OpenNebulaはデータセンター内で最初に利用可能なリソースプールを使用します。以下の内容を完全に理解するため、本章の「OpenNebulaにおけるリソースプール」セクションをまだご覧になっていない場合は、ぜひご参照ください。

このテンプレートは現在、デフォルトのリソースプールで仮想マシンを起動するよう設定されています。

この動作を維持する場合は y、新しいリソースプールを選択する場合は n、選択権をユーザーに委譲する場合は d を入力してください。

←選択をユーザーに委ねるかどうかの確認です ([y]/n/d) ?

新しいリソースプールを選択される場合は、利用可能なリソースプールのリストが表示されますので、その中から一つお選びください：

利用可能なリソースプールのリストは以下の通りです：

- TestResourcePool/NestedResourcePool
- TestResourcePool

新しいデフォルトのリソースプール名を入力してください:

ユーザーが選択できるリソースプールのリストを作成したい場合、インポートツールが生成したリストを受け入れるか、値をカンマで区切ってリソースプールの参照を入力する機会があります:

ユーザーに提示される利用可能なリソースプールのリストは以下の通りです:

→ "TestResourcePool/NestedResourcePool, TestResourcePool"

編集に同意される場合は「y」を押してください。または、編集対象のリソースプールをカンマ区切りでリスト入力してください ([y] /  
→カンマ区切りリスト)

リストを選択された場合、そのリスト内のデフォルトのリソースプールの参照を選択するよう求められます:

エンドユーザーに提示されるデフォルトのリソースプールは 「TestResourcePool/

→NestedResourcePool」 に設定されています。

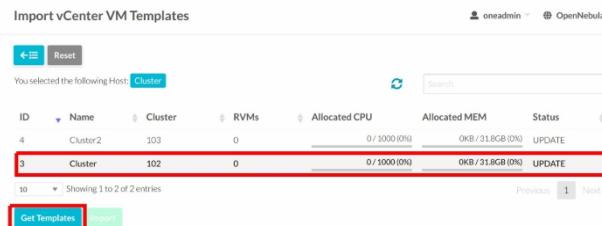
同意する場合は y を押すか、新しいリソースプールを入力してください ([y] /リソースプール名)

### Sunstone を使用して VM テンプレートをインポートする

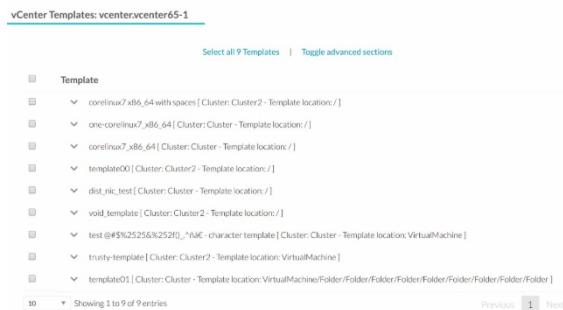
Sunstone にて、[テンプレート] メニュー内の [VM] をクリックし、[インポート] ボタンをクリックすると、新しいウィンドウが開きます。



新しいウィンドウで、この vCenter インスタンスへの認証に使用するクラスターを選択し、「Get Templates」をクリックしてください。



OpenNebula は、まだインポートされていないテンプレートを検索します。



テンプレートをインポートする前に、テンプレート名の横にある下向き矢印をクリックし、本章の「OpenNebulaにおけるリソースプール」セクションで説明した通り、リソースプールを指定することができます。



**注記：**vCenterクラスタでDRSが有効化されていない場合、リソースプールはご利用いただけません。そのため、下矢印には一切の内容が表示されません。

インポートしたいテンプレートを選択し、最後に「インポート」ボタンをクリックしてください。OpenNebulaはテンプレート内に存在するディスクとネットワークインターフェースをインポートし、それらを表現するイメージとネットワークを作成するため、この処理には時間がかかる場合があります。



テンプレートのインポートが完了すると、テンプレートのIDが表示されます。



**注記：**vCenterテンプレートは、そのテンプレートのmorefおよびvCenterインスタンスのuuidがOpenNebulaのテンプレートプール内で見つからない場合、インポートされていないものとみなされます。

**警告：**OpenNebulaが新しいテンプレートを見つけられない場合は、それらのテンプレートを含むvCenterクラスタを事前にインポート済みであることをご確認ください。

**警告：**テンプレートでリンククローンを使用したい場合は、前のセクションで説明したように `onevcenter` ツールを使用してインポートしてください。

**注記：**vCenterテンプレート内の仮想ディスクを表すイメージが作成されると、VCEN-TER\_IMPORTED属性が自動的にYESに設定されます。この属性により、OpenNebulaからイメージが削除された場合でも、OpenNebulaがvCenterデータストアからファイルを削除することを防止します。

vCenter VMテンプレートがOpenNebula VMテンプレートとしてインポートされた後、CPUやメモリの容量、名前、権限などを変更するために修正することができます。また、以下の機能を追加して拡張することもできます：

- 新規ディスク
- 新規ネットワークインターフェース
- コンテキスト情報の追加

**重要：**VMテンプレートを変更し、テンプレートインポート時にOpenNebulaが検出したディスクまたはNICを編集する場合は、以下

の注意事項をご確認ください：

- vCenter テンプレートで検出されたディスクおよび NIC には、OPENNEBULA\_MANAGED という特別な属性が設定されており、その値は NO です。
- この OPENNEBULA\_MANAGED=NO は、vCenter テンプレート内に存在する DISK および NIC 要素にのみ設定してください。
  - OpenNebula は、vCenter テンプレートの一部ではないディスクや NIC に対しては、同様の操作を適用しないためです。

- 仮想マシンテンプレート内のディスクまたはNIC要素を編集する際、OPENNEBULA\_MANAGED属性がNOに設定されている場合、vCenterテンプレートに含まれない新しいリソースに関する付けられたイメージまたは仮想ネットワークを変更される際は、SunstoneのAdvancedビューまたはCLIのonetemplate updateコマンドを使用して、仮想マシンテンプレートのディスクまたはNICセクションからOPENNEBULA\_MANAGED属性を削除することをお忘れなく。

OpenNebulaクラウドをご利用になる前に、vCenterに関する詳細についてお読みいただくことをお勧めいたします。

## 6.2.6 稼働中の仮想マシンのインポート

vCenterクラスタの監視が開始されると、OpenNebulaは既存の仮想マシンをすべて「Wild」として表示します。ホストの取得が正常に完了次第、これらの仮想マシンはOpenNebulaを通じてインポートおよび管理が可能となります。

コマンドラインでは、one host show コマンドを使用して、動作中の仮想マシンの一覧を表示することができます：

```
$ onehost show 0 HOST 0
INFORMATION ID
NAME
CLUSTER [ : 0
           : MyvCenterHost
           : -
           ]
```

未割り当て仮想マシン

名前	インポートID	CPU
メモリ	vm-2184	1
test-rp-removeme - クラスター		
256		
[.....]		

Sunstoneでは、ホスト情報に「Wild」タブがございます：

VM name	Remote ID
.....	vm-2351
.....	vm-2187
.....	vm-1931
.....	vm-12
.....	vm-34
test-rp-removeme - Cluster	vm-2184

稼働状態の仮想マシンはインポート可能です。また、vCenterで定義されているが電源投入状態ではない仮想マシンもインポートできます（この場合、OpenNebulaでは電源オフ状態としてインポートされます）。

**重要：**ワイルドVMをインポートする前に、前述の通り、VMのハードディスクファイルが配置されているデータストアを必ずインポートしておく必要があります。OpenNebulaでは、既存の仮想ハードディスクを表すイメージを作成する前に、データストアが存在していることが必要です。

**警告：**仮想マシンのインポート中は、OpenNebulaが仮想ディスクと仮想NICを検査し、仮想マシンで使用されているディスクとポートグループを参照するイメージと仮想ネットワークを作成します。この処理には時間がかかる場合がありますので、し

既存の仮想マシンをインポートするには、「onehost importvm」コマンドをご利用いただけます。

```
$ onehost importvm 0 "test-rp-removeme - Cluster"
$ onevm list
```

ID	ユーザー	グループ	NAME	STAT	UCPU	UMEM	HOST	TIME
3	oneadmin	oneadmin	test-rp-removem 実行中	0.00		20M	[vcenter.v]	0d 01h02

また、ホストの [Wilds] タブから Sunstone ユーザーインターフェイスをご利用いただけます。リストから仮想マシンを選択し、[インポート] ボタンをクリックしてください。

VM name	Remote ID
...	vm-2351
...	vm-2187
...	vm-1931
...	vm-12
...	vm-34
...	vm-2184

仮想マシンをインポートした後、そのライフサイクル（スナップショットの作成を含む）はOpenNebulaを通じて制御できます。インポートされた仮想マシンに対しては、以下の操作は実行できません：

- リカバリ（再作成）
- アンデプロイ（およびアンデプロイ - ハード）
- 移行（および移行 - ライブ）
- 停止

ワイルドVMがインポートされると、OpenNebulaはvCenter VMを再構成し、VMの監視開始後にVNC接続を確立できるようにします。

また、ネットワーク管理操作（ネットワークインターフェースの接続/切断機能など）、容量（CPUおよびメモリ）のサイズ変更操作、および事前にポートが開かれている場合のVNC接続も利用可能です。

仮想マシンをインポートした後、OpenNebula から削除し、再度インポートすることができます。OpenNebula は vCenter の仮想マシンメタデータに削除が必要な情報を設定します。この情報は `onevcenter cleartags` コマンドで削除できます：

- `opennebula.vm.running`
- `opennebula.vm.id`
- `opennebula.disk.*`

仮想マシンがvCenterで変更され、OpenNebulaがディスクとNICを「再スキャン」する必要がある場合に、以下の手順が有用です：

- 削除対象の VM で `onevcenter cleartags` を実行します：

```
$ onevcenter cleartags <vmid>
```

**vmid** は、属性がクリアされる仮想マシンの ID です。

- 仮想マシンの登録解除

```
$ onevm recover --delete-db <vmid>
```

- 仮想マシンの再インポート：次のホストの監視サイクルにおいて、この仮想マシンは「Wilds」タブに表示され、安全にイ

---

ンポートできます。

## 6.2.7 vCenter ネットワークのインポート

OpenNebula は、既存の vCenter ネットワーク（標準ポートグループおよび分散ポートグループ）の仮想ネットワーク表現を作成できます。OpenNebula は、これらの表現の上に、イーサネット、IPv4、IPv6 の 3 種類のアドレス範囲を処理できます。このネットワーク情報は、コンテキスト化プロセスを通じて VM に渡すことができます。

vCenter のポートグループまたは分散ポートグループをインポートすると、OpenNebula はその vCenter ネットワークを表す OpenNebula 仮想ネットワークを作成します。

インポートツール（onevcenter ツールまたは Sunstone）は、各データセンターで検出されたネットワークに関する情報を提供します：

- ネットワーク名
- ネットワークの種類（標準ポートグループまたは分散ポートグループ）
- ポートグループが使用されている vCenter クラスターの名称と、適切な vCenter クラスターを参照する OpenNebula ホストの ID です。

OpenNebula クラスターの ID が存在しない場合、このポートグループを使用する vCenter クラスターをまだインポートしていないことを意味します。ネットワークを正しくインポートするには、まず vCenter クラスターをインポートする必要があります。

---

**注記：**OpenNebula ではマルチクラスタネットワークがサポートされており、1つの vCenter クラスターにまたがるポートグループおよび分散ポートグループは適切にインポート可能です。OpenNebula は関連する vCenter クラスターを表示し、ネットワークインポート処理を進める前に少なくとも1つはインポートする必要があります。たとえ1つの vCenter クラスターのみインポートされたマルチクラスター ネットワークをインポートすることが可能であっても、ネットワークに関連するすべての vCenter クラスターをまず OpenNebula にインポートし（OpenNebula クラスターに配置し）、その後でネットワークインポートプロセスを進めることができます。

---

vCenter のネットワーク名はデータセンター内で一意であるため、異なるデータセンター や vCenter インスタンス間で同一名称のネットワークが存在する可能性があります。ネットワークをインポートする際、OpenNebula は既存のネットワークプールを確認し、必要に応じて衝突を回避する名称を生成します。この名称は仮想ネットワークのインポート後に変更可能です。以下のスクリーンショットに例を示します：



**警告：**仮想ネットワークをインポートするには、事前に vCenter クラスターをインポートしておく必要があります。そうしない場合、OpenNebula は関連する vCenter クラスターのインポートが必要であると警告します（前の項目を参照）。

### 1つの vCenter でのネットワークインポート

### **OpenNebula 5.10 導入ガイド、リリース 5.10.5**

インポートツールは各データセンター内のポートグループを検出します。検出結果には、ポートグループ名、ポートグループの種類（ポートグループまたは分散ポートグループ）、そのポートグループを使用するクラスター、およびこの仮想ネットワークが追加されるOpenNebulaクラスターIDが表示されます。

クラスタ名に色が付いていることに気付かれるかと思います。これは以下の2つの意味があります：

ネットワークに複数のvCenterクラスターが関連付けられている場合、listコマンドはOpen-Nebulaクラスターの一覧を表示します。

コマンドラインインターフェースを使用して標準ポートグループまたは分散ポートグループをインポートする方法を示す例を以下に示します：

常にそうであるように、まずインポート可能なオブジェクトの一覧を取得する必要があります：

```
$ onevcenter list -o networks -h 0

# vCenter: vcenter.Server

  IMID  REF          NAME           CLUSTERS
  0    ネットワーク-12   仮想マシンネットワーク [100, 102]
  1    ネットワーク-12245 テスト00      [100, 102]
  2    ネットワーク-12247 テスト03      [102]
  3    ネットワーク-12248 テスト02      [102]
  4    ネットワーク-12246 テスト01      [100, 102]
```

---

**注記：**CLUSTERS 列が -1 に設定されたネットワークが存在する場合があります。これは、そのネットワークに関連する vCenter クラスターがまだインポートされていないことを意味します。表示されている -1 の数に応じて、上記の注記をご参照ください。

---

この情報をもとに、今回は「Testing0\*」ネットワークをインポートします（複数のネットワークを同時にインポートすることは一般的です）。リストコマンドを使用すると、テストネットワークがIMID 1から4に含まれていることが簡単に確認できます。

```
$ onevcenter import 1..4 -o networks -h 0
```

または

```
$ onevcenter import "network-12245, network-12247, network-12246, network-12248" -o
←networks -h 0
```

たとえ上記の2番目のオプションが長すぎる場合でも、連続していない複数のネットワークをインポートしたい場合には非常に便利です。その後、いくつかの質問が表示され、回答に応じて異なるアクションが実行されます。

ネットワークをインポートする場合、vnet に VLAN ID が設定されている場合は最初に表示されます。次のステップはアドレス範囲の割り当てです。アドレス範囲の詳細については、「アドレス範囲の管理」セクションをご参照ください。

まず、アドレスプールのサイズを指定する必要があります：

このネットワークに配置予定の仮想マシン台数 **[255]** は何台ですか？

次に、アドレスプールの種類を指定する必要があります：

作成する仮想ネットワークの種類 (**IPv4** [4]、**IPv6** [6]、**[E]thernet**) をお選びください。

イーサネットプールを選択された場合、プールの最初のMACアドレスを選択いただけます（任意です）：

範囲内の最初のMACアドレスを入力してください [**デフォルトの場合はEnterキーを押してください**] :

IPv4アドレスプールを選択された場合、初期IPアドレスとプール内の最初のMACアドレスを指定する必要があります（任意です）。

。

範囲内の最初のIPアドレスを入力してください：**10.0.0.0**

範囲内の最初のMACアドレスを入力してください [**デフォルトの場合はEnterキーを押してください**] :

IPv6アドレスプールを選択される場合、プールの最初のMACアドレスを指定する必要があります（任意）。また、SLAACを使用さ

れる場合は：

範囲内の最初のMACアドレスを入力してください [デフォルトの場合はEnter] :

SLAAC (ステートレスアドレス自動設定) を使用しますか? ([y]/n)

SLAAC自動設定をご利用になる場合は、GLOBAL PREFIXおよびULA\_PREFIXを指定いただくか、デフォルト値をご利用ください。

グローバルプレフィックスを入力してください [デフォルト値の場合はEnterキーを押してください]

: ULAプレフィックスを入力してください [デフォルト値の場合はEnterキーを押してください] :

SLAAC自動設定をご利用にならない場合は、IPv6アドレスとプレフィックス長を指定する必要があります。

IPv6アドレスを入力してください (空欄不可) : プレフィックス長を入力してください

さい (空欄不可) :

最後に、ネットワークが正常に作成されると、ネットワーク名（前述の通りOpenNebulaによって自動生成されます）と数値IDが表示されます。

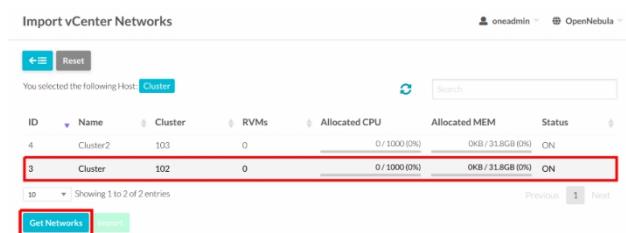
OpenNebula virtual network TestPG - Cluster [vcenter.vcenter3 - Datacenter] b2bfcced7734 with ID 140 created with size 255!

### Sunstone でのネットワークのインポート

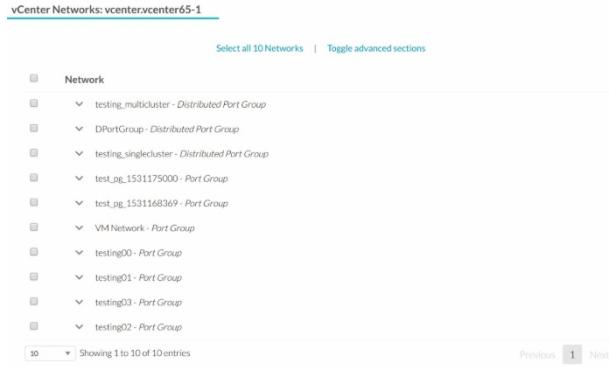
Sunstone でも同様の手順となります。ネットワークメニューの「仮想ネットワーク」をクリックし、次に「インポート」ボタンをクリックすると、新しいウィンドウが開きます。



新しいウィンドウで、このvCenterインスタンスに認証するためのクラスターを選択し、「ネットワークを取得」をクリックしてください。

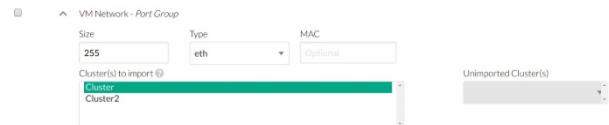


「ネットワークを取得」をクリックすると、ポートグループの一覧が表示されます。ポートグループ名、タイプ、クラスター、クラスターの場所、およびこの仮想ネットワークが割り当てられる既存のOpenNebulaクラスターのIDが表示されます。OpenNebulaクラスタIDが-1の場合、インポートツールがデータストアをグループ化できるOpenNebulaクラスタを見つけられなかったことを意味します。後で手動で割り当てる必要があるか、データストアインポートツールの操作をキャンセルし、vCenterクラスタを先にインポートしてみてください。



ネットワークをインポートする前に、ネットワーク名の横にある下向き矢印をクリックし、設定したいアドレスプールのタイプを指定することができます：

- イーサネットアドレス範囲プールには eth。
- \* ipv4 : IPv4 アドレス範囲プール。
- SLAAC を使用した IPv6 アドレス範囲プールの場合は「ipv6」。
- ipv6\_static は、SLAAC を使用しない IPv6 アドレス範囲プール用です（IPv6 アドレスとプレフィックス長が必要です）。



ネットワークをインポートする際、デフォルトのアドレス範囲は255個のMACアドレスプールとなります。

最後に「インポート」ボタンをクリックすると、作成された仮想ネットワークのIDが表示されます：



**警告：** OpenNebula が新しいネットワークを検出しない場合は、それらのポートグループを使用している vCenter クラスターを事前にインポート済みかどうかご確認ください。

## 6.2.8 vCenterイメージのインポート

OpenNebulaは、vCenterデータストア内に存在するvCenter VMDKおよびISOファイルのイメージ表現を作成することができます。

VMDK または ISO ファイルは、データストア内の異なる場所に同じ名前で存在する場合があります。インポートツールは、見つかった各ファイルについて以下の情報を提供します：

- データストア内のパス。
- VMDKファイルのサイズです。これは仮想マシン側から見たVMDKファイルの容量サイズとなります。例えば、シンプロビジョニングされている場合、VMDKファイルの実際のサイズは数KB程度である可能性がありますが、そのファイルが仮想マシンに接続された場合、仮想マシンが報告するサイズは異なります。そのため、利用可能な場合には容量が表示され、そうでない場合にはファイルの実際のサイズが表示されます。

- 
- ファイルの種類：VmDiskFileInfo または IsoImageFileInfo。

イメージをインポートする際、OpenNebulaは衝突を避けるための名前を生成します。この名前にはイメージ名が含まれ、同じ名前のイメージが既に存在する場合、サフィックスが付加されます。イメージのインポート完了後、この名前はより分かりやすい名前に変更することができます。以下にサンプル名を示します：

## one-11-corelinux7\_x86\_64-11\_2

インポートツールは、以前にインポートされていないファイルを検索し、同じPATHおよびDATASTORE\_ID属性を持つファイルが存在するかを確認します。

### onevcenter によるイメージのインポート

onevcenterツールおよびSunstoneインターフェースを使用して、この種のファイルをインポートすることができます。

onevcenterツールは、引数としてOpenNebulaのIMAGEデータストア名を指定する必要があります。OpenNebulaはデータストアをスキヤンし、VMDKおよびISOファイルを検索します。これは、適切なvCenterイメージデータストアをOpenNebulaにインポートすることが必須であることを意味します。この情報はonevcenterツールの-dオプションで渡すことができますので、イメージインポート操作の前に必ずご確認ください：

利用可能なvCenterデータストアを確認する簡単な方法は以下の通りです：

```
onedatastore list | grep -E 'img.* vcenter'

100  datastore2 (IM)          924G 100%    102           1 img      vcenter   vcenter オン
102  データストア1 (IM)        924G 88%     -            0 img      vcenter   vcenter オン
106  NFS (イメージ)           4.5T 39%     100,102       24 img     vcenter   vcenter オン
```

コマンドラインインターフェースを使用してVMDKファイルをインポートする方法の例を以下に示します。この例では、データストア1(102)とホスト0を使用します：

```
onevcenter list -o images -h 0 -d 102
# vCenter: vcenter.vcenter65-1

IMID REF                                パス
  0 one-21                               one_223304/21/one-21.vmdkone_223304/22/Core-
  1 Core-current.iso.iso                  current.iso.iso
```

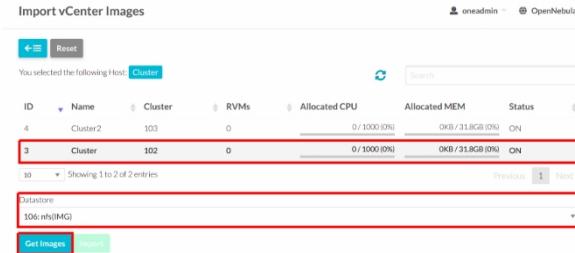
画像のインポートが完了しましたら、OpenNebulaの画像IDが表示されます。

### Sunstone を使用したイメージのインポート

Sunstoneからも画像をインポートできます。ストレージメニューの「Images」をクリックし、「Import」ボタンをクリックしてください。



新しいウィンドウで、このvCenterインスタンスへの認証に使用するクラスターを選択し、「イメージを取得」をクリックしてください。



OpenNebulaは、まだインポートされていないVMDKおよびISOファイルを検索します。

Select all 25 Images			
	Path	Size	Type
■	vOneCloud-1.8.0/vOneCloud-1.8.0.vmdk	10GB	VmDiskFileInfo
■	vOneCloud-1.8.0/vOneCloud-1.8.0-000001.vmdk	10GB	VmDiskFileInfo
■	one-template-vc_silitaz_template/one-template-vc_silitaz_template.vmdk	256MB	VmDiskFileInfo
■	one-corelinux7_x86_64/one-corelinux7_x86_64.vmdk	5MB	VmDiskFileInfo
■	one-corelinux7_x86_64/one-corelinux7_x86_64_3.vmdk	70MB	VmDiskFileInfo
■	one-corelinux7_x86_64/one-corelinux7_x86_64_2.vmdk	10MB	VmDiskFileInfo
■	one-11-test_instantiate_to_persist_hook_2/one-11-test_instantiate_to_persist_hook_2.vmdk	70MB	VmDiskFileInfo
■	jav-one-28-one-template-vc_silitaz_template - devel-28/jav-one-28-one-template-vc_silitaz_template - devel-28-000001.vmdk	256MB	VmDiskFileInfo
■	jav-one-27-one-template-vc_silitaz_template - devel-27/jav-one-27-one-template-vc_silitaz_template - devel-27-000001.vmdk	256MB	VmDiskFileInfo
■	jav-one-24-one-template-vc_silitaz_template - devel-24/jav-one-24-one-template-vc_silitaz_template - devel-24-000004.vmdk	256MB	VmDiskFileInfo

10 | Showing 1 to 10 of 25 entries

Previous 1 2 3 Next

インポートしたい画像を選択し、[インポート]ボタンをクリックしてください。インポートされた画像のIDが表示されます。

**注記：**インポートツールを使用してイメージを作成した場合、VCENTER\_IMPORTED 属性は自動的に YES に設定されます。この属性により、OpenNebulaからイメージが削除されても、OpenNebulaがvCenterデータストアからファイルを削除することを防止します。これにより、仮想ハードディスクがvCenterテンプレートから誤って削除されるのを防ぐことができます。このデフォルト動作は、/var/lib/one/etc/remotes/vmm/vcenter/vcenterrc 内のDELETE\_IMAGESをYesに設定することで変更可能です。

## 6.2.9 OpenNebulaを使用したvCenter仮想マシンの移行

vCenterドライバーは、異なるvCenterクラスター間（すなわちOpenNebulaホスト間）および/または異なるデータストア間での仮想マシンの移行を可能にします。移行の種類（コールド移行：仮想マシンが電源オフまたは保存状態である場合、またはライブ移行：仮想マシンが稼働中に移行される場合）や移行先（クラスターおよび/またはデータストア）に応じて、マシンを移行するにはいくつかの要件を満たす必要があります。

### vCenterクラスタ間（OpenNebulaホスト間）でのVM移行要件（ライブ移行・コールド移行共通）

)

- 対象の仮想マシンに接続されているすべてのネットワークは、両方のvCenterクラスタおよびOpenNebulaクラスタに存在している必要があります
- VMが使用するすべてのデータストアは、vCenterクラスタとOpenNebulaクラスタの両方に存在している必要があります
- ターゲットとなるOpenNebulaホストでは、ESX\_MIGRATION\_LIST属性を指定することが可能です：

#### **OpenNebula 5.10 導入ガイド、リリース 5.10.5**

- 指定がない場合、移行先のESXホストは明示的に宣言されず、移行が失敗する可能性があります。
- 空文字列（""）に設定された場合、OpenNebulaはvCenterターゲットクラスターに属する全てのESXからランダムにターゲットESXを選択します

- スペース区切りのESXホスト名リスト（vCenterターゲットクラスタに属している必要があります）に設定された場合、OpenNebulaはリストからランダムにターゲットESXを選択します

---

**注記：**仮想マシンが OpenNebula の要件を満たしているかどうかを確認するには、仮想マシンの「AUTO-MATIC\_REQUIREMENTS」属性を参照することをお勧めいたします（これはテンプレート情報タブで確認できます）。対象の OpenNebula クラスタが含まれているかどうかを確認してください（OpenNebula のクラスタはホスト、仮想ネットワーク、データストアの集合体であり、vCenter のクラスタは OpenNebula ではホストとして表現されることをご留意ください）。

---

#### 要件（ライブマイグレーションのみ）

- 両方のvCenterクラスターでvMotionインターフェースが有効になっていること（有効でない場合、互換性の問題についてドライバーから警告が表示されます）
- OpenNebulaのライブマイグレーションは稼働中の仮想マシンでのみ動作します。そのため、必ず状態を確認してから実行してください。

#### 使用方法（CLI）

##### コールド移行

```
$ onevm migrate "<VM名>" <移行先ホストID>
```

##### ライブマイグレーション

```
$ onevm migrate --live "<VM名>" <移行先ホストID>
```

#### データストア間の仮想マシンの移行

仮想マシンの移行時には、ターゲットとなるデータストアを変更することができます。仮想マシンに属するディスクは、移行先のデータストアへ移行されます。これは、データストア間でのリソース使用量の再調整に有用です。

#### 要件（コールド移行およびライブ移行の両方）

- VM が使用するすべてのデータストアは、vCenter クラスタと OpenNebula クラスタの両方に存在している必要があります

#### 使用方法（CLI）

##### コールド移行

```
$ onevm migrate "<VM名>" <移行先ホストID> <移行先データストアID>
```

##### ライブマイグレーション

```
$ onevm migrate --live "<VM名>" <移行先ホストID> <移行先データストアID>
```

## **6.2.10 vCenter フック**

OpenNebula には、vCenter および [NSX](#) におけるネットワーク管理のための 2 つのフックが用意されています。

フック名	フックの説明
vcenter_net_create	vCenter および NSX ネットワークの作成／インポートを可能にします
vcenter_net_delete	vCenter および NSX ネットワークを削除できます

これらのフックは、vCenterクラスタを追加する際に自動的に作成されるはずです。万が一削除してしまった場合でも、手動で再度作成することができます。

*vCenter フックの作成に移動し、手順に従って新しいフックを作成してください。*

---

**注記：**フックの動作に関する詳細情報は、こちらでご確認いただけます。

---

### vCenter フックの一覧表示

登録済みのフックを一覧表示するには、次のコマンドを入力してください：

```
$ onehook list
```

コマンドの出力は、以下のような内容になります：

ID	NAME	TYPE
1	vcenter_net_delete	api
0	vcenter_net_create	api

### vCenter フックの作成

新しいフックを作成するには、以下のコマンドをご利用いただけます：

```
$ onehook create <テンプレートファイル>
```

テンプレートファイルは、フックテンプレート情報を含むファイルの名前です。ネットワーク作成用のフックテンプレートは以下の通りです：

```
NAME = vcenter_net_create TYPE =
api
COMMAND = vcenter/create_vcenter_net.rb CALL =
"one.vn.allocate"
ARGUMENTS = "$API"
ARGUMENTS_STDIN = yes
```

最新版 <[https://raw.githubusercontent.com/OpenNebula/one/master/share/hooks/vcenter/templates/create\\_vcenter\\_net.tpl](https://raw.githubusercontent.com/OpenNebula/one/master/share/hooks/vcenter/templates/create_vcenter_net.tpl)>`

ネットワーク削除用のフックテンプレートは以下の通りです：

```
NAME = vcenter_net_delete TYPE =
api
COMMAND = vcenter/delete_vcenter_net.rb CALL =
"one.vn.delete"
引数 = "$API" 引数_標準入力 =
はい
```

フック削除テンプレートの最新版は、こちらでご確認いただけます。

#### vCenter フックの削除

フックは、そのIDが判明している場合に削除できます。IDはonehook listコマンドで取得でき、その後以下のコマンドで削除できます。

```
$ onehook delete <hook_id>
```

#### 6.2.11 ドライバーの調整

ドライバーは簡単にカスタマイズできます。詳細は統合ガイドのvCenterドライバーセクションをご参照ください。

ドライバの動作の一部は、`/var/lib/one/remotes/etc/vmm/vcenter/vcenterrc` で設定可能です：

- **delete\_images**: OpenNebulaがインポートされたvCenterイメージを削除することを許可します。デフォルト: **no**。
- **vm\_poweron\_wait\_default**: デプロイ操作のタイムアウト時間。デフォルト: **300**。
- **debug\_information**: 詳細なログを出力します。デフォルト: **false**。
- **retries**: 失敗が発生した場合に一部のドライバ操作が再試行をサポートします。このパラメータは再試行回数を設定します。デフォルト: **3**。
- **retry\_interval**: 再試行間の待機時間（秒単位）。デフォルト: **1**。

### 6.3 vCenter データストア

vCenter データストアは、VMDK ファイルやその他のファイルタイプをホストし、仮想マシンやテンプレートがそれらを利用できるようにします。これらのデータストアは、OpenNebulaにおいて「イメージ データストア」および「システム データストア」の両方として表現されます：

- イメージ・データストア。イメージ・リポジトリを格納します。VMDKファイルは、このデータストアに保存されたOpenNebulaイメージとして表現されます。
- システム・データストア。イメージ・データストアからコピーまたはクローンされた仮想マシンを実行するためのディスクを保持します。

例えば、vCenter データストアに「nfs」という名前がある場合、この vCenter データストアを OpenNebula にインポートすると、同じ vCenter データストアを指す 2 つの OpenNebula データストア（イメージ データストアとシステム データストア）が作成されます：

```
* Datastore found:
- Name      : nfs
- Datacenter : Datacenter
- Total MB  : 4691989
- Free MB   : 1564950
- Import name: nfs [vcenter.vcenter5-devel - Datacenter]
Import this datastore [y/n]? y

NOTE: For each vCenter datastore a SYSTEM and IMAGE datastore
will be created in OpenNebula except for a StorageDRS which is
represented as a SYSTEM datastore only.

OpenNebula datastore 315 created!
OpenNebula datastore 316 created!
```

ID	NAME	SIZE	AVAIL	CLUSTERS	IMAGES	TYPE	DS	TM	STAT
313	LocalDatastor	225.3G	17%	0,120	0	img	vcenter	vcenter	on
314	LocalDatastor	225.3G	17%	0,120	0	sys	-	vcenter	on
315	nfs [vcenter.	4.5T	33%	0,120	0	img	vcenter	vcenter	on
316	nfs [vcenter.	4.5T	33%	0,120	0	sys	-	vcenter	on

---

**重要:** vCenterインスタンスに読み取り専用データストアが含まれる場合、OpenNebulaがそこに仮想マシンを展開しようとしないよう、インポート後に当該データストアのSYSTEM表現を無効化する必要があります。

---

### 6.3.1 イメージとディスク

vCenterハイパーバイザーを使用する場合、OpenNebulaには以下の3種類のイメージタイプが存在します：

- OS: 起動可能なディスクイメージです。すべての仮想マシンテンプレートは、このタイプのイメージを参照するディスクを1つ定義する必要があります。
- CDROM: これらのイメージは読み取り専用データです。
- DATABLOCK: データブロックイメージは、データを保存するためのストレージです。これらのイメージは、既存のデータ（例: VMDKファイルのアップロード）から作成することも、空のドライブとして作成することも可能です。

OpenNebula のイメージは、永続イメージと非永続イメージにも分類されます：

- 非永続イメージ。これらのイメージは少なくとも1台のVMで使用されます。他のVMでも引き続き使用可能です。非永続イメージを使用する新しいVMがデプロイされると、VMDKファイルのコピーが作成されます。
- 永続イメージ。永続イメージは特定のVMのみが使用可能です。新規VMでは使用できません。元のファイルが使用され、コピーは作成されません。

仮想マシンに接続されたディスクは、非永続的または永続的イメージによってバックアップされますが、揮発性ディスクもサポートされています。揮発性ディスクはターゲットホスト上でオンザフライで作成され、仮想マシンのシャットダウン時に破棄されます。

### 6.3.2 制限事項

- vCenterテンプレートまたはワイルドVMをOpenNebulaにインポートする際、仮想ディスクがインポートされ、それらの仮想ディスクを表すイメージがOpenNebula内に作成されます。これらのイメージはデータストアに既に存在するファイルを表していますが、OpenNebulaはインポートされたイメージのサイズを新規作成されたファイルと同様に扱います。そのため、vCenterデータストアの実際の領域がOpenNebulaイメージによって使用されていないにもかかわらず、データストアの容量は減少します。例えば、OpenNebula が空き容量不足を報告してイメージのインポートが失敗する場合や、ディスククオータを使用している場合などには、この制限事項を理解しておく必要があります。
- vCenterデータストアにおけるディスクスナップショットはサポートされていません。
- 画像の名前およびパスには、スペースや非ASCII文字を含めることはできません。

### 6.3.3 vCenter Transfer Manager

OpenNebulaのvCenter Transfer Manager ドライバーは、ディスクイメージを以下の方法で扱います：

- OpenNebulaによって作成された新しいディスクイメージは、Imagesデータストアに配置されます。これらは永続イメージまたは非永続イメージとして作成できます。

- ・永続イメージは、そのイメージが作成されたデータストアから vCenter VM によって使用されます。
- ・非永続イメージは、作成元の「Images」データストアから、OpenNebula のスケジューラが選択した「System」データストアにコピーされます。
- ・一時的なイメージは、スケジューラが選択したシステムデータストアに作成され、不要になった時点で（例：ディスクのデータタッチや仮想マシンの終了操作時）そのデータストアから削除されます。
- ・空のデータブロックの作成とVMDKイメージのクローン作成、ならびにイメージの削除がサポートされております。

スケジューラは、操作ガイドに記載されている通り、/etc/one/sched.conf の設定に基づいてデータストアを選択します：

- デフォルトでは、空き容量が少ないデータストアを選択することでストレージ使用量の最適化を試みます。
- 利用可能なデータストアに仮想マシンを分散させることで、I/O を最適化することも可能です。

OpenNebulaにおけるvCenterデータストアは、vCenterインスタンスに紐付けられています。つまり、データストアで実行されるすべての操作は、そのインスタンスで定義された認証情報を使用して、vCenterインスタンスを通じて実行されます。

vCenterデータストアはOpenNebula内で表現され、以下の仮想マシン操作を実現します：

- VMDK* ファイルのアップロード
- ISO* ファイルのアップロード
- 空のデータブロックを作成する
- VMDKイメージのクローンを作成する
- VMDKイメージの削除

## OpenNebulaクラスタ

クラスターとは、ホストのグループであり、関連付けられたデータストアや仮想ネットワークを持つことができます。vCenterクラスターをインポートする際、インポートツールは自動的に、そのvCenterクラスターを表すOpenNebulaホストにクラスターを割り当てます。

**重要：** vCenterデータストアをOpenNebulaにインポートする際、OpenNebulaは既存のOpenNebulaクラスタにデータストアを追加しようと試みます。 そのデータストアを使用するvCenterクラスタを以前にインポートしたことがない場合、自動割り当ては適切なOpenNebulaクラスタを見つけられないため、スケジューラはVMをデプロイする際に使用できる適切なデータストアを認識できません。この場合、「クラスタへのリソースの追加」セクションで説明されているように、OpenNebulaホスト(vCenterクラスタを表す)が存在するクラスタにデータストアを追加する必要があります。

## 転送マネージャーが使用するファイルの場所

VMDKファイルまたはISOファイルは、以下の規則に従いvCenterデータストア内に配置・命名されます：

- 永続イメージ。これらのイメージは、以下のパターンに従って配置されます：IMAGE\_DIR/IMAGE\_ID/one-IMAGE\_ID.vmdk  
例：one/258/one-258.vmdk。IMAGE\_DIRはデフォルトではディレクトリ **one** ですが、VCENTER\_DS\_IMAGE\_DIR属性により別のディレクトリを使用することも可能です。
- 非永続イメージ。これらのイメージは次のパターンに従って配置されます：IMAGE\_DIR/IMAGE\_ID/one-IMAGE\_ID.vmdk、  
例：one/259/one-259.vmdk。IMAGE\_DIRはデフォルトでディレクトリ **one** ですが、VCENTER\_DS\_IMAGE\_DIR属性により別のディレクトリを使用できます。
- 仮想マシンで使用される非永続イメージ。非永続イメージのコピーは次のパターンに従います：  
IMAGE\_DIR/IMAGE\_ID/one-VMID-IMAGE\_ID-DISK\_NUMBER.vmdk。ここで、VMIDは仮想マシンの数値識別子に置き換えられ、IMAGE\_IDは元のイメージの識別子となり、DISK\_NUMBERは仮想マシン内のディスクの位置に置き換えられます。
- 仮想マシンに接続された揮発性ディスク。これらのイメージは次のパターンで配置されます：VOLATILE\_DIR/one-VMID-

---

DISK\_NUMBER.vmdk（例：one-volatile/285/one-285-2.vmdk）。VOLATILE\_DIRはデフォルトではone-volatileディレクトリですが、VCENTER\_DS\_VOLATILE\_DIR属性により別のディレクトリを使用することも可能です。

以下の例では、OpenNebulaのID 8を持つイメージに関連付けられたファイルが、前述の配置ロジックに基づいてVMDKファイルを含んでいることが確認できます。



**重要：**OpenNebulaは、データストア上に不要なフォルダやファイルを残さないよう努めておりますが、vCenter 6.5のAPIにバグが検出されており、現在OpenNebulaが作成した空のディレクトリを削除できない状況です。そのため、前述の場所にあるデータストア内に空のフォルダが残っている可能性がございます。これらのフォルダが何らかの理由で気になる場合は、手動で削除する必要があります。

### 6.3.4 要件

vCenter データストアをご利用いただくには、以下の要件を満たす必要があります：

- vCenterによって管理されているすべてのESXサーバーは、同じ名前のVMFSデータストアをマウントする必要があります。
- ESXサーバーは、OpenNebulaによって管理されるクラスターの一部である必要があります。
- IMAGE データストアにイメージを作成する前に、そのデータストアが監視対象となっており、サイズと使用状況情報が報告されていることをご確認ください。監視対象となるまでは、データストアにイメージを作成することはできません。

### 6.3.5 VMDK ファイルのアップロード

仮想マシンに仮想ハードディスクとして接続可能なVMDKファイルをアップロードできます。VMDKを

含むファイルは、以下の2つの方法でアップロードできます：

- ウェブブラウザからファイルを追加する方法、または
- URLを使用してパスを指定する方

法です。アップロード可能なファイルは

以下の通りです：

- スタンドアロンのVMDKファイルです。このファイルはgzipまたはbzip2で圧縮することも可能です。
- フラットファイルとVMDK記述子をアーカイブされたtarファイルに格納します。両ファイルはアーカイブされたtarファイルの第一階層に配置する必要があります。OpenNebulaはtarファイル内のフォルダおよびサブフォルダをサポートしております。tarファイルはgzipまたはbzip2で圧縮することも可能です。

### CLIの使用方法

oneimage CLI コマンドをご利用いただけます。以下に、スタンドアロンの vmdk ファイルを ID 154 の IMAGE データストアにアップロードする例を示します。

vcenter ドライバー、タイプ、名前、説明を指定します。タイプパラメータには OS (イメージにオペレーティングシステムが含まれていることを OpenNebula に通知する場合)、DATABLOCK、CDROM が指定可能です。他のオプションを指定したい場合は、パラメータなしで oneimage を実行すると、パラメータの一覧と使用例が表示されます。

```
$ oneimage create -d 153 --type OS --name test_standalone --path /tmp/tinycore-2.1-  
↪x86.vmdk --driver vcenter --description "アップロードテスト" ID: 134
```

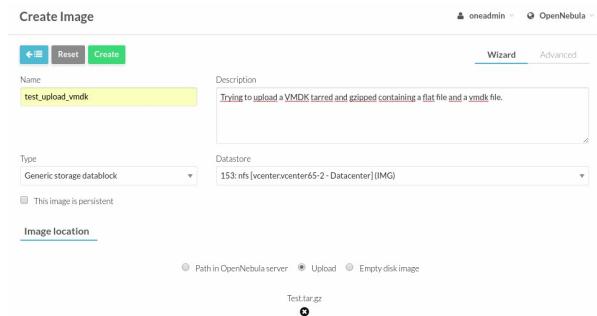
コマンドを実行するとイメージIDが返されます。イメージのアップロード中はイメージステータスが「LOCKED」となります。後ほどステータスが「READY」または「ERROR」に変更されたかどうかをご確認ください。

## Sunstone の使用

ファイルをアップロードするIMAGEデータストアを選択し、仮想マシンで使用する際のイメージを永続的（persistent）または非永続的（non-persistent）のいずれかに指定する必要があります。

VMDKファイルをアップロードする際、イメージのタイプを「オペレーティングシステムイメージ」または「汎用ストレージデータブロック」のいずれかに割り当てることができます。

以下の例では、フラットファイルと vmdk 記述子を含む tar.gz ファイルをブラウザを使用してアップロードします。OpenNebula はファイルを一時的な場所にアップロードし、そのファイルを解凍・展開した後、選択した vCenter データストアにその内容をアップロードします。



**警告：**SunstoneサーバーがApacheまたはNGINXサーバー（Passenger搭載）の背後で動作していない場合、アップロード操作が30秒以上かかる際に「**サーバーに接続できません。サーバーは稼働中ですか、または到達可能ですか？**」というエラーメッセージが表示される可能性があります。その場合は、画像ウィンドウを更新してください。新しい画像が「ロック中」状態になっていることが確認できますが、アップロード操作は継続中です。後ほど再度確認し、画像が「準備完了」状態か「エラー」状態になっているかご確認ください。

ファイルのアップロードプロセスを開始するには、[作成]ボタンをクリックしてください。

画像のアップロード中はステータスが「ロック中」となります。後ほど「画像」タブを更新して、ステータスが「使用可能」または「エラー」になっているかご確認ください。

132	test_upload_vmdk	oneadmin	oneadmin	nfs [vcenter:vcenter65-2 - Datacenter][IMG]	DATABLOCK	READY	0
-----	------------------	----------	----------	---	-----------	-------	---

### 6.3.6 ISOファイルのアップロード

仮想マシンに接続可能な CDROM イメージとして使用できる ISO ファイルをアップロードできます。

**ご注意：**CDROMイメージファイルは、仮想マシンが電源オフ状態の場合にのみ接続可能です。ISOファイルは仮想マシンにIDE CD-ROMドライブとして接続されるため、ホットプラグ対応デバイスではないためです。

ISOファイルは、以下の2つの方法でアップロードできます：

- ウェブブラウザからファイルを追加する方法、または
- URLを使用してパスを指定する方法です。

## CLIの使用

oneimage CLI コマンドをご利用いただけます。以下は、スタンドアロンの vmdk ファイルを ID 154 の IMAGE データストアにアップロードする例です。

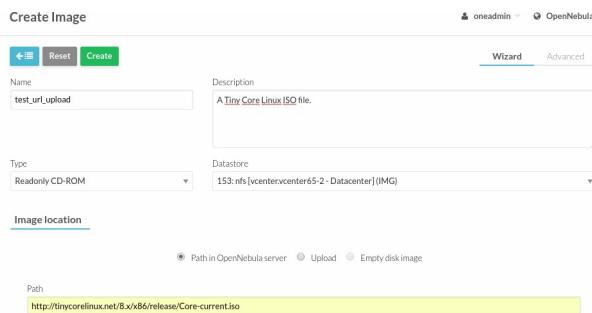
vcenter ドライバーを指定し、タイプは CDROM、名前と説明を指定します。

```
$ oneimage create -d 153 --name test_iso_file --type CDROM --path http://
˓→tinycorelinux.net/8.x/x86/release/Core-current.iso --driver vcenter --description
˓→「ISOテストアップロード」ID:
135
```

コマンドはイメージIDを返します。ISOイメージのアップロード中は、イメージステータスはLOCKEDとなります。後ほどステータスがREADYまたはERRORに変更されたかどうかをご確認ください。

### Sunstone の使用

以下の例では、インターネット上の URL を使用しています。OpenNebula は ISO ファイルを一時ファイルにダウンロードし、その後、選択された vCenter データストアにアップロードします。



**警告：**SunstoneサーバーがApacheまたはNGINXサーバー（Passenger搭載）の背後で動作していない場合、アップロード操作が30秒以上かかる際に「**サーバーに接続できません。サーバーは稼働中ですか、または到達可能ですか？**」というエラーメッセージが表示される可能性があります。その場合は、画像ウィンドウを更新してください。新しい画像が「ロック中」状態になっていることが確認できますが、アップロード操作は継続中です。後ほど再度確認し、画像が「準備完了」状態か「エラー」状態になっているかご確認ください。

ファイルのアップロードプロセスを開始するには、[作成]ボタンをクリックしてください。

画像のアップロード中はステータスが「ロック中」となります。後ほど「画像」タブを更新して、ステータスが「使用可能」または「エラー」になっているかご確認ください。

131	test_upload_cdrom	oneadmin	oneadmin	nfs[vcentervcenter65-2 - Datacenter](IMG)	CDROM	READY	0
-----	-------------------	----------	----------	---	-------	-------	---

### 6.3.7 空のデータブロックの作成

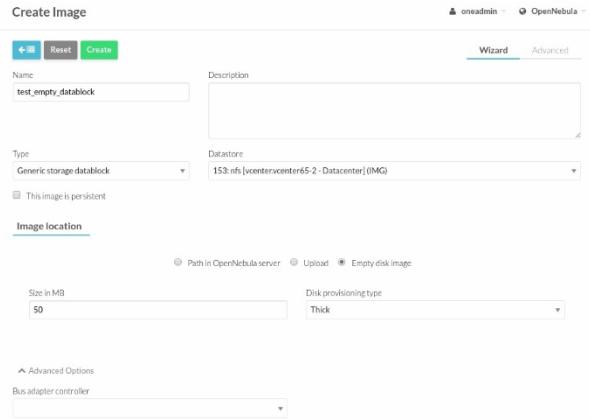
OpenNebulaから簡単に空のVMDKデータブロックを作成できます。Sunstone

では以下の手順に従ってください：

- データブロックに名前を付けます。説明は任意です。
- ドロップダウンメニューの「タイプ」で「汎用ストレージデータブロック」を選択してください。
- OpenNebula に空のデータブロックを作成させたい IMAGE データストアを選択します。

- 
- ・「空のディスクイメージ」を選択します。
  - ・データブロックのサイズをMB単位で指定してください。

- ディスクの種類を選択してください（任意）。ディスクの種類の一覧は、設定セクションで説明されている VCENTER\_DISK\_TYPE 属性の説明に記載されています。
- バスアダプターを選択してください（任意）。コントローラーのタイプの一覧は、設定セクションで説明されている VCENTER\_ADAPTER\_TYPE 属性の説明に記載されています。



最後に「作成」をクリックしてください。

VMDK ファイルがデータストアに作成されている間、イメージのステータスは「ロック中」となります。後ほど「イメージ」タブを更新し、ステータスが「使用可能」または「エラー」になっているかご確認ください。

---

**注記：** ディスクタイプおよび/またはバスアダプタコントローラタイプを指定しない場合、デフォルト値が適用されます。  
`/etc/one/vcenter_driver.default` ファイルの設定が適用されます。詳細はこちらをご参照ください。

---

### 6.3.8 設定

vCenter VMFS データストアを表す OpenNebula vCenter データストアを作成するには、以下の属性を持つ新しい OpenNebula データストアを作成する必要があります。onevcneter インポートツールは、必要な属性を持つデータストア表現を作成します。

属性	説明
DS_MAD	
TM_MAD	
TYPE	TYPE が SYSTEM_DS の場合、vcenter に設定する必要があります。TYPE が SYSTEM_DS または IMAGE_DS の場合は、必ず SYSTEM_DS または IMAGE_DS に設定する必要があります。
VCENTER_DAefDaAuPltTaEdRap_tTerYtPyEpe	VCENTER_DAefDaAuPltTaEdRap_tTerYtPyEpe は、データストア内のイメージに対して仮想マシンに継承される仮想ディスクに使用されます。これはイメージによって継承され、イメージ内で明示的に指定された場合、上書きされる可能性があります。可能な値（大文字小文字に注意）：lsiLogic、ide、busLogic。詳細は VMware のドキュメントをご参照ください。Sunstone では「バスアダプタコントローラ」として知られています。
VCENTER_TyDpIeSKo_fTYdPisEk	VCENTER_TyDpIeSKo_fTYdPisEk は、DATABLOCK が要求された際に作成されます。この値はデータストアからイメージへ継承されますが、明示的に上書きすることも可能です。ディスクのタイプは、以下の点に影響を及ぼします。 performance occupied space. 値（注意 場合）: delta、eagerZeroedThick、flatMonolithic、preallocated、raw、rdm、rdmp、seSparse、sparse2Gb、sparseMonolithic、thick、thick2Gb、thin。 詳細については、VMware のドキュメントをご参照ください。Sunstone では「ディスクプロビジョニングタイプ」として知られています。
VCENTER_MDaSn_agReEdFObject	VCENTER_MDaSn_agReEdFObject は vCenter データストアの参照です。これらの参照に関する詳細は、マネージドオブジェクトリファレンスセクションをご覧ください。
VCENTER_NDamS_eNoAf M	VCENTER_NDamS_eNoAf M
VCENTER_MDaCn_agReEdFObject	VCENTER_MDaCn_agReEdFObject vCenter データセンターの参照情報です。これらの参照情報に関する詳細は、マネージドオブジェクトリファレンスセクションをご覧ください。
VCENTER_NDamC_eNoAf M	VCENTER_NDamC_eNoAf M
VCENTER_TIheNSvCTAenNtCerEi_nIstDance ID	VCENTER_TIheNSvCTAenNtCerEi_nIstDance ID。これらの参照に関する詳細は、マネージドオブジェクトリファレンスセクションをご覧ください。
VCENTER_HHoOstSnTame	VCENTER_HHoOstSnTame または vCenter ホストの IP アドレス
VCENTER_GSEp_eDcIfiRes	VCENTER_GSEp_eDcIfiRes データストアのルートディレクトリ配下において、永続的および非永続的イメージを格納するフォルダを指定します。例：one
VCENTER_AS	データストアのルートディレクトリ配下にある VCENTER_AS フォルダには、揮発性ディスクがホストされます。

すべての OpenNebula データストアは常に監視されており、空き容量が不足している vCenter データストアへの仮想マシンのデプロイはスケジューラによって拒否されます。

### 6.3.9 Storage DRS を使用したデータストアクラスター

OpenNebula のスケジューラにより、負荷分散機能を備えたデータストアクラスターの管理が可能ですが、既に vCenter の Storage DRS 機能をご利用中かもしれません。 Storage DRS では、データストアクラスターの集約リソースを管理できます。 Storage DRS をご利用の場合、OpenNebula はデータストアの選択判断を Storage DRS クラスタ（SDRS）に委譲できますが、この動作は OpenNebula のスケジューラと干渉し、vSphere の API がいくつかの制限を課すため、OpenNebula における Storage DRS のサポートには制限が生じます。

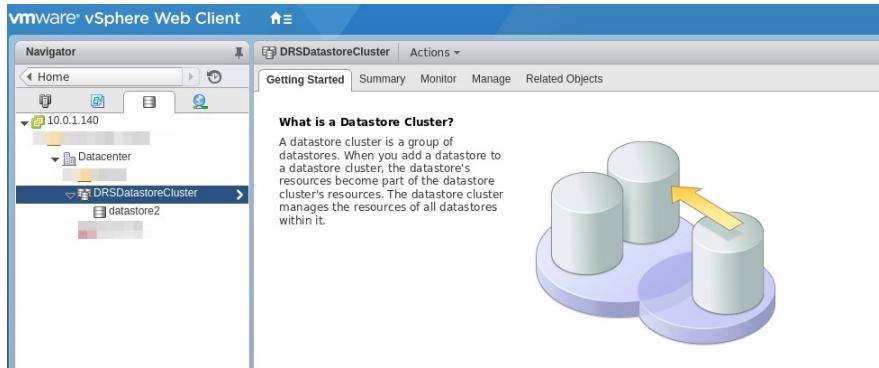
onevcneter または Sunstone を使用して SDRS クラスタをインポートする場合：

- 当該クラスターは SYSTEM データストアとしてのみインポートされます。 vSphere の API では SDRS クラスターへのファイルの直接アップロードや作成が提供されていないため、 IMAGE データストアとして使用することはできません。
- OpenNebula は、 SDRS クラスタごとにグループ化されたデータストアを検出いたします。そのため、それらのデータストアを IMAGE データストアおよび SYSTEM データストアの両方としてインポートすることが可能です。

- 非永続イメージは、vSphere の API が SDRS クラスター全体に対するファイルの作成、コピー、削除を提供していないため、SDRS ではサポートされていません。ただし、SDRS を基盤とする仮想マシンでは、永続イメージおよび揮発性イメージをご利用いただけます。
- OpenNebula では、SDRS 上でのリンクドクローンはサポートされていません。そのため、VM クローンを作成する際にはフルクローンが実行されます。

データストアの選択をSDRSクラスタに委任するには、OpenNebulaのスケジューラに対し、ストレージクラスタを表すSYSTEMデータストアを特に使用したい旨を通知する必要があります。VMテンプレートを編集し、

以下の式を追加してください：ID=DATASTORE\_ID を属性 SCHED\_DS\_REQUIREMENTS に追加します。ここで、DATASTORE\_ID は OpenNebula がデータストアに割り当てた数値 ID に置き換える必要があります。この属性により、OpenNebula は仮想マシンを展開する際、常にこのデータストアを使用します。SCHED\_DS\_REQUIREMENTS を使用しない代替手段として、同じ OpenNebula クラスター内に vCenter クラスターに関連付けられた複数の SYSTEM データストアが存在する場合、StorageDRS クラスターを表すデータストア以外のすべての SYSTEM データストアを無効化することができます。スケジューラは無効化されたデータストアを使用しないためです。



### 6.3.10 vCenter データストアを使用したマーケットプレイス

vCenter データストアは、OpenNebula HTTP および S3 マーケットプレイスと互換性があります。汎用 VM テンプレートが必要となります。OpenNebula マーケットプレイスに関する詳細情報は、こちらでご確認いただけます。

### 6.3.11 チューニングと拡張

各データストアドライバーの専用ガイド、またはストレージサブシステム開発者ガイドをご参照いただければ、ドライバーのカスタマイズは容易に行えます。

必要なファイルは、以下の場所でご確認いただけます：

- /var/lib/one/remotes/datastore/vcenter
- /var/lib/one/remotes/tm/vcenter

## 6.4 vCenter ネットワークの概要

vCenter の仮想ネットワークは、OpenNebula の仮想ネットワークを使用して表現することが可能です。OpenNebula の仮想ネットワークと vSphere のポートグループの間には一対一の関係が存在します。OpenNebulaにおいて、VM テンプレートに NIC を追加する場合、または稼働中の VM に NIC を接続（ホットプラグ）する場合、ネットワークインターフェースを OpenNebula の仮想ネットワークに接続することができます。

OpenNebula はポートグループを利用することも、ポートグループを作成することも可能です。

vSphere の用語では、ポートグループは VLAN タグ付けなどの特定の仕様セットを持つ仮想ポートを作成するためのテンプレートと見なすことができます。VM のネットワークインターフェースは、ポートグループを介して vSphere の仮想スイッチに接続されます。vSphere では 2 種類のポートグループを提供しています：

- 
- ポートグループ（または標準ポートグループ）。このポートグループはvSphere標準スイッチに接続されます。
  - 分散ポートグループ。ポートグループはvSphere分散スイッチに接続されています。[VMWareの](#)

vSphereネットワークガイドによれば、仮想スイッチには2種類あります：

- vSphere 標準スイッチ。物理イーサネットスイッチと同様に動作します。vSphere 標準スイッチは、物理イーサネットアダプター（アップリンクアダプターとも呼ばれます）を使用して物理スイッチに接続し、仮想ネットワークと物理ネットワークを結合することができます。この仮想スイッチを利用可能にしたい各 ESXi ホスト上で、仮想標準スイッチを作成および設定します。
- vSphere 分散スイッチ。データセンター内の関連するすべてのホストにまたがる単一のスイッチとして機能し、仮想ネットワークの集中的なプロビジョニング、管理、監視を提供します。vSphere 分散スイッチは vCenter Server システム上で設定され、その設定はスイッチに関連付けられたすべてのホストに反映されます。これにより、仮想マシンが複数のホスト間で移行する際にも、一貫したネットワーク設定を維持することが可能となります。

---

**注記：**vSphere 分散スイッチは、VMware の vSphere Enterprise Plus ライセンスでのみご利用いただけます。

---

OpenNebula の仮想ネットワークを vSphere のポートグループに関連付けたい場合：

- vSphere の Web クライアントでポートグループを作成し、インポートツールを使用してそれらを利用する方法、または
- OpenNebula から仮想ネットワーク定義を使用して直接ポートグループを作成し、属性 `VN_MAD=vcenter` をネットワークテンプレートに設定し、OpenNebula にネットワーク要素の作成をさせる方法です。

#### 6.4.1 既存の vCenter ポートグループの活用

既存の vCenter ネットワークは、OpenNebula 仮想ネットワークを使用して表現することができます。ただし、仮想ネットワークの BRIDGE 属性が、vCenter で定義されたネットワーク（ポートグループ）の名前と一致する必要があることにご留意ください。

OpenNebula は「ポートグループ」と「分散ポートグループ」の両方をサポートしており、vCenter で定義されたあらゆるネットワークリソースの作成または利用が可能です。

ネットワークは、VLAN などの特定の構成を含め、vSphere の Web クライアントを使用して作成できます。OpenNebula は、これらのネットワークを定義された特性で仮想ネットワークとして表現し、利用します。さらに OpenNebula は、これらのネットワーク上に、イーサネット、IPv4、IPv6 の 3 種類のアドレス範囲を扱うことができます。

vCenter VM テンプレートは独自の NIC を定義することができ、OpenNebula がそれらを管理します。その情報（IP、MAC など）は OpenNebula によって認識されます。OpenNebula VM テンプレート内に存在する NIC、または `attach_nic` 操作を通じて追加された NIC は、OpenNebula によって処理されるため、切断の対象となります。

既存のポートグループを利用する OpenNebula 仮想ネットワークには、`VN_MAD=dummy` 属性が付与されます。

vCenter ネットワークのインポートについては、「*vCenter ネットワークのインポート*」セクションで説明されているインポートツールを使用して、簡単に利用することができます。

セクションで説明されているインポートツールを使用して簡単に vCenter ネットワークを利用できます。

#### 6.4.2 OpenNebula からのポートグループの作成

OpenNebula は、vCenter ネットワークドライバが使用されている場合（属性 `VN_MAD=vcenter` により）、仮想ネットワークテンプレートから vCenter ネットワークを作成することができます。

OpenNebula が vCenter ネットワークを作成する必要がある場合のワークフローは以下の通りです：

1. OpenNebulaのoned設定ファイルにおいて、VNETフックを有効にしてください。
2. 新しいOpenNebula仮想ネットワークテンプレートを作成します。テンプレートに必要な属性を追加し、ネットワーク要素が作成されるvCenterクラスターを表すOpenNebulaのホストIDを含めます。
3. 仮想ネットワークが作成されると、フックにより、指定された vCenter クラスタのメンバーである各 ESX ホスト上で必要なネットワーク要素が作成されます。

4. 仮想ネットワークは、OpenNebulaホストとして表現されるvCenterクラスタを含むOpenNebulaクラスタに自動的に割り当てられます。
5. フックは非同期で動作するため、VCENTER\_NET\_STATE 属性が確認できるまで仮想ネットワーク情報を更新する必要がある場合があります。操作が正常に完了すると、この属性は READY に設定され、仮想マシンやテンプレートから利用可能になります。フックが失敗した場合、VCENTER\_NET\_STATE は ERROR に設定され、VCENTER\_NET\_ERROR 属性により詳細情報が提供されます。

### 6.4.3 フック情報

最初の vCenter クラスタが作成されると、ネットワークの作成と削除を処理するために、OpenNebula に 2 つのフックが登録されます。

- vcenter\_net\_create
- vcenter\_net\_delete

これらのフックは、vCenter ネットワーク要素の作成、および OpenNebula 仮想ネットワークテンプレートの削除時にそれらを削除する責任を担うスクリプトです。

作成フックは、テンプレートに割り当てられたクラスタ内で検出された各ESXホストに対して、標準ポートグループが選択されている場合に以下の処理を実行します：

- ポートグループが存在しない場合、作成します。
- ポートグループまたはスイッチ名が既に存在する場合、接続が切断される可能性のある予期せぬ変更から保護するため、新しい属性を無視して更新は行われません。

分散ポートグループが選択された場合、作成フックは以下の処理を実行します：

- OpenNebula は、分散スイッチが存在しない場合に作成します。スイッチが存在する場合は、設定された属性を無視して更新されません。
- OpenNebula は、vCenter クラスタに関連付けられたデータセンター内に分散ポートグループが存在しない場合、これを自動的に作成いたします。既に分散ポートグループが存在する場合、予期せぬ変更から保護するため、更新は行われません。
- テンプレートに割り当てられたクラスタ内で検出された各ESXホストについて、そのESXホストを分散スイッチに追加します。

作成フックは非同期で実行されます。そのため、VCENTER\_NET\_STATE 属性が設定されたかどうかを確認する必要があります。フックが完了すると、VCENTER\_NET\_STATE は READY 値または ERROR 値のいずれかになります。エラーが発生した場合は、問題の原因を確認できます。

以下は、フックが完了しネットワークが準備完了状態になった際のスクリーンショットです：

vCenter information	
VCENTER_INSTANCE_ID	8C3875CD-275B-4718-BFE4-99739AE06F78
VCENTER_NET_ERROR	
VCENTER_NET_REF	network-1131
VCENTER_NET_STATE	READY
VCENTER_ONE_HOST_ID	66
VCENTER_PORTGROUP_TYPE	Port Group
VCENTER_SWITCH_NAME	[REDACTED]

削除フックは以下の処理を実行します：

- 
- OpenNebulaはvCenterサーバーと通信します。
  - テンプレートに割り当てられた vCenter クラスタ内で検出された各 ESX ホストについて、ポートグループとスイッチの両方の削除を試みます。スイッチにポートグループが残っていない場合、スイッチ自体も削除されます。

この場合、フックは非同期で動作します。成功または失敗を確認するには、以下のコマンドを実行してください:

```
grep EXECUTE /var/log/one/oned.log | grep vcenter_net_delete
```

スクリプトが失敗した場合、/var/log/one/oned.log 内の EXECUTE FAILURE の前の行を確認することで、失敗の詳細情報を得ることができます。削除フックが失敗した場合は、vCenter サーバーを確認し、自動削除できなかったリソースを削除する必要があるかもしれません。

**警告：**ポートグループまたはスイッチが使用中の場合（例：VMが稼働中で、そのポートグループにNICが接続されている場合）、削除操作は失敗します。仮想ネットワークの表現を削除しようとする前に、そのポートグループを使用しているVMやテンプレートがないことを必ずご確認ください。

#### 6.4.4 vCenter ネットワーク属性

Sunstoneから仮想ネットワーク定義を簡単に作成できますが、テンプレートを作成し、onevnetコマンドで適用することも可能です。以下は、TEMPLATEセクション内に追加する必要がある属性をまとめた表です：

属性	タイプ	必須	説明
VN_MAD	string	必須	vcenter に設定する必要があります
BRIDGE	string	必須	ポートグループ名です。
PHYDEV	string	任意	スイッチにアップリンクを割り当てる場合、使用するESXiホストの物理ネットワークインターフェースカード（NIC）の名前を指定できます。複数の物理 NIC 名を使用する場合は、カンマで区切ってください（例：vmnic0,vmnic1）。ただし、2つのスイッチが同じ物理 NIC を共有することはできず、クラスタ内のすべてのESXホストで同じ物理インターフェイス名が存在し、利用可能であることを確認する必要があります。スイッチが既に存在する場合、この属性は無視されます
VCENTER	string	任意	VCENTER_ipOgRYTeGsROUPTThYePreEare には Port Group と Distributed Port Group の2つの値が指定可能です。Port Group は標準ポートグループを意味します。
VCENTER	integer	必須	VCENTER_ipOgRYTeGsROUPTThYePreEare の値を表します。
VCENTER	string	任意	VCENTER_ipOgRYTeGsROUPTThYePreEare の値を表します。
VCENTER	integer	必須	VCENTER_ipOgRYTeGsROUPTThYePreEare の値を表します。
MTU	integer	必須	仮想スイッチの最大伝送単位設定です。スイッチが既に存在する場合、この属性は無視されます。
VLAN_ID	integer	はい (た)	VLAN IDは、定義されていない場合、かつAUTOMATIC_VLAN_IDがYESに設定されている場合に生成されます
AUTOMATIC_VLAN_ID	boolean	ただし (た) ただし	AUTOMATIC_VLAN_ID が設定されている場合 AUTOMATIC_VLAN_ID 必須であり、VLAN_ID が定義されていない場合は YES に設定する必要があります。そうすることで、OpenNebula 自動的に VLAN ID を作成した場合
VLAN_ID	integer	なし	VLAN_ID
VCENTER	boolean	必須	VCENTER_ipOgRYTeGsROUPTThYePreEare この属性は、vcen-ter_vnet_delete フックによる誤削除を防ぐための保護メカニズムです

す

### 6.4.5 OpenNebulaによって作成された仮想スイッチおよびポートグループに適用される設定

OpenNebulaは、vCenter内で仮想スイッチおよびポートグループを作成する際、vSphere Webクライアントが同等の操作で使用する値に基づき、以下の値を採用しております：

- VLAN ID は 0 に設定されます。これは VLAN を使用しないことを意味します。
- MTU 値は 1500 に設定されます。

OpenNebulaによって作成される標準ポートグループには、以下の設定が適用されます：

- ポート数は「Elastic」に設定されています。VMWare のドキュメントによれば、Elastic モードは ESXi ホスト上で仮想スイッチのポートを動的に拡張・縮小し、リソースの効率的な利用を確保するために使用されます。いずれにせよ、標準スイッチのデフォルトポート番号は 128 です。
- セキュリティ - プロミスキャスマードは「拒否」に設定されています。これは、仮想ネットワークアダプタが自身宛てのフレームのみを受信することを意味します。
- セキュリティ - MAC アドレス変更は「許可」に設定されています。これにより、ESXi ホストは初期 MAC アドレス以外の有効な MAC アドレスへの変更要求を受け入れます。
- セキュリティ - 偽装送信は「許可」に設定されています。これは、ESXi ホストが送信元 MAC アドレスと実効 MAC アドレスを比較しないことを意味します。
- ポートグループにおける帯域幅およびバーストサイズを制御するトラフィックシェーピングポリシーは無効化されています。テンプレート内の各 NIC に対して QoS を設定することは可能です。
- 物理 NIC。アップリンクとして使用される物理 NIC は、チーム機能を備えたボンディングブリッジでブリッジ接続されています。OpenNebula によって作成された分散ポートグループには、以下の設定が適用されます：

- ポート数は「Elastic」に設定されています。VMWare のドキュメントによれば、Elastic モードは仮想スイッチのポートが動的にスケールアップ/ダウンされる ESXi ホスト上で、リソースの効率的な使用を確保するために使用されます。分散スイッチのデフォルトのポート番号は 8 です。
- 静的バイニング。仮想マシンを分散ポートグループに接続すると、ポートが直ちに割り当てられ、予約されます。これにより、常に接続性が保証されます。ポートは、仮想マシンがポートグループから削除された場合にのみ切断されます。
- 自動拡張が有効化されています。ポートグループがポート不足になりそうな場合、事前に定義されたわずかな余裕分だけ自動的に拡張されます。
- 早期バイニングが有効化されています。仮想マシンがポートグループに接続するよう再構成された際、空き状態の分散仮想ポートが選択され、仮想マシンに割り当てられます。

### 6.4.6 OpenNebula 仮想ネットワークテンプレート (Sunstone)

本セクションでは、Sunstone ユーザーインターフェースを使用して仮想ネットワーク定義を作成する方法について説明し、vcenter ネットワークドライバで利用可能な属性を紹介します。

最初のステップでは、仮想ネットワークの名前を入力する必要があります：

### Create Virtual Network

The screenshot shows the 'Create Virtual Network' interface. At the top, there are buttons for 'Reset' and 'Create'. Below them is a navigation bar with tabs: General (selected), Conf, Addresses, Security, QoS, and Context. Under the General tab, there are two input fields: 'Name' containing 'vcenter\_network\_test' and 'Description' which is empty.

[Conf]タブで、[Network Mode]メニューから[vCenter]を選択します。これによりvcenterネットワークドライバが使用されます（OpenNebulaのテンプレートにVN\_MAD=vcenter属性が追加されます）。[Bridge name]はポートグループ名となります。デフォルトでは仮想ネットワーク名と同じですが、別のポートグループ名を選択することも可能です。



vCenterのネットワークモードを選択すると、Sunstoneでは定義可能な複数のネットワーク属性が表示されます。

The screenshot shows the Sunstone interface for vCenter mode. It includes fields for VLAN ID (Automatic VLAN ID), Physical device, MTU of the interface, Switch name, Number of ports, Port group type (Port group), and OpenNebula's Host ID (Please select).

これらの属性に関する詳細は「[vCenter ネットワーク属性](#)」セクションに記載されていますが、一部について説明いたします：

#### OpenNebulaホストのID

vCenterドライバーを使用して仮想ネットワークを作成するには、この仮想ネットワークを関連付ける対象となるvCenterクラスター（OpenNebulaホストとして表現されます）を選択する必要があります。OpenNebulaは、vCenterクラスターのメンバーである各ESXホストに対して動作を行います。

#### 物理デバイス

スイッチにアップリンクを割り当てる場合、使用する ESXi ホストの物理ネットワークインターフェースカード名を指定できます。複数の物理 NIC 名はカンマで区切って指定可能です（例: vm-nic0,vmnic1）。ただし、2つのスイッチが同一の物理 NIC を共有しないこと、またクラスタ内の全 ESX ホストで同一の物理インターフェース名が存在し利用可能であることを必ず確認してください。

具体例をご説明いたします。新しい仮想スイッチにポートグループを作成する場合、まずvCenterクラスタ内のホストで空き状態かつ未割り当てる物理アダプタを確認します。私のクラスタには2台のホストがあります：

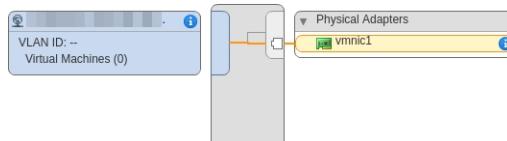
最初のホストでは、vmnic1アダプターは空き状態であり、いずれのvSwitchにも割り当てられていません：

Device	Actual Speed	Configured Speed	Switch	MAC Address
<b>Intel Corporation 82576 Gigabit Network Connection</b>				
vmnic0	1000 Mb	Auto negotiate	vSwitch0	78:e3:b5:11:ab:4a
<b>vmnic1</b>	1000 Mb	Auto negotiate	--	78:e3:b5:11:ab:4b

2番目のホストでは、vmnic1、vmnic2、vmnic3インターフェースが空き状態です：

Device	Actual Speed	Configured Speed	Switch	MAC Address
<b>Intel Corporation 82571EB Gigabit Ethernet Controller</b>				
vmnic2	Down	Auto negotiate	--	00:23:8b:ce:a8:7e
vmnic3	Down	Auto negotiate	--	00:23:8b:ce:a8:7f
<b>Intel Corporation 80003ES2LAN Gigabit Ethernet Controller</b>				
vmnic0	1000 Mb	Auto negotiate	vSwitch0	00:23:8b:ce:a8:7c
<b>vmnic1</b>	Down	Auto negotiate	--	00:23:8b:ce:a8:7d

したがって、アップリンクを指定する場合、両方のESXホストで使用可能な唯一のアダプターは**vmnic1**となり、OpenNebulaは必要に応じてスイッチとアップリンクを作成します：



## ポート数

この属性はオプションです。この属性を使用すると、仮想スイッチが使用するポート数を指定できます。ここに値を設定する場合は、お使いのvSphereプラットフォームがサポートする最大値を必ずご確認の上、ご理解ください。

## VLAN ID

この属性はオプションです。手動でVLAN IDを設定する、OpenNebulaに自動VLAN IDの生成を強制する、またはVLANを使用しない設定が可能です。この値はVLAN\_ID属性に割り当てられます。

## アドレス範囲

仮想ネットワークを作成するには、[アドレス]タブでアドレス範囲を追加する必要があります。詳細は「仮想ネットワーク定義」セクションをご参照ください。

## 6.4.7 制限事項

OpenNebulaはESXホストの同期を行いません。仮想ネットワーク作成後に追加または削除されたESXホスト上のポートグループやスイッチについては、OpenNebulaによる作成・削除は行われません。例えば、vMotionやDPMをご利用中でESXホストが電源投入された場合、そのホストにはOpenNebulaが作成したスイッチやポートグループが存在しないため、仮想マシンを当該ホストへ移行

---

することはできません。

**仮想ネットワークの更新はサポートされていません。**仮想ネットワーク定義を更新した場合、OpenNebula は既存のポートグループやスイッチの属性を更新しません。そのため、仮想ネットワークを削除し、新しい属性で新規に作成する必要があります。

**セキュリティグループについて。**セキュリティグループはvSphereスイッチモードではサポートされていません。

**ネットワークエイリアスについて。**vCenterではネットワークインターフェースのエイリアスを使用することが可能ですが、仮想マシンが稼働中にエイリアスを割り当てた場合、その設定は次の再起動時（OpenNebulaデプロイ時）に有効となります。マシンを再起動たくない場合は、マシンのプロンプトで次のコマンドを手動で実行してください：

```
$ /usr/sbin/one-contextd all reconfigure
```

**ネットワークのインポートについて。**OpenNebulaは、いずれのホストにも属していないネットワークはインポートしません。分散ポートグループの場合、DVSに接続されているホストが存在しない場合も同様です。

## 6.4.8 ネットワーク監視

OpenNebulaは各仮想マシン（VM）のネットワーク監視情報を収集します。パフォーマンスマネージャーによりvCenterからリアルタイムデータが取得され、20秒ごとにデータを収集し1時間保持します。リアルタイムサンプルを使用するため、vCenterの統計設定に変更を加える必要はありません。送信および受信トラフィックのネットワークメトリクスはKB/s単位の平均値として提供されます。

Sunstoneが提供するグラフは、vCenterの「監視」→「パフォーマンス」タブにおいて、時間範囲ドロップダウンメニューで「リアルタイム」を選択した場合、または詳細ビューで「ネットワークビュー」を選択した場合に表示されるグラフとは異なります。その理由は、Sunstoneがグラフの時間基準としてポーリング時間を使用するのに対し、vCenterはサンプリング時間を使用するためです。そのため、vCenterのサンプリングデータをポーリング間隔で集計し、実値に近似させる必要があります。その結果、アップロードとダウンロードのピーク値は異なる値となり、ポーリング間の異なるピークは表示されません。Sunstoneのグラフはネットワーク動作に関する有用な情報を提供し、後ほどvCenterでより詳細に検証することが可能です。

## 6.5 NSX セットアップ

NSXは、VMwareが提供するネットワークおよびセキュリティソフトウェアであり、仮想クラウドネットワークを通じて、データセンター、マルチクラウド、ベアメタル、コンテナインフラストラクチャにまたがるアプリケーションの接続と保護を実現します。VMware NSX Data Centerは、完全なレイヤー2からレイヤー7までのネットワークおよびセキュリティ仮想化プラットフォームを提供し、ソフトウェアのみのソリューションによる俊敏性、自動化、そして大幅なコスト削減を実現します。

OpenNebulaは、以下の方法で NSX-V および NSX-T の論理スイッチを管理できます：

- 既存のトランスポートゾーン内に新しい論理スイッチを作成します。
- インポートされた vCenter クラスタから論理スイッチをインポートします。
- OpenNebulaに作成またはインポートされた論理スイッチを削除します。
- 作成またはインポートされた論理スイッチを仮想マシンに接続します。
- 作成またはインポートされた論理スイッチを仮想マシンから切り離すこと。

### 6.5.1 要件

## **NSX Manager**

NSX アプライアンスは、1 つの IP アドレスのみでデプロイされている必要があります。OpenNebula のインストール環境は、必要な認証情報を使用して NSX Manager に接続できる必要があります。

## コントローラノード

少なくとも1つのコントローラノードがデプロイされている必要があります。

## ESXiホスト

クラスタ内のすべての ESXi は、NSX に対応した準備が整っている必要があります。

## トランSPORTゾーン

少なくとも1つのトランSPORTゾーンを作成する必要があります。

**警告 :**輸送ゾーン名には英数字 (a-z、 A-Z、 0-9) およびアンダースコア (\_) のみを含めることができます。

## 論理スイッチ

OpenNebula NSX-V 統合の使用開始前に論理スイッチを設定することは必須ではありませんが、論理スイッチが正常に機能していることを確認するため、vCenter からトランSPORTゾーンに論理スイッチを作成し、2台の仮想マシンにアタッチしてオーバーレイネットワークが動作することをテストすることをお勧めいたします。

### 6.5.2 NSX ドライバーの制限事項

- トランSPORTゾーンの作成/変更/削除はできません
- 論理スイッチ作成時には、すべてのパラメータが利用可能ではありません
- ユニバーサル論理スイッチはサポートされていません
- vCenter Server 1台につき、NSX Manager は1つのみサポートされます
- NSXクラスタの準備プロセスは、NSX Managerから行う必要があります
- インポートされたネットワークは、NSX IDではなくvCenter IDで動作します
- トランSPORTゾーン名には英数字とアンダースコアのみを含めることができます

### 6.5.3 OpenNebulaへのNSX Managerの追加

これは半自動のプロセスです。vCenterがNSX Managerに接続されると、次の監視実行時にOpenNebulaがこれを検出し、UIに「NSX」という新しいタブが表示されます。このタブでは、NSX Managerへの接続に必要な認証情報（ユーザー名とパスワード）の設定ができます。NSX-VまたはNSX-Tと連携する準備が整った新しいvCenterクラスターをインポートする場合も、同様のプロセスが適用されます。

本セクションでは、vCenterクラスターのインポートからOpenNebulaがNSX情報を正しく取得しているか確認するまでの全プロセスを網羅し、OpenNebulaをNSXと連携させるための設定方法を詳細に説明します。

### **OpenNebulaへのvCenterクラスターの追加**

最初のステップは、OpenNebulaにESXiクラスターを追加することです。このクラスターは、NSX-VまたはNSX-Tと連携するために必要なすべての要件を満たしている必要があります。クラスターの追加方法は、従来通り以下の2通りです：

## Sunstoneからのインポート

## CLIからのインポート:

```
$ onevcenter hosts --vcenter <vcenter_fqdn> --vuser <vcenter_user> --vpass <vcenter_
˓パスワード>
```

## フックの確認

vCenterクラスタがOpenNebulaにインポートされると、以下の2つのフックが作成されます：

- vcenter\_net\_create
- vcenter\_net\_delete

これらのvCenterフックの一覧表示、作成、削除に関する詳細情報は、[vCenter フック](#)をご覧ください。

## NSX Manager の自動検出属性を確認する

vCenter クラスタのインポートが完了し、監視サイクルが終了すると、そのクラスタに登録されている NSX Manager が検出されます。この情報は以下の場所でご確認いただけます：

インフラストラクチャ > ホスト

目的の OpenNebula ホストをクリックすると、[属性] セクションに以下の情報が表示されます。特に以下の属性が取得され

ます：

- **NSX\_MANAGER**: 該当する NSX Manager の URL を格納しています

- 
- **NSX\_TYPE:** NSX-V または NSX-T のいずれかを示します

### Attributes

AVAILHOST	3	 
CPUSPEED	2100.0	 
CUSTOMIZATION		 
NAME	customLinux01	 
TYPE	Linux	 
HYPERSVISOR	vcenter	 
IM_MAD	vcenter	 
NSX_LABEL	NSX Manager	 
NSX_MANAGER	https://10.0.1.30:443	 
NSX_STATUS	Missing NSX_USER	 
NSX_TYPE	NSX-V	 
NSX_VERSION	6.4.5.13282012	 
STATUS	green	 
TOTALHOST	3	 
TOTAL_WILDS	17	 
VM_MAD	vcenter	 

- **NSX\_VERSION**: その NSX Manager のバージョン
- **NSX\_STATUS**: 最後の NSX Manager チェックの状態を記述します

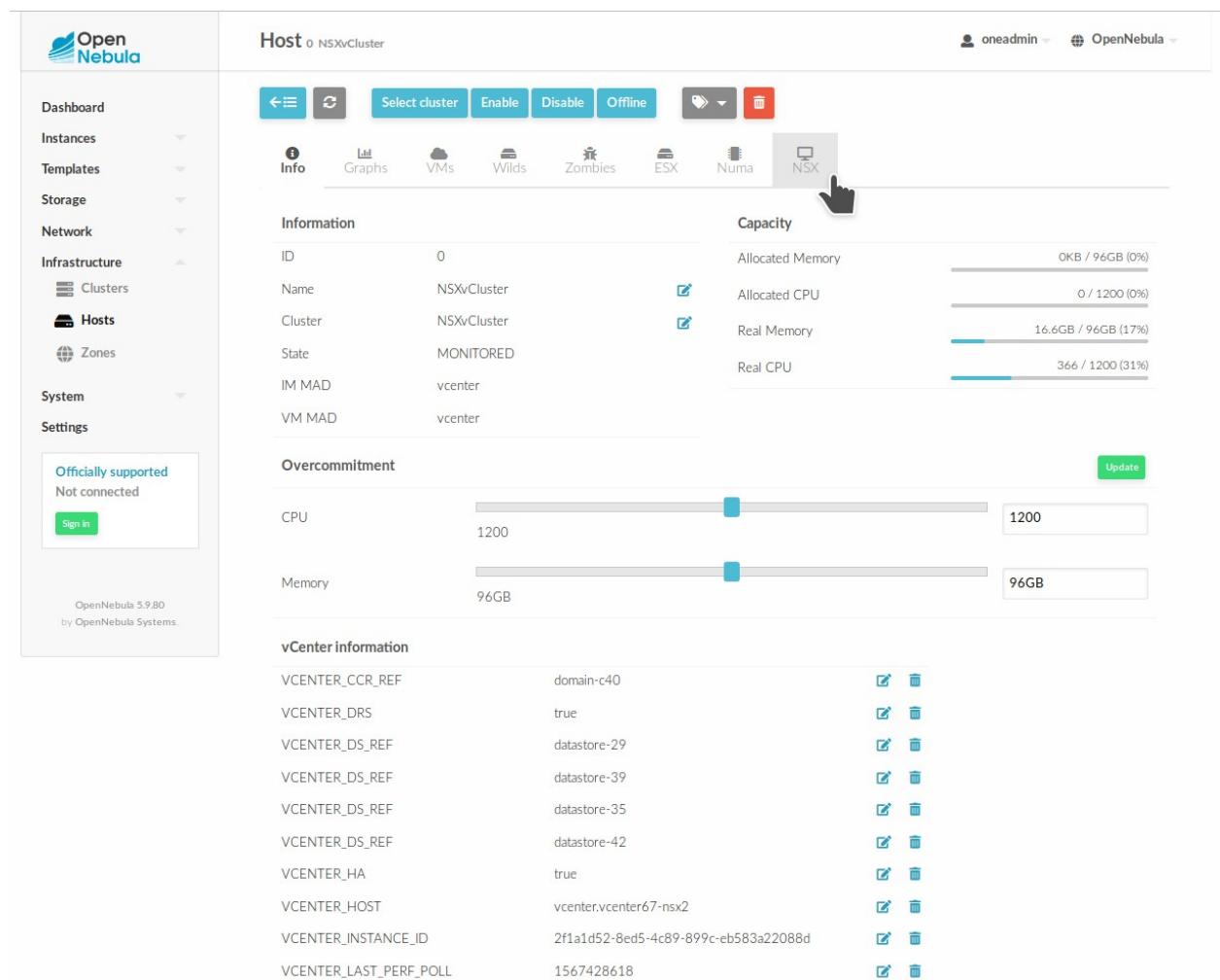
これらのパラメータの詳細な説明は、[NSX 属性のセクション](#)に記載されております。

### NSX Manager 認証情報の設定

vCenter クラスターが OpenNebula ホストとしてインポートされたら、次のステップとして NSX 認証情報を設定します。ホストに「NSX」という新しいタブが表示されます：

インフラストラクチャ > ホスト

該当するホストをクリックした後：



The screenshot shows the OpenNebula web interface with the following details:

- Left Sidebar:** Includes 'Dashboard', 'Instances', 'Templates', 'Storage', 'Network', 'Infrastructure' (with 'Clusters', 'Hosts', 'Zones' sub-options), 'System', and 'Settings'. A note says 'Officially supported' and 'Not connected'. A 'Sign in' button is present.
- Header:** Shows 'Host 0 NSXvCluster', user 'oneadmin', and dropdown for 'OpenNebula'.
- Top Bar Buttons:** Includes 'Select cluster', 'Enable', 'Disable', 'Offline', and several icons for 'Info', 'Graphs', 'VMs', 'Wilds', 'Zombies', 'ESX', 'Numa', and 'NSX'.
- Information Section:** Displays host details like ID (0), Name (NSXvCluster), Cluster (NSXvCluster), State (MONITORED), IM MAD (vcenter), and VM MAD (vcenter). It also shows capacity metrics: Allocated Memory (0KB / 96GB 0%), Allocated CPU (0 / 1200 0%), Real Memory (16.6GB / 96GB 17%), and Real CPU (366 / 1200 31%).
- Overcommitment Section:** Shows CPU and Memory overcommitment levels. CPU is set to 1200, and Memory is set to 96GB.
- vCenter Information Section:** Lists vCenter-related variables such as VCENTER\_CCR\_REF, VCENTER\_DRS, VCENTER\_DS\_REF, VCENTER\_DS\_REF, VCENTER\_DS\_REF, VCENTER\_DS\_REF, VCENTER\_HA, VCENTER\_HOST, VCENTER\_INSTANCE\_ID, and VCENTER\_LAST\_PERF\_POLL, each with edit and delete icons.

NSXタブをクリックし、NSXの認証情報を入力してください：

「送信」をクリックすると、認証情報が NSX Manager に対して検証されます。

- 認証情報が有効な場合、メッセージが表示され、認証情報が保存されます。
- 資格情報が無効な場合、エラーが表示されます。こ

---

れで、NSX 資格情報は 2 つの新しい属性に保存されます：

- **NSX\_USER:** NSX Manager ユーザー名
- **NSX\_PASSWORD:** NSX Managerの暗号化パスワード

OpenNebulaからはトランSPORTゾーンを作成できませんので、ご注意ください。トランSPORTゾーンの作成は必須要件となります。ただし、NSX ManagerでのトランSPORTゾーンの追加はサポートされており、OpenNebulaは次の監視サイクル後にそれらを検出いたします。

#### NSX ステータスの確認

NSX のステータスを確認するには、

以下の手順に従ってください

:

インフラストラクチャ > ホス  
ト

目的のホストをクリックし、「属性」セクションをご確認ください

正常に動作している場合、以下の2つの属性が表示されます：

- **NSX\_STATUS=OK**
- **NSX\_TRANSPORT\_ZONES** = 利用可能なトランSPORTゾーンを格納します。

#### 6.5.4 NSX の編集不可属性

これらの属性は自動検出され、手動で変更することはできません。

**Attributes**

AVAILHOST	3	 
CPUSPEED	2100.0	 
CUSTOMIZATION		
NAME	customLinux01	 
TYPE	Linux	 
HYPERSVISOR	vcenter	 
IM_MAD	vcenter	 
NSX_LABEL	NSX Manager	 
NSX_MANAGER	https://10.0.1.30:443	 
NSX_PASSWORD	019WN+Ga8YIBwwLTnynbMg==	 
NSX_STATUS	Missing NSX_USER	 
NSX_TYPE	NSX-V	 
NSX_USER	admin	 
NSX_VERSION	6.4.5.13282012	 
STATUS	green	 
TOTALHOST	3	 
TOTAL_WILDS	17	 
VM_MAD	vcenter	 

## 属性 AVAILHOST

CPUSPEED	3100.0		
カスタマイズ			
NAME	カスタムLinux01		
タイプ	Linux		
イパーバイザ	vcenter vcenter		
IM	https CIO.O.1.3O:443		
MADNSX_LABEL	SCX+mis ZZSX--		
NSX マネージャー	OK		
NSX_パスワード NSX ステータス	vdriscope—3 vdnscope-4 vdnscope-S		
NSX_トランスポートゾーン TZ	NSX-V		
TZ2	管理者		
TZ3	6.4.5.132B2012		
NSX_タイプ	緑		
NSX_ユーザー	3		
NSX_バージョン	17		
ステータス 総ホスト数	vcenter		
ストレージ 総サイズ			
ルドホスト数			
VM MAD			



属性	タイプ	値	説明
NSX_LABEL	文字列	「NSX - マネージャー」   「NSX-T Manager」	NSX Manager のタイプを示すラベル
NSX_MANAGER	文字列	エンドポイントのURL	NSX Manager の URL を含むエンドポイントです。OpenNebula はコマンドを送信するためにその URL にアクセスする必要があります
NSX_STATUS	文字列	可能な値は以下の通りです：	最新の NSX ステータスを説明します
	OK	NSX_USER および NSX_PASSWORD は正しく、検証クエリが正常に実行されました	
	NSX_USER が不足しています	NSX_USER 属性が設定されていません	
	NSX_PASSWORD が欠落しています	NSX_PASSWORD 属性が設定されていません	
	NSX_TYPE が欠落しています	NSX_TYPE 属性が検出されませんでした	
	欠落 NSX_MANAGER	属性 NSX_MANAGER が検出されませんでした	
	応答コードが不正です 正	検証クエリの応答に問題がありました。通常は無効なユーザー名またはパスワードによるものです	
	NSX_MANAGERへの接続エラー NSX_MANAGER	NSX_MANAGER の IP アドレスが正しくないか、通信に問題が発生しています NSX Manager との通信に問題があります	
NSX_TRANSPOR	TH_ZAOSHNEASR	R[A T Y Z_N A M E => TZ_ID, ... ]	検出されたすべてのトランスポートゾーンのリスト
NSX_TYPE	文字列		NSX-V または NSX-T のインストールであるかどうかを判断します
NSX_VERSION	文字列		NSX のインストール済みバージョン

## 6.5.5 NSX の編集可能な属性

これらのパラメータは、NSXタブから手動で入力する必要があります

パラメータ	タイプ	必須	説明
NSX_USER	文字列	はい	NSX Manager ユーザー
NSX_PASSWORD	文字列	はい	NSX Manager パスワード

## 6.5.6 ドライバの調整

ドライバーは簡単にカスタマイズできます。統合ガイドの「NSX ドライバー」セクションをご参照ください。



## オープンクラウドホストの設定

### 7.1 概要

ホストとは、ハイパーバイザ（KVM または LXD）がインストールされたサーバーであり、稼働中の仮想マシンを実行します。これらのホストは KVM または LXD ドライバーによって管理され、仮想マシンとそのライフサイクルを管理するために必要な操作を実行します。本章では KVM または LXD ドライバーを詳細に分析し、OpenNebula クラウドへの KVM または LXD ホストの設定および追加方法など、さまざまな手法をご提供いたします。

#### 7.1.1 本章の読み方

本章をお読みになる前に、[フロントエンド](#)、[KVMホスト](#)および[またはLXDホスト](#)のインストールが完了し、少なくとも1つの仮想化ノードを備えたOpenNebulaクラウドが稼働している状態であることを前提とします。

本章では、ホストの設定オプションに焦点を当てて説明いたします。

- KVMホストの設定および管理手順を理解するため、[KVM ドライバ](#)のセクションをお読みください。
- LXDホストの設定および管理手順について理解するには、[LXD ドライバ](#)のセクションをお読みください。
- 監視セクションでは、OpenNebula がホストおよび仮想マシンをどのように監視しているか、またそのサブシステムの設定で変更可能な点についてご確認いただけます。
- [PCI/バススルーリー](#)の実行にご興味がある場合は、このセクションをお読みください。この

章をお読みになった後は、[Open Cloud Storage](#)の章をお読みください。

#### 7.1.2 ハイパーバイザの互換性

本章はKVMおよびLXDに適用されます。

vCenter に関する同様のガイドについては、「[vCenter ノード](#)」セクションに従ってください。

### 7.2 KVM ドライバー

KVM（カーネルベース仮想マシン）は、[OpenNebula](#)の[Open Cloud Architecture](#)におけるハイパーバイザです。KVMはLinux向けの

完全な仮想化システムであり、各仮想マシンが独自の仮想化ハードウェアと相互作用する完全仮想化を提供します。本ガイドでは、OpenNebulaにおけるKVMの使用方法について説明します。

## 7.2.1 要件

仮想化をサポートするためには、ホストに Intel VT または AMD の AMD-V 機能を備えた CPU が必要です。KVM の「[KVM を使用するための準備](#)」ガイドでは、お使いのハードウェアが KVM をサポートしているかどうかに関する疑問を解消いたします。

KVMのインストールと設定は、「[KVMホストのインストール](#)」セクションの手順に従って行います。

## 7.2.2 考慮事項と制限事項

可能な限り、ネットワークとディスクの両方で [virtio](#)をご利用ください。ネットワークとディスクの両方でエミュレートされたハードウェアを使用すると、パフォーマンスに影響が生じ、利用可能な機能のすべてが提供されません。例えば、ディスクドライバに [virtio](#)を使用しない場合、コントローラに接続できるデバイス数が制限されます。これはディスクの接続数に上限が生じ、仮想マシン稼働中のディスク接続（ライブディスク接続）が不可能になることを意味します。

## 7.2.3 設定

### KVM の設定

OpenNebula パッケージは KVM を自動的に設定しますので、追加の手順は不要です。

### ドライバ

OpenNebulaでは、KVM ドライバーはデフォルトで有効化されています：

```
-----  
#-----  
#   KVM仮想化ドライバ管理者の設定  
#  
#   -r ホスト監視時の再試行回数  
#  
#   -t スレッド数、すなわち同時に監視するホストの数  
#  
#   -l <アクション[=コマンド名]> ローカルで実行されるアクション。各アクションごとにコマンドを上書きできます。  
#  
#       有効なアクション: deploy, shutdown, cancel, save, restore, migrate, poll例: "-l  
#  
#           migrate=migrate_local,save"  
#  
#   -p ホストごとに複数のアクションを並行実行します。ハイパーバイザーのサポートが必要です  
#  
#   -s <シェル> リモートコマンドを実行します。デフォルトはbashです  
#-----  
  
注記: CPUに仮想化拡張機能がない場合や、ネストされたQemu-KVMホストを使用する場合などには、type = "qemu" を指定してQemuエミュレーションゲストをご利用いただけます。  
-----  
VM_MAD = [  
    NAME      =  
    "kvm", SUNSTONE_NAME = "KVM",  
    実行可能ファイル = "one_vmm_exec", 引数  
    = "-t 15 -r 0 kvm",  
    デフォルト     = "vmm_exec/vmm_exec_kvm.conf", TYPE =  
    "KVM",  
    スナップショット保持= "いいえ",  
    インポートされたVMSアクション = "終了, 強制終了, 保留, 解放, 一時停止, 再開, 削除, 再起動, 強制再起動, スケ  
    ジュール変更, スケジュール解除, ディスク接続, ディスク切断, NIC接続, NIC切断, スナップショット作成, ス  
    ナップショット削除"  
]
```

設定パラメータ: -r、-t、-l、-p、および -s は、すでに適切なデフォルト値で事前設定されています。これらを変更する場合は、OpenNebula を再起動する必要があります。

これらのパラメータの詳細、およびドライバのカスタマイズや拡張方法については、仮想マシンドライバリファレンスをご参照ください。

### ドライバのデフォルト設定

KVMが仮想マシンを起動するために必要な属性があります。これらに適したデフォルト値を設定することで、すべての仮想マシンに必要な値が適用されます。これらの属性は/etc/one/vmm\_exec/vmm\_exec\_kvm.confで設定されます。KVM向けに設定可能な項目は以下の通りです：

- EMULATOR: KVM実行ファイルへのパス。
- OS: KERNEL、INITRD、BOOT、ROOT、KERNEL\_CMD、MACHINE、ARCH、SD\_DISK\_BUS の属性。
- VCPU
- FEATURES: ACPI、PAE の属性。
- CPU\_MODEL: MODEL 属性を指定します。
- ディスク: 属性 DRIVER および CACHE。すべてのディスクは、そのドライバーとキャッシュアルゴリズムを使用します。
- NIC: FILTER 属性。
- RAW: ドメイン XML ファイルに libvirt 属性を追加します。
- HYPERV: Hyper-V 拡張機能を有効化します。
- SPICE: SPICE用のデフォルトデバイスを追加します。

**警告:** これらの値は仮想マシン作成時のみ使用されます。NICやディスクの接続/切断などの他の操作については、デフォルト値を/var/lib/one/remotes/etc/vmm/kvmrcに設定する必要があります。詳細は「[ファイルとパラメータ](#)」セクションをご確認ください。

例：

```

OS          = [ ARCH = "x86_64" ]
機能      = [ PAE = "無効", ACPI = "有効", APIC = "無効", HYPERV = "無効", GUEST_AGENT = "無効"
             ]
ディスク    = [ ドライバ = "raw" , キャッシュ = "なし" ]
HYPERV_OPTIONS=<リラックス状態='on' /><vapic状態='on' /><スピンドルロック状態='on' リトライ=
             ↵'4096' />
SPICE_OPTIONS=""
<video>
  <model type='qxl' heads='1' />
</video>
<サウンド モデル='ich6' />
<channel type='spicevmc'>
  <ターゲット タイプ='virtio' 名前='com.redhat.spice.0' />
</channel>
<redirdev bus='usb' type='spicevmc' />
<redirdev bus='usb' type='spicevmc' />
<redirdev bus='usb' type='spicevmc' />
```

---

注記：これらの値は、クラスター、ホスト、および VM テンプレートで上書きすることができます。

### その他のキャッシング設定に対するライブマイグレーション

キャッシング設定が「none」以外になっているディスクをご利用の場合、libvirtのバージョンによってはライブマイグレーションに問題が生じる可能性があります。virshコマンドに--unsafeパラメータを追加することで、マイグレーションを有効にできます。変更が必要なファイルは/var/lib/one/remotes/etc/vmm/kvm/kvmrcです。以下の行のコメントを外し、その後onehost sync --forceを実行してください：

```
MIGRATE_OPTIONS==unsafe
```

### タイムアウトの設定（任意）

必要に応じて、仮想マシンのシャットダウン操作に対するタイムアウトを設定することができます。この機能は、仮想マシンがシャットダウン中に停止した状態になる場合（あるいは単にシャットダウンコマンドを認識しない場合）に有用です。デフォルトでは、タイムアウト時間経過後、仮想マシンは実行状態に戻りますが、猶予時間経過後に仮想マシンを破棄するように設定することも可能です。この設定は

/var/lib/one/etc/remotes/vmm/kvm/kvmrc で設定します。

```
シャットダウン後のタイムアウトまでの待機時間 (秒)
export SHUTDOWN_TIMEOUT=300

# シャットダウンタイムアウト後にVMの強制終了を行う場合は、この行のコメントを外してください#export
FORCE_DESTROY=yes
```

### cgroups の操作（オプション）

cgroups は、カーネルの機能の一つであり、特定のプロセスに割り当てられるリソースの量を制御することを可能にします（その他の機能も含まれます）。OpenNebula テンプレートで定義された通り、VM に割り当てられる CPU 量を強制するために使用できます（例：CPU=0.5 の VM は、CPU=1.0 の VM よりも物理 CPU サイクルの半分を取得します）。cgroupsの設定は、フロントエンドではなく、各ハイパーバイザーホスト上で（必要な場合に）行います。

**注記：**systemd を実行している現在のオペレーティングシステムでは、cgroups は libvirt/KVM によって自動的に有効化され、使用されます。設定は不要です。ツール lscgroup（RHEL/CentOS では配布パッケージ libcgrouptools に、Debian/Ubuntu では cgroup-tools に含まれています）を使用して、システム上の cgroups の状態を確認できます。ツールの実行でエラー出力（例：以下）が表示される場合、cgroups は利用できません。

```
$ lscgroup
cgroups can't be listed: Cgroup is not mounted
```

cgroups を有効化し設定するには、お使いのオペレーティングシステムのドキュメントに従ってください。

Cgroupsを使用することで、仮想マシンが使用できる物理RAMの総量を制限することが可能です。これにより、常に一定量をホストOSに確保しておくことができます。この場合、ホストまたはクラスタテンプレートにおいてRESERVED\_MEMパラメータも設定されることをお勧めいたします。

OpenNebulaは、VMテンプレートのCPU属性に比例したCPUシェア数を自動的に生成します。例えば、2台のVM（ID 73と74、CPU=0.5およびCPU=1）を実行するホストは、以下の方法で設定する必要があります：

```
/sys/fs/cgroup/cpu,cpuacct/machine.slice/
|-- cgroup.clone_children
|-- cgroup.event_control
7.2. KVM ドライバー
...
|-- cpu.shares
|-- cpu.stat
|-- machine-qemu\x2d1\x2done\x2d73.scope
```

```

|   |-- cgroup.clone_children
|   |-- cgroup.event_control
|   |-- cgroup.procs
|   |-- cpu.shares
|
|   ...
|
|   -- vcpu0
|       |-- cgroup.clone_children
|       ...
|
|   -- machine-qemu\x2d2\x2done\x2d74.scope
|       |-- cgroup.clone_children
|       |-- cgroup.event_control
|       |-- cgroup.procs
|       |-- cpu.shares
|
|       ...
|
|       -- vcpu0
|           |-- cgroup.clone_children
|
|           ...
|
|   -- 解放時の通知
`-- タスク

```

各VMのCPUシェアは以下の通りです:

```

$ cat '/sys/fs/cgroup/cpu,cpuacct/machine.slice/machine-qemu\x2d1\x2done\x2d73.scope/
`cpu.shares'
512
$ cat '/sys/fs/cgroup/cpu,cpuacct/machine.slice/machine-qemu\x2d2\x2done\x2d74.scope/
`cpu.shares'
1024

```

---

**注記:** cgroups (ディレクトリ) のレイアウトは、ご利用のオペレーティングシステムや設定によって異なる場合があります。

libvirtのドキュメントでは、すべてのケースと、libvirt/KVMによるcgroupsの管理方法について説明されています。

---

仮想CPUは固定されていないため、仮想マシンのプロセスが使用する物理コアは頻繁に変更される可能性が高いです。仮想マシンが物理ホスト上で単独で動作している理想的なケースでは、消費されるCPUの総量は仮想CPU数に仮想化オーバーヘッド（例：ネットワーク処理）を加えた値となります。一方、物理ノード内に複数の仮想マシンが存在し、かつ高負荷状態にある場合、仮想マシンは物理CPU時間を競合して使用することになります。この場合、cgroupsが仮想マシン間でCPU時間を公平に分配します（CPU=2の仮想マシンはCPU=1の仮想マシンと比較して2倍の時間を割り当てられます）。

オーバーコミット (CPU=VCPU) を行わない場合、すべての仮想CPUは（ピン留めされていなくても）1つの物理CPUを割り当たられるため、割り当たされたVCPU数から仮想化オーバーヘッドとホストOS上で実行中のプロセスによる消費分を差し引いた分を使用できます。

## 7.2.4 使用方法

### KVM固有の属性

以下は KVM に固有のテンプレート属性です。VM を定義するためにサポートされている属性の完全なリストについては、テンプレートリファレンスドキュメントをご参照ください。

### ディスク

- **TYPE:** この属性は、仮想マシンに公開されるメディアの種類を定義します。可能な値は、disk (デフォルト) またはcdrom

---

です。この属性は、kvmコマンドの-driver引数のmediaオプションに対応します。

- DRIVER: ディスクイメージのフォーマットを指定します。可能な値は raw、qcow2 などです。この属性は、kvm コマンドの -driver 引数の format オプションに対応します。
- CACHE: オプションのキャッシュ機構を指定します。可能な値は、default、none、writethrough、および writeback です。
- IO: IOポリシーの設定が可能です。設定可能な値は「threads」と「native」です。
- DISCARD: TRIMコマンドの処理方法を制御します。オプションは無視またはアンマップです。virtio-scsiでのみ使用可能です。
- IO スロットリングのサポート：TOTAL/READ/WRITE スループットまたは IOPS を制限できます。また、この IO 操作のバースト制御をディスクごとに設定できます。属性名と目的についてはリファレンスガイドをご参照ください。

## NIC

- TARGET: VM用に作成されるtunデバイスの名称です。これはkvmコマンドの"-net"引数におけるifnameオプションに対応します。
- SCRIPT: VM 用の tun デバイス作成後に実行するシェルスクリプトの名前です。これは kvmコマンドの「-net」引数におけるスクリプトオプション。
- ネットワークトラフィックを制御するためのQoSです。ネットワークトラフィックに対して、以下のような異なる種類の制御を定義できます：
  - INBOUND\_AVG\_BW
  - INBOUND\_PEAK\_BW
  - INBOUND\_PEAK\_KW
  - OUTBOUND\_AVG\_BW
  - アウトバウンドピーク帯域幅
  - アウトバウンドピークキロワット
- MODEL: エミュレートするイーサネットハードウェア。利用可能なモデルのリストは、以下のコマンドで取得できます:

```
$ kvm -net nic,model=? -nographic /dev/null
```

- FILTER は、インターフェースに対するネットワークフィルタリングルールを定義します。Libvirt には、clean-traffic など、使用可能な事前定義ルールが含まれています。詳細については [Libvirt のドキュメントをご確認ください](#)。また、システム内のルールを一覧表示するには、以下のコマンドを実行できます:

```
$ virsh -c qemu:///system nwfilter-list
```

- VIRTIO\_QUEUES は、CPU とネットワークドライバ間の通信に使用するキューの数を定義します。この属性は MODEL = ‘virtio’ の場合のみ利用可能です。

## グラフィックス

適切に設定されていれば、libvirt および KVM は SPICE と連携可能です（詳細は [こちらをご確認ください](#)）。選択するには、GRAPHICS 属性に以下を追加してください：

- TYPE = SPICE

スパイスを有効にすると、ドライバーはこれらのマシン向けに特定の設定を注入します。設定はドライバー設定ファイルの変数

---

SPICE\_OPTIONS で変更可能です。

## Virtio

VirtioはKVMにおけるI/O仮想化のフレームワークです。ゲスト用にvirtioドライバを搭載したLinuxカーネルが必要となります。詳細はKVMのドキュメントをご確認ください。

virtio ドライバを使用する場合は、デバイスに以下の属性を追加してください：

- DISK には、属性 `DEV_PREFIX="vd"` を追加してください。
- NIC (ネットワークインターフェースカード) には、`MODEL="virtio"` 属性を追加してください。

ディスクについては、SCSI バス (`sd`) を使用することも可能です。この場合、virtio-scsi コントローラが使用されます。このコントローラは、実際のハードウェアをエミュレートしないため高速であるだけでなく、ディスクに `DISCARD="unmap"` 属性が設定されている場合にディスク領域を解放する trim コマンドのサポートも提供します。必要に応じて、vCPU キューの数を以下の方法で変更できます：

```
FEATURES = [
    VIRTIO_SCSI_QUEUES = 4
]
```

## 追加属性

`raw` 属性は、OpenNebulaが認識しない属性をKVMに渡すことをエンドユーザーに可能にします。基本的に、ここに記述された内容はすべて、KVMデプロイメントファイルに文字通り書き込まれます（libvirt XML形式とセマンティクスを使用）。

```
RAW = [ type = "kvm",
        data = "<devices><serial type=\"pty\"><source path=\"/dev/pts/5\"/><target
        <port=\"0\"/></serial><console type=\"pty\" tty=\"/dev/pts/5\"><source path=\"/dev/
        pts/5\"/><target port=\"0\"/></console></devices>" ]
```

## Libvirt メタデータ

以下の OpenNebula 情報は、Libvirt ドメインのメタデータセクションに追加されます。具体的な属性は下記の通りです：

- `system_datastore`
- `name`
- `uname`
- `uid`
- `gname`
- グループID
- オープンネブラバージョン
- `stime`
- デプロイメント時間

これらは、VM の XML 表現における OpenNebula 相当の値に対応しています。`opennebula_version` および `deployment_time` は、それぞれ、デプロイ時に使用された OpenNebula のバージョンと、エポック形式でのデプロイ時間です。

また、VM名はLibvirt XMLのtitleフィールドに含まれます。そのため、`--title`オプションを使用してLibvirtドメインを一覧表示する場合、ドメイン名とともにVM名も表示されます。

## ディスク/NICのホットプラグ

KVMは、virtioバスおよびSCSIバスへのホットプラグをサポートしております。ディスクに関しては、ディスクが接続されるバスは、ディスクテンプレートの`DEV_PREFIX`属性から推測されます。

- `vd: virtio` (推奨)。
- `sd: SCSI` (デフォルト)。
- `hd: IDE`

**注記:** CDROM およびフロッピーディスクについては、ホットプラグはサポートされていません。

`DEV_PREFIX` の代わりに `TARGET` が渡された場合も、同様のルールが適用されます（内部的には、`TARGET` が指定されていない場合、Open-Nebula は `DEV_PREFIX` に基づいて `TARGET` を生成します）。

新規に接続されたディスクおよび NIC のデフォルト設定は、`/var/lib/one/remotes/etc/vmm/kvm/kvmrc` にあります。関連するパラメータには `DEFAULT_ATTACH_` が接頭辞として付きます。詳細は以下の「[ファイルとパラメータ](#)」で説明されています。

ディスクおよび NIC に関して、ゲスト OS が Linux 系の場合、ゲストに PCI バスを再スキャンするよう明示的に指示する必要があります。これは root 権限で以下のコマンドを実行することで実現できます：

```
# echo 1 > /sys/bus/pci/rescan
```

## QEMUゲストエージェントの有効化

QEMUゲストエージェントは、ゲストOSとの間で特定の操作を通信可能にします。このエージェントはvirtioシリアル接続を使用してコマンドを送受信します。特に有用な機能の一つとして、スナップショット作成前にファイルシステムを凍結することが挙げられます。これにより、スナップショットに書きかけのデータが含まれることを防げます。ファイルシステムの凍結機能は、CEPH およびqcow2ストレージドライバでのみご利用いただけます。

ゲストOSに必要なエージェントパッケージは、ほとんどのディストリビューションで利用可能です。多くは`qemu-guest-agent`と呼ばれています。詳細情報が必要な場合は、以下のリンクをご参照ください：

- [https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7/html/Virtualization\\_Deployment\\_and\\_Administration\\_Guide/chap-QEMU\\_Guest\\_Agent.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Virtualization_Deployment_and_Administration_Guide/chap-QEMU_Guest_Agent.html)
- [http://wiki.libvirt.org/page/Qemu\\_guest\\_agent](http://wiki.libvirt.org/page/Qemu_guest_agent)
- <http://wiki.qemu.org/Features/QAPI/GuestAgent>

ゲストエージェントとの通信チャネルは、VMテンプレートで`GUEST_AGENT`機能を選択した場合、ドメインXML内で有効化されます。

## 仮想マシンのインポート

OpenNebulaを介さずに起動されたKVMハイパーバイザー上で動作する仮想マシン（VM）は、OpenNebulaにインポートすることができます。ただし、ホストガイドに記載されている制限事項に加え、KVM環境においてインポートされたVMに対しては「電源オフ」操作が利用できない点に留意が必要です。

### 7.2.5 チューニングと拡張

## ホストごとの複数アクション

**警告：**この機能は実験段階です。推奨される設定とするには、コードへの一部修正が必要です。

デフォルトでは、ドライバはlibvirtデーモンとの通信にUNIXソケットを使用します。この方法は、同時に1つのプロセスでのみ安全に使用できます。これを確実にするため、ドライバはホストごとに一度に1つのアクションのみを送信するよう設定されています。例えば、特定の時点において、ホストごとにデプロイメントは1回のみ実行されます。

この制限は、libvirtがTCP接続を受け入れるように設定し、OpenNebulaがこの通信方法を使用するように設定することで解決できます。

### Libvirtの設定

ここでは、CentOS 7 マシンにおいて libvirtd が暗号化および認証なしの TCP 接続を受け入れるように設定する方法を説明します。その他の環境については、ご利用のディストリビューションおよび libvirt のドキュメントをご確認ください。

各ハイパーバイザーの /etc/libvirt/libvirtd.conf ファイルを変更し、以下のパラメータが設定され、以下の値になっていくことをご確認ください：

```
listen_tls = 0
listen_tcp = 1
tcp_port = "16509"
auth_tcp =
= "none"
```

また、/etc/sysconfig/libvirtd を修正し、以下の行のコメントを外す必要があります：

```
LIBVIRTD_ARGS="--listen"
```

これらのファイルを変更した後、libvirtデーモンを再起動する必要があります：

```
$ sudo systemctl restart libvirtd
```

### OpenNebulaの設定

VMMドライバは、1つのホストにつき複数のアクションを実行できるように設定する必要があります。これは、ドライバ実行ファイルに-pパラメータを追加することで実現できます。設定は/etc/one/oned.confのVM\_MAD設定セクションで行います：

```
VM_MAD = [
    name      = "kvm",
    executable = "one_vmm_exec", arguments = "-t 15
    -r 0 kvm -p",
    default   = "vmm_exec/vmm_exec_kvm.conf", タイプ =
    "kvm" ]
```

ファイル /var/lib/one/remotes/etc/vmm/kvm/kvmrc を変更し、libvirt 通信用の TCP エンドポイントを設定してください：

```
export LIBVIRT_URI=qemu+tcp://localhost/system
```

また、スケジューラの設定を変更し、ホストあたり複数の仮想マシンを展開できるようにする必要があります。該当ファイルは次の場所にあります：  
`/etc/one/sched.conf` ファイル内の変更対象値は `MAX_HOST` です。例えば、スケジューラがホストごとに 10 台の仮想マシンを提出できるようにするには、以下の行をご使用ください：

```
MAX_HOST = 10
```

この更新後、ノード内のリモートファイルを更新し、OpenNebulaを再起動してください：

```
$ onehost sync --force
$ sudo systemctl restart opennebula
```

## ファイルとパラメータ

本ドライバは以下のファイルで構成されております：

- `/usr/lib/one/mads/one_vmm_exec` : 汎用 VMM ドライバー。
- `/var/lib/one/remotes/vmm/kvm` : アクションを実行するために実行されるコマンドです。

また、以下のドライバ設定ファイルがあります：

- `/etc/one/vmm_exec/vmm_exec_kvm.conf` : このファイルには、ドメイン定義（つまり OpenNebula テンプレート）のデフォルト値が格納されています。

一般的に、KVM固有の属性、すなわち他のハイパーバイザでは必須ではないがKVMドライバーでは必須となる属性については、デフォルト値を設定しておくことが推奨されます。KVMでは必須ではないもののKVM固有の属性についても、デフォルト値を設定しておくことが望ましいです。このファイルを変更した場合は、**OpenNebulaの再起動が必要となります。**

- `/var/lib/one/remotes/etc/vmm/kvmrc` : このファイルには、特定のタスクを実行するため、または環境変数をドライバに渡すために、実際のドライバロード前に実行される指示が記述されています。前者の場合に使用される構文は、ドライバ実行前に評価されるプレーンなシェルスクリプトです。後者の場合、構文はおなじみの以下の形式となります：

```
ENVIRONMENT_VARIABLE=VALUE
```

ここで変更可能なパラメータは以下の通りです：

パラメータ	説明
<code>LIBVIRT_URI</code>	libvirtd への接続文字列
<code>QEMU_PROTOCOL</code>	ライブマイグレーションに使用されるプロトコル
<code>シャットダウンタイムアウト</code>	シャットダウン後のタイムアウトまでの待機時間（秒）
<code>VIRSH_RETRIES</code>	必要な場合の「virsh」コマンドの再試行回数。現在、
<code>SYNC_TIME</code>	再開時および移行後にRTCからVMの時刻同期をトリガーします
<code>強制削除</code>	シャットダウンタイムアウト後に仮想マシンの強制終了を実行します
<code>CANCEL_NO_ACPI</code>	ACPIが有効でないVMは、シャットダウン時に強制的に破棄されます
<code>MIGRATE_OPTIONS</code>	virsh migrate コマンドのオプションを設定します
<code>DEFAULT_ATTACH_CACHE</code>	このパラメータは、新しく接続されるディスクのデフォルトのキャッシングタイプを設定します。
<code>DEFAULT_ATTACH_DISCARD</code>	新規に接続されたディスクに対するデフォルトの破棄オプション（属性が欠落している場合）
<code>DEFAULT_ATTACH_IO</code>	新規に接続されたディスクに対するデフォルトのI/Oポリシー（属性が指定されていない場合）
<code>DEFAULT_ATTACH_TOTAL_BYTES_SEC</code>	新規に接続されたディスクに対するデフォルトの総バイト/秒 I/O スロットリング（属性が存在しない場合）
<code>DEFAULT_ATTACH_TOTAL_BYTES_SEC_MAX</code>	新規に接続されたディスクに対するデフォルトの最大総バイト/秒 I/O スロットリング

### OpenNebula 5.10 導入ガイド、リリース 5.10.5

DEFAULT_ATTACH_TOTAL_BYTES_SEC_MAX_LENGTH	新規に接続されたディスクに対するデフォルトの最大総バイト/秒 I/O スロットリングの長さ
DEFAULT_ATTACH_READ_BYTES_SEC	新規に接続されたディスクに対するデフォルトの読み取りバイト/秒のI/Oスロットリング (attrが設定されている場合)
DEFAULT_ATTACH_READ_BYTES_SEC_MAX	新規に接続されたディスクに対するデフォルトの最大読み取りバイト/秒 I/O スロットリング
DEFAULT_ATTACH_READ_BYTES_SEC_MAX_LENGTH	新規接続ディスクに対するデフォルトの最大読み取りバイト/秒 I/O スロットリング

表 7.1 – 続き

パラメータ	説明
DEFAULT_ATTACH_WRITE_BYTES_SEC	新規接続ディスクに対する書き込みバイト/秒のデフォルトI/Oスロットリング（接続時）
DEFAULT_ATTACH_WRITE_BYTES_SEC_MAX	新規に接続されたディスクに対する書き込みバイト/秒のデフォルト最大 I/O スロットリング
DEFAULT_ATTACH_WRITE_BYTES_SEC_MAX_LENGTH	新規接続ディスクに対する書き込みバイト/秒のデフォルト最大長スロットリング
DEFAULT_ATTACH_TOTAL_IOPS_SEC	新規に接続されたディスクに対するデフォルトの総 IOPS スロットリング（属性が設定されている場合）
DEFAULT_ATTACH_TOTAL_IOPS_SEC_MAX	新規に接続されたディスクに対するデフォルトの最大総IOPSスロットリングは、
DEFAULT_ATTACH_TOTAL_IOPS_SEC_MAX_LENGTH	新規に接続されたディスクに対するデフォルトの最大総IOPSスロットリング期間
DEFAULT_ATTACH_READ_IOPS_SEC	新規に接続されたディスクに対するデフォルトの読み取り IOPS スロットリング（属性が設定されている場合）
DEFAULT_ATTACH_READ_IOPS_SEC_MAX	新規に接続されたディスクに対するデフォルトの最大読み取り IOPS スロットリング。
DEFAULT_ATTACH_READ_IOPS_SEC_MAX_LENGTH	新規に接続されたディスクに対するデフォルトの最大長読み取り IOPS スロットリング
新規接続ディスクに対する書き込み IOPS スロットリングのデフォルト値	新規に接続されたディスクに対する書き込み IOPS スロットリングのデフォルト値（属性が設定されている場合）
DEFAULT_ATTACH_WRITE_IOPS_SEC_MAX	新規に接続されたディスクに対するデフォルトの最大書き込みIOPSスロットリングは、
DEFAULT_ATTACH_WRITE_IOPS_SEC_MAX_LENGTH	新規に接続されたディスクに対する書き込み IOPS スロットリングのデフォルト最大長
DEFAULT_ATTACH_NIC_MODEL	新規に接続された NIC のデフォルト NIC モデル（属性が欠落している場合）
DEFAULT_ATTACH_NIC_FILTER	新規に接続された NIC に対するデフォルトの libvirt NIC フィルター（属性が存在しない場合）

詳細については、仮想マシンドライバのリファレンスをご参照ください。

## 7.2.6 トラブルシューティング

### イメージマジックが不正です

サスペンド状態から仮想マシンを復元しようとすると、次のエラーが返されます:

libvirtd1021: 操作に失敗しました: イメージマジックが不正です。以下の設定を適用する

ことで修正できます:

```
options kvm_intel nested=0
オプション kvm_intel emulate_invalid_guest_state=0 オプション
kvmignore_msrs=1
```

## 7.3 LXD ドライバー

LXD は、LXC コンテナを駆動するための REST API を提供するデーモンです。コンテナは軽量な OS レベルの仮想化インスタンスであり、仮想マシンのように動作しますが、ホストとカーネルを共有することでハードウェアエミュレーションによる処理の遅延の影響を受けません。

### 7.3.1 要件

LXD ドライバーは、snap パッケージを介した LXD の使用をサポートしております。snap がインストールされている場合、それを検出し、そのインストールパスを使用します。

ホストシステムは Ubuntu 16.04 以降、または Debian 10 以降である必要があります。

### 7.3.2 考慮事項と制限事項

ゲストOSは仮想化ノードとLinuxカーネルを共有するため、Linux以外のOSを起動することはできません。

まだ実装されていない標準的な機能がいくつかあります：

- スナップショット
- ライブマイグレーション
- 保存/復元
- ライブディスクリサイズ
- LVMデータストア
- PCI/バススルー
- 振発性ファイルシステム
- オフラインディスクのサイズ変更：
  - 複数のパーティションイメージではサポートされていません
  - XFS および ext4 ファイルシステムのみ対応
- データブロック：OpenNebula上で作成されたデータブロックは、コンテナにアタッチする前にフォーマットが必要となります
- 複数のパーティションイメージ：パーティションをマウントするには、いずれかのパーティションに有効な /etc/fstab が必要です
- lxc exec \$container -- CentOS コンテナへのログインではログインシェルが出力されません

### 7.3.3 設定

LXD と OpenNebula 間の相互作用に関する設定は、/var/lib/one/remotes/etc/vmm/lxd/lxdrc に記述されています

```
#####
# VNC オプション #####
# コンテナへのVNCアクセスをカスタマイズするためのオプション：
#   - VNC 端末で実行されるコマンド。 #           - :width: ターミナルの幅
#   - :height: ターミナルの高さ
#   - :timeout: 入力がない場合に端末を閉じるまでの秒数
:vnc:
:command: /bin/login
:width: 800
:height: 600
:timeout: 300

#####
# OpenNebula 設定オプション
#####
# データストアのデフォルトパスです。oned.conf 内の対応する値が変更された場合のみ変更が必要です。
:datastore_location: /var/lib/one/datastores
```

より複雑なcgroup構成の場合、コンテナのcgroupはデフォルトのルートcgroupではなく、別のスライスに配置することができます。  
環境変数で設定できます。

```
export LXC_CGROUP_PREFIX=system.slice
```

## LXD デーモン

既存のすべてのコンテナには、以下の制限値が定義されている必要があります：`limits.cpu.allowance`、`limits.cpu`、および`limits.memory`。opennebula-node-lxdパッケージでは、これらの制限値のデフォルトプロファイルを100%、1、および512MBに設定しております。

## ドライバー

OpenNebulaでは、LXD ドライバーはデフォルトで有効化されています：

```
#-----
# LXD仮想化ドライバ管理者の設定 #      -r ホスト監視時の再試行回数
#      -t スレッド数、すなわち同時に監視するホストの数 #      -l <アクション[=コマンド名]> ローカルで実行されるア
クション。コマンドは各アクションごとに上書き可能です。          各アクションごとに上書き可能です。
#      有効なアクション: deploy, shutdown, cancel, save, restore, migrate, poll #      例: "-l
migrate=migrate_local,save"
#      -p ホストごとに複数のアクションを並列実行する場合、ハイパーバイザのサポートが必要です #      -s <シェル> リモー
トコマンドを実行します。デフォルトはbashです
#      -w 外部コマンドの実行タイムアウト(秒単位) (デフォルトは無制限) #
#-----
```

```
VM_MAD = [
    NAME           =
    "lxd", SUNSTONE_NAME = "LXD",
    実行可能ファイル =
        = "one_vmm_exec", 引数
        = "-t 15 -r 0 lxd",
    # デフォルト
        = "vmm_exec/vmm_exec_lxd.conf",
    タイプ           =
        = "xml", ス
    ナップショット保持 = "いいえ",
    IMPORTED_VMS_ACTIONS = "terminate, terminate-hard, reboot, reboot-hard, poweroff,
    ↳強制電源オフ、サスPEND、レジューム、停止、削除、NIC接続、NIC 切断"
]
```

設定パラメータ:`-r`、`-t`、`-l`、`-p` および `-s` は、すでに適切なデフォルト値で事前設定されています。これらを変更される場合は、OpenNebula を再起動する必要があります。

これらのパラメータの詳細、およびドライバのカスタマイズや拡張方法については、仮想マシンドライバリファレンスをご参照ください。

### 7.3.4 使用方法

#### LXD 固有の属性

以下は LXD に固有のテンプレート属性です。VM を定義するためにサポートされている属性の完全なリストについては、テンプレートリファレンスドキュメントをご参考ください。

#### VNC

Sunstone上で表示されるVNC接続は、`lxc exec`経由で実行されるコマンドの結果です。デフォルトではこのコマンドは`login`であり、`lxdrc`ファイルによってノードごとに設定されています。変更するには、GRAPHICSセクション内のCOMMANDキーで設定して

---

ください。

```

CONTEXT = [
    NETWORK = "YES",
    SET_HOSTNAME = "$NAME",
    SSH_PUBLIC_KEY = "$USER[SSH_PUBLIC_KEY]" ]
CPU = "1"
DISK = [
    IMAGE_ID = "14" ]
GRAPHICS = [
    COMMAND = "bash",
    LISTEN = "0.0.0.0",
    TYPE = "vnc" ]
LXD_SECURITY_PRIVILEGED = "true"
MEMORY = "768"

```

## セキュリティ

コンテナは特権モードまたは非特権モードで動作可能であり、ネストされたコンテナも許可されます。OpenNebulaテンプレートでのこの設定を定義するには、以下を追加してください：

```
LXD_SECURITY_PRIVILEGED=true
LXD_SECURITY_NESTING=true
```

デフォルトでは、OpenNebula は非特権イメージを作成します。

## プロファイル

LXDデーモンは複数の定義済みプロファイルを保持できます。各コンテナはデフォルトでデフォルトプロファイルからプロパティを継承します。ただし、VMテンプレート内で継承するカスタムプロファイルを設定することも可能です。

```
LXD_PROFILE=<プロファイル名>
```

テンプレートは、プロファイルと一致するキーをすべて上書きすることをご留意ください。ノード上でプロファイルが見つからない場合、デフォルトのプロファイルが適用され、VMログにエラーが表示されます。

各ノードではプローブが実行され、フロントエンドに対してノード上に存在するプロファイルを報告します。これにより、手動で検索することなく、プロファイルを容易に適用することが可能となります。

## ディスク

接続されたディスクはタイプ別に扱われます：コンテナ内のディスクデバイスです。これはKVMとは動作が異なり、ディスクはLXDホストにマウントされ、そのマウントポイントがユーザー定義のマウントポイントとしてコンテナに渡されます。

### ディスク接続プロセスについて、概要を説明いたします：

- コンテナディレクトリ内でその内容が参照可能となるべきイメージファイルが存在します
- LXDにディスクを処理させるためには、このファイルをホストディレクトリにマウントする必要があります。  
\$DATASTORE\_LOCATION/\$system\_datastore\_id/\$vm\_id/mapper/disk.\$disk\_id
- ディスクには様々なタイプがありますが、現在サポートされているのはrawおよびqcow2イメージファイル、ならびにcephです  
**rbd**です。
- マウントするためには、まず各イメージをホストデバイスにマッピングする必要があります。
- イメージの種類に応じて、異なるユーティリティが使用されます。生イメージには`losetup`、  
qcow2イメージには`qemu-nbd`、ceph rbdには`rbd-nbd`が使用されます。
- イメージに複数のパーティションがある場合、/etc/fstab ファイルが見つかるまで各パーティションがマウントされ、有効なファイルシステムを持つ各パーティションが適切にマウントされます。

## 追加属性

raw 属性により、エンドユーザーは OpenNebula が認識しない属性を LXD に渡すことが可能となります。基本的に、ここに記述された内容はすべて LXD 展開ファイルにそのまま書き込まれます。

```
RAW = [ type = "lxd",
        "boot.autostart": "true", "limits.processes": "10000"]
```

## 仮想マシンのインポート

LXDはリソース制限なくコンテナを展開できますが、OpenNebulaは明示的な容量指定なしに仮想マシンを作成できません。そのため、ワイルドコンテナにはこれらのキーを定義する必要があります。インポート後、コンテナは以下の利点を得られます：

- start
- 停止（強制停止）
- restart hard also
- ネットワークインターフェースの接続/切断
- vnc connection

コンテナはデータストアに有効なイメージが存在しないため、ストレージ関連の操作による恩恵は受けられません。インポートされたコンテナを削除すると、再び未管理状態となります。

### 7.3.5 チューニングと拡張

#### マルチハイパーバイザー

LXD は仮想化拡張機能を必要としないため、同じ仮想化ノード内で KVM やその他の HVM ハイパーバイザーと問題なく共存できます。

## イメージ

LXD ドライバーは、KVM と同じ形式のイメージ（例：qcow2 イメージ）からコンテナを作成できます。

### 独自のイメージを作成する

基本的には、ファイルを作成し、それをブロックデバイスにマッピングし、デバイスをフォーマットしてパーティションを作成し、そこにデータをダンプすれば、イメージが完成します。

ここでは、LXD CLI を使用してコンテナを作成し、後ほど OpenNebula データストアで使用するためにブロックデバイスにダンプします。このタイミングでコンテナをコンテキスト化 (<kvm\_contextualization>) すると良いでしょう。手順は KVM と同様です。

```
# truncate -s 2G container.img
# block=$(losetup --find --show container.img) # mkfs.ext4
$block
# mount $block /mnt
# lxc init my-container ubuntu:18.04
# cp -rpa /var/lib/lxd/containers/my-container/rootfs/* /mnt# umount $block
# losetup -d $block
```

これでイメージが使用可能な状態になりました。qemu-img を使用してイメージ形式を変換することも可能です。コンテナのベースイメージとして、Linux 標準のファイルシステムやパーティションレイアウトを自由に使用できる点にご留意ください。これにより、raw lxc からのイメージ、KVM イメージのルートパーティション、proxmox テンプレートからのイメージを簡単にインポートすることが可能となります。

### linuxcontainers.org マーケットプレイスのご利用

通常のLXD環境では、デフォルトでパブリックイメージサーバーへの読み取りアクセス権が付与されており、コンテナイメージを取得することができます。

# lxc リモート一覧		URL	プロトコル	認証
→	+-----+-----+			
	NAME			
←	タイプ   公開   静的			
→	+-----+-----+-----+			
	images	https://images.linuxcontainers.org	simplestreams	
→	はい   い			
←	+-----+-----+-----+			
	local (デフォルト)	unix://	LXD	TLS
→	いいえ   はい			
←	+-----+-----+-----+			
	ubuntu	https://cloud-images.ubuntu.com/releases   simplestreams		
→	はい   はい			
←	+-----+-----+-----+			
	ubuntu-daily	https://cloud-images.ubuntu.com/daily   simplestreams		
→	はい   はい			
←	+-----+-----+-----+			

---

OpenNebulaは、既存のイメージサーバーをマーケットプレイスのバックエンドとして活用することができます。

## KVMディスクイメージを使用します

LXDドライバーは、パーティションテーブルを持つイメージからコンテナを作成できます。ただし、そのイメージに有効なfstabファイルが存在する場合に限ります。LXDコンテナのセキュリティは、このUUIDマッピングに基づいています。コンテナを起動すると、そのUUIDはLXD設定に従ってマッピングされます。ただし、コンテナのルートファイルシステムがマッピングできない場合があります。この問題はマーケットプレイスのイメージで発生することがあり、これを使用するには、コンテナVMテンプレートでLXD\_SECURITY\_PRIVILEGEDをtrueに設定する必要があります。

この種の画像は、OpenNebula Marketplaceから直接入手できます。

## カスタムストレージバックエンド

サポートされているイメージ（例：vmdkファイル）をカスタマイズしたい場合、LXDドライバーにはマッパーと呼ばれるモジュールが用意されており、これによりドライバーはraw、qcow2、rbdデバイスなどの複数のイメージ形式とやり取りすることが可能となります。

マッパーは基本的に、2つのメソッドが定義されたRubyクラスです。`do_map` メソッドはディスクファイルをシステムロックデバイスにロードし、`do_unmap` メソッドはこれを元に戻します（例：

```
disk.qcow2      -> マップ -> /dev/nbd0
disk.raw        -> マップ -> /dev/loop0
one/one-7-54-0 -> マップ -> /dev/nbd0
```

ただし、パーティションテーブル付きのイメージを扱う場合、状況が複雑になることがあります。マッパーデバイスのコードはこちらでご確認いただけます。

### 7.3.6 トラブルシューティング

- oneadminユーザーの\$HOMEディレクトリが/home/\$USER以外の場所に存在します。これにより、snapの制限によりoneadminアカウントがLXD CLIを正常に使用できません。sudoを使用して他のアカウントでlxdコマンドを実行することができます。
- VNC設定のコマンドパラメータは、コンテナ進入時にnoVNCに表示されるコマンドを指定します。/bin/bashを設定すると、ユーザーログインをスキップし、コンテナ上でrootアクセス権を取得できます。
- 再起動の問題**が発生した場合、ネットワークフックパッチを適用することができます。具体的には、ファイル「/usr/share/one/examples/network\_hooks/99-lxd\_clean.rb」を「/var/lib/one/remotes/vnm/clean.d」にコピーし、「onehost sync --force」を実行してください。この操作は、クラウドで使用されているすべてのネットワークドライバに対して実施する必要があります。  
<ネットワークドライバ>/clean.dにコピーし、onehost sync --forceを実行することでネットワークフックパッチを適用できます。この操作は、クラウドで使用されているすべてのネットワークドライバに対して実施する必要があります。

## 7.4 監視

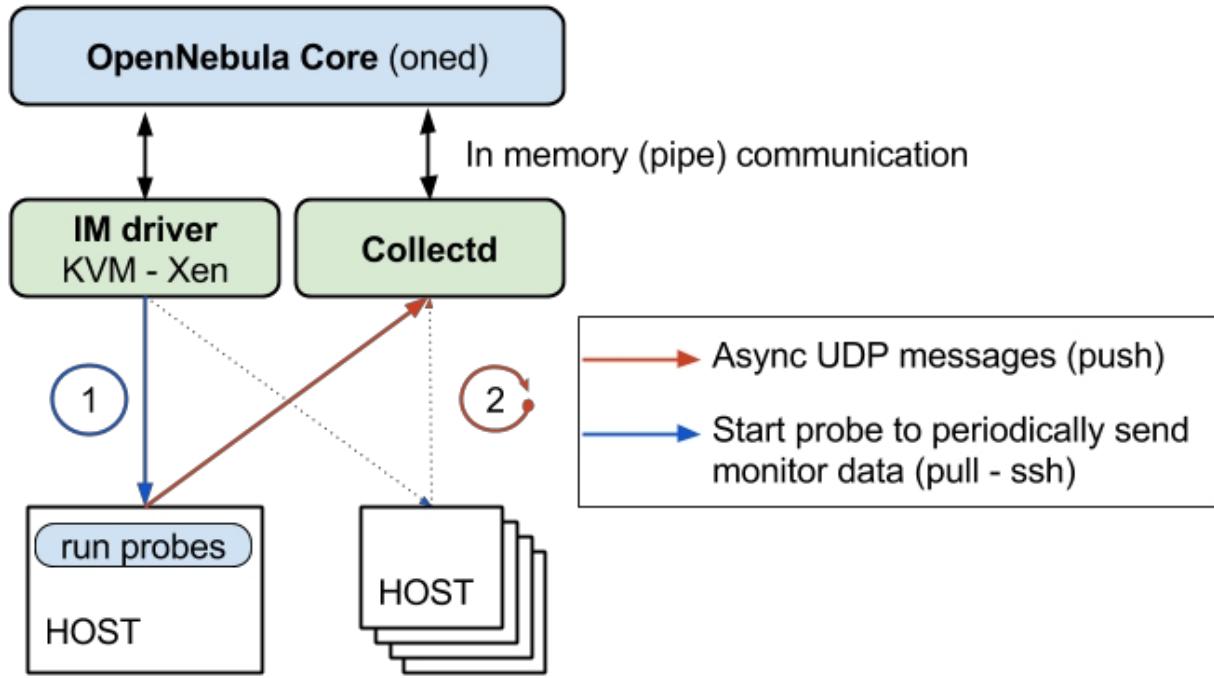
本セクションでは、OpenNebulaの監視サブシステムについて概要を説明します。監視サブシステムは、ホストの状態や基本的なパフォーマンス指標、仮想マシンの状態や容量使用率など、ホストおよび仮想マシンに関する情報を収集します。この情報は、

OpenNebulaが提供する一連の静的プローブを実行することで収集されます。これらのプローブの結果は、プッシュ方式を用いてOpenNebulaに送信されます。

#### **7.4.1 概要**

各ホストは定期的に監視データをUDP経由でフロントエンドに送信し、フロントエンドは専用のモジュールで収集・処理を行います。この分散型監視システムは、軽量な通信プロトコルとプッシュモデルを採用した専用監視システムのアーキテクチャに類似しています。

OpenNebulaは、フロントエンド上でcollectdデーモンを起動し、ポート4124でUDP接続を待機させます。最初の監視サイクルにおいて、OpenNebulaはSSHを使用してホストに接続し、プローブスクリプトを実行するデーモンを起動します。このデーモンは、収集したデータをフロントエンドのcollectdデーモンへ、特定の秒数間隔（`collectd IM_MAD`の`-i`オプションで設定可能）で送信します。これにより、監視サブシステムはデータ受信のために新たなSSH接続を確立する必要がありません。



特定のホストでエージェントが停止した場合、OpenNebulaはそのホストから監視データが受信されていないことを検知し、SSHによるプローブを再起動します。

定期的な監視とは別に、一部の特定ドライバー（SSH および LVM 転送マネージャー）は、アクションが完了した時点で即座にデータ送信をトリガーします。これは、ホスト上の collectd クライアントに HUP シグナルを送信することで実現されます。``onehost forceupdate`` コマンドを使用して、手動で情報更新をトリガーすることも可能です。

## 7.4.2 要件

- フロントエンドのファイアウォール（有効化されている場合）は、ホストからのポート4124へのUDPパケットの受信を許可する必要があります。

## 7.4.3 OpenNebulaの設定

### ドライバの有効化

この監視システムを有効にするには、`/etc/one/oned.conf` を以下のスニペットで設定する必要があります：

`collectd` は KVM に対して有効にする必要があります:

**注記：**すべての設定はLXDにも適用可能です

```

# このドライバはホストに割り当てることができません。KVM
#
# -h このヘルプを表示します。
# -a collectd ソケットをバインドするアドレス（デフォルトは 0.0.0.0）モニター情報をリッスン
# -f ンする UDP ポート（デフォルトは 4124）
# -t 収集した情報をフラッシュする間隔（秒単位、デフォルト値 5）サーバーのスレッド数（デフォルト値 50）
# -i 監視パッシュサイクルの時間間隔（秒単位）。このパラメータは
    MONITORING_INTERVAL_HOST よりも小さく設定してください。そうでない場合、パッシュモニタリングは効果を発揮しません
#
#-----
IM_MAD = [
    NAME      = "collectd",
    EXECUTABLE = "collectd",
    引数      = "-p 4124 -f 5 -t 50 -i 20" ]
#-----
```

このドライバで有効な引数は以下の通りです：

- **-a:** collectd ソケットをバインドするアドレス（デフォルトは 0.0.0.0）
- **-p:** ポート番号
- **-f:** 収集した情報を OpenNebula にフラッシュする間隔（秒単位、デフォルトは 5）
- **-t:** collectd サーバーのスレッド数（デフォルト 50）
- **-i:** 監視パッシュサイクルの時間間隔（秒単位）。このパラメータは MONITORING\_INTERVAL\_HOST（下記参照）よりも小さく設定する必要があります。そうでない場合、監視パッシュは効果を発揮しません。

#### KVM:

```

#-----
# KVM UDPパッシュ情報ドライバーマネージャー設定 #           -r ホスト監視時の再試行
回数
#       -t スレッド数、すなわち同時に監視されるホストの数
#-----
IM_MAD = [
    NAME      =
    "kvm", SUNSTONE_NAME = "KVM",
    実行可能ファイル = "one_im_ssh", 引数
    = "-r 3 -t 15 kvm" ]
#-----
```

このドライバに渡される引数は以下の通りです：

- **-r:** ホスト監視時の再試行回数
- **-t:** スレッド数、すなわち同時に監視するホストの数

#### 監視設定パラメータ

OpenNebulaでは、監視サブシステム全体の動作をカスタマイズすることができます：

パラメータ	説明
MONITOR- ING_INTERVAL_HOST	ホスト監視間の間隔（秒単位）。この値は
MONITOR- 監視間隔DB更新	VM監視情報のDB書き込み間隔（秒単位）。-Iで EDBの更新を無効化します。0を指定すると、更新のたびに書き込みを行います
MONITOR- ING_INTERVAL_VM	VM モニタリングの間隔（秒単位）。この値は MONITOR-
MONITOR- ING_INTERVAL_DATASTOR	データストアの監視間隔（秒単位）。この値は マネージャータイマーよりも大きい値である必要があります
MONITOR- ING_INTERVAL_MARKET	マーケットプレイスの監視間隔（秒単位）。この値は、マネージャータイマーよりも大きい 値である必要があります。
HOST_PER_INTERVAL	マネージャータイマーよりも大きい値である必要があります
HOST_PER_INTERVAL	各間隔で監視対象となるホストの数。

#### 7.4.4 トラブルシューティング

##### 健全な監視システム

OpenNebulaは、約monitoring\_push\_cycle秒ごとに、各仮想マシンおよびホストの監視データを以下のように受信します：

```
2016年5月24日(火) 16:21:47 [Z0][InM][D]: ホスト thost087 (0) の監視に成功しました。
2016年5月24日(火) 16:21:47 [Z0][VMM][D]: VM 0 の監視に成功しました: STATE=a CPU=0.0
→メモリ使用量=113404 受信パケット数=648 送信パケット数=398
2016年5月24日(火) 16:22:07 [Z0][InM][D]: ホスト thost087 (0) の監視に成功しました。
2016年5月24日(火) 16:22:07 [Z0][VMM][D]: VM 0 の監視に成功しました: STATE=a CPU=0.0
→メモリ使用量=113516 受信ネットワーク転送量=648 送信ネットワーク転送量=468
2016年5月24日(火) 16:22:11 [Z0][VMM][D]: VM 0 の監視に成功しました: DISK_SIZE=[ID=0,
→SIZE=27] DISK_SIZE=[ID=1,SIZE=1]
2016年5月24日(火) 16:22:27 [Z0][InM][D]: ホスト thost087 (0) の監視に成功しました。
2016年5月24日(火) 16:22:27 [Z0][VMM][D]: VM 0 の監視に成功しました: STATE=a CPU=0.0
→メモリ使用量=113544 受信ネットワークパケット=648 送信ネットワークパケット=468
```

しかしながら、oned.logにおいてホストが定期的に（MONITORING\_INTERVAL\_HOST 秒ごと）  
秒ごと）監視されている場合、監視は正しく機能していません：

```
2016年5月24日(火) 16:24:23 [Z0][InM][D]: ホスト thost087 (0) の監視を開始しました
2016年5月24日(火) 16:25:23 [Z0][InM][D]: ホスト thost087 の監視 (0)
2016年5月24日(火) 16:26:23 [Z0][InM][D]: ホスト thost087 の監視 (0)
```

この場合、OpenNebulaがプローブを処理できる速度よりも速く受信しているためと考えられます。この問題を解決するには、チューニングのセクションをご参照ください。

##### プローブの監視

監視プローブの実行中に発生したエラーのトラブルシューティングについては、[トラブルシューティング](#)のセクションをご参照ください。

## 7.4.5 チューニングと拡張

## 監視間隔時間の調整

OpenNebula のインストールを適切な監視パラメータ値で調整するには、`collectd_IM_MAD`（監視プッシュサイクル）の `-i` オプションを調整する必要があります。

システムが正常に動作していない場合、その原因はデータベースのスループットにある可能性があります。OpenNebula は、VM 1 台あたり約 4KB の監視情報をデータベースに書き込むためです。仮想マシンの数が多くて、監視プッシュサイクルが低すぎる場合、OpenNebula はその量のデータをデータベースに書き込むことができません。

## ドライバファイル

プローブは、監視メトリクスを取得するための専用プログラムです。プローブは各ハイパーバイザーごとに定義されており、KVM の場合は `/var/lib/one/remotes/im/kvm-probes.d` に配置されています。

独自のプローブを作成したり、既存のプローブを修正したりすることは容易です。詳細は「情報マネージャードライバガイド」をご参照ください。ホスト管理ガイドに記載されている通り、`onehost sync` を使用してホスト間でモニタープローブを同期せることをお忘れなく。

## 7.5 PCIバススルー

KVMハイパーバイザーでは、ホスト内のPCIデバイスを検出し、仮想マシンに割り当てることが可能です。

設定および環境情報は [ここから](#) 取得されます。グラフィックカード以外の PCI デバイス、またはそれらからビデオ信号を出力しない場合は、VGA 関連のセクションはすべて無視して問題ありません。

**警告：**全体の設定状態は、事前設定済みの Fedora 22 マシンから抽出されたものです。お使いのディストリビューションの設定は異なる場合があります。

### 7.5.1 必要条件

- 仮想化に使用されるホストマシンは、I/O MMUをサポートしている必要があります。IntelプロセッサではVT-d、AMDプロセッサではAMD-Viと呼ばれています。以下の手順はIntelブランドのプロセッサ向けに作成されていますが、AMDプロセッサでもほぼ同様の手順で対応可能です。
- カーネルバージョン 3.12 以上

### 7.5.2 マシン構成（ハイパーバイザー）

#### カーネル設定

カーネルは、I/O MMUをサポートするように設定し、仮想マシンで使用するPCIデバイスにアクセスする可能性のあるドライバをすべてブラックリストに登録する必要があります。I/O MMUを有効にするパラメータは次のとおりです：

```
intel_iommu=on
```

#### **OpenNebula 5.10 導入ガイド、リリース 5.10.5**

また、カーネルにvfio-pciドライバをロードするよう指示し、選択したカード用のドライバをブラックリストに登録する必要があります。例えば、NVIDIA GPUの場合は以下のパラメータを使用できます：

```
rd.driver.pre=vfio-pci
rd.driver.blacklist=nouveau
```

### initrd への vfio ドライバのロード

vfio のモジュールは initrd に追加する必要があります。モジュールのリストは以下の通りです：vfio vfio\_iommu\_type1 vfio\_pci vfio\_virqfd。例えば、お使いのシステムが dracut を使用している場合、/etc/dracut.conf.d/local.conf ファイルに以下の行を追加してください：

```
add drivers+="vfio vfio_iommu_type1 vfio_pci vfio_virqfd"
```

その後、initrdを再生成してください:

```
# dracut --force
```

### ドライバのブラックリスト登録

カーネルパラメータで行ったブラックリスト登録は、システム設定でも同様に行う必要があります。NVIDIA GPU用の /etc/modprobe.d/blacklist.conf は以下の通りです：

```
blacklist nouveau
blacklist lbm-nouveau
blacklist nouveau_uoptions
blacklist nouveau_modeset=0
alias nouveau off
alias lbm-nouveau off
```

この設定と併せて、仮想マシンに接続したいPCIカードのIDを指定してvfioドライバをロードする必要があります。例えば、nvidia Grid K2 GPUの場合はID 10de:11bfを指定します。ファイル /etc/modprobe.d/local.conf:

```
options vfio-pci ids=10de:11bf
```

### vfio デバイスバインディング

I/O MMU は、デバイス間および VM 間のメモリ操作を分離するため、PCI カードをグループに分割します。カードを vfio に追加し、グループを割り当てるには、[前述の Web ページ](#)で共有されているスクリプトを使用できます。

このスクリプトはカードをvfioにバインドします。/usr/local/bin/vfio-bindに配置します：

```
#!/bin/shmodprobe
vfio-pci
for dev in "$@"; do
    ベンダー=$(cat /sys/bus/pci/devices/$dev/ベンダー) デバイス
    =$(cat /sys/bus/pci/devices/$dev/デバイス)
    if [ -e /sys/bus/pci/devices/\$dev/driver ]; then
        echo $dev > /sys/bus/pci/devices/$dev/driver/unbind
    fi
    $vendor $device を /sys/bus/pci/drivers/vfio-pci/new_id に書き出します
done
```

設定は /etc/sysconfig/vfio-bind に記述されます。カードは PCI アドレスで指定します。アドレスは lspci コマンドで取得できます。通常 0000 であるドメインを必ず先頭に付加してください。例：

```
DEVICES="0000:04:00.0 0000:05:00.0 0000:84:00.0 0000:85:00.0"
```

以下は、このスクリプトを実行するsystemdスクリプトです。/etc/systemd/system/vfio-bindに記述し、サービスとして有効化できます：

```
[Unit]
Description=デバイスをvfio-pciにバインドしますAfter=syslog.target

[Service]
EnvironmentFile=-/etc/sysconfig/vfio-bindType=oneshot
終了後も残留=はい
実行開始=-/usr/local/bin/vfio-bind $DEVICES

[Install]
WantedBy=multi-user.target
```

### qemuの設定

次に、PCIカードに割り当てられたグループ用のvfioデバイスへのアクセス権をqemuに付与する必要があります。以下のコマンドでPCIカードとそのI/O MMUグループのリストを取得できます：

```
# find /sys/kernel/iommu_groups/ -type l
```

この例では、カードにはグループ45、46、58、59が割り当てられています。そのため、以下の設定を/etc/libvirt/qemu.confに追加します：

```
cgroup_device_acl = [
    "/dev/null", "/dev/full", "/dev/zero",
    "/dev/random", "/dev/urandom", "/dev/ptmx",
    "/dev/kvm", "/dev/kqemu", "/dev/rtc", "/dev/hpet",
    "/dev/vfio/vfio",
    "/dev/vfio/45", "/dev/vfio/46", "/dev/vfio/58", "/dev/vfio/59"
]
```

### 7.5.3 ドライバ設定

必要な設定は、PCIカードのリストを取得する監視プローブのフィルターのみです。デフォルトでは、プローブはホストからカードを表示しません。リストを絞り込むには、/var/lib/one/remotes/etc/im/kvm-probes.d/pci.conf で設定を変更できます。以下の設定属性が利用可能です：

パラメータ	説明
filter	PCIベンダー:デバイス:クラスのパターンによるリストフィルタリング (lspciと同様)
short_address	短いPCIアドレスによるフィルタリング:bus:device.function
デバイス名	大文字小文字を区別しない正規表現パターンによるデバイス名でのリストフィルタリング

すべてのフィルタは、最終的なPCIカードリストに適用

されます。例：

```
# このオプションは、PCIカード監視の主要なフィルタを指定します。フォーマットは、lspciコマンドがベンダー:デバイス(:クラス)識別子でPCIカードをフィルタリングする際に使用する形式と同じです。複数のフィルタはリストとして追加するか、カンマで区切って指定できます。NULLフィルタはすべてのPCIカードを取得します。
# 区切ることができます。NULL フィルターはすべての PCI カードを取得します。 #
# lspci のヘルプより:
```

```

#      -d [<ベンダー>]:[<デバイス>][:<クラス>]
#
#          指定されたベンダー、デバイス、クラス ID を持つデバイスのみを表示します。ID は 16 進数で指定さ
#          れ、省略することも「*」と指定することも可能です。いずれも「任意の値」を意味します。
#
# 例:
#      :filter:
#          - '10de:*'           # すべての NVIDIA VGA カード
#          - '10de:11bf'        # GK104GL [GRID K2] のみ
#          - '*:10d3'           # 82574L ギガビットネットワークカードのみ# Intel
#          - '8086::0c03'       USB コントローラのみ
#
# また
#      # :filter: '* :*'    すべてのデバイス
#
# また
#      # :filter: '0:0'      # デバイスがありま
#                           せん
:filter: '* :*'

上記の :filter オプションで制限された PCI カードの一覧は、正確な PCI アドレス (bus:device.func) の一覧によって
さらに絞り込むことが可能です。
#
# 例えば:
#      :short_address:
#          - '07:00.0'
#          - '06:00.0' #
:short_address:
- '00:1f.3'

# 上記の :filter オプションで制限された PCI カードの一覧は、デバイス名を正規表現の大文字小文字を区別しないパターン
一覧と照合することで、さらに絞り込むことが可能です。
#
# 例えば:
#      :device_name:
#          - '仮想関数' #          - 'ギガ
ビットネットワーク'
#          - 'USB.* ホストコントローラ' #
#          - '^MegaRAID'
#
:デバイス名:
- 'イーサネット'
- 'オーディオコントローラ'

```

## 7.5.4 使用方法

基本的なワークフローは、CLIまたはSunstoneでホスト情報を確認し、利用可能なPCIデバイスを特定した後、必要なデバイスをテンプレートに追加することです。PCIデバイスは、ベンダー、デバイス、クラスを指定するか、単にクラスを指定することで追加できます。OpenNebulaは、利用可能なPCIデバイスを備えたホストでのみVMを展開することに注意してください。一致するホストがない場合、スケジューラのログにエラーメッセージが表示されます。

## CLI

onehost show コマンドの新しいテーブルでは、ホストごとの PCI デバイス一覧を確認できます。例：

PCI デバイス	
仮想マシンアドレス	名前
00:00.0 8086:0a04:0600	Haswell-ULT DRAM コントローラ
00:02.0 8086:0a16:0300	Haswell-ULT 統合グラフィックス コントローラ
123 00:03.0 8086:0a0c:0403	Haswell-ULT HD オーディオコントローラ 00:14.0
8086:9c31:0c03 8	シリーズ USB xHCI HC
00:16.0 8086:9c3a:0780 8	シリーズ HECI #0
00:1b.0 8086:9c20:0403 8	シリーズ HD オーディオコントローラー 00:1c.0
8086:9c10:0604 8	シリーズ PCI Express ルートポート 100:1c.2 8086:9c14:0604 8 シ
リーズ PCI Express ルートポート 3 00:1d.0 8086:9c26:0c03 8	シリーズ USB EHCI #1
00:1f.0 8086:9c43:0601 8	シリーズ LPC コントローラ
00:1f.2 8086:9c03:0106 8	シリーズ SATA コントローラ 1 [AHCI モード] 00:1f.3
8086:9c22:0c05 8	シリーズ SMBus コントローラ
02:00.0 8086:08b1:0280	ワイヤレス 7260

- **VM:** その特定のデバイスを使用している仮想マシン (VM) の識別子です。該当デバイスを使用している VM が存在しない場合は空欄となります。
- **ADDR:** PCI アドレス。
- **TYPE:** デバイスを記述する値です。これらは VENDOR:DEVICE:CLASS の形式です。これらの値は、パスルートを行うために PCI デバイスを選択する際に使用されます。
- **NAME:** PCI デバイスの名称です。

仮想マシン内の PCI デバイスを利用するには、使用するデバイスを選択する新しいオプションを追加できます。例えば、以下は Haswell-ULT HD オーディオコントローラーを指定する例です：

```
PCI = [
    ベンダー = "8086", デ
    バイス = "0a0c", クラ
    ス = "0403"
]
```

デバイスは、すべてのタイプ値を指定せずに指定することも可能です。例えば、任意の PCI Express ルートポートを取得するには、VM テンプレートに以下を追加できます：

```
PCI = [
    CLASS = "0604"
]
```

複数の PCI デバイスを仮想マシンに接続するには、複数の PCI オプションを追加することができます。

## サンストーン

サンストーンでは、情報は PCI タブに表示されます：

VM	PCI Address	Type	Name
	00:00.0	8086:1604:0600	Broadwell-U Host Bridge -OPI
	00:02.0	8086:1616:0300	Broadwell-U Integrated Graphics
	00:03.0	8086:160c:0403	Broadwell-U Audio Controller
	00:04.0	8086:1603:1180	Broadwell-U Camarillo Device
	00:14.0	8086:9cb1:0c03	Wildcat Point-LP USB xHCI Controller
	00:16.0	8086:9cba:0780	Wildcat Point-LP MEI Controller #1
	00:1b.0	8086:9ca0:0403	Wildcat Point-LP High Definition Audio Controller
	00:1c.0	8086:9c90:0604	Wildcat Point-LP PCI Express Root Port #1
	00:1c.3	8086:9c96:0604	Wildcat Point-LP PCI Express Root Port #4
	00:1d.0	8086:9ca6:0c03	Wildcat Point-LP USB EHCI Controller

テンプレートにPCIデバイスを追加するには、[その他]タブを選択してください：

## 7.5.5 ネットワークインターフェースとしての使用

OpenNebulaでは、PCIデバイスをNICインターフェースとして直接使用することが可能です。これを行うには、本ガイドに記載されている設定手順、具体的にはデバイスドライバーの変更を行う必要があります。

PCIバススルーニックに使用するネットワークを定義する際は、ダミーネットワークドライバ、またはVLANをご利用の場合は802.1Qをご利用ください。いずれの場合も、BRIDGEフィールドには任意の値を入力していただければ、その値は無視されます。802.1Qをご利用の場合、PHYDEVは空白のままにしておくことも可能です。

コンテキストパッケージでは、以下の属性の設定をサポートしております：

- **MAC**: 対応するネットワークインターフェースのMACアドレスを、OpenNebulaが割り当てるMACアドレスに変更します。
- **IP**: インターフェースにIPv4アドレスを割り当てます。/24ネットマスクを前提とします。
- **IPV6**: インターフェースにIPv6アドレスを割り当てます。ネットマスクは/128を想定します。
- **VLAN\_ID**: 指定された場合、タグ付きインターフェースを作成し、IPアドレスをそのインターフェースに割り当てます。

## CLI

テンプレート内の `PCI` に `TYPE="NIC"` 属性が含まれている場合、それは `NIC` として扱われ、OpenNebula は PCI デバイスに MAC アドレス、`VLAN_ID`、IP アドレスなどを割り当てます。

以下は、NICとして扱われるインターフェースのPCIセクションの例です：

```
PCI=[  
    ネットワーク設定は「パススルー」です。ネット  
    ワークユーザー名は「oneadmin」です。デ  
    バイス種別は「NIC」です。  
    CLASS="0200",  
    DEVICE="10d3",  
    VENDOR="8086" ]
```

テンプレート内におけるPCI要素とNIC要素の出現順序は重要です。これらは、PCIのNICであるかどうかにかかわらず、出現順に NICにマッピングされます。

## Sunstone

ネットワークタブの「詳細オプション」で、PCIパススルーオプションにチェックを入れ、PCIアドレスを入力してください。残りのダイアログは通常通り、テーブルからネットワークを選択して操作してください。

### Hardware

PCI passthrough

Device name	Vendor	Device	Class
82574L Gigabit Network Connection	8086	10d3	0200

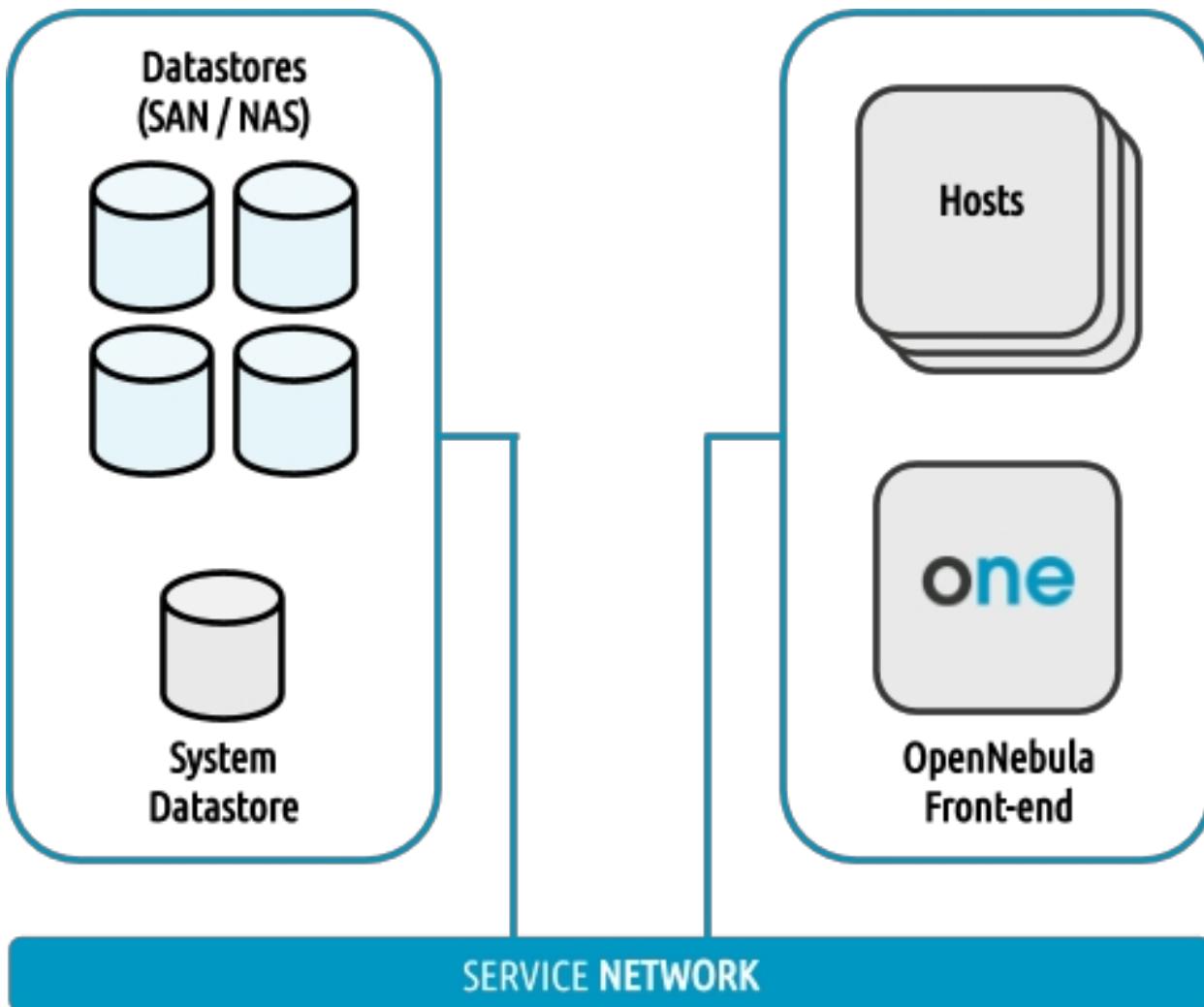
## オープンクラウドストレージの設定

### 8.1 概要

#### 8.1.1 データストアの種類

OpenNebulaのストレージは、データストアの概念を中心に構成されています。データストアとは、ディスクイメージを保存するためのあらゆるストレージ媒体を指します。OpenNebulaでは、以下の3種類のデータストアを提供しています：

- ・ **イメージデータストア**は、イメージリポジトリを格納します。
- ・ **システムデータストア**は、仮想マシンを実行するためのディスクを保持します。ディスクは、仮想マシンのデプロイ時や終了時、あるいはディスクの接続時やスナップショット作成時に、イメージデータストアとの間で移動または複製されます。
- ・ **ファイル&カーネルデータストア**は、ディスクイメージではなく通常のファイルを保存します。これらの通常のファイルは、カーネル、RAMディスク、またはコンテキストファイルとして使用できます。[詳細はこちらをご覧ください](#)。



## イメージデータストア

基盤となるストレージ技術における画像の保存方法に応じて、異なる画像データストアが存在します：

- ファイルシステム：イメージをファイル形式で保存します。
- *LVM*：イメージをLVM論理ボリュームに保存します。
- *Ceph*：Cephブロックデバイスを使用してイメージを保存します。
- *Raw Device Mapping*：ノード内の既存ブロックデバイスを仮想マシンに直接接続します。
- *iSCSI - Libvirt データストア*：組み込みの qemu サポートを通じて iSCSI デバイスにアクセスします。

ディスクイメージは、転送マネージャ（TM）ドライバによってイメージ・データストアとシステム・データストア間で転送されます。これらのドライバは、低レベルのストレージ操作を実行する特殊なソフトウェアです。以下の表は、各データストアで利用可能な転送モードをまとめたものです：

データストア	イメージからシステムデータストアへのディスク転送方法
ファイルシステム	<ul style="list-style-type: none"> <li><b>共有</b>、イメージは共有ファイルシステムにエクスポートされます</li> <li>SSH、SSHプロトコルを使用してイメージをコピーします</li> <li><b>qcow2</b>、<b>共有</b>と同様ですが、qcow2 フォーマットに特化しています</li> </ul>
Ceph	<ul style="list-style-type: none"> <li><b>ceph</b>、すべてのイメージは Ceph プールにエクスポートされます</li> <li><b>共有</b>、揮発性ディスクおよびコンテキストディスクは共有ファイルシステムでエクスポートされます。</li> </ul>
LVM	<ul style="list-style-type: none"> <li><b>fs_lvm</b>、イメージは共有ファイルシステムにエクスポートされますが、LV にダンプされます</li> </ul>
Rawデバイス	<ul style="list-style-type: none"> <li><b>dev</b>、イメージはノード内に存在するブロックデバイスです</li> </ul>
iSCSI libvirt	<ul style="list-style-type: none"> <li><b>iscsi</b>、イメージはiSCSIターゲットとなります</li> </ul>

### 8.1.2 本章の読み方について

本章をお読みになる前に、必ず「*Open Cloud Host*」の章をお読みください。その後、ご興味のあるデータストアに関する特定のセクションに進んでください。

本章をお読みになった後は、*Open Cloud Networking* の章をお読みください。

### 8.1.3 ハイパーバイザーの互換性

本章はKVMおよびLXDに適用されます。

---

**注記：**LXDドライバーはファイルシステムとCephのみをサポートしております。

---

vCenterに関する同様のガイドについては、*vCenter Storage* セクションをご参照ください。

## 8.2 ファイルシステム・データストア

ファイルシステム・データストアでは、仮想マシンイメージをファイル形式で保存できます。ファイルベースのディスクイメージを使用すると、デバイスベースのディスクに比べていくつかの利点があります（例：イメージのバックアップが容易、共有ファイルシステムの利用など）。ただし、場合によってはパフォーマンスが低下する可能性があります。

通常、複数のファイルシステム・データストアを用意することは、以下の目的で有効です：

- ストレージサーバー間のI/O操作の負荷分散
- 異なるクラスター・ホストには異なるデータストアをご利用ください
- 異なるイメージに異なる転送モードを適用する

- 異なるSLAポリシー（例：バックアップ）を異なるVMタイプやユーザーに適用できます
- クラウドに新しいストレージを簡単に追加できます

ファイルシステムデータストアでは、以下の3種類の転送モードをご利用いただけます：

- 共有モード**：イメージは共有ファイルシステムにエクスポートされます
- ssh**：イメージはSSHプロトコルを使用してコピーされます
- qcow2**：**共有モード**と同様ですが、qcow2形式に特化しています

**警告**：LXDデプロイメントでは、データストアのバインドマウントはサポートされていません。

## 8.2.1 データストアのレイアウト

イメージは対応するデータストアディレクトリ（/var/lib/one/datastores/<DATASTORE\_ID>）に保存されます。また、稼働中の各仮想マシンには、対応するシステムデータストア内に（VM ID を名前に持つ）ディレクトリが存在します。これらのディレクトリには、VM ディスクやチェックポイント、スナップショットなどの追加ファイルが含まれます。

例えば、イメージデータストア（1）に3つのイメージがあり、システムデータストア0から3台の仮想マシン（VM 0と2が稼働中、VM 7が停止中）が実行されているシステムの場合、以下のレイアウトとなります：

```
/var/lib/one/datastores
|-- 0/
|   |-- 0/
|   |   |-- disk.0
|   |   `-- disk.1
|   |-- 2/
|   |   '-- disk.0
|   '-- 7/
|       |-- チェックポイント
|       '-- disk.0
`-- 1
    |-- 05a38ae85311b9dbb4eb15a2010f11ce
    |-- 2bbec245b382fd833be35b0b0683ed09
    '-- d0e0df1fb8cfa88311ea54dfbcfc4b0c
```

---

**注記**： /var/lib/one/datastores の正規パスは、oned.conf ファイル内の DATASTORE\_LOCATION 設定属性で変更できます

---

### 共有および Qcow2 転送モード

共有転送ドライバーは、データストアがクラスター内の全ホストにマウントされていることを前提とします。通常、これはNFS、GlusterFS、Lustreなどの分散ファイルシステムを通じて実現されます。

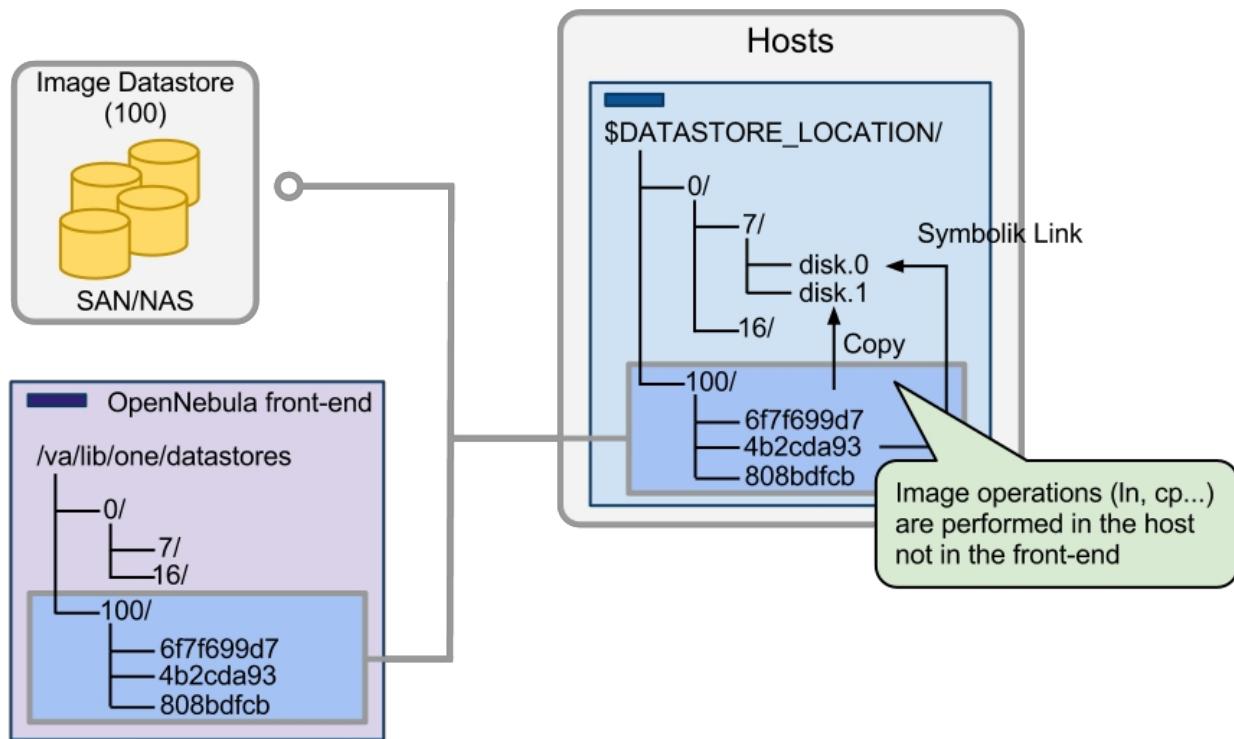
仮想マシンが作成されると、そのディスク（disk.iファイル）はシステムデータストアの対応するディレクトリにコピーまたはリンクされます。これらのファイル操作は常にターゲットホスト上でリモートで実行されます。

この転送モードは通常、仮想マシンの展開時間を短縮し、**ライブマイグレーションを可能**にしますが、仮想化されたサービスがディスク集約型のワークロードを実行する場合、インフラストラクチャのボトルネックとなり、仮想マシンのパフォーマンスを低下

させる可能性があります。通常、この制限は次の方法で克服できます：

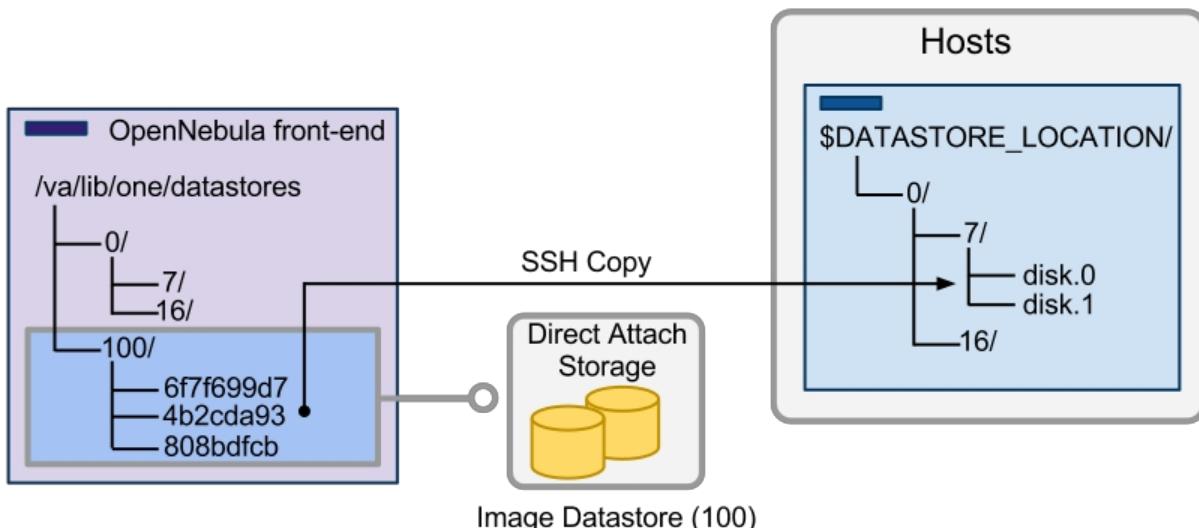
- イメージ用データストアに異なるファイルシステムサーバーを使用し、実際のI/O帯域幅を分散させる

- 代わりに SSH システムデータストアを使用し、イメージを各ホストにローカルにコピーする
- ファイルシステムサーバーのチューニングまたは改善



#### SSH転送モード

この場合、システムデータストアは各ホストに分散されます。SSH転送ドライバーは、稼働中の仮想マシンのイメージをホストのローカルストレージに配置します。すべての操作はローカルで実行されますが、イメージは常にホストにコピーする必要があります。これが非常にリソースを消費する操作となる可能性があります。また、このドライバーではホスト間のライブマイグレーションが利用できません。



## 8.2.2 フロントエンドの設定

フロントエンドでは、以下のストレージ領域を準備する必要があります：

- ・イメージを保存するためのイメージデータストア
- ・システムデータストア：停止中およびデプロイ解除された仮想マシン用の一時ディスクおよびファイルを保持します。

### 共有モードと Qcow2 転送モード

以下の手順で設定を行います：  
 イメージ データストア ディレクトリ フロントエンド /var/lib/one/datastores/  
`<datastore_id>`。すべてのデータストアが同一タイプの場合、/var/lib/one/datastores ディレクトリ全体をマウントすることも可能です。

**警告：**フロントエンドはイメージデータストアのみをマウントする必要があり、システムデータストアはマウントする必要はありません。

**注記:** NFS ボリュームのマウントに関するヒント。NFS 共有をマウントするには、以下のオプションを推奨します: `soft, intr, rsize=32768, wsize=32768`。libvirt/kvm の文書化された構成では、イメージファイルは `oneadmin` ユーザーとしてアクセスされます。ファイルを `root` として読み取る必要がある場合は、`no_root_squash` オプションを追加する必要があります。

### SSH転送モード

/var/lib/one/datastores ディレクトリの下に、イメージおよび停止中・デプロイ解除済みの仮想マシンのディスクを保存するのに十分な空き容量があることをご確認ください。なお、/var/lib/one/datastores はネットワーク内の任意の NAS/SAN サーバーからマウントすることが可能です。

## 8.2.3 ノード設定

### 共有および Qcow2 転送モード

設定は上記のフロントエンドと同様です。各ノードで以下のデータストアディレクトリをマウントしてください。

/var/lib/one/datastores/<datastore\_id> を各ノードにマウントしてください。

### SSH 転送モード

/var/lib/one/datastores 配下に、そのホスト上で稼働中の仮想マシンのディスクを保存するのに十分な空き容量があることをご確認ください。

**警告：** フロントエンドを含むすべてのホストが、他の任意のホスト（自身を含む）に SSH 接続できることを必ずご確認ください。そうしない場合、移行処理が正常に動作しません。

## **8.2.4 OpenNebulaの設定**

ファイルシステムストレージの設定が完了したら、OpenNebula の設定は以下の 2 つのステップで構成されます:

- システムデータストアの作成

- イメージデータストアの作成

### システムデータストアの作成

新しいシステムデータストアを作成するには、そのタイプをシステムデータストアとして指定し、転送モードを設定する必要があります：

属性	説明
NAME	データストアの名前
TYPE	SYSTEM_DS
TM_MAD	共有転送モード用の共有 qcow2転送モードの場合はqcow2 SSH転送モードの場合はSSH

これはSunstoneまたはCLIを通じて行うことが可能です。例えば共有モードでシステムデータストアを作成するには、以下のように単純に実行します：

```
$ cat systemds.txt NAME
    = nfs_system
TM_MAD = shared TYPE
    = SYSTEM_DS

$ onedatastore create systemds.txt ID: 101
```

**注記：**複数のシステムデータストアが利用可能な場合、データストアを選択した後に TM\_MAD\_SYSTEM 属性が設定されます。

### イメージデータストアの作成

同様に、イメージデータストアを作成するには、以下の設定が必要です：

属性	説明
NAME	データストアの名前
DS_MAD	fs
TM_MAD	共有転送モード用の共有 qcow2 転送モード用 qcow2 ssh 転送モード用

例えば、以下は共有転送ドライバを使用したファイルシステムデータストアの作成例です。

```
$ cat ds.conf
NAME      =
nfs_imagesDS_MAD = fs
TM_MAD = shared

$ onedatastore create ds.conf ID: 100
```

なお、追加で設定可能な属性もございますので、データストアテンプレートの属性をご確認ください。

**警告：**システムデータストアとイメージデータストアの両方で、必ず同じ TM\_MAD を使用してください。異なる転送モードを組み合わせる場合は、以下のセクションをご確認ください。

## 追加設定

- CONVERT: yes (デフォルト) または no。イメージデータストアで DRIVER が設定されている場合、このオプションは、異なる形式のイメージがインポート時に内部的に DRIVER 形式に変換されるかどうかを制御します。
- QCOW2\_OPTIONS: qemu-img クローン操作用のカスタムオプションです。qcow2 ドライバーは、ディスクイメージの qcow2 形式に対応するため、共有ドライバーを特化したものです。イメージは、元のイメージをバックアップファイルとして使用し、qemu-img コマンドを通じて作成されます。カスタムオプションは、/etc/one/tmrc 内の変数 QCOW2\_OPTIONS を通じて qemu-img クローン操作に送信できます。
- DD\_BLOCK\_SIZE: dd 操作 の ブ ロ ッ ク サ イ ズ ( デ フ ォ ル ツ : 64kB ) は、/var/lib/one/remotes/etc/datastore/fs/fs.conf で設定できます。

## 共有モードと SSH 転送モードの組み合わせ

共有モードをご利用の際には、ディスクをホストのローカルストレージ領域に配置することで、仮想マシンのパフォーマンスを向上させることができます。これにより、イメージのリポジトリ（共有ファイルシステムを使用してホスト間で分散）を維持しつつ、仮想マシンはローカルディスクから実行されます。これは、上記の共有モードとSSHモードを効果的に組み合せた形となります。

**重要：**この場合でも純粋な共有モードは引き続きご利用いただけます。これにより、同一イメージを共有モードまたはSSHモード（VM単位）で展開することが可能となります。

**警告：**この設定は、デプロイ時間を増加させる代償としてパフォーマンスを向上させます。

このシナリオを設定するには、上記で説明したように共有イメージおよびシステムデータストアを設定してください（TM\_MAD=shared）。その後、SSHシステムデータストアを追加します（TM\_MAD=ssh）。イメージデータストアに登録されたイメージは、共有データストアまたはSSHシステムデータストアを使用してデプロイできるようになります。

**警告：**共有データストアをクラスターに追加した場合、新しいSSHシステムデータストアを同じクラスターに追加する必要があります。

(代替) デプロイメントモードを選択するには、仮想マシンテンプレートに以下の属性を追加してください：

- TM\_MAD\_SYSTEM="ssh"

## 8.3 Ceph データストア

Ceph データストアドライバーは、OpenNebula ユーザーが Ceph ブロックデバイスを仮想イメージとして使用することを可能にしま

す。

**警告：**このドライバーをご利用になるには、Cephドライバーを使用するOpenNebulaノードが稼働中のCephクラスターのクライアントである必要があります。詳細は[Cephのドキュメント](#)をご参照ください。

### 8.3.1 データストアのレイアウト

イメージは、OpenNebula ID 「one-<IMAGE\_ID>」にちなんで命名されたCephプールに保存されます。仮想マシンのディスクは、デフォルトでは同じプールに保存されます（Cephモード）。また、SSHモードを使用してイメージのrbdをハイパーバイザーのローカルストレージにエクスポートすることも可能です。

---

**重要：**各イメージは一度登録するだけで結構です。その後は、CephモードまたはSSHモードのいずれでもデプロイが可能です。

---

#### Cephモード（デフォルト）

このモードでは、仮想マシンはディスク用に同じイメージの rbd ボリューム（永続イメージ）を使用するか、one-<IMAGE\_ID>-<VM\_ID>-<ディスク\_ID> の形式で作成されたイメージの新しいスナップショット（非永続イメージ）を使用します。

例えば、one という名前の Ceph プールでバックアップされたイメージおよびシステムデータストアを使用するシステムを考えてみましょう。1つのイメージ（ID 0）と、このイメージを仮想ディスク 0 として使用する 2 台の仮想マシン 14 および 15 を持つプールは、以下のようなものになります。

```
$ rbd ls -l -p one --id libvirt NAME
          SIZE PARENT          FMT PROT LOCK 2
one-0      10240M           2  はい
one-0@snap 10240M
one-0-14-0 10240M one/one-0@snapone-0-15-
0 10240M one/one-0@snap
```

---

**注：**この場合、コンテキストディスクおよび補助ファイル（デプロイメントの説明とチェックポイント）は、ノードにローカルで保存されます。

---

#### SSH モード

このモードでは、各ディスクに関連付けられた rbd ファイルがファイルにエクスポートされ、ハイパーバイザーのローカルファイルシステムに保存されます。

例えば、先の例でVM 14をSSHシステムデータストア（例：100）にデプロイする場合、ハイパーバイザー上のデータストアのレイアウトは以下のようになります：

```
$ ls -l /var/lib/one/datastores/100/14total 609228
-rw-rw-r-- 1 oneadmin oneadmin          1020 Dec 20 14:41 deployment.0
-rw-r--r-- 1 oneadmin oneadmin 10737418240 12月 20日 15:19 disk.0
-rw-rw-r-- 1 oneadmin oneadmin        372736 12月 20日 14:41 disk.1
```

---

**注記：**この場合、disk.0 は rbd export one/one-0@snap disk.0 のようなコマンドで生成されます。

---

### 8.3.2 Cephクラスタの設定

本ガイドでは、既に機能するCephクラスターが構築済みであることを前提としております。加えて、以下の作業が必要となります：

- OpenNebulaのデータストア用にプールを作成してください。データストア定義に含めるため、プールの名前をメモしてください。

```
$ ceph osd pool create one 128
$ ceph osd lspools
0 data,1 metadata,2 rbd,6 one,
```

- データストアプールにアクセスするためのCephユーザーを定義してください。このユーザーは、libvirtがディスクイメージにアクセスする際にも使用されます。例えば、libvirtというユーザーを作成します：

Ceph Jewel (v10.2.x) およびそれ以前のバージョンでは：

```
$ ceph auth get-or-create client.libvirt \
    mon 'allow r' osd 'allow class-read object_prefix rbd_children, allow wxr'
    mon 'allow r' osd 'allow class-read object_prefix rbd_children, allow rwx'
    ↪pool=one'
```

Ceph Luminous (v12.2.x) 以降では：

```
$ ceph auth get-or-create client.libvirt \
    mon 'profile rbd' osd 'profile rbd pool=one'
```

**警告**：Ceph Luminous リリースでは、RBD の機能が簡素化されております（ユーザー管理および認証機能に関する詳細は、[Ceph ドキュメント](#)をご参照ください）。既存の Ceph 環境を Luminous 以降にアップグレードされる際は、選択されたユーザーが適切な新機能を有していることをご確認ください。例えば、上記ユーザー libvirt に対しては、以下のコマンドを実行してください：

```
$ ceph auth caps client.libvirt \
    mon 'profile rbd' osd 'profile rbd pool=one'
```

- このユーザーのキーのコピーを取得し、後ほどOpenNebulaノードに配布してください。

```
$ ceph auth get-key client.libvirt | tee client.libvirt.key
$ ceph auth get client.libvirt -o ceph.client.libvirt.keyring
```

- RBDフォーマット1はサポートされていますが、フォーマット2の使用を強く推奨いたします。ceph.confに以下の設定が含まれていることをご確認ください。  
以下の設定が含まれていることをご確認ください：

```
[global] rbd_default_format = 2
```

- クラスタ内のクライアントノードの中から、ストレージブリッジとして機能するノード群を選択してください。これらのノードは、OpenNebula から Ceph クラスタへイメージをインポートするために使用されます。これらのノードには qemu-img コマンドがインストールされている必要があります。

---

**注記**：本番環境では、Cephサービス（モニター、OSD）をOpenNebulaノードやフロントエンド

---

### 8.3.3 フロントエンドとノードの設定

Cephクラスターをご利用いただくには、ノードを以下の通り設定する必要があります：

- マシンには Ceph クライアントツールがインストールされている必要があります
- すべてのノードにおいて、mon デーモンを ceph.conf に定義する必要があります。これにより、Ceph コマンド内でホスト名やポートを明示的に指定する必要がなくなります。

- Ceph ユーザーキーリング (`ceph.client.libvirt.keyring`) を各ノードの `/etc/ceph` ディレクトリにコピーし、ユーザーキー (`client.libvirt.key`) を `oneadmin` のホームディレクトリにコピーしてください。

```
$ scp ceph.client.libvirt.keyring root@node:/etc/ceph  
$ scp client.libvirt.key oneadmin@node:
```

#### 8.3.4 ノードの設定

ノードでは、libvirtで認証情報を設定するために追加の手順が必要です：

- Ceph ユーザー用のシークレットを生成し、ノードの oneadmin ホームディレクトリ下にコピーしてください。後で使用するため、`UUID` をメモしておいてください。

```
$ UUUID=`uuidgen` ; echo $UUID c7bdeabf-5f2a-  
4094-9413-58c6a9590980  
  
$ cat > secret.xml <<EOF  
<secret ephemeral='no' private='no'>  
  <uuid>$UUID</uuid>  
  <usage type='ceph'>  
    <name>client.libvirt secret</name>  
  </usage>  
</secret>  
EOF  
  
$ scp secret.xml oneadmin@node:
```

- libvirtのシークレットを定義し、ノード内のキーファイルを削除します：

```
$ virsh -c qemu:///system secret-define secret.xml  
  
$ virsh -c qemu:///system secret-set-value --secret $UUID --base64 $(cat client.  
˓→libvirt.key)  
  
$ rm client.libvirt.key
```

- oneadmin アカウントは、上記で定義した libvirt Ceph ユーザーを使用して Ceph クラスターにアクセスする必要があります。これには、ceph ユーザーのキーリングへのアクセス権が必要です。ノードで Ceph クライアントが正しく設定されていることをテストしてください。

```
$ ssh oneadmin@node  
$ rbd ls -p one --id libvirt
```

これに関する詳細情報は、Cephガイド「[Cephでのlibvirtの使用](#)」をご確認いただけます。

- ・コンテキストディスク、デプロイメントファイル、チェックポイントファイルなどの仮想マシン補助ファイルは、ノードの /var/lib/one/datastores/ 配下に作成されます。これらのファイル用に十分なストレージがノードにプロビジョニングされていることをご確認ください。
  - ・SSHモードをご利用になる場合は、システムデータストアに必要なスペースを考慮する必要があります。  
/var/lib/one/datastores/<ds\_id> (ds\_id はシステムデータストアの ID です) に必要なスペースを考慮する必要があります。

## I XDノードの設定

仮想化ノードが KVM の場合は、これらの手順をスキップして

---

ください。rbd-nbd をインストールします

```
apt install rbd-nbd
```

**警告:** LXD ドライバの rbd バックエンドには、ceph Luminous およびカーネル 4.15 が必要です。

### 8.3.5 OpenNebula の設定

OpenNebula で Ceph クラスターをご利用になるには、システムおよびイメージのデータストアを定義する必要があります。各イメージ/システム データストアのペアは、以下の Ceph 構成属性を共有します：

属性	説明	マンダ-トリー
NAME	データストアの名前	はい
プール名	Ceph プールの名称	はい
CEPH_USER	libvirt および rbd コマンドで使用される Ceph ユーザー名です。	はい
CEPH_KEY	ユーザー用のキーファイル。設定されていない場合、デフォルトの場所が使用されます。	いいえ
CEPH_CONF	必要に応じて、デフォルト以外のCeph設定ファイルをご使用ください。	いいえ
RBD_FORMAT	デフォルトでは RBD フォーマット 2 が使用されます。	いいえ
BRIDGE_LIST	Ceph クラスターにアクセスするためのストレージブリッジの一覧	はい
CEPH_HOST	Ceph モニターのスペース区切りリスト。例：host1 host2:port2 host3 host4:port4	はい
CEPH_SECRET	libvirt シークレットの UUID です。	はい
EC_POOL_NAME	Ceph エラクションコーディングプールの名称	いいえ
CEPH_TRASH	指定されたデータストア (Luminous+) でトラッシュ機能を有効にします。値：yes no	いいえ

**注記 :**Cephにおいて、異なる割り当て/レプリケーションポリシーを持つ他のプールを指す別のイメージおよびシステムデータストアを追加することができます。

**注記 :**Ceph Luminous リリースでは、RBD イメージに対してエラー訂正符号化 (Erasure Coding) の使用が可能となりました。一般的に、エラー訂正符号化されたイメージはより少ない容量で保存できますが、I/O パフォーマンスは低下します。イメージおよび/またはシステムデータストアでエラー訂正符号化を有効にするには、EC\_POOL\_NAME をエラー訂正符号化データプールの名前で設定します。イメージのメタデータには、通常のレプリケーションされた Ceph プール POOL\_NAME が依然として必要です。詳細は [Ceph ドキュメント](#) をご参照ください。

### システムデータストアの作成

システムデータストアには以下の属性も必要です：

属性	説明	マンダ-トリー
属性		
タイプ	SYSTEM_DS	はい
TM_MAD	ceph (完全なcephモードを使用するには、上記を参照してください) ssh (ローカルホストストレージを使用するには、sshモード上記参照)	はい

---

Sunstone または CLI を通じてシステムデータストアを作成します。例：

```
$ cat systemds.txt NAME =
ceph_system TM_MAD = ceph
TYPE      = SYSTEM_DS

POOL_NAME      = one
CEPH_HOST      = "ホスト1 ホスト2:ポート
2"CEPH_USER    = libvirt
CEPH_SECRET= "6f88b54b-5dae-41fe-a43e-b2763f601cfc"BRIDGE_LIST= cephfrontend

$ onedatastore create systemds.txt ID: 101
```

**注記:** 複数のシステムデータストアが利用可能な場合、データストアを選択後に TM\_MAD\_SYSTEM 属性が設定されます。

### イメージデータストアの作成

関連するシステムデータストアと同じである必要がある前述の属性とは別に、イメージデータストアには以下を設定できます。

属性	説明	必須
NAME	データストアの名前	必須
DS_MAD	ceph	はい
TM_MAD	ceph	はい
ディスクタイプ	RBD	はい
ステージングディレクトリ	ブリッジにおけるイメージ操作のデフォルトパス	いいえ

データストアの例:

```
> cat ds.conf
NAME =
= "cephds" DS_MAD =
ceph TM_MAD = ceph

DISK_TYPE = RBD

POOL_NAME      = one
CEPH_HOST      = "ホスト1 ホスト2:ポート2"
CEPH_USER      = libvirt
CEPH_SECRET = "6f88b54b-5dae-41fe-a43e-b2763f601cfc"BRIDGE_LIST =
cephfrontend

> onedatastore create ds.conf ID:
101
```

**警告:** SSH モードで TM\_MAD\_SYSTEM 属性を使用する場合は、SSH タイプのシステムデータストアが設定されている必要があります。

---

注記: Ceph バックエンドストレージのクオータに関する詳細については、クオータをご覧ください。

---

## 追加設定

Ceph ドライバーのデフォルト値は、`/var/lib/one/remotes/etc/datastore/ceph/ceph.conf` で設定できます。

- `POOL_NAME`: デフォルトのボリュームグループ
- `STAGING_DIR`: ストレージブリッジにおけるイメージ操作のデフォルトパス
- `RBD_FORMAT`: RBD ボリュームのデフォルト形式。
- `DD_BLOCK_SIZE`: `dd` 操作におけるブロックサイズ（デフォルト: 64kB）。

## 異なるモードの使用

VMテンプレートを作成する際、ディスクのデプロイ方法としてデフォルトのCephモードまたはSSHオンモードを選択できます。なお、選択したモードはVMの全ディスクに適用されます。デプロイモードを設定するには、VMテンプレートに以下の属性を追加してください：

- `TM_MAD_SYSTEM="ssh"`

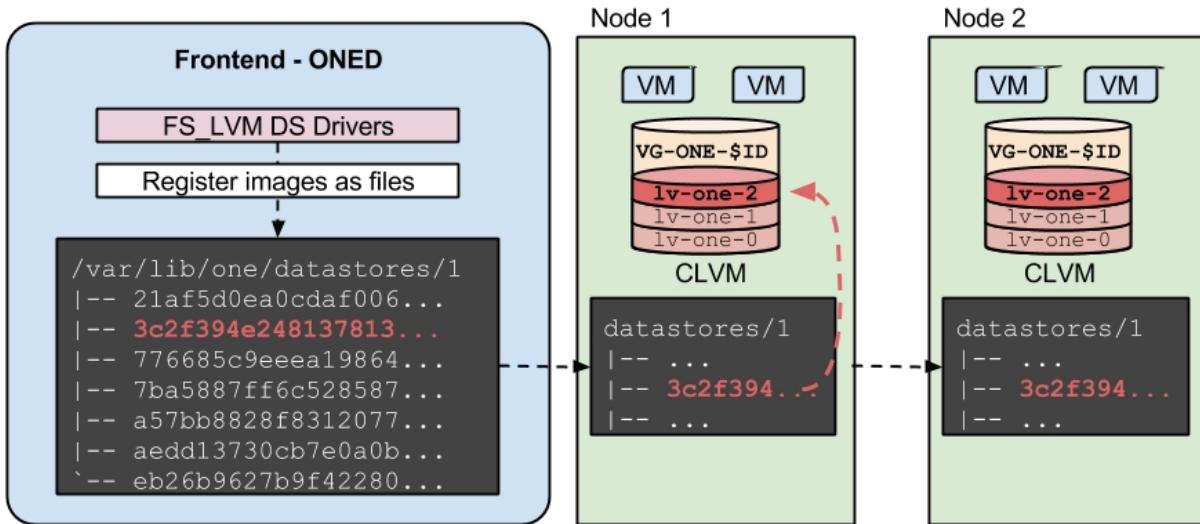
サンストーンをご利用の際には、ストレージタブにてデプロイメントモードの設定が必要となります。

## 8.4 LVM データストア

LVM データストアドライバは、仮想イメージを保存するために単純なファイルではなく LVM ボリュームを使用する可能性を OpenNebula に提供します。これにより、ファイルシステムを設置するオーバーヘッドが削減され、I/O パフォーマンスが向上する可能性があります。

### 8.4.1 データストアのレイアウト

イメージはイメージデータストア内に通常のファイルとして保存されます（通常のパス：`/var/lib/one/datastores/<id>`）。ただし、仮想マシンの作成時には論理ボリューム（LV）にダンプされます。仮想マシンはノード内の論理ボリュームから実行されます。



ハイエンドSANが利用可能な場合、こちらのドライバーの使用をお勧めいたします。同一のLUNを全ホストにエクスポートすることが可能であり、仮想マシンはSANから直接動作いたします。

**注記：**LVMデータストアをご利用の場合、クラスター内でCLVMを設定する必要はございません。ドライバーは、別のホストでイメージが必要となるたびにLVMメタデータを更新いたします。

例えば、ID 0 の LVM データストア上で動作するディスクを使用する 2 台の仮想マシン（9 と 10）が存在するシステムを考えてみましょう。ノードは共有 LUN を設定し、vg-one-0 というボリュームグループを作成しています。この場合、データストアのレイアウトは以下のようになります：

```
# lvs
  LV      VG      属性          LSize Pool Origin Data% Meta% Move
  lv-one-10-0 vg-one-0 -wi----- 2.20g
  lv-one-9-0 vg-one-0 -wi----- 2.20g
```

**警告：**イメージはファイル形式で共有ストレージ（例：NFS、GlusterFSなど）に保存されます。データストアディレクトリとマウントポイントは、通常の共有イメージデータストアとして設定する必要があります。詳細は[ファイルシステムデータストアガイド](#)をご参照ください。まず共有ファイルシステムデータストアを展開し、正常に動作することを確認した後、関連するシステムデータストアをLVMデータストアに置き換えることをお勧めいたします。その際、下記の手順に従い共有マウントポイントを保持してください。

## 8.4.2 フロントエンドの設定

- フロントエンドは、イメージデータストアへのアクセス権を持ち、関連するディレクトリをマウントする必要があります。

## 8.4.3 ノードの設定

ノードは以下の要件を満たす必要があります：

- ホストではLVM2が利用可能である必要があります。

#### **OpenNebula 5.10 導入ガイド、リリース 5.10.5**

- 
- lvmtdは無効化する必要があります。/etc/lvm/lvm.conf ファイル内で以下の設定を行ってください：use\_lvmetad = 0。また、lvm2-lvmetad.service が実行中の場合は停止してください。
  - oneadmin はディスクグループに属している必要があります。

- すべてのノードが同じ LUN にアクセスできる必要があります。
- LVM VG には、以下の設定が必要です。 共有 LUN 各 データストア 以下の 名前:  
vg-one-<system\_ds\_id> となります。これはノード1台で実行すれば十分です。
- 仮想マシンのディスクはブロックデバイスへのシンボリックリンクです。ただし、チェックポイントやデプロイメントファイルなどの追加のVMファイルは、/var/lib/one/datastores/<id>の下に保存されます。十分なローカル領域が確保されていることをご確認ください。
- すべてのノードは、関連するディレクトリをマウントすることで、イメージおよびシステムデータストアへのアクセス権を持つ必要があります。
  - 。

**注記:** 仮想化ホストの再起動時には、ボリュームを再度アクティブ化してハイパーバイザーが利用可能にする必要があります。ノードパッケージがインストールされている場合、アクティブ化は自動的に行われます。そうでない場合、再起動前にホスト上で稼働していた仮想マシンの各ボリュームデバイスは、lvchangeコマンドを実行して手動でアクティブ化する必要があります。  
-ay \$DEVICE (または、リモートスクリプト内のアクティベーションスクリプト /var/tmp/one/tm/fs\_lvm/activate をホスト上で実行して処理を行うことも可能です)。

#### 8.4.4 OpenNebula の設定

ストレージの設定が完了しましたら、OpenNebula の設定は以下の 2 つのステップで構成されます：

- システムデータストアの作成
- イメージデータストアの作成

##### システムデータストアの作成

LVM システムデータストアは、以下の値で作成する必要があります:

属性	説明
NAME	データストアの名前
TM_MAD	fs_lvm
タイプ	SYSTEM_DS
ブリッジリスト	監視対象のLVにアクセス可能なノードの一覧

例：

```
> cat ds.conf
NAME      = lvm_system
TM_MAD   = fs_lvm TYPE
          = SYSTEM_DS
ブリッジリスト = "node1.kvm.lvm node2.kvm.lvm"

> onedatastore 作成 ds.conf ID: 100
```

##### イメージデータストアの作成

---

イメージデータストアを作成するには、名前を定義し、以下の設定を行う必要があります：

属性	説明
NAME	データストアの名前
タイプ	IMAGE_DS
DS_MAD	fs
TM_MAD	fs_lvm
ディスクタイプ	ブロック

たとえば、以下の例は、設定ファイルを使用して LVM データストアを作成する方法を示しています。この場合、OpenNebula の LVM 対応ホストの 1 つとして host01 ホストを使用します。

```
> cat ds.conf
NAME =
production DS_MAD =
fs TM_MAD = fs_lvm
ディスクタイプ = "ブロック"タイ
プ = IMAGE_DS
安全なディレクトリは「/var/tmp /tmp」です

> onedatastore による ds.conf の作成
ID: 101
```

### ドライバ設定

デフォルトでは、LVM ドライバーは LVM ボリュームをすべてゼロクリアするため、VM データが他のインスタンスに漏洩することはありません。ただし、この処理には時間がかかり、VM のデプロイが遅延する可能性があります。ドライバーの動作は、ファイル /var/lib/one/remotes/etc/fs\_lvm/fs\_lvm.conf で設定できます。具体的には以下の通りです：

属性	説明
ZERO_LVM_ON_CREATE	LVMボリュームの作成時/サイズ変更時にゼロクリアを行う
ZERO_LVM_ON_DELETE	仮想マシンディスクが削除される際にLVMボリュームをゼロクリアします
DD_BLOCK_SIZE	dd 操作におけるブロックサイズ (デフォルト: 64kB)

例:

```
# LVMボリュームの作成時またはサイズ変更時に初期化
ZERO_LVM_ON_CREATE=no

VMディスクが破棄される際に、LVMボリュームを削除しない設定
ZERO_LVM_ON_DELETE=yes

# ddコマンドのブロックサイズ
DD_BLOCK_SIZE=32M
```

## 8.5 Raw Device Mapping (RDM) データストア

RDMデータストアは、ノードのブロックデバイスへの直接アクセスを可能にするイメージデータストアです。このデータストアでは、イメージデータストアからシステムデータストアへの転送操作が存在しないため、仮想マシンの迅速な展開を実現します。

**警告:** このデータストアは管理者だけが使用できる状態にしておく必要があります。ユーザーがこのデータストアでイメージを作成できるようにすると、セキュリティ上の問題が発生します。例えば、/dev/sda というイメージを登録し、ホストのファイルシステムを読み取ることが可能になります。

### 8.5.1 データストアのレイアウト

RDMデータストアは、ノード内に既に存在するブロックデバイスを登録するために使用されます。デバイスは事前に設定済みで利用可能な状態である必要があります、これらのデバイスを利用する仮想マシンは、それらに対応したノード上で実行されるように固定されている必要があります。デプロイメントファイルや揮発性ディスクなどの追加の仮想マシンファイルは、通常のファイルとして作成されます。

### 8.5.2 フロントエンドの設定

追加の設定は不要です。RDM データストア用に `/etc/one/oned.conf` に以下の設定がされていることをご確認ください：

```
TM_MAD_CONF = [
    NAME = "dev", LN_TARGET = "NONE", CLONE_TARGET = "NONE", SHARED = "YES", TM_MAD_SYSTEM = "ssh,shared",
    LN_TARGET_SSH = "SYSTEM", CLONE_TARGET_SSH = "SYSTEM", DISK_TYPE_SSH = "BLOCK", LN_TARGET_SHARED = "NONE",
    クローン対象共有 = "自身", 共有ディスクタイプ = "ブロック"
]
```

### 8.5.3 ノードの設定

仮想マシンに接続するデバイスは、ハイパーバイザーからアクセス可能な状態にしてください。KVM は通常 `oneadmin` として実行されるため、このユーザーがディスクアクセス権限を持つグループ（例：`disk`）に所属し、そのグループに対して読み書き権限が与えられていることを確認してください。

### 8.5.4 OpenNebulaの設定

ストレージの設定が完了したら、OpenNebula の設定は次の 2 つのステップで構成されます。

- システムデータストアの作成
- イメージデータストアの作成

#### システムデータストアの作成

RDM データストアは、以下のシステム データストアと連携することが可能です：

- ファイルシステム、共有転送モード
- ファイルシステム、SSH転送モード

詳細については、[ファイルシステム・データストアのセクション](#)をご参照ください。なお、システム・データストアは揮発性ディスクおよびコンテキスト・デバイス専用となります。

#### イメージデータストアの作成

イメージデータストアを作成するには、名前を定義し、以下の設定を行う必要があります：

属性	説明
----	----

NAME	データストアの名前
タイプ	IMAGE_DS
DS_MAD	dev
TM_MAD	dev
ディスクタイプ	ブロック

データストアの例：

```
> cat rdm.conf
NAME      = rdm_datastore
TYPE      = "IMAGE_DS"
DS_MAD    = "dev"
TM_MAD    =
"dev"DISK_TYPE = "BLOCK"

> onedatastore create rdm.conf ID:
101
```

### 8.5.5 データストアの使用方法

新しいイメージは、パスを指定することで他のイメージと同様に追加できます。例として、ノードディスク /dev/sdb を追加するためのイメージテンプレートを以下に示します：

```
cat image.tpl

NAME=scsi_device
PATH=/dev/sdb
PERSISTENT=YES

oneimage create image.tpl -d 101
```

CLIの省略形パラメータをご利用の場合は、sourceを使用してイメージを定義してください：

```
oneimage create -d 101 --name nbd --source /dev/sdc --driver raw --prefix vd --
--persistent --type OS --size 0MB
```

---

**注記：**このデータストアは既存のデバイスイメージを格納するコンテナに過ぎないため、それ自体はいかなるサイズも占有しません。登録された全デバイスはサイズ0として表示され、デバイスデータストア全体の空き容量は1MBとなります。

---

## 8.6 iSCSI - Libvirt データストア

このデータストアは、ハイパーバイザーノードが利用可能な既存のiSCSIボリュームを登録するために使用されます。

**警告：**このデータストアは管理者だけが使用できる状態にしておく必要があります。ユーザーがこのデータストアにイメージを作成できるようにすると、セキュリティ上の問題を引き起こす可能性があります。

### 8.6.1 フロントエンドの設定

追加の設定は必要ありません

### 8.6.2 ノードの設定

ノードは以下の要件を満たす必要があります：  
 \* VM に接続するデバイスは、ハイパーバイザーからアクセス可能である必要があります。  
 \* Qemu は Libiscsi サポート付きでコンパイルされている必要があります。

### iSCSI CHAP認証

CHAP認証をご利用いただくには、すべてのハイパーバイザーにlibvirtシークレットを作成する必要があります。シークレットの登録には、こちらのLibvirtシークレットXMLフォーマットガイドに従ってください。以下の点にご留意ください：

- iSCSI認証ファイルのincominguserフィールドは、データストアのISCSI\_USERパラメータと一致させる必要があります。
- シークレット XML ドキュメント内の <target> フィールドには、ISCSI\_USAGE パラメータが含まれます。
- すべてのハイパーバイザーでこの操作を行ってください。

### 8.6.3 OpenNebulaの設定

ストレージの設定が完了したら、OpenNebula の設定は次の 2 つのステップで構成されます：

- システムデータストアの作成
- イメージデータストアの作成

#### システムデータストアの作成

RDM データストアは、以下のシステムデータストアと連携可能です：

- ファイルシステム、共有転送モード
- ファイルシステム、SSH転送モード

詳細については、[ファイルシステムデータストアのセクション](#)をご参照ください。なお、システムデータストアは揮発性ディスクおよびコンテキストデバイス専用となります。

#### イメージデータストアの作成

イメージデータストアを作成するには、名前を定義し、以下の設定を行う必要があります：

属性	説明
NAME	データストアの名前
タイプ	IMAGE_DS
DS_MAD	iscsi_libvirt
TM_MAD	iscsi_libvirt
ディスクタイプ	ISCSI
ISCSI_HOST	ISCSIホスト。例：ホスト名またはホスト名:ポート番号。

CHAP認証（オプション）をご利用になる場合は、データストアに以下の属性を追加してください：

属性	説明
ISCSI_USAGE	CHAP認証文字列として使用するシークレットです。
ISCSI_USER	iSCSI CHAP認証で使用されるユーザーです。

データストアの例：

#### 8.6.4 iSCSI Libvirt データストア

```
NAME = iscsi
```

```
DISK_TYPE = "ISCSI"
```

```

DS_MAD = "iscsi_libvirt" TM_MAD =
"iscsi_libvirt"

ISCSI_HOST = "the_iscsi_host" ISCSI_USER =
"the_iscsi_user" ISCSI_USAGE =
"the_iscsi_usage"

> onedastore による iscsi.ds の作成
ID: 101

```

**警告:** このデータストアで作成されるイメージは永続化する必要があります。イメージを非永続化に設定すると、複数の仮想マシンがこのデバイスを使用できるようになり、問題やデータ破損を引き起こす可能性があります。

#### 8.6.4 データストアの使用方法

新しいイメージは、パスを指定して他のイメージと同様に追加できます。CLI を使用する場合は、**短縮パラメータを使用しないでください**。CLI はファイルの存在を確認しますが、デバイスはフロントエンドに存在しない可能性が高いです。

例 例 こちら は ある 画像 テンプレート です 追加する a ノード ディスク 例:  
storage:diskarrays-sn-a8675309:

```

NAME = iscsi_device
PATH = iqn.1992-01.com.example:storage:diskarrays-sn-a8675309PERSISTENT = YES

```

**警告:** このデータストアは既存のデバイスイメージを格納するコンテナに過ぎないため、そこからサイズを消費することはありません。登録されたすべてのデバイスはサイズ0として表示され、デバイスデータストア全体の空き容量は1MBと表示されます。

---

**注記:** イメージテンプレート内の以下のパラメータ (ISCSI\_HOST、ISCSI\_USER、ISCSI\_USAGE、ISCSI\_IQN) は、いずれでも上書き設定が可能です。上書きされたパラメータは、新規に作成される仮想マシンに適用されます。

以下は、iSCSI転送マネージャーを使用するiSCSI LUNテンプレートの例です。

```

oneadmin@onedv:~/exampletemplates$ more
iscsiimage.tpl NAME=iscsi_device_with_lun PATH=iqn.2014.01.192.168.50.61:test:7cd2cc1e/
0ISCSI_HOST=192.168.50.61
PERSISTENT=YES

```

IQN ターゲットパスの末尾に明示的に「/0」が付いている点にご注意ください。これは iSCSI LUN ID です。

#### 8.7 カーネルとファイルデータストア

ファイルデータストアは、VMカーネル、RAMディスク、またはコンテキストファイルとして使用するブレーンファイルを保存するために利用できます。ファイルデータストアは特別なストレージ機構を提供するものではありませんが、VMテンプレート内でファイルをシンプルかつ安全に利用する方法を提供します。OpenNebulaでは、ファイルデータストア（データストアID: 2）がすぐに利用可能です。

## 8.7.1 要件

ファイルデータストアをご利用いただくにあたり、特別な要件やソフトウェア依存関係はございません。推奨されるドライバは、標準的なファイルシステムユーティリティ（cp、ln、mv、tar、mkfsなど）を利用しておられ、これらはシステムにインストールされている必要があります。

## 8.7.2 構成

ディスクイメージデータストアで使用される設定上の考慮事項のほとんどは、ファイルデータストアにも適用されます（例：ドライバの設定、クラスタの割り当て、データストアの管理など）。

このデータストアドライバー固有の属性は下記の表にまとめられております。また、共通のデータストア属性についても設定が必要です。

属性	説明
タイプ	FILE_DS を使用してファイルデータストアを設定します
DS_MAD	データストアのタイプ。fsを使用するとファイルベースのドライバーが使用されます
TM_MAD	データストアの転送ドライバー。ファイル転送にはSSHをご利用ください

例えば、以下の例はファイルデータストアの作成を示しています。

```
> cat kernels.ds.confNAME
= kernels
DS_MAD = fsTM_MAD =
sshTYPE = FILE_DS
SAFE_DIRS = /var/tmp/files

> onedatastore create kernels.ds.conf ID: 100

> onedatastore list
```

ID	名前	クラスタ	イメージ	タイプ	DS	TM
0	システム	-	0	シス	-	ダミー
1	デフォルト	-	0	画像	ダミー	ダミー
2	ファイル	-	0	ファイ	fs	SSH
100	カーネル	-	0	ファイ	fs	SSH

DSおよびTM MADは、onedatastore updateコマンドを使用して後から変更することができます。onedatastore showコマンドを実行することで、データストアの詳細情報を確認いただけます。

## 8.7.3 ホスト構成

ファイルデータストアに推奨されるSSHドライバーは、ホスト側での特別な設定を必要としません。フロントエンドおよびホスト上でVMファイルを格納するのに十分な空き容量が\$DATASTORE\_LOCATION配下に確保されていることをご確認ください。

詳細については、[ファイルシステムデータストアガイド](#)をご参考ください。同じ設定ガイドラインが適用されます。

## オープンクラウドネットワーキングの設定

### 9.1 概要

新しい仮想マシンが起動されると、OpenNebulaはそのネットワークインターフェース（NIC属性で定義）を仮想ネットワークで定義されたハイパーバイザの物理デバイスに接続します。これにより、仮想マシンはパブリックまたはプライベートの異なるネットワークにアクセスできるようになります。

OpenNebulaは4つの異なるネットワークモードをサポートしております：

- *ブリッジモード*。仮想マシンはハイパーバイザ内の既存ブリッジに直接接続されます。このモードでは、セキュリティグループやネットワーク分離の設定が可能です。
- *VLAN*。仮想ネットワークは802.1Q VLANタグ付けを通じて実装されます。
- *VXLAN*。仮想ネットワークは、UDPカプセル化とIPマルチキャストに依存するVXLANプロトコルを使用してVLANを実装します。
- *Open vSwitch*。VLANモードと類似していますが、Linuxブリッジの代わりにopenvswitchを使用します。
- *Open vSwitch on VXLAN*。VXLANモードと同様ですが、Linuxブリッジの代わりにopenvswitchを使用します。

新しいネットワークを作成する際には、上記のネットワークモードのいずれを使用するか指定するため、テンプレートに属性 `VN_MAD` を追加する必要があります。

---

**注記：**Open vSwitchモードではセキュリティグループはサポートされていません。

---

各ネットワークドライバには、VMの起動前 (*pre*)、起動後 (*post*)、および VM がホストから離脱するとき (*clean*) に実行される3種類の設定アクションがあります。これらのドライバアクションは、ネットワークドライバディレクトリ内の対応するアクションフォルダ (*pre.d*, *post.d*, *clean.d*) に実行ファイルを配置することで、カスタムプログラムで拡張することができます。

最後に、OpenNebulaのネットワークスタックは、外部IPアドレス管理システム（IPAM）と統合することができます。これを行うには、必要な連携機能を開発する必要があります。詳細については、IPAM ドライバガイドをご参照ください。

#### 9.1.1 本章の読み方について

この章をお読みになる前に、必ず「[オープンクラウドストレージ](#)」の章をお読みください。

まず、共通の [ノート](#) 設定セクションをお読みいただき、ホストの設定方法をご確認ください。その後、ご希望のネットワークモード

に対応した特定のセクションに進んでください。

本章をお読みになった後は、オプションで[外部認証](#)を有効化するか、[Sunstone](#)を設定することでOpenNebulaのインストールを完了できます。それ以外の場合は、クラウドの運用を開始する準備が整っています。

## 9.1.2 ハイパーバイザの互換性

本章はKVMおよびLXDに適用されます。

## 9.2 ノードの設定

本ガイドには、各ネットワークモードを有効にするための具体的なノード設定手順が含まれています。選択したモードに対応するセクションのみを適用してください。

### 9.2.1 ブリッジド・ネットワーキング・モード

#### 前提条件

- OpenNebulaノードパッケージがインストールされていること。詳細は[KVMノードのインストールセクション](#)をご参照ください。
- デフォルトでは、ネットワーク分離は `ebtables` を通じて提供されます。このパッケージはノードにインストールする必要があります。

#### 設定

- 仮想マシンに公開される各ネットワークに対して、Linuxブリッジを作成してください。すべてのノードで同じ名前を使用してください。
- 物理ネットワークインターフェースをブリッジに追加してください。

例えば、パブリックIPアドレス用（`eth0`に接続）とプライベート通信用（NIC `eth1`）の2つのネットワークを持つノードの場合、以下の2つのブリッジが必要です：

```
$ ip link show type bridge4: br0:
...
5: br1: ...

$ ip link show master br0 2:
eth0: ...

$ ip link show master br1 3:
eth1: ...
```

**注記：**この設定は永続化することをお勧めいたします。設定方法につきましては、お使いのシステムのネットワーク設定ガイドをご参照ください。

### 9.2.2 VLAN ネットワークモード

#### 要件

- OpenNebulaノードパッケージがインストールされていること。詳細は[KVMノードのインストールセクション](#)をご参照ください。
- カーネルに `8021q` モジュールがロードされている必要があります。

- 
- VLAN タグ付きトラフィックの転送が可能なネットワークスイッチが必要です。物理スイッチポートは VLAN トランクである必要があります。

## 設定

追加の設定は必要ありません。

### 9.2.3 VXLAN ネットワークモード

#### 要件

- OpenNebulaノードパッケージのインストールが完了しました。詳細については、[KVMノードのインストールセクション](#)をご参照ください。
- ノードは、VXLANプロトコルおよび関連するiproute2パッケージをネイティブでサポートするLinuxカーネル（バージョン3.7.0以上）を実行している必要があります。
- すべてのノードが同一のブロードキャストドメインに接続されている場合、ノード内のiptableルールによってマルチキャストトラフィックがフィルタリングされていないことをご確認ください。なお、マルチキャストトラフィックがルーターを経由する必要がある場合、ネットワーク上でIGMPなどのマルチキャストプロトコルを設定する必要があります。

## 設定

追加の設定は必要ありません。

### 9.2.4 Open vSwitch ネットワークモード

#### 要件

- OpenNebulaノードパッケージがインストールされている必要があります。詳細はKVMノードの[インストールセクション](#)をご参照ください。
- 各ノードにOpen vSwitchをインストールする必要があります。インストール方法については、Open vSwitchのドキュメントをご参照ください。

例えば、enp0s8 ネットワークインターフェースを介して仮想ネットワークのトラフィックを転送するノードでは、以下のように openvswitch を作成する必要があります：

```
# ovs-vsctl show
c61ba96f-fc11-4db9-9636-408e763f529e Bridge "ovsbr0"
    ポート "ovsbr0"
        インターフェース "ovsbr0" タイ
            ブ: 内部
    ポート "enp0s8"
        インターフェース "enp0s8"
```

## 設定

- 仮想マシンに公開される各ネットワークごとにOpenVSwitchを作成してください。すべてのノードで同じ名前を使用してください。
- 物理ネットワークインターフェースをopenvswitchに追加してください。

**注記：**この設定は永続化することをお勧めいたします。永続化の方法につきましては、お使いのシステムのネットワーク設定ガイドをご参照ください。

---

## 9.3 ブリッジドネットワーク

本ガイドでは、ブリッジドネットワークの展開方法について説明します。このモードでは、仮想マシンのトラフィックはノード内のLinuxブリッジを介して直接ブリッジされます。ブリッジドネットワークは、OpenNebulaによる追加のトラフィックフィルタリングに応じて、以下の4つの異なるモードで動作します：

- **ダミーブリッジド**：フィルタリングなし、ブリッジ設定なし（レガシーのノーオペレーションドライバ）。
- **ブリッジド**：フィルタリングなし、管理対象ブリッジ。
- **セキュリティグループ付きブリッジド**：iptablesルールをインストールし、セキュリティグループルールを実装します。
- **ebtables VLAN を使用したブリッジド**：上記に加え、各仮想ネットワークを分離（L2）するための追加の ebtables ルールを適用します。

### 9.3.1 考慮事項と制限事項

トラフィック分離に関して、以下の点をご考慮ください：

- **ダミーブリッジド、ブリッジド、およびセキュリティグループ付きブリッジド**の各モードでは、タグ付きネットワークインターフェースを追加することでネットワークの分離を実現できます。このモードにおける本番環境での推奨デプロイメント戦略となります。
- **ebtables を使用したブリッジド VLAN モード**は、VLAN を実装するための適切なハードウェアサポートがない小規模な環境を対象としています。このモードは /24 ネットワークに制限され、仮想ネットワーク間で IP アドレスが重複することはできません。このモードはテスト目的でのみ推奨されます。

### 9.3.2 OpenNebulaの設定

以下の設定属性は、`/var/lib/one/remotes/etc/vnm/OpenNebulaNetwork.conf` で調整可能です：

パラメータ	説明
<code>ipset_maxelem</code>	IP セット（セキュリティグループルールに使用）の最大エントリ数
<code>keep_empty_bridge</code>	仮想インターフェースが残っていないブリッジを保持するには、 <code>true</code> に設定してください。
<code>bridge_conf</code>	<code>brctl</code> の / リッシュオプション（非推奨、 <code>ip-route2</code> オプションに変換されます）
<code>ip_bridge_conf</code>	<code>ip-route2</code> の / リッシュオプション <code>ip link add &lt;bridge&gt; type bridge ...</code>

### 9.3.3 ブリッジドネットワークの定義

仮想ネットワークを作成するには、以下の情報を含めてください：

属性	値	必須
<b>VN_MAD</b>	<ul style="list-style-type: none"> <li>ダミー ダミーブリッジモード</li> <li>ブリッジモード用ブリッジ</li> <li>セキュリティグループ付きブリッジ接続用ファイアウォール</li> <li>ebtables によるブリッジモード (ebtables による分離機能付き)</li> </ul>	はい
<b>ブリッジ</b>	Linuxブリッジの名称 ノード	はい
<b>PHYDEV</b>	物理ネットワークデバイスの名称 ブリッジに接続される物理ネットワークデバイスの名称 (ダミードライバには適用されません)	いいえ

以下の例は、セキュリティグループモードを使用したブリッジネットワークを定義しています：

```
NAME      = "bridged_net"
VN_MAD   = "fw"
ブリッジ = vbr1
...
```

### 9.3.4 ebtables VLAN モード: デフォルトルール

このセクションでは、設定のデバッグが必要な場合に備え、作成される ebtables ルールを一覧表示します

```
# ネットワークのMACアドレスに一致しないパケットを破棄します
-s ! <mac_address>/ff:ff:ff:ff:ff:0 -o <tap_device> -j DROP# MACアドレスの偽装
を防止します
-s ! <mac_address> -i <tap_device> -j DROP
```

## 9.4 802.1Q VLAN ネットワーク

本ガイドでは、ホスト管理型VLANを通じて提供されるネットワーク分離機能を有効化する方法について説明します。このドライバーは、OpenNebula仮想ネットワークごとにブリッジを作成し、VLANタグ付きネットワークインターフェースをそのブリッジに接続します。この仕組みはIEEE 802.1Qに準拠しています。

VLAN ID は、特定のネットワーク内のすべてのインターフェースで共通となり、OpenNebula によって自動的に計算されます。また、仮想ネットワークテンプレートで VLAN\_ID パラメータを指定することで、強制的に設定することも可能です。

### 9.4.1 OpenNebulaの設定

VLAN\_ID は、oned.conf の以下の設定オプションに基づいて計算されます：

```
# VLAN_IDS: 自動VLAN_ID割り当て用のVLAN_IDプールです。このプールは802.1Qネットワーク (Open vSwitchおよび
9.4.802.1Q VLANネットワーク
802.1Qトライアル) 向けです。ドライバはまずVLAN_IDS[START] + VNET_IDの割り当てを試みます
# start: 最初に使用するVLAN_ID
```

```

# reserved: VLAN_ID または範囲をカンマ区切りで列挙します。コロンで区切られた 2 つの数値は範囲を示
# します。

VLAN_IDS = [
    START      = "2",
    予約済み   = "0, 1, 4095"
]

```

このパラメータを変更することで、一部のVLANを予約し、仮想ネットワークに割り当たられないようにすることができます。また、最初のVLAN\_IDを定義することも可能です。新しい分離ネットワークが作成されると、OpenNebulaはVLANプールから空きVLAN\_IDを検索します。このプールはグローバルであり、[Open vSwitch](#)ネットワークモードとも共有されています。

以下の設定属性は、`/var/lib/one/remotes/etc/vnm/OpenNebulaNetwork.conf` で調整可能です：

パラメータ	説明
<code>validate_vlan_id</code>	他のVLANがブリッジに接続されていないことを確認するには、 <code>true</code> に設定してください
<code>keep_empty_bridge</code>	<code>true</code> に設定すると、仮想インターフェイスが残っていないブリッジを保持します。
<code>bridge_conf</code>	<code>brctl</code> のパラメータオプション（非推奨、 <code>ip-route2</code> オプションに変換されます）
<code>ip_bridge_conf</code>	<code>ip-route2</code> のパラメータオプション <code>ip link add &lt;bridge&gt; type bridge ...</code>
<code>ip_link_conf</code>	<code>ip link add</code> に渡されるパラメータ引数

例:

```

# ブリッジ作成時に以下のオプションが追加されます。例：#
#       ip link add name <ブリッジ名> type bridge stp_state 1 #
# :ip_bridge_conf:
#   :stp_state: on

# これらのオプションは ip link add コマンドに追加されます。例：#
#       sudo ip link add lxcbr0.260 type vxlan id 260 group 239.0.101.4 \ #
#       16 dev lxcbr0 udp6zerochecksumrx tos 3                                ttl
# :ip_link_conf:
#   :udp6zerochecksumrx:
#   :tos: 3

```

#### 9.4.2 802.1Q ネットワークの設定

802.1Q ネットワークを作成するには、以下の情報を含めてください：

属性	値	必須
<b>VN_MAD</b>	802.1Q	はい
<b>PHYDEV</b>	物理ネットワークデバイスの名称 ブリッジに接続される物理ネットワークデバイスの名称です。	はい
<b>BRIDGE</b>	Linux ブリッジの名前は、デフォルトで onebr<net_id> または onebr.<vlan_id>	NO
<b>VLAN_ID</b>	VLAN ID。定義されていない場合、自動的に生成され、AUTO-MATIC_VLAN_ID が YES に設定されている場合	YES (ただし AUTO-MATIC_VLAN_ID)
<b>AUTO-MATIC_VLAN_ID</b>	必須であり、VLAN_ID が定義されていない場合、必須であり、必ず YES に設定してください	YES (VLAN_ID 以外の場合)
<b>MTU</b>	タグ付きインターフェイスおよびブリッジの MTU	NO

以下の例は、802.1Q ネットワークを定義しています

```
NAME      = "hmnet"
VN_MAD   = "802.1Q"
PHYDEV   = "eth0"
VLAN_ID  = 50          # オプションです。VLAN_IDを設定しない場合は、AUTOMATIC_VLAN_ID = "YES" を設定してください
BRIDGE   = "brhm"       # オプション
```

このシナリオでは、ドライバは brhm ブリッジの存在を確認します。存在しない場合は作成されます。 eth0 eth0.50 というタグが付けられ、 brhm に接続されます（既に接続されている場合は除きます）。

## 9.5 VXLAN ネットワーク

本ガイドでは、VXLAN カプセル化プロトコルを通じて提供されるネットワーク分離機能を有効化する方法について説明します。このドライバは、OpenNebula 仮想ネットワークごとにブリッジを作成し、VXLAN タグ付きネットワークインターフェースをそのブリッジに接続します。

特定のネットワーク内のすべてのインターフェースで VLAN ID は同一となり、OpenNebula によって自動的に計算されます。また、仮想ネットワークテンプレートで VLAN\_ID パラメータを指定することで強制的に設定することも可能です。

さらに、各 VLAN には、L2 ブロードキャストおよびマルチキャストトラフィックをカプセル化するためのマルチキャストアドレスが関連付けられています。このアドレスは、RFC 2365（管理スコープ IP マルチキャスト）で定義されている 239.0.0.0/8 範囲にデフォルトで割り当てられます。具体的には、マルチキャストアドレスは、239.0.0.0/8 のベースアドレスに VLAN\_ID を附加することで取得されます。

### 9.5.1 考慮事項と制限事項

本ドライバは、デフォルトの UDP サーバー ポート 8472 で動作します。

VXLAN トラフィックは物理デバイスに転送されます。このデバイスは（オプションで）VLAN タグ付きインターフェースとして設定できますが、その場合は、タグ付きインターフェースをすべてのホストで事前に手動で作成しておく必要があります。

物理デバイスとして機能する物理デバイスには、IP アドレスが必要です。

#### 同時接続可能な VXLAN ホスト数の上限

#### **OpenNebula 5.10 導入ガイド、リリース 5.10.5**

各VXLANは1つのマルチキャストグループに関連付けられます。物理ホストが同時にメンバーとなることができるマルチキャストグループの数には制限があります。これは、物理ホスト上で同時に使用できる異なるVXLANの数も意味します。デフォルト値は20であり、カーネルランタイムパラメータnet.ipv4.igmp\_max\_membershipsを介したsysctlで変更可能です。

例えば150に恒久的に変更する場合は、以下の設定を/etc/sysctl.confに記述してください：

```
net.ipv4.igmp_max_memberships=150
```

その後、設定を再読み込みしてください。

```
$ sudo sysctl -p
```

### 9.5.2 OpenNebula の設定

/etc/one/oned.conf を設定することで、開始 VLAN ID を指定することが可能です:

```
# VXLAN_IDS: 自動 VXLAN ネットワーク ID (VNI) 割り当て。これは # VXLAN ネットワークに使用されます
。
#
#      start: 最初に使用する VNI

VXLAN_IDS = [
    START = "2"
]
```

その 以下の 設定 属性 以下の調整 調整 設定ファイル  
 /var/lib/one/remotes/etc/vnm/ OpenNebulaNetwork.conf:

パラメータ	説明
vxlan_mc	各 VLAN の基本マルチキャストアドレスです。マルチキャストアドレスは vxlan_mc + vlan_id で構成されます。
vxlan_ttl	ルーティングされたマルチキャストネットワーク (IGMP) では、Time To Live (TTL) は1以上である必要があります
validate_vlan_id	他のVLANがブリッジに接続されていないことを確認するには、trueに設定してください
keep_empty_bridge	仮想インターフェースが残っていないブリッジを保持するには、true に設定してください。
bridge_conf	brctl のハッシュオプション (非推奨、ip-route2 オプションに変換されます)
ip_bridge_conf	ip-route2 のハッシュオプション ip link add <bridge> type bridge ...
ip_link_conf	ip link add に渡されるハッシュ引数

例:

```
# ブリッジ作成時に以下のオプションが追加されます。例: #
#      ip link add name <ブリッジ名> type bridge stp_state 1 #
# :ip_bridge_conf:
#   :stp_state: on

# これらのオプションは ip link add コマンドに追加されます。例: #
#      sudo ip link add lxcbr0.260 type vxlan id 260 group 239.0.101.4 \# TTL
16dev lxcbr0udp6zerocsumrx tos 3
#
:ip_link_conf:
  :udp6zerocsumrx:
  :tos: 3
```

### 9.5.3 VXLAN ネットワークの定義

VXLANネットワークを作成するには、以下の情報を含めてください：

属性	値	Manda- 必須
<b>VN_MAD</b>	vxlan	はい
<b>PHYDEV</b>	ブリッジに接続される物理ネットワークデバイスの名称です。	はい
<b>ブリッジ</b>	Linux ブリッジの名前。デフォルトは onebr<net_id> または onebr.<vlan_id> となります。	いいえ
<b>VLAN_ID</b>	VLAN ID。定義されていない場合は生成されます	いいえ
<b>AUTO-MATIC_VLAN_ID</b>	VLAN_ID が定義されていない場合、OpenNebula に IDを割り当てさせるために必須です	NO
<b>MTU</b>	タグ付きインターフェースおよびブリッジのMTU	NO

以下の例は、VXLAN ネットワークを定義します

```
NAME      =
"vxlan_net"VN_MAD =
"vxlan"PHYDEV = "eth0"
VLAN_ID = 50          # オプション
BRIDGE = "vxlan50" # オプション
...
```

このシナリオでは、ドライバはvxlan50ブリッジの存在を確認します。存在しない場合は作成されます。eth0はタグ付けされ（eth0.50）、vxlan50に接続されます（既に接続されている場合を除く）。OpenNebulaのVXLANトラフィックを分離したい場合、eth0を802.1Qタグ付きインターフェースにすることも可能です。

#### 9.5.4 BGP EVPN を使用した VXLAN の運用

デフォルトでは、VXLAN はマルチキャストに依存してトンネルエンドポイントを検出します。代わりに、制御プレーンに MP-BGP EVPN を使用することで、ネットワークのスケーラビリティを向上させることができます。このセクションでは、そのような設定を展開するための主な構成手順について説明します。

##### ハイパーバイザの設定

ハイパーバイザでは、FFRouting (FRR) のようなBGP EVPN対応のルーティングソフトウェアを実行する必要があります。その主な目的は、ホスト上で動作する各VXLANトンネルエンドポイント（すなわちVXLANネットワーク内の仮想マシンインターフェース）のMACアドレスとIPアドレス（任意）を含むBGP更新情報を送信することです。完全なルート到達性を実現するためには、この更新情報をクラウド内の他のすべてのハイパーバイザに配布する必要があります。この第二段階は通常、1つ以上のBGPルートリフレクターによって実行されます。

例として、2台のハイパーバイザ（10.4.4.11と10.4.4.12）と、10.4.4.13にあるルートリフレクターを考えます。ハイパーバイザ用のFRR設定ファイルは（すべてのVXLANネットワークをアナウンスする場合）、以下になります：

```
router bgp 7675
bgp router-id 10.4.4.11
no bgp default ipv4-unicastneighbor
10.4.4.13 remote-as 7675
neighbor 10.4.4.13 capability extended-nexthop address-
family l2vpn evpn
neighbor 10.4.4.13 activate
advertise-all-vni
exit-address-family exit
```

また、AS 7675 および 10.4.4.0/24 内のハイパーバイザ向けのレフクターは以下の通りです：

```

router bgp 7675
  bgp ルーターID 10.4.4.13
    BGPクラスタID 10.4.4.13
    no bgp default ipv4-unicast neighbor
    kvm_hostspeer-group neighbor
    kvm_hostsremote-as 7675
    neighbor kvm_hosts capability extended-nexthopneighbor
    kvm_hosts update-source 10.4.4.13
    BGP リスト範囲 10.4.4.0/24 ピアグループ kvm_hosts アドレスファミリー
    L2VPN EVPN
      neighbor fabric activate
      neighbor fabric route-reflector-client exit-
      address-family
終了

```

なお、これはすべてのVNIに同じ設定を使用するシンプルなシナリオです。ルーティングソフトウェアの設定が完了すると、各ハイパーバイザ上で動作する仮想マシン（VM）の設定更新が確認できるはずです。例えば：

```

10.4.4.11# show bgp evpn route
  ネットワーク          ネクストホップ          メトリック LocPrf ウェイト パス
ルート識別子: 10.4.4.11:2
* > [2]:[0]:[0]:[48]:[02:00:0a:03:03:c9]
          10.4.4.11                      32768 i
* > [3]:[0]:[32]:[10.4.4.11]
          10.4.4.11                      32768 i
ルート識別子: 10.4.4.12:2
* >i[2]:[0]:[0]:[48]:[02:00:0a:03:03:c8]
          10.4.4.12                      0       100        0 i
* >i[3]:[0]:[32]:[10.4.4.12]
          10.4.4.12                      0       100        0 i

```

## OpenNebulaの設定

/var/lib/one/remotes/vnm/OpenNebulaNetwork.conf ファイルを以下の内容に更新する必要があります：

1. BUMトラフィックの制御プレーンとしてBGP EVPNを設定し、vxlan\_modeを指定してください。
2. ハイパーバイザがトラフィックを送信するVTEPを選択します。これは、仮想ネットワークテンプレートで定義されたPHY\_DEVインターフェースを介してトラフィックを転送する「dev」、またはPHY\_DEVで設定された最初のIPを使用してトラフィックをルーティングする「local\_ip」のいずれかになります。
3. 最後に、VXLANリンクにnolearningオプションを追加することをお勧めいたします。

```

# マルチディステイニーションBUMトラフィック用のマルチキャストプロトコル。オプション: # -
multicast: IPマルチキャスト用

# - evpn: BGP EVPN制御プレーン用
:vxlan_mode: evpn

# トンネルエンドポイント通信タイプ。evpn vxlan_modeの場合のみ適用されます。 # - dev: トンネルエンドポイント間の通信は PHYDEV 宛てに送信されます
# - local_ip: PHYDEVの最初のIPアドレスを通信アドレスとして使用します
:vxlan_tep: local_ip

# 追加のIPリンクオプション。以下のコメントを外すとEVPNモードでの学習を無効化します
:ip_link_conf:
  :nolearning:

```

設定ファイルを更新した後は、変更内容を反映させるために必ず `onehost sync -f` を実行してください。

## 9.6 Open vSwitch ネットワーク

本ガイドでは、Open vSwitch ネットワークドライバの使用方法について説明します。これらは、ポートのタグ付けによる VLAN を使用したネットワーク分離と、OpenFlow を使用した基本的なネットワークフィルタリングを提供します。Open vSwitch を通じて設定可能なその他のトラフィック属性は変更されません。

特定のネットワーク内のすべてのインターフェースで VLAN ID は同一となり、OpenNebula によって自動的に計算されます。また、仮想ネットワークテンプレートで `VLAN_ID` パラメータを指定することで強制的に設定することも可能です。

**警告：**このドライバはセキュリティグループと互換性がありません。

### 9.6.1 OpenNebulaの設定

`VLAN_ID` は、`oned.conf` の以下の設定オプションに基づいて計算されます：

```
# VLAN_IDS: 自動VLAN_ID割り当て用のVLAN_IDプールです。このプールは802.1Qネットワーク (open vSwitchおよび802.1Q ドライバ) 向けです。ドライバはまずVLAN_IDS[START] + VNET_IDの割り当てを試みます
#       start: 使用する最初のVLAN_ID
#       予約: カンマ区切りのVLAN_IDまたは範囲のリスト。コロンで区切られた2つの数値は範囲を示します。範囲を
示します。
VLAN_IDS = [
    START      = "2",
    予約済み = "0, 1, 4095"
]
```

このパラメータを変更することで、一部のVLANを予約し、仮想ネットワークに割り当たられないようにすることができます。また、最初のVLAN\_IDを定義することも可能です。新しい分離ネットワークが作成されると、OpenNebulaはVLANプールから空きVLAN\_IDを検索します。このプールはグローバルであり、[802.1Q VLAN](#) ネットワークモードとも共有されています。

以下の設定属性は、`/var/lib/one/remotes/etc/vnm/OpenNebulaNetwork.conf` で調整可能です：

パラメータ	説明
<code>arp_cache_poisoni</code>	有効なARPキャッシュポイズニング防止ルール（仮想ネットワークのIP/MACスプーフィングフィルターが有効な場合のみ有効です） スプーフィング フィルターが有効な場合のみ有効です）。
<code>keep_empty_bridge</code>	true に設定すると、仮想インターフェースが残っていないブリッジを保持します。
<code>ovs_bridge_conf</code>	Open vSwitch ブリッジ作成時のノックオフオプション

**注記：**ファイルを全ノードに展開するには、`onehost sync` を実行することをお忘れなく。

### 9.6.2 Open vSwitch ネットワークの定義

---

Open vSwitch ネットワークを作成するには、以下の情報を含めてください：

属性	値	Manda- 必須
<b>VN_MAD</b>	ovswitch	はい
<b>PHYDEV</b>	ブリッジに接続される物理ネットワークデバイスの名称	いいえ
<b>ブリッジ</b>	使用する Open vSwitch ブリッジの名前	はい
<b>VLAN_ID</b>	VLAN ID。この属性が定義されていない場合、	NO
<b>AUTO-MATIC_VLAN_ID</b>	VLAN_ID が定義されている場合、この属性は無視されます。 OpenNebula に自動 VLAN ID を生成させたい場合、YES に設定してください。	NO

以下の例は、Open vSwitch ネットワークを定義します

```
NAME      = "ovswitch_net"
VN_MAD   = "ovswitch" BRIDGE =
vbr1
VLAN_ID  = 50 # オプション
...
```

### 複数の VLAN (VLAN トランкиング)

VLAN トランкиングは、VM テンプレートの `NIC` 要素または仮想ネットワーク テンプレートに以下のタグを追加することでサポートされます：

- `VLAN_TAGGED_ID`: タグ付けする VLAN の範囲を指定します。例: `1,10,30,32,100-200`。

### 9.6.3 VXLANネットワークにおけるOpen vSwitchの使用方法

本セクションでは、VXLANネットワークにおけるOpen vSwitchの使用方法について説明いたします。VXLANをご利用になるには、VXLAN ドライバーの機能を組み込んだ専用バージョンのOpen vSwitch ドライバーを使用する必要があります。これらの2つのドライバー、その設定オプション、利点、および欠点について理解しておくことが重要です。

VXLANオーバーレイネットワークは基盤として使用され、その上にOpen vSwitch（通常のLinuxブリッジではなく）が構築されます。最下層のトラフィックはVXLANカプセル化プロトコルによって分離され、Open vSwitchはカプセル化されたトラフィック内部で802.1Q VLANタグによる第2レベルの分離を依然として可能にします。主な分離機能は常にVXLANによって提供され、802.1Q VLANによって提供されるものではありません。VXLANを分離するために802.1Qが必要な場合、ユーザーが作成した802.1Qタグ付き物理インターフェースでドライバを設定する必要があります。

この階層構造を理解することは重要です。

#### OpenNebula の設定

このドライバに固有の設定は、上記および [VXLAN ガイド](#)で指定されているオプションを除き、ありません。

#### Open vSwitch - VXLAN ネットワークの定義

ネットワークを作成するには、以下の情報を含めてください：

属性	値	Manda- 必須
<b>VN_MAD</b>	ovswitch_vxlan	はい
<b>PHYDEV</b>	ブリッジに接続される物理ネットワークデバイスの名称です。	はい
<b>ブリッジ</b>	使用する Open vSwitch ブリッジの名前	いいえ
<b>OUTER_VLAN_ID</b>	外部 VXLAN ネットワーク ID。	いいえ
<b>オートマティック・アウタ -- • VLA MATIC_OUTER_VLAN</b>	OUTER_VLAN_ID が定義されている場合、この属性は無視されます。 OpenNebula に自動 ID の生成を望まない場合。	NO
<b>VLAN_ID</b>	内部の 802.1Q VLAN ID です。この属性が定義されていない場合、VLAN ID は生成されます。	いいえ
<b>AUTO- MATIC_VLAN_ID</b>	VLAN_ID が定義されている場合、この属性は無視されます。OpenNebula に自動 VLAN ID を生成させたい場合は YES に設定してください。 OpenNebula に自動 VLAN ID を生成させたい場合、YES に設定してください。	NO
<b>MTU</b>	VXLANインターフェースおよびブリッジのMTU	NO

以下の例は、Open vSwitch ネットワークを定義します

```
NAME      = "ovsvx_net" VN_MAD =
"ovswitch_vxlan" PHYDEV = eth0
BRIDGE   = ovsvxbr0.10000 OUTER_VLAN_ID = 10000
# VXLAN VNI
VLAN_ID = 50          # オプション
...
```

このシナリオでは、ドライバーはブリッジ ovsvxbr0.10000 の存在を確認します。存在しない場合、作成されます。また、VXLAN インターフェース eth0.10000 が作成され、Open vSwitch ブリッジ ovsvxbr0.10000 に接続されます。仮想マシンがインスタンス化されると、そのブリッジポートには802.1Q VLAN 50のタグが付けられます。

#### 9.6.4 DPDK を使用した Open vSwitch

本セクションでは、Open vSwitch ドライバと DPDK データパスを併用する方法について説明します。DPDK バックエンドを使用する場合、OpenNebula ドライバは自動的にブリッジとポートを設定します。

**警告：**このセクションは KVM ゲストにのみ関連します

##### 要件と制限事項

Open vSwitch 向けに DPDK データパスをご利用の際は、以下の点をご考慮ください：

- DPDK サポート付きでコンパイルされた Open vSwitch バージョンが必要です。
- このモードは非DPDKスイッチとの併用はできません。
- 仮想マシン (VM) のネットワークインターフェースカード (NIC) には virtio インターフェースを使用する必要があります。
- 動作には必須ではありませんが、ホストでの NUMA ピンニングおよび hugepages の設定に関心をお持ちかもしれません。詳細は [こちら](#)をご覧ください。

##### OpenNebulaの設定

OpenNebula を設定するには、以下の手順に従ってください：

- スイッチ用のDPDKバックエンドを選択してください。/etc/one/oned.conf 内のopenvswitch ドライバの設定を以下のように編集してください：

```
VN_MAD_CONF = [
    NAME = "ovswitch",
    BRIDGE_TYPE = "openvswitch_dpdk"
]
```

この変更を行った後は、OpenNebulaを再起動する必要があります。

- ブリッジのデータパスタイプを設定します。/var/lib/one/remotes/etc/OpenNebulaNetwork.conf 内のブリッジ設定オプションを編集してください：

```
:ovs_bridge_conf:
  :datapath_type: netdev
```

この変更を行った後、onehost sync コマンドを使用してホストとの変更内容を同期させる必要があります。なお、vhost インターフェースで使用されるソケットは VM ディレクトリ (/var/lib/one/datastores/<ds\_id>/<vm\_id>) 内に作成され、スイッチポート名に基づいて命名されます。  
<ds\_id>/<vm\_id> に作成され、スイッチポート名に基づいて命名されます。

### 仮想ネットワークにおける DPDK の使用

追加の変更は不要です。以下の手順に従ってください：

- ovswitch ドライバーを使用してネットワークを作成してください。詳細は上記をご参照ください。
- NICモデルがvirtioに設定されていることをご確認ください。この設定は、/etc/one/vmm\_exec/vmm\_exec\_kvm.conf にデフォルトとして追加できます。

ホスト上のドメイン定義を確認することで、仮想マシンがvhostインターフェースを使用していることを確認できます。以下のような記述が表示されるはずです：

```
<domain type='kvm' id='417'>
  <name>one-10</name>
  ...
  <devices>
    ...
    <インターフェース タイプ='vhostuser'>
      <mac address='02:00:c0:a8:7a:02' />
      <ソース タイプ='unix' パス='/var/lib/one//datastores/0/10/one-10-0' モード='サーバー' />
    </interface>
    <target dev='one-10-0' />
    <モデル タイプ='virtio' />
    <alias name='net0' />
    <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0' />
  </interface>
  ...
</domain>
```

また、qemu vhost インターフェースを使用したブリッジ内の関連ポートは以下の通りです：

```
ブリッジ br0
  ポート "one-10-0" インターフェース
    "one-10-0"
      type: dpdkvhostuserclient
    オプション: {vhost-server-path="/var/lib/one//datastores/0/10/one-10-0"}
```

## 参考文献

### 10.1 概要

本章では、OpenNebula が他のデータセンターコンポーネントと円滑かつ効率的に連携するための設定に関する参照情報、ならびにログファイルの保存場所、データベースツール、および OpenNebula コアデーモンの全設定パラメータに関する情報を扱います。

#### 10.1.1 本章の読み方

*oned.conf* ファイルは OpenNebula のメイン設定ファイルであり、OpenNebula のパフォーマンスや動作を調整する上で不可欠です。「大規模な導入環境」セクションには、OpenNebula のパフォーマンスを最適化するための有益なヒントがさらに記載されています。

ログファイルの使用方法、詳細レベルの調整、ログサブシステムの設定に関する完全なリファレンスについては、「[ログとデバッグ](#)」をお読みください。

データベースのメンテナンス作業には、[コマンドラインツール onedb](#) をご利用ください。OpenNebula データベースからの情報取得、アップグレード、不整合問題の修正などに使用できます。

#### 10.1.2 ハイパーバイザの互換性

本章の内容は、すべてのハイパーバイザに適用されます。

### 10.2 ONED 構成

OpenNebula のデーモン *oned* は、クラスタノード、仮想ネットワーク、仮想マシン、ユーザー、グループ、およびストレージデータストアを管理します。このデーモンの設定ファイルは *oned.conf* と呼ばれ、/etc/one ディレクトリ内に配置されます。このリファレンス文書では、*oned.conf* で指定可能なすべての形式とオプションについて説明します。

#### 10.2.1 デーモン設定属性

- **MANAGER\_TIMER:** コアが定期的な機能を評価するために使用する時間（秒単位）。

MONITORIN

G\_INTERVAL は MANAGER\_TIMER よりも小さい値に設定できません。

- **MONITORING\_INTERVAL\_HOST:** 各ホスト監視サイクル間の間隔（秒単位）。

- MONITORING\_INTERVAL\_VM: 各仮想マシン監視サイクル間の間隔（秒単位）。
- MONITORING\_INTERVAL\_DATASTORE: 各データストア監視サイクル間の間隔（秒単位）。

- MONITORING\_INTERVAL\_MARKET: マーケットプレイスの各監視サイクル間の間隔（秒単位）。
- MONITORING\_INTERVAL\_DB\_UPDATE: VM 監視情報の DB 書き込み間隔（秒単位）。-1 で DB 更新を無効化、0 で更新ごとに書き込みます。
- DS\_MONITOR\_VM\_DISK: VM ディスクを監視する MONITORING\_INTERVAL\_DATASTORE 間隔の数。無効にする場合は 0 。fs および fs\_lvm データストアにのみ適用されます。
- MONITORING\_THREADS: 監視メッセージの処理に使用するスレッドの最大数です。
- HOST\_PER\_INTERVAL: 各間隔で監視されるホストの数。
- HOST\_MONITORING\_EXPIRATION\_TIME: 監視情報の有効期限（秒単位）。HOST監視の記録を無効にする場合は0をご使用ください。
- VM\_INDIVIDUAL\_MONITORING: VM 監視情報は通常、ホスト情報とともに取得されます。VM 監視プロセスを個別に起動する必要があるカスタム監視ドライバを使用する場合は、これを「YES」に設定してください。
- VM\_PER\_INTERVAL: 各間隔で監視される仮想マシンの数。
- VM\_MONITORING\_EXPIRATION\_TIME: 監視情報の有効期限（秒単位）。0 を指定すると VM 監視の記録を無効にします。
- SCRIPTS\_REMOTE\_DIR: 監視および VM 管理スクリプトを保存するリモートパス。
- PORT: oned が XML-RPC 呼び出しを待機するポート番号です。
- LISTEN\_ADDRESS: XML-RPC 呼び出しをリッスンするホスト IP (デフォルト: すべての IP)。
- DB: データベースバックエンドの設定属性のベクトルです。
  - backend: sqlite または mysql に設定してください。詳細については、[MySQL 設定ガイド](#)をご覧ください。
  - server (MySQLのみ): MySQLサーバーのホスト名またはIPアドレス。
  - user (MySQLのみ) : MySQLユーザーのログインIDです。
  - passwd (MySQLのみ) : MySQLユーザーのパスワードです。
  - db\_name (MySQLのみ) : MySQLデータベース名。
  - connections (MySQLのみ) : MySQLサーバーへの最大接続数。
- VNC\_PORTS: 自動VNCポート割り当て用のVNCポートプールです。可能な場合、ポートはSTART + VMID に設定されます。詳細については、VM テンプレートのリファレンスをご参照ください:
  - start: 割り当て開始ポート番号
  - reserved: 予約済みポートまたは範囲をカンマ区切りで指定します。コロンで区切られた2つの数値は範囲を示します。
- VM\_SUBMIT\_ON\_HOLD: VMの作成を保留状態 (pending) ではなく保留状態 (hold) で強制します。値: YES または NO。
- API\_LIST\_ORDER: リストAPI呼び出し (例: onevm list) における要素の順序 (ID順) を設定します。値: ASC (昇順) またはDESC (降順) 。
- ログ: ログシステムの設定
  - SYSTEM: ファイル (デフォルト)、syslog、または std のいずれかを選択できます
  - DEBUG\_LEVEL: ログメッセージの詳細度を設定します。設定可能な値は以下の通りです:

DEBUG_LEVEL	意味
0	エラー
1	警告
2	情報
3	デバッグ

このセクションの例：

```
#***** # デーモン設定属性
#***** *****

ログ = [ シ
          = "ファイル",
スレム デバッグレベル = 3
]

#MANAGER_TIMER = 15

監視間隔ホスト = 180
監視間隔(ホスト) = 180
MONITORING_INTERVAL_DATASTORE = 300
監視間隔_マーケット = 600

MONITORING_THREADS =
50#DS_MONITOR_VM_DISK = 10
#HOST_PER_INTERVAL = 15
#ホスト監視有効期限 = 43200

#VM_INDIVIDUAL_MONITORING#VM_PER_INTERVAL = "no"
= 5
#VM_MONITORING_EXPIRATION_TIME = 14400

スクリプトリモートディレクトリ=/var/tmp/one ポート=2633

LISTEN_ADDRESS = "0.0.0.0"DB = [
BACKEND = "sqlite"]

# MySQL のサンプル設定 # DB = [ BACKEND =
"mysql",
#     サーバー = "localhost",
#     ポート = 0,
#     ユーザー名 = "oneadmin",
#     パスワード =
"opennebula", CONNECTIONS = 50
]

VNC_PORTS = [
    START = 5900
#    RESERVED = "6800, 6801, 9869"
]

#VM_SUBMIT_ON_HOLD =
"NO"#API_LIST_ORDER = "DESC"
```

```
... oned.conf.federation:
```

### 10.2.2 フェデレーション構成属性

onedのフェデレーション機能を制御します。フェデレーション環境での運用には、特別なデータベース設定が必要となります。

- FEDERATION: フェデレーション属性。
  - MODE: このonedの動作モード。
    - \* STANDALONE: フェデレーション未参加。これがデフォルトの動作モードです。
    - \* MASTER: このonedはフェデレーションのマスターゾーンです。
    - \* SLAVE: このonedはスレーブゾーンです。
- ZONE\_ID: onezoneコマンドで返されるゾーンIDです。
- MASTER\_ONED: マスター oned の XML-RPC エンドポイントです。例: <http://master.one.org:2633/> RPC2.

```
***** # フェデレーション構成属性
*****  

FEDERATION = [
    MODE = "STANDALONE",
    ZONE_ID = 0, MASTER_ONED =
    ""
]
```

### 10.2.3 Raft構成属性

OpenNebulaはRaftアルゴリズムを採用しております。設定ファイル/etc/one/oned.conf内の複数のパラメータにより調整が可能です。以下のオプションが利用可能です：

- LIMIT\_PURGE: 各ページ時に削除されるDBログレコードの数。
- LOG\_RETENTION: 保持するデータベースログレコードの数。サーバー間の同期ウィンドウと必要な追加ストレージ容量を決定します。
- LOG\_PURGE\_TIMEOUT: ログ保持期間値に基づき、適用済みレコードがページされる間隔（秒単位）。
- ELECTION\_TIMEOUT\_MS: リーダーからハートビートまたはログを受信しない場合の選挙プロセス開始までのタイムアウト（ミリ秒単位）。
- BROADCAST\_TIMEOUT\_MS: フォロワーへのハートビート送信間隔（ミリ秒単位）。
- XMLRPC\_TIMEOUT\_MS: Raft関連のAPI呼び出しにおけるタイムアウト時間（ミリ秒単位）。無制限のタイムアウトを設定する場合は、この値を0に設定してください。

```
RAFT = [
    LIMIT_PURGE = 100000,
    LOG_RETENTION = 500000,
    LOG_PURGE_TIMEOUT = 600,
    ELECTION_TIMEOUT_MS = 2500,
    ブロードキャストタイムアウトミリ秒 = 500,
```

```

XMLRPC_TIMEOUT_MS      = 450
]

```

### 10.2.4 デフォルトのショーバックコスト

以下の属性は、CPU、メモリ、ディスクのコストが設定されていない仮想マシンに対するデフォルトのコストを定義します。これは oneshowback の計算メソッドで使用されます。

```

***** # デフォルトのショーバック
コスト *****

DEFAULT_COST = [ CPU_COST
                  = 0,
    MEMORY_COST = 0,
    ディスクコスト = 0
]

```

### 10.2.5 XML-RPC サーバー設定

- MAX\_CONN: サーバーが維持する同時TCP接続の最大数
- MAX\_CONN\_BACKLOG: サーバーがオペレーティングシステムから受け入れることなく、オペレーティングシステムがサーバーに代わって受け入れる TCP 接続の最大数
- KEEPALIVE\_TIMEOUT: RPC 間の接続がオープン状態を維持できる最大時間（秒単位）
- KEEPALIVE\_MAX\_CONN: サーバーが单一接続上で実行するRPCの最大数
- TIMEOUT: RPC処理中にサーバーがクライアントの応答を待機する最大時間（秒単位）。このタイムアウトは、フェデレーション環境でプロキシがマスターに呼び出しを行う場合にも適用されます。
- RPC\_LOG: XML-RPCリクエスト用の別個のログファイルを、/var/log/one/one\_xmlrpc.log に作成します。
- MESSAGE\_SIZE: XML-RPC応答用のバッファサイズ（バイト単位）。
- LOG\_CALL\_FORMAT: XML-RPC呼び出しをログに記録するためのフォーマット文字列。解釈される文字列:
  - %i – リクエスト ID
  - %m – メソッド名
  - %u – ユーザー ID
  - %U – ユーザー名
  - %l[number] – パラメータリストと各パラメータの出力文字数（オプション）、デフォルトは20文字です。  
例: %l300
  - %p – ユーザーパスワード
  - %g – グループ ID
  - %G – グループ名
  - %a – 認証トークン
  - %% – %

```
***** # XML-RPC サーバー設定
*****
#MAX_CONN          = 15
#MAX_CONN_BACKLOG = 15
#キープアライブタイムアウト = 15
#キープアライブ最大接続数 = 30
#タイムアウト        = 15
#RPC_LOG            = NO
#メッセージサイズ     = 1073741824
#ログ呼び出し形式   = "要求: %i UID:%u %m が %l を呼び出しました"
"
```

**警告:** この機能は、xmlrpc-c ライブラリ 1.32 以上でコンパイルした場合にのみ利用可能です。現在、OpenNebula によって配布されているパッケージのみがこのライブラリとリンクされています。

## 10.2.6 仮想ネットワーク

- **NETWORK\_SIZE:** ここでは仮想ネットワークのデフォルトサイズを定義できます
  - **MAC\_PREFIX:** 自動生成されるMACアドレスの作成に使用されるデフォルトのMACプレフィックスです。（これは仮想ネットワークテンプレートによって上書きされる場合があります。）
  - **VLAN\_IDS:** 自動VLAN\_ID割り当て用のVLAN IDプールです。このプールは802.1Qネットワーク（Open vSwitchおよび802.1Q ドライバ）向けです。ドライバはまずVLAN\_IDS[START] + VNET\_IDの割り当てを試みます
    - **start:** 使用する最初のVLAN\_ID
    - **reserved:** カンマ区切りのVLAN\_IDまたは範囲のリストです。コロンで区切られた2つの数値は範囲を示します。
  - **VXLAN\_IDS:** 自動VXLANネットワークID（VNI）割り当て。これはVXLANネットワークに使用されます。
    - **start:** 最初に使用するVNI
    - **reserved:** カンマ区切りのVXLAN\_IDまたは範囲のリストです。コロンで区切られた2つの数値は範囲を示します。
- 
- 注記:** このプールでは予約済みIDはサポートされていません

設定例：

```
***** # 物理ネットワークの設定
*****
NETWORK_SIZE = 254

MAC_PREFIX     =

"02:00"VLAN_IDS = [
    START      = "2",
    予約済み   = "0, 1, 4095"
]

VXLAN_IDS = [
    START = "2"
]
```

## 10.2.7 データストア

ストレージサブシステムでは、仮想マシンで使用するイメージ（オペレーティングシステムやデータなど）を簡単に設定できます。これらのイメージは複数の仮想マシンで同時に使用でき、他のユーザーと共有することも可能です。

ここでは、データストアおよびイメージテンプレートのデフォルト値を設定できます。テンプレートの構文に関する詳細情報は、こちらをご覧ください。

- **DATASTORE\_LOCATION:** データストアのパス。すべてのホストおよびフロントエンドで共通です。デフォルトは `/var/lib/one/datastores` です（自己完結モードの場合は `$ONE_LOCATION/var/datastores` がデフォルトとなります）。各データストアには、以下の形式の専用ディレクトリ（`BASE_PATH` と呼ばれる）が割り当てられます：  
`$DATASTORE_LOCATION/<datastore_id>` 必要に応じて、このディレクトリを任意のパスへシンボリックリンクできます。  
。 `BASE_PATH` は `oned` の起動時に毎回この属性から生成されます。
- **DATASTORE\_CAPACITY\_CHECK:** 新規イメージ作成前に十分な容量があるか確認します。デフォルトは Yes です。
- **DEFAULT\_IMAGE\_TYPE:** テンプレート内で TYPE フィールドが省略された場合のデフォルト値です。以下の値が有効です：
  - OS: オペレーティングシステムを格納したイメージファイル
  - CDROM: CDROM を保持するイメージファイル
  - DATABLOCK: 空のブロックとして作成されたデータブロックを保持するイメージファイル
- **DEFAULT\_DEVICE\_PREFIX:** テンプレート内で省略された場合、DEV\_PREFIX フィールドのデフォルト値です。欠落している DEV\_PREFIX 属性はイメージ作成時に補完されるため、このプレフィックスを変更しても既存のイメージには影響しません。設定可能な値は以下の通りです：

プレフィックス	デバイス種別
HD	IDE
sd	SCSI
vd	KVM仮想ディスク

- **DEFAULT\_CDROM\_DEVICE\_PREFIX:** 上記と同様ですが、CDROM デバイス用です。
- **DEFAULT\_IMAGE\_PERSISTENT:** イメージのクローン作成または保存 (`oneimage clone`、`onevm disk-saveas`) における PERSISTENT 属性のデフォルト値を制御します。省略された場合、イメージはベースイメージから PERSISTENT 属性を継承します。
- **DEFAULT\_IMAGE\_PERSISTENT\_NEW:** イメージ作成時 (`oneimage create`) の PERSISTENT 属性のデフォルト値を制御します。設定されていない場合、デフォルトではイメージは永続化されません。

イメージリポジトリに関する詳細は、「仮想マシンイメージの管理」ガイドをご参照ください。設定例：

## OpenNebula 5.10 導入ガイド、リリース 5.10.5

```
#***** # イメージリポジトリ設定
#***** #***** #***** #***** #***** #***** #***** #***** #***** #
#DATASTORE_LOCATION = /var/lib/one/datastores

DATASTORE_CAPACITY_CHECK = "yes"

DEFAULT_IMAGE_TYPE = "OS"
DEFAULT_DEVICE_PREFIX = "hd"

デフォルトCDROMデバイスプレフィックス = "hd"
```

```
#デフォルトイメージの永続性 = ""
#デフォルトイメージ永続化新規 = "NO"
```

### 10.2.8 情報収集モジュール

このドライバはホストに割り当てることができません。KVM ドライバと組み合わせてご利用ください。設定可能なオプションは以下の通りです：

- **-a:** collectd ソケットをバインドするアドレス（デフォルト 0.0.0.0）
- **-p:** 監視情報を待機する UDP ポート（デフォルト 4124）
- **-f:** 収集した情報をフラッシュする間隔（秒単位）（デフォルト値：5）
- **-t:** サーバーのスレッド数（デフォルト値 50）
- **-i:** 監視プッシュサイクルの時間間隔（秒単位）。このパラメータは MONITOR-ING\_INTERVAL よりも小さく設定する必要があります。そうでない場合、プッシュ監視は効果を発揮しません。

設定例：

```
IM_MAD = [
    name      = "collectd",
    実行可能ファイル = "collectd",
    引数 = "-p 4124 -f 5 -t 50 -i 20" ]
```

### 10.2.9 情報ドライバー

情報ドライバーはクラスターノードから情報を収集するために使用され、ご利用の仮想化環境によって異なります。複数の情報マネージャーを定義できますが、それぞれ異なる名前を付ける必要があります。定義するには、以下の設定が必要です：

- **name:** この情報収集ドライバの名前。
- **実行可能ファイル:** 情報収集ドライバーの実行可能ファイルのパスを、絶対パスまたは /usr/lib/one/mads/ からの相対パスで指定します。
- **引数:** ドライバー実行ファイル用の引数。通常はプローブ設定ファイルの絶対パス、または /etc/one/ からの相対パスで指定します。

情報および監視システムの設定に関する詳細情報と拡張のヒントについては、情報ドライバ設定ガイドをご参照ください。

設定例：

```
#-----
# KVM UDP プッシュ情報ドライバー管理者の設定 # -r ホスト監視時の再試行回数
#      -t スレッド数、すなわち同時に監視するホストの数
#-----
IM_MAD = [
    NAME      =
    "kvm", SUNSTONE_NAME = "KVM",
    実行可能ファイル = "one_im_ssh", 引数
    = "-r 3 -t 15 kvm" ]
#-----
```

### 10.2.10 仮想化ドライバー

仮想化ドライバーは、ホスト上で仮想マシンを作成、制御、監視するために使用されます。複数の仮想化ドライバーを定義することも可能です（例：複数のホストで異なる仮想化ソフトウェアを使用している場合）。ただし、各ドライバーには異なる名前を付ける必要があります。定義するには、以下の設定が必要です：

- **NAME:** 仮想化ドライバの名前
- **SUNSTONE\_NAME:** Sunstone に表示される名称
- **実行ファイル:** 仮想化ドライバーの実行ファイルのパス（絶対パス、または `/usr/lib/ one/mads/` からの相対パス）
- **引数:** ドライバー実行ファイル用の引数
- **TYPE:** ドライバーの種類。サポートされているドライバー: xen、kvm、または xml
- **DEFAULT:** ドライバーのデフォルト値および設定パラメータを含むファイルの絶対パス、または `/etc/one/` からの相対パス
- **スナップショット保持:** ハイパーバイザが対応している場合、電源オン/オフサイクルおよびライブマイグレーション時にスナップショットを削除しない
- **IMPORTED\_VMS\_ACTIONS:** インポートされた仮想マシンでサポートされているアクションのカンマ区切りリストです。  
利用可能なアクションは以下の通りです：
  - 移行
  - ライブ移行
  - terminate
  - 強制終了
  - アンデプロイ
  - ハードアンデプロイ
  - 保留
  - リリース
  - 停止
  - 一時停止
  - 再開
  - 削除
  - 削除・再作成
  - 再起動
  - ハード再起動
  - 再スケジュール
  - 再スケジュール解除

- 
- 電源オフ
  - ハード電源オフ
  - ディスク接続
  - ディスク切断
  - NIC接続

- NICの取り外し
- スナップ作成
- スナップ削除

仮想マシンマネージャードライバーの設定および構成に関する詳細情報は、該当するセクションをご確認ください：

- *KVM ドライバー*
- *vCenter ドライバー*

サンプル設定：

```
#-----
# 仮想化ドライバーの設定
#-----

VM_MAD = [
    NAME          = "kvm",
    SUNSTONE_NAME = "KVM",
    実行可能ファイル = "one_vmm_exec",
    ARGUMENTS     = "-t 15 -r 0 kvm",
    デフォルト      = "vmm_exec/vmm_exec_kvm.conf", タイプ
                      = "KVM",
    スナップショットの保持 = "いいえ",
    インポートされたVMSアクション = "終了, 強制終了, 保留, 解放, 一時停止, 再開, 削除, 再起動, 強制再起動, スケ
                                ジュール変更, スケジュール解除, ディスク接続, ディスク切断, NIC接続, NIC切断, スナップショット作成, ス
                                ナップショット削除"
]
```

## 10.2.11 転送ドライバ

転送ドライバーは、仮想マシンイメージの転送、複製、削除、作成に使用されます。デフォルトのTM\_MADドライバーには、サポートされているすべてのストレージモード用のプラグインが含まれています。カスタムプラグインを追加するには、TM\_MADを修正する必要がある場合があります。

- **実行ファイル:** 転送ドライバーの実行ファイルのパスを、絶対パスまたは /usr/lib/one/mads/ からの相対パスで指定してください。
- **引数:** ドライバ実行ファイルに対する引数:
  - **-t:** スレッド数、すなわち同時に実行される転送の数
  - **-d:** 転送ドライバのリスト（カンマ区切り）。指定しない場合、利用可能な全ドライバが有効となります各種ストレージ代替手段の設定に関する詳細は、[ストレージ設定ガイド](#)をご参照ください。設定例：

```
#-----
# 転送マネージャードライバ設定
#-----

TM_MAD = [
    EXECUTABLE = "one_tm",
    引数 = "-t 15 -d dummy,lvm,shared,fs_lvm,qcow2,ssh,ceph,dev,vcenter,iscsi_
              libvirt"
]
```

各ドライバの設定は、TM\_MAD\_CONF セクションで定義されます。これらの値は、新しいデータストアを作成する際に使用され、データストアの動作を定義するため、変更しないでください。

- **NAME:** 転送ドライバの名前。TM\_MAD セクションの -d オプションにリストされます。
- **LN\_TARGET:** 新しい仮想マシンがインスタンス化される際に、永続イメージがどのように複製されるかを決定します:
  - **NONE:** イメージはリンクされ、追加のストレージ容量は使用されません
  - **SELF:** イメージは Images データストアにクローンされます
  - **SYSTEM:** イメージは System データストアにクローンされます
- **CLONE\_TARGET:** 新しい仮想マシンがインスタンス化される際に、非永続イメージがどのように複製されるかを決定します:
  - **NONE:** 画像はリンクされ、追加のストレージ容量は使用されません
  - **SELF:** イメージは Images データストアにクローンされます
  - **SYSTEM:** イメージはシステムデータストアにクローンされます
- **SHARED:** システムデータストアを保持するストレージが、異なるホスト間で共有されるかどうかを決定します。有効な値: yes または no。
- **DS\_MIGRATE:** このTMでシステムデータストアの移行を許可する場合はYESに設定します。システムデータストアのTMでのみ有効です。
- **ALLOW\_ORPHANS:** 親のないスナップショットを許可するかどうか。3つの値 (YES、NO、MIXED) が設定可能です。最後のモードであるMIXEDでは、孤立スナップショットの作成を許可しますが、Cephデータストアでのスナップショット復元操作後に発生する可能性のある依存関係を考慮します。

設定例：

```
TM_MAD_CONF = [
    name          = "lvm",
    ln_target     = "NONE",
    clone_target  = "SELF", shared
                  = "はい", 孤立オブジェクトを許可 = "いいえ"
]

TM_MAD_CONF = [
    name          = "共有",
    ln_target     = "NONE",
    clone_target  = "SYSTEM",
    shared        = "はい",
    ds_migrate   = "はい"
]
```

## 10.2.12 データストア・ドライバー

データストアドライバは、ストレージバックエンドを管理するための一連のスクリプトを定義します。

- **実行ファイル:** 転送ドライバの実行ファイルのパスを、絶対パスまたは /usr/lib/one/mads/ からの相対パスで指定してください。

- 
- 引数: ドライバ実行ファイル用
    - **-t** スレッド数、すなわち同時リポジトリ操作の数
    - **-d** データストア MADs (カンマ区切り)
    - **-s** システムデータストアのTMドライバー。共有システムデータストアの監視に使用されます

設定例：

```
DATASTORE_MAD = [
    EXECUTABLE = "one_datastore",
    ARGUMENTS = "-t 15 -d dummy,fs,lvm,ceph,dev,iscsi_libvirt,vcenter -s shared,ssh,
    ↳ceph,fs_lvm"
]
```

このドライバーの詳細およびカスタマイズ方法については、[リファレンスガイド](#)をご覧ください。

### 10.2.13 マーケットプレイスドライバーの設定

各種マーケットプレイスを管理するためのドライバー（ストレージバックエンドに特化）

- **実行ファイル:** 転送ドライバーの実行ファイルのパスを、絶対パスまたは `/usr/lib/one/mads/` からの相対パスで指定してください。
- **引数:** ドライバー実行ファイル用:
  - `-t` スレッド数、すなわち同時リポジトリ操作の数
  - `-m` マーケットプレイス名（カンマ区切り）
  - `--proxy` インターネットアクセスに必要なプロキシURI。例: `--proxy http://1.2.3.4:5678`
  - `-w` 外部コマンドを実行する秒単位のタイムアウト（デフォルトは無制限）設定

例：

```
MARKET_MAD = [
    実行可能ファイル = "one_market", 引数 = "-t 15 -m
    http,s3,one"
]
```

### 10.2.14 フックシステム

OpenNebulaにおけるフックとは、仮想マシン（VM）またはホストの状態変化によって実行がトリガーされるプログラム（通常はスクリプト）です。フックは、VMまたはホストが稼働しているノードに対して、ローカルまたはリモートで実行することが可能です。フックシステムを設定するには、OpenNebulaの設定ファイルで以下の項目を設定する必要があります：

- **EXECUTABLE:** フックドライバ実行ファイルのパスを絶対パス、または `/usr/lib/one/mads/` からの相対パスで指定します。
- **引数:** ドライバー実行ファイル用の引数。絶対パス、または `/etc/one/` からの相対パスで指定します。

設定例：

```
HM_MAD = [
    実行可能ファイル = "one_hm" ]
```

仮想マシンフック (VM\_HOOK) は以下によって定義されます：

- **NAME:** フックの名前。フックの追跡に便利です（オプション）。
- **ON:** フックを実行すべきタイミング：

---

- **CREATE:** 仮想マシンが作成された時 (`onevm create`)

- **PROLOG:** VM がプロログ状態にあるとき

- **RUNNING**: VM が正常に起動した後
- **UNKNOWN**: VM が不明状態にあるとき
- **シャットダウン後**: 仮想マシンがシャットダウンされた後
- **停止後**: 仮想マシンが停止した後（仮想マシンイメージの転送を含む）
- **完了**: VM が削除またはシャットダウンされた後
- **カスタム**: フックをトリガーする状態として、ユーザーが定義した特定の STATE と LCM\_STATE の組み合わせ
- **コマンド**: 絶対パス、または /usr/share/one/hooks からの相対パスとして指定
- **引数**: フック用の引数です。仮想マシン情報を以下で置換できます:
  - **\$ID**: 仮想マシンのID
  - **\$TEMPLATE**: ベース64エンコードされたXML形式のVMテンプレート
  - **PREV\_STATE**: 仮想マシンの前回の状態
  - **PREV\_LCM\_STATE**: 仮想マシンの前回のLCM状態
- **REMOTE**: 値:
  - **YES**: フックは仮想マシンが割り当てられたホスト上で実行されます
  - **NO**: フックは OpenNebula サーバーで実行されます（デフォルト）

**ホストフック (HOST\_HOOK)** は以下で定義されます:

- **NAME**: フックの識別名（追跡に有用、オプション）
- **ON**: フックを実行すべきタイミング
  - ホストの作成時 (onehost create)
  - **ERROR**: ホストがエラー状態に入ったとき
  - **DISABLE**: ホストが無効化されたとき
- **COMMAND**: 絶対パス、または /usr/share/one/hooks からの相対パスとして
- **引数**: フック用の引数です。以下のホスト情報を使用できます:
  - **\$ID**: ホストのID
  - **\$TEMPLATE**: ホストテンプレート (base64エンコードされたXML形式)
- **REMOTE**: 以下の値
  - **YES**: フックはホスト上で実行されます
  - **いいえ**: フックはOpenNebulaサーバー上で実行されます（デフォルト）

ト) 設定例：

```
VM_HOOK = [
    name      = "advanced_hook",
    on        = "CUSTOM",
    状態      = "ACTIVE", LCM状態 =
    "BOOT_UNKNOWN", コマンド = "log.rb",
    引数      = "$ID $PREV_STATE $PREV_LCM_STATE" ]
```

## 10.2.15 認証マネージャーの設定

- **AUTH\_MAD:** OpenNebulaのリクエストを認証および認可するために使用されるドライバーです。定義されていない場合、OpenNebulaは組み込みの認証ポリシーを使用します。
  - **EXECUTABLE:** 認証ドライバーの実行ファイルのパスを、絶対パスまたは /usr/lib/one/mads/ からの相対パスで指定します。
  - **AUTHN:** 認証モジュールのリストをコンマで区切って指定します。定義されていない場合、利用可能なすべてのモジュールが有効になります。
  - **AUTHZ:** 権限付与モジュールのリスト（カンマ区切り）。
- **SESSION\_EXPIRATION\_TIME:** 認証済みトークンの有効期限（秒単位）。この期間中はドライバーが使用されません。セッションキヤッショを無効にするには0を指定してください。
- **ENABLE\_OTHER\_PERMISSIONS:** 「その他」の権限を有効化するかどうか。oneadminグループのユーザーは引き続きこれらの権限を変更できます。値: YES または NO。
- **DEFAULT\_UMASK:** Unix の umask に類似しています。デフォルトのリソース権限を設定します。その形式は必ず 3 衝の 8 進数でなければなりません。例えば、umask が 137 の場合、新規オブジェクトの権限は 640 (u- u-- ---) に設定されます。

設定例：

```
AUTH_MAD = [
    executable = "one_auth_mad",
    authn = "ssh,x509,ldap,server_cipher,server_x509"
]

SESSION_EXPIRATION_TIME = 900

#ENABLE_OTHER_PERMISSIONS = "YES"

DEFAULT_UMASK = 177
```

DEFAULT\_AUTH は、希望するデフォルト認証ドライバ（例：ldap）を指定するために使用できます。

```
DEFAULT_AUTH = "ldap"
```

## 10.2.16 VM操作の権限

以下のパラメータは、ADMIN、MANAGE、USE 権限に関連する操作を定義します。一部の仮想マシン操作では、他のオブジェクトに対する追加の権限が必要となる場合があります。また、一部の操作は一連のアクションを指します：

- disk-snapshot: 作成、削除、復元アクションを含みます
- disk-attach: 接続および切断アクションを含みます
- nic-attach: 接続および切断アクションを含みます
- snapshot: 作成、削除、および復元アクションを含みます
- resched: スケジュール変更およびスケジュール解除アクションを含みます

---

リストおよび表示操作にはUSE権限が必要となります。これは設定変更できません。

以下の例では、VM に対して移行、削除、復元などの操作を行うには ADMIN 権限が必要となります。一方、アンデプロイ、保留などの操作には MANAGE 権限が必要となります：

```

VM_ADMIN_OPERATIONS = "移行、削除、復旧、再試行、デプロイ、再スケジュール"

VM_MANAGE_OPERATIONS = "アンデプロイ、保留、解放、停止、一時停止、再開、再起動、電源オフ、ディスク接続、NIC接続、デ
    ィスクスナップショット、終了、ディスクサイズ変更、スナップショット、構成更新、名前変更、サイズ変更、更新、ディスク
    保存"

VM_USE_OPERATIONS      = ""

```

## 10.2.17 制限付き属性の設定

oneadmin グループに属さないユーザーは、oneadmin グループに属さないユーザーが作成したテンプレートで、以下の属性によって制限されているものをインスタンス化することはできません：

- **VM\_RESTRICTED\_ATTR**: oneadmin グループ外のユーザーに対して制限する仮想マシン属性
- **IMAGE\_RESTRICTED\_ATTR**: oneadmin グループ外のユーザーに対して制限するイメージ属性
- **VNET\_RESTRICTED\_ATTR**: 予約を更新する際、oneadmin グループ外のユーザーに対して制限する仮想ネットワーク属性。  
通常の仮想ネットワーク作成時にはこれらの属性は考慮されません。

VMテンプレートがoneadminグループの管理者によって作成された場合、oneadminグループ外のユーザーは

これらのテンプレートをイン

スタンス化できます。

設定例：

```

VM_RESTRICTED_ATTR = "CONTEXT/FILES" VM_RESTRICTED_ATTR =
"NIC/MAC" VM_RESTRICTED_ATTR =
"NIC/VLAN_ID" VM_RESTRICTED_ATTR =
"NIC/BRIDGE" VM_RESTRICTED_ATTR = "NIC_DEFAULT/MAC"
VM_RESTRICTED_ATTR =
"NIC_DEFAULT/VLAN_ID" VM_RESTRICTED_ATTR =
"NIC_DEFAULT/BRIDGE" VM_RESTRICTED_ATTR =
"DISK/TOTAL_BYTES_SEC" VM_RESTRICTED_ATTR =
"DISK/READ_BYTES_SEC" VM_RESTRICTED_ATTR = "ディスク/書き
込みバイト/秒" VM_RESTRICTED_ATTR = "ディスク/総I/O操作数/秒"
"VM_RESTRICTED_ATTR = "ディスク/読み取りI/O操作数/秒"
"VM_RESTRICTED_ATTR = "ディスク/書き込みI/O操作数/秒"
#VM_RESTRICTED_ATTR = "ディスク容量" VM_RESTRICTED_ATTR =
"ディスクの元の容量" VM_RESTRICTED_ATTR = "CPUコスト"
VM_RESTRICTED_ATTR = "メモリコスト" VM_RESTRICTED_ATTR =
"ディスクコスト" VM_RESTRICTED_ATTR =
"PCI" VM_RESTRICTED_ATTR = "ユーザー入力"

#VM_RESTRICTED_ATTR = "ランク" #VM_RESTRICTED_ATTR = "スケ
ジューリングランク" #VM_RESTRICTED_ATTR = "要件"
#VM_RESTRICTED_ATTR = "スケジューリング要件"

IMAGE_RESTRICTED_ATTR = "SOURCE"

VNET_RESTRICTED_ATTR =
"VN_MAD" VNET_RESTRICTED_ATTR =
"PHYDEV" VNET_RESTRICTED_ATTR =
"VLAN_ID" VNET_RESTRICTED_ATTR = "BRIDGE"

```

```
VNET_RESTRICTED_ATTR = "AR/VN_MAD" VNET_RESTRICTED_ATTR =
"AR/PHYDEV" VNET_RESTRICTED_ATTR = "AR/VLAN_ID"
VNET_RESTRICTED_ATTR = "AR/BRIDGE"
```

OpenNebula は以下の属性を評価します：

- VMテンプレートのインスタンス化時 (`onetemplate instantiate`)
- 仮想マシン作成時 (`onevm create`)
- 仮想マシンへのネットワークインターフェースカード接続時 (`onevm nic-attach`)、例えばNIC/MACの使用を防止する場合

### 10.2.18 暗号化属性設定

これらの属性は OpenNebula コアによって暗号化および復号化されます。サポートされている属性は以下の通りです：

- CLUSTER\_ENCRYPTED\_ATTR**
- DATASTORE\_ENCRYPTED\_ATTR**
- HOST\_ENCRYPTED\_ATTR**
- VM\_ENCRYPTED\_ATTR**: これらの属性はユーザーTemplateにも適用されます。
- VNET\_ENCRYPTED\_ATTR**: これらの属性は、仮想ネットワークに属するアドレス範囲にも適用されます。

設定例：

```
CLUSTER_ENCRYPTED_ATTR = "PROVISION/PACKET_TOKEN"

DATASTORE_ENCRYPTED_ATTR = "PROVISION/PACKET_TOKEN"

HOST_ENCRYPTED_ATTR = "EC2_ACCESS"
HOST_ENCRYPTED_ATTR = "EC2_SECRET"
HOST_ENCRYPTED_ATTR = "AZ_ID" HOST_ENCRYPTED_ATTR =
"AZ_CERT" HOST_ENCRYPTED_ATTR = "VCENTER_PASSWORD"
HOST_ENCRYPTED_ATTR = "NSX_PASSWORD"
HOST_ENCRYPTED_ATTR = "ONE_PASSWORD"

VM_ENCRYPTED_ATTR = "ワンパスワード" VM_ENCRYPTED_ATTR = "コンテキス
ト/管理者パスワード"

VNET_ENCRYPTED_ATTR = "PROVISION/PACKET_TOKEN" VNET_ENCRYPTED_ATTR =
"PROVISION/PACKET_PROJECT"

VNET_ENCRYPTED_ATTR = "AR/PROVISION/PACKET_PROJECT"
```

OpenNebula は以下の属性を暗号化します：

- オブジェクト作成時 (`onecluster/onedatastore/onehost/onevm/onevnet` の作成時)
- オブジェクト更新時 (`onecluster/onedatastore/onehost/onevm/onevnet update`)

属性を復号するには、パラメータとして `true` を指定した `info` API メソッドをご利用ください。`onevm show`、`onehost show`、`onevnet show` の各コマンドでは、`--decrypt` オプションを使用して属性を復号することができます。

## 10.2.19 繙承属性の設定

以下の属性は、リソーステンプレートからインスタンス化された VM にコピーされます。複数の属性を定義することが可能です。

- `INHERIT_IMAGE_ATTR`: イメージテンプレートから各 VM/ディスクにコピーされる属性。
- `INHERIT_DATASTORE_ATTR`: データストアテンプレートから各 VM/ディスクにコピーされる属性。
- `INHERIT_VNET_ATTR`: ネットワークテンプレートから各 VM/NIC にコピーされる属性。設定例:

```
#INHERIT_IMAGE_ATTR = "EXAMPLE"
#INHERIT_IMAGE_ATTR = "SECOND_EXAMPLE"
#INHERIT_DATASTORE_ATTR = "COLOR" #INHERIT_VNET_ATTR =
"帯域幅制御"

INHERIT_DATASTORE_ATTR = "CEPH_HOST" INHERIT_DATASTORE_ATTR =
"CEPH_SECRET" INHERIT_DATASTORE_ATTR = "CEPH_USER"
INHERIT_DATASTORE_ATTR = "CEPH_CONF" INHERIT_DATASTORE_ATTR =
"POOL_NAME"

INHERIT_DATASTORE_ATTR = "ISCSI_USER" INHERIT_DATASTORE_ATTR =
"ISCSI_USAGE" INHERIT_DATASTORE_ATTR = "ISCSI_HOST"

INHERIT_IMAGE_ATTR = "ISCSI_USER"
INHERIT_IMAGE_ATTR = "ISCSI_USAGE"
INHERIT_IMAGE_ATTR = "ISCSI_HOST"
INHERIT_IMAGE_ATTR = "ISCSI_IQN"

INHERIT_DATASTORE_ATTR = "GLUSTER_HOST"
INHERIT_DATASTORE_ATTR = "GLUSTER_VOLUME"

INHERIT_DATASTORE_ATTR = "ディスクタイプ"
"INHERIT_DATASTORE_ATTR = "アダプタタイプ"

INHERIT_IMAGE_ATTR = "ディスクタイプ"
INHERIT_IMAGE_ATTR = "アダプタタイプ"

INHERIT_VNET_ATTR = "VLAN_タグ付きID"
INHERIT_VNET_ATTR = "IPスルーフィングのフィルタリング"
"INHERIT_VNET_ATTR =
"MACスルーフィングフィルタリング"
"INHERIT_VNET_ATTR =
"MTU"
```

## 10.2.20 OneGate 設定

- `ONEGATE_ENDPOINT`: OneGate が待機するエンドポイントです。オプションです。設定

例:

```
ONEGATE_ENDPOINT = "http://192.168.0.5:5030"
```

### 10.2.21 VDC ACL ルールに対するデフォルトの権限

VDC にリソースが追加された際に作成されるデフォルトの ACL ルールです。以下の属性により、各リソースタイプに対して VDC グループに付与される権限を設定します:

- DEFAULT\_VDC\_HOST\_ACL: VDC に追加されたホストに対して付与される権限。
- DEFAULT\_VDC\_NET\_ACL: VDC に追加された仮想ネットワークに対して付与される権限。
- DEFAULT\_VDC\_DATASTORE\_ACL: VDC に追加されたデータストアに対して付与される権限。
- DEFAULT\_VDC\_CLUSTER\_HOST\_ACL: クラスタが VDC に追加された際に、クラスタホストに対して付与される権限です。
- DEFAULT\_VDC\_CLUSTER\_NET\_ACL: クラスタが VDC に追加された際にクラスタの仮想ネットワークに付与される権限。
- DEFAULT\_VDC\_CLUSTER\_DATASTORE\_ACL: クラスタに追加されたデータストアに対して付与される権限です。

権限を定義する際、"" または "--" を使用することで、特定のリソースにルールを追加しないようにできます。また、複数の権限を "+" で組み合わせることも可能です。例えば "MANAGE+USE" のように指定します。有効な権限は USE、MANAGE、または ADMIN です。

例:

```
DEFAULT_VDC_HOST_ACL = "MANAGE"
# ホストがVDCに追加された際に、@<gid> HOST/#<hid> MANAGE #<zid> を追加します。
onevdc addhost <VDC> <ZID> <HID>

デフォルトVDCネットワークアク = "USE"
# 仮想ネットワークがVDCに追加された際に、@<gid> NET/#<netid> USE #<zid> を追加します。
onevdc addvnet <VDC> <ZID> <VNetID>

DEFAULT_VDC_DATASTORE_ACL = "USE"
# 仮想ネットワークがVDCに追加された際に、@<gid> DATASTORE/#<dsid> USE #<zid> を追加します。
onevdc adddatastore <vdc> <zid> <dsid>

DEFAULT_VDC_CLUSTER_HOST_ACL = "MANAGE"
DEFAULT_VDC_CLUSTER_NET_ACL = "USE"

DEFAULT_VDC_CLUSTER_DATASTORE_ACL = "USE"

追加内容:
#@<gid> ホスト/%<cid> 管理 #<zid> #@<gid> データストア
+ネットワーク/%<cid> 使用 #<zid> #VDCにクラスターを追加す
る場合。 onevdc addcluster <vdc> <zid> <cid>
```

## 10.3 ログ記録とデバッグ

OpenNebulaは、多くのリソースに関するログを提供します。ファイルベースのログシステム、syslogログ、標準エラーストリームへのログ出力という3つのログシステムをサポートしております。

ファイルベースのロギングの場合、OpenNebulaはアクティブなコンポーネントごとに個別のログファイルを保持し、それらすべてを /var/log/one に保存されます。ユーザーや管理者が問題を発見・解決するのを支援するため、CLI や *Sunstone GUI* から一部のエラーメッセージにアクセスすることも可能です。

syslog または標準エラーへのロギングでは、ロギングメッセージの形式が syslog ロギングの規約とリソース情報に従って若干変更

---

される点を除き、ロギング戦略はほぼ同一です。

### 10.3.1 ロギングシステムの設定

ロギングシステムは、`/etc/one/oned.conf` の LOG セクションで変更できます。変更可能なパラメータは 2 つあります。SYSTEM は「syslog」、「file」（デフォルト）または「std」から選択でき、DEBUG\_LEVEL はロギングの詳細度です。

スケジューラの場合も、まったく同じ方法でロギングシステムを変更できます。この場合の設定は `/etc/one/sched.conf` にあります。

### 10.3.2 ログリソース

OpenNebulaの各コンポーネントに対応する異なるログリソースが存在します：

- **ONEデーモン**: OpenNebulaの中核コンポーネントは、すべてのログ情報を `/var/log/one/oned.log` に記録します。その詳細度は `/etc/one/oned.conf` 内の DEBUG\_LEVEL で制御されます。デフォルトでは、ONE起動スクリプトは最新の `oned.log` ファイルを現在の時刻を用いてバックアップします（例：`oned.log.20121011151807`）。あるいは、このリソースを syslog に記録することも可能です。
- **スケジューラ**: スケジューラに関するすべての情報は `/var/log/one/sched.log` ファイルに収集されます。このリソースも syslog に記録することが可能です。
- **仮想マシン**: 仮想マシン固有の情報は、ログファイル `/var/log/one/` に書き込まれます。  
`<vmid>.log`。OpenNebulaによって管理されるすべての仮想マシンには、独自のディレクトリが用意されています。  
`/var/lib/one/vms/<VID>`  
 syslog/stderr が有効化されていない場合。このディレクトリには以下の情報が記録されます：
  - **デプロイメント記述ファイル**: `deployment.<EXECUTION>` に保存されます。`<EXECUTION>` は VM の実行履歴におけるシーケンス番号です（最初のホストは `deployment.0`、2番目は `deployment.1` のように続きます）。
  - **転送記述ファイル**: `transfer.<EXECUTION>.<OPERATION>` に保存されます。  
`<EXECUTION>` は仮想マシン（VM）の実行履歴におけるシーケンス番号であり、`<OPERATION>` はスクリプトが使用された段階を表します。例えば、`transfer.0.prolog`、`transfer.0.epilog`、または `transfer.1.cleanup` などです。
- **ドライバ**: 各ドライバは、RCファイル内で ONE\_MAD\_DEBUG 変数を有効化できます。有効化されている場合、エラー情報は `/var/log/one/` ドライバ実行ファイル名.log にダンプされます。ドライバからのログ情報は `oned.log` に記録されます。

そうでない場合、情報は syslog/stderr に送信されます。

### 10.3.3 ログ形式

ファイルベースのロギングシステムにおける OpenNebula メッセージの構造は以下の通りです：

日付 [z<ゾーンID>] [モジュール] [ログレベル]: メッセージ本文
--

syslog の場合、標準に従います：

日付 ホスト名 プロセス [プロセスID]: [z<ゾーンID>] [モジュール] [ログレベル]: メッセージ
--

ここで、`zone_id` はフェデレーション内のゾーンの ID であり、単一ゾーン構成の場合は 0 となります。`module` は OpenNebula の内部コンポーネント（VMM、ReM、TM など）のいずれかです。`log_level` はログレベルを示す单一文字であり、I は情報、D はデバッ

グなどを表します。

syslog については、OpenNebula は仮想マシンのイベントも以下のように記録します：

```
date hostname process[pid]: [VM id] [Z<zone_id>] [module] [log_level]: メッセージ
```

同様に、標準エラー出力（stderr）のログ記録も行われます。

oned および VM イベントのフォーマットは以下の通りです：

```
date [Z<ゾーンID>] [モジュール] [ログレベル]: メッセージ
日付 [VM ID] [Z<ゾーンID>] [モジュール] [ログレベル]: メッセージ
```

### 10.3.4 仮想マシンのエラー

仮想マシンのエラーは、所有者または管理者が `onevm show` の出力を確認することで確認できます：

```
$ onevm show 0
仮想マシン 0 情報 ID      : 0
NAME          : one-0
ユーザー       : oneadmin
グループ       : oneadmin
状態          : ACTIVE
LCM_STATE     : プロローグ失敗
開始時刻     : 07/19 17:44:20
終了時刻     : 07/19 17:44:31
デプロイ ID    : -
仮想マシン監視 NET_TX      : 0
受信データ量   : 0
使用済みメモリ : 0
中古CPU       : 0
仮想マシンテンプレートコンテキスト= [
  ファイル=/tmp/some_file, ターゲット
  =hdb ]
CPU=0.1
マークス=[
  メッセージ="イメージ転送スクリプトの実行中にエラーが発生しました: /tmp/some_file を /var/
  ↪lib/one/0/images/isofiles" のコピー中にエラーが発生しました",
  タイムスタンプ="2011年7月19日(火) 17:44:31" ] メモリ=64
NAME=one-0
VMID=0
仮想マシンの履歴      ホスト名 アクション          開始          時間          開始時刻
SEQ                host01   なし  07/19 17:44:31 00 00:00:00 00 00:00:00
```

ファイルをコピーできなかったというエラーメッセージは、おそらくファイルが存在しないことを意味します。

あるいは、仮想マシンのログファイル（`/var/log/one/<vmid>.log`）を確認することもできます。

---

**注記：**障害状態にある仮想マシンを復旧する方法については、「仮想マシンの高可用性ガイド」をご参照ください。

---

### 10.3.5 ホストエラー

---

ホストエラーは、`onehost show` コマンドを実行して確認できます：

```
$ onehost show1 HOST 1
INFORMATION ID
NAME
STATE : 1
IM_MAD : host01
VM_MAD エラー
TM_MAD : im_kvm
          : vmm_kvm
          : tm_shared

ホスト共有
最大メモリ : 0
使用済みメモリ (実使用量) : 0 使
用済みメモリ (割り当て済み) : 0 最大
CPU : 0
使用済み CPU (実使用) : 0 使用済
み CPU (割り当て済み) : 0 合計 VMS
                           : 0

監視情報 エラー=[

MESSAGE="ホスト 1 の監視エラー : MONITOR FAILURE 1 リモートを更新できませんでした", TIMESTAMP="2011年7月19日(火) 17:17:22"
]
```

エラーメッセージは、モニタリングのERROR値に表示されます。詳細な情報については、`/var/log/one/oned.log`をご確認ください。例えば、このエラーについては、ログファイルに以下のように記録されます：

```
2011年7月19日(火) 17:17:22 [InM][I]: ホスト host01 (1) の監視
2011年7月19日(火) 17:17:22 [InM][I]: コマンド実行失敗: scp -r /var/lib/one/
←remotes/. host01:/var/tmp/one
2011年7月19日(火) 17:17:22 [InM][I]: ssh: ホスト名 host01 を解決できませんでした: ノード名
←ホスト名またはサービス名が指定されていません、または不明です
2011年7月19日(火) 17:17:22 [InM][I]: 接続が失われました
2011年7月19日(火) 17:17:22 [InM][I]: 終了コード: 1
2011年7月19日(火) 17:17:22 [InM][E]: ホスト 1 の監視エラー: MONITOR FAILURE 1
←リモートを更新できません
```

実行結果から、ホスト名が不明であることがわかります。おそらくホスト名の命名ミスが原因と思われます。

## 10.4 Onedb Tool

このセクションでは、`onedb` CLIツールについて説明します。このツールは、OpenNebulaデータベースから情報を取得したり、データベースをアップグレードしたり、不整合の問題を修正したりするために使用できます。

### 10.4.1 接続パラメータ

`onedb` コマンドは、SQLite または MySQL データベースに接続できます。完全なリファレンスについては、`onedb` のマニュアルページをご参照ください。デフォルトのデータベースに対する 2 つの使用例を以下に示します：

```
$ onedb <コマンド> -v --sqlite /var/lib/one/one.db
$ onedb <コマンド> -v -S localhost -u oneadmin -p oneadmin -d opennebula
```

## 10.4.2 onedb fsck

DB（データベース）の一貫性をチェックし、問題があれば修正します。たとえば、OpenNebula が実行されているマシンがクラッシュした場合や、データベースへの接続が失われた場合、ホストで実行されている VM の数が間違っていたり、一部のユーザーのクオータが不正確になったりすることがあります。

```
$ onedb fsck --sqlite /var/lib/one/one.db
SQLiteデータベースのバックアップは/var/lib/one/one.db.bckに保存されます。
データベースを復元するには、「onedb restore」を使用するか、ファイルをコピーして元に戻してください。

ホスト 0 の稼働中仮想マシンには 12      は 11 5      1441792
ホスト下の CPU 使用率は 260% です      1100 で
ユーザー 2 のクオータ: CPU_USED は 12      す      11.0
ホスト 0 の MEM_USAGE は 572864 です
ユーザー 2 のクオータ: 使用済みメモリは 1536 ユーザー      は 1408
2 のクオータ: 使用済み仮想マシンは 12 です ユーザー 2 11
のクオータ: イメージ 0 の使用済み仮想メモリは 6 です      5
ダルニブ 1 の割り当て: CPU_USED は 136 です グループ 1 の割
り当て: VMS_USED は 12 です      11
ダルニブ 1 割当量: 画像 0 RVMS は 6 です      5
検出されたエラーの総数: 12
```

onedb fsck で以下のエラーメッセージが表示された場合：

```
[修復不可] VM <<vid>> のシーケンス番号 <<seq>> に関する履歴レコードが閉じられていません (etime = 0)
```

これは修正済みバグ #4000 によるものです。会計処理やショーバック機能を使用する際、当該履歴レコードの終了時刻 (etime) が設定されず、本来停止しているべき仮想マシンが稼働中と見なされていたことを意味します。この問題を修正するには、ログから仮想マシンのシャットダウン時刻を特定し、以下のパッチを適用して時刻を手動で編集してください。

```
$ onedb patch -v --sqlite /var/lib/one/one.db /usr/lib/one/ruby/onedb/patches/history_times.rb
```

読み取りバージョン：

共有テーブル 4.11.80 : OpenNebula 5.0.1 デーモンブートストラップローカルテーブル  
4.13.85 : OpenNebula 5.0.1 デーモンブートストラップ

SQLite データベースのバックアップは /var/lib/one/one.db\_2015-10-13\_12:40:2.bck に保存されています。データベースを復元するには、「onedb restore」を使用するか、ファイルを元の場所に戻してください。

> /usr/lib/one/ruby/onedb/patches/history\_times.rb のパッチを実行中  
このツールを使用すると、VM の履歴レコードのタイムスタンプを編集できます。

←会計処理およびショーバックの計算に使用されます。VM

履歴 1 シーケンス番号: 0

STIME	開始時刻	:	2015年10月8日	15:24:06	UTC
PST	プロローグ開始時刻	:	2015年10月8日	15:24:06	UTC
現地時間	プロローグ終了時刻	:	2015年10月8日	15:24:29	UTC
RSTIME	開始時刻 :		2015年10月8日	15:24:29	UTC
RETIME	実行終了時刻	:	2015年10月8日	15:42:35	UTC
ESTIME	エピローグ開始時刻	:	2015年10月8日	15:42:35	UTC
EETIME	エピローグ終了時刻	:	2015年10月8日	15:42:36	UTC
ETIME	終了時刻	:	2015年10月8日	15:42:36	UTC

新しい値を設定するには：

```
現在の値を使用する場合は空欄のままにします。UTC形式の<YYYY-MM-DD HH:MM:SS>を指定するか、設定しない場合は0を指定します（オーバー
ーブン
新規登録）。STIME 開始時刻 : 2015-10-08 15:24:06 UTC
新しい値 : 
終了時刻 終了時刻 新しい値 : 2015年10月8日 15時42分36秒 UTC
          : 
```

**VM 1 の履歴レコード #0** は、以下の新しい値で更新されます：

```
開始時刻 開始時刻 : 2015-10-08 15:24:06 UTC
PSTIME プロローグ開始時刻 : 2015-10-08 15:24:06 UTC PETIME プロローグ終了時刻 : 2015-10-08 15:24:29 UTC
RSTIME 実行開始時刻 : 2015-10-08 15:24:29 UTC RETIME 実行終了時刻 : 2015-10-08 15:42:35 UTC
UTCEETIME エピローグ開始時刻 : 2015-10-08 15:42:35 UTC ESTIME エピローグ終了時刻 : 2015-10-08 15:42:36 UTC
ETIME エピローグ終了時刻 : 2015-10-08 15:42:36 UTC
```

データベースへの書き込みを確認します [Y/n]：y

> 完了

> 合計時間: 27.79秒

### 10.4.3 onedb のバージョン

現在のデータベースのバージョンを表示します。

```
$ onedb version --sqlite /var/lib/one/one.db 3.8.0
```

完全なバージョン情報とコメントを表示するには、-v フラグをご利用ください。

```
$ onedb version -v --sqlite /var/lib/one/one.dbバージョン:
3.8.0
タイムスタンプ: 10/19 16:04:17
コメント: データベースは onedb コマンドにより 3.7.80 から 3.8.0 (OpenNebula 3.8.0) へ移行されました。
```

MySQLデータベースのパスワードに@や#などの特殊文字が含まれている場合、onedbコマンドはデータベースへの接続に失敗します。

回避策として、一時的に oneadmin のパスワードを英数字のみの文字列に変更してください。以下の set password ステートメントをご利用いただけます：

```
$ mysql -u oneadmin -p
mysql> SET PASSWORD = PASSWORD('newpass');
```

### 10.4.4 onedb history

データベースがアップグレードされるたびに、そのプロセスはログに記録されます。historyコマンドを使用すると、アップグレード履歴を取得できます。

```
$ onedb history -S localhost -u oneadmin -p oneadmin -d opennebula バージョン: 3.0.0
タイムスタンプ: 10/07 12:40:49
コメント: OpenNebula 3.0.0 デーモン ブートストラップ
...
バージョン: 3.7.80
タイムスタンプ: 10月8日 17時36分15秒
コメント: データベースを3.6.0から3.7.80 (OpenNebula 3.7.80) へ移行いたしました (onedbによる処理)
←コマンドにより移行されました。
バージョン: 3.8.0
タイムスタンプ: 10月19日 16:04:17
コメント: データベースを 3.7.80 から 3.8.0 (OpenNebula 3.8.0) へ onedb コマンドにより移行いたしました。
```

#### 10.4.5 onedb アップグレード

アップグレード手順の詳細は、「以前のバージョンからのアップグレード」ガイドに完全に記載されております。

#### 10.4.6 onedb backup

OpenNebula データベースをファイルにダンプします。

```
$ onedb backup --sqlite /var/lib/one/one.db /tmp/my_backup.db SQLite データベースのバックアップは /tmp/my_backup.db に保存されます。
データベースを復元するには、「onedb restore」を使用するか、ファイルを元の場所に戻してください。
```

#### 10.4.7 onedb restore

バックアップファイルからデータベースを復元します。ご注意ください、このツールは同一のバックエンドから生成されたバックアップのみ復元可能です。つまり、SQLiteデータベースをバックアップした後、MySQLデータベースにデータを移行することはできません。

#### 10.4.8 onedb sqlite2mysql

このコマンドはSQLiteデータベースからMySQLデータベースへの移行を行います。手順は以下の通りです：

- OpenNebula を停止します
- /etc/one/oned.conf 内の DB ディレクティブを SQLite から MySQL に変更します
- MySQLデータベースの初期化: oned -i
- データベースを移行します: onedb sqlite2mysql -s <SQLITE\_PATH> -u <MYSQL\_USER> -p <MYSQL\_PASS> -d <MYSQL\_DB>
- OpenNebulaを起動します

## 10.4.9 onedb purge-history

非完了状態の仮想マシンから、最後の2件を除くすべての履歴レコードを削除します。すべての履歴を削除したくない場合は、開始日と終了日を指定できます：

```
$ onedb purge-history --start 2014/01/01 --end 2016/06/15
```

**警告:** この操作は OpenNebula の稼働中に実行されます。実行前にデータベースのバックアップを作成してください。

#### 10.4.10 onedb purge-done

DONE状態のマシンから情報を削除します。--startおよび--endパラメータは、purge-historyと同様に--startおよび--endパラメータを使用できます：

```
$ onedb purge-done --end 2016/01
```

**警告:** この操作は OpenNebula の稼働中に実行されます。実行前にデータベースのバックアップを作成してください。

#### 10.4.11 onedb change-body

オブジェクト本体から値を変更します。対象となるオブジェクトは以下の通りです：vm、host、vnet、image、cluster、document、group、marketplace、marketplaceapp、secgroup、template、vrouter、zone。

以下のいずれかのオプションを使用して、変更対象のオブジェクトを絞り込むことができます：

- **--id:** オブジェクト ID、例: 156
- **--xpath:** XPath式、例: TEMPLATE[count(NIC)>1]
- **--expr: XPath式。** 演算子 =、!=、<、>、<=、>= を使用できます。例：  
UNAME=oneadmin,  
TEMPLATE/NIC/NIC\_ID>0

値を変更したい場合は、第三引数をご利用ください。削除をご希望の場合は、**--delete** オプションをご利用ください

```
$ onedb change-body vm --expr UNAME=user '/VM/TEMPLATE/NIC[NETWORK="service"] /NETWORK  
↳' new network
```

◦

すべてのディスクのキャッシュ属性を削除し、XMLを書き込みますが、データベースは変更しません：

```
$ onedb change-body vm '/VM/TEMPLATE/DISK/CACHE' --delete --dry
```

電源オフ状態のすべてのディスクのキャッシュ属性を削除します：

```
$ onedb change-body vm --expr LCM_STATE=8 '/VM/TEMPLATE/DISK/CACHE' --delete
```

**警告:** この操作は OpenNebula の稼働中に実行されます。実行前にデータベースのバックアップを作成してください。

### 10.5 スケーラビリティのテストと調整

クラウドのスケーラビリティを判断し、それを向上させるには、多くの変数をバランスよく考慮する必要があります。ストレージ

からネットワークバックエンドに至るまで、クラウドのスケーラビリティを制限する要因は複数存在し、あらゆる方の拡張性目標を満たす単一の解決策は存在しません。

本ガイドでは、まず単一OpenNebulaインスタンス（単一ゾーン）におけるスケール限界を提示し、その後、大規模展開に向けたチューニングに関する推奨事項を提供します。

一般的な推奨事項として、单一インスタンスで管理するサーバーは2,500台以下、仮想マシン（VM）は10,000台以下に抑えることが望ましいです。データベースなどの他のコンポーネントを最適化したり、より高性能なハードウェアを導入したりすることで、より優れたパフォーマンスと高いスケーラビリティを実現できます。いずれにせよ、クラウドの規模をこれらの制限を超えて拡大するには、フェデレーション環境内で新たなOpenNebulaゾーンを追加することにより、クラウドを水平方向にスケールさせることができます。最大規模のOpenNebula導入事例では、16のデータセンターと30万コアで構成されています。

### 10.5.1 スケーラビリティテスト

#### フロントエンド (oned) のスケーラビリティ

本セクションでは、単一の OpenNebula ゾーンにおけるコントローラコンポーネントである oned のスケール限界に焦点を当てます。ここで推奨する限界値は特定の API 負荷に関連付けられています。実際の API リクエストに基づいて、限界値の増減をご検討ください。単一の OpenNebula インスタンスが管理できるサーバー（仮想化ホスト）の最大数は、基盤となるプラットフォームインフラストラクチャ、主にストレージサブシステムのパフォーマンスとスケーラビリティに大きく依存することに留意してください。以下の結果は、以下の仕様の物理サーバー上で動作する oned に負荷をかける合成ワークロードを用いて得られたものです。

CPUモデル：	Intel(R) Atom(TM) CPU C2550 @ 2.40GHz、4コア、HT非対応
RAM:	8GB、DDR3、1600 MT/s、シングルチャネル
HDD:	インテル SSDSC2BB150G7
OS:	Ubuntu 18.04
OpenNebula:	バージョン5.8
データベース:	MariaDB v10.1（デフォルト設定）

単一ゾーンにおいて、OpenNebula (oned) は以下の制限で動作いたします：

ホスト数	2,500
仮想マシン数	10,000
仮想マシンテンプレートの平均サイズ	7 KB

本番環境においては、システムの過度な遅延を避けるため、同一インストール環境におけるサーバー数（2,500台）および仮想マシン数（10,000台）の上限、ならびにAPIリクエストの負荷（30リクエスト/秒）を超えないようお勧めいたします。データベースなどの他コンポーネントの最適化や、より高性能なハードウェアの導入により、さらなるパフォーマンス向上が見込めます。

最も頻繁に使用される4つのAPI呼び出しを、実際のデプロイ環境で観測される比率とほぼ同等の比率で同時に実行し、コアに負荷をかけました。秒あたりのAPI呼び出し総数は、10、20、30回でした。これらの条件下で、ホスト監視間隔を20ホスト/秒、ホスト数2,500のプール、各ホストの監視期間を125秒とした場合、最も一般的なXMLRPC呼び出しに対するonedプロセスの応答時間（秒単位）は以下の通りです：

応答時間 (秒)			
API呼び出し - 比率	API負荷：10リクエスト/秒	API負荷：20リクエスト/秒	API負荷：30リクエスト/秒
host.info (30%)	0.06	0.50	0.54
Hostpool.info (10%)	0.14	0.41	0.43

vm.info (30%)	0.07	0.57	0.51
vmpool.info (30%)	1.23	2.13	4.18

## ハイパーバイザのスケーラビリティ

仮想化ノードが実行できる仮想マシン（またはコンテナ）の数は、監視プローブによって制限されます。本セクションでは、KVM および LXD ドライバーにおける監視プローブのパフォーマンスを評価しました。ホストの仕様は以下の通りです：

CPUモデル：	Intel(R) Atom(TM) CPU C2550 @ 2.40GHz、4コア、HT非対応
RAM:	8GB、DDR3、1600 MT/s、シングルチャネル
HDD:	INTEL SSDSC2BB150G7
OS:	Ubuntu 18.04
ハイパーバイザ ー:	Libvirt (4.0)、Qemu (2.11)、lxd (3.03)

監視テストに使用したコンテナおよび仮想マシンの設定：

ハイパーバイ ザー	OS	RAM	CPU	最大VM/ホスト
KVM	なし (空のディスク)	32MB	0.1	250
LXD	Alpine	32MB	0.1	250

注記：両タイプのハイパーバイザに対して250インスタンスがデプロイされました。これは、両テンプレートが同一の容量要件で作成されたためです。これにより割り当てリソースが同率で増加したため、スケジューラはKVM仮想マシンとLXDコンテナを同数インスタンス化しました。注目すべき点は、KVM仮想マシンのオーバーヘッドがLXDコンテナよりも大きいことです。したがって、本レポートが示す通り、後者のハイパーバイザ技術はKVMと比較してより高い密度を実現します。

各タイプのインスタンスの監視時間は以下の通りです：

ハイパーバイ ザー	インスタンスあたりの監視時間 (秒)
KVM	0.42
LXD	0.1

なお、単一ホスト上で実行できる仮想マシンやコンテナの数を制限する要因は他にも存在する可能性があります。

## 10.5.2 大規模環境向けのチューニング

### 監視の調整

KVM環境において、OpenNebulaは2つのネイティブ監視システムをサポートしております：ssh-pullとudp-pushです。前者であるssh-pullは、OpenNebula 4.2以前のデフォルト監視システムでしたが、OpenNebula 4.4以降ではudp-pushがデフォルトとなりました。このモデルは高いスケーラビリティを有し、その限界（1秒あたりに監視可能な仮想マシンの数）は、onedを実行するサーバーとデータベースサーバーの性能によって決まります。詳細は[監視ガイド](#)をご覧ください。

vCenter環境においては、OpenNebulaはvCenterが提供するVI APIを利用し、インポートされた全てのvCenterクラスター内で稼働するハイパーバイザおよび仮想マシンの状態を監視します。このドライバーは、一般的な仮想マシン情報をキャッシュするよう最適化されています。

いずれの環境においても、当社のスケーラビリティテストでは数分で数万台の仮想マシンの監視を実現しています。

## コアチューニング

OpenNebulaは、定義された期間の監視履歴をデータベーステーブルに保持します。これらの値は、Sunstoneでのプロット作成に使用されます。これらの監視エントリは、データベース内でかなりのストレージ容量を占める可能性があります。使用されるストレージ容量は、クラウドの規模およびoned.conf内の以下の構成属性によって異なります：

クラウドの規模と、*oned.conf*内の以下の設定属性によって異なります：

- MONITORING\_INTERVAL\_HOST: 各監視サイクル間の間隔（秒単位）。デフォルト: 180。このパラメータは、標準のudp-pushモデルにおいて、監視プローブをプロアクティブに再起動するタイムアウトを設定します。
- collectd IM\_MAD -i 引数（KVM のみ）：監視プッシュサイクルの時間（秒単位）。デフォルト: 60。
- HOST\_MONITORING\_EXPIRATION\_TIME: 監視情報の有効期限（秒単位）。デフォルト: 12時間。
- VM\_MONITORING\_EXPIRATION\_TIME: 監視情報の有効期限（秒単位）。デフォルト: 4時間。
- MONITORING\_INTERVAL\_DB\_UPDATE: VM 監視情報のデータベース書き込み間隔（秒単位）。デフォルト: 0（更新ごとに書き込み）。

---

**重要:** 5,000 台以上の仮想マシンを実行する場合、データベース書き込みスレッドの負荷を軽減するため、MONITORING\_INTERVAL\_DB\_UPDATE および IM\_MAD-collectd -i 引数の値を大きくすることを強く推奨いたします。多数の仮想マシンを実行する際のAPI応答遅延は、通常この要因によるものです。

---

Sunstoneをご利用でない場合は、監視履歴を無効化することをご検討ください。その場合、両方の有効期限を0に設定します。

各監視エントリは、ホストごとに約2KB、仮想マシンごとに約4KBとなります。必要なデータベースストレージの目安として、以下の例をご参照ください：

監視間隔	ホストの有効期限	ホスト数	ストレージ
20秒	12時間	200	850 MB
20秒	24時間	1,000   8.2 GB	

監視間隔	VMの有効期限	VM数	ストレージ
20秒	4時間	2,000   1.8 GB	
20秒	24時間	10,000   7 GB	

## APIチューニング

XML-RPC呼び出しが多数発生する大規模な展開環境では、XML-RPCサーバーのデフォルト設定は控えめすぎます。変更可能な設定値とその意味については、*oned.conf*およびxmlrpc-cライブラリのドキュメントに記載されています。当社の経験上、以下の設定値を変更することで、多数のクライアント呼び出しが発生する環境におけるサーバーの動作が改善されます：

```
MAX_CONN = 240
MAX_CONN_BACKLOG = 480
```

コアは一部のプール回答をページ分割することができます。これによりメモリ消費量が減少し、場合によっては解析速度が向上します。デフォルトのページ分割値は2,000オブジェクトですが、環境変数ONE\_POOL\_PAGE\_SIZEを使用して変更できます。この値は2以上である必要があります。例えば、ページサイズ5,000でVMを一覧表示するには以下のように設定します：

```
$ ONE_POOL_PAGE_SIZE=5000 onevm list
```

ページネーションを無効化するには、数値以外の値を使用できます：

```
$ ONE_POOL_PAGE_SIZE=disabled onevm list
```

この環境変数はSunstoneでも使用可能です。また、OpenNebulaのスケーリングにおける主な障壁の一つは、大規模プールにおける

list操作です。OpenNebula 5.8以降、VMプールは要約形式で一覧表示されます。しかしながら

返されるプールサイズを縮小するため、検索操作の利用をお勧めいたします。検索操作はバージョン5.8以降、VMプールで利用可能です。

### APIサーバーのスケーリング

OpenNebulaをスケーリングするためには、複数のonedプロセス間でクライアントリクエストを分散させることをお勧めいたします。これは、既存のRAFTフォロワーを利用するか、APIサーバー専用モードでonedを追加することで実現できます。

onedが読み取り専用（またはキャッシュ）モードで起動された場合、読み取り専用操作はデータベースに直接アクセスして解決されます。特に、以下のAPI呼び出しはキャッシュモードにおいてサーバーが直接処理します：

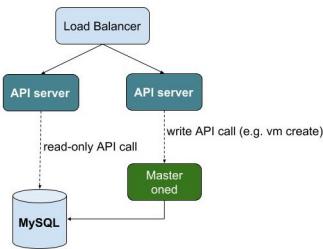
one.vmpool.info	one.clusterpool.info	one.group.info
one.vmpool.accounting	one.zonepool.info	one.user.info
one.vmpool.showback	one.secgrouppool.info	one.データストア.info
one.vmpool 監視	one.VDCPool.info	one.クラスタ情報
one.テンプレートプール.info	one.vrouterpool.info	one.ドキュメント.info
one.vnpool.info	ワン・マーケットプール・インフォ	one.zone.info
one.vntemplatepool.info	one.marketapppool.info	one.secgroup.info
ワン・イメージプール・インフォ	one.vmgrouppool.info	one.vdc.info
one.hostpool.info	one.template.info	one.vrouter.info
one.hostpool.monitoring	one.vn.info	ワン・マーケット・インフォ
one.grouppool.info	one.vntemplate.info	one.marketapp.info
one.userpool.info	ワン・イメージ・インフォ	one.vmgroup.info
one.データストアプール.info	one.host.info	one.zone.raftstatus

---

**注記：**読み取り専用操作では、ACL 制限や所有権チェックが適用されます。

---

その他のAPI呼び出しは、アクティブなonedプロセスに転送されます。この場合、キャッシュサーバーは単純なプロキシとして機能します。キャッシュサーバーと組み合わせて使用することを推奨するアーキテクチャは、以下の図に示されています：



マスター ノードが実際にRAFTクラスターである場合、APIサーバーをクラスターのVIPアドレスに指定するだけで構築です。なお、各RAFTサーバー内のMySQLサーバーは、APIサーバーがデータベースにクエリを実行できるよう、VIPアドレスでリスニングするように設定する必要があります。

### 設定

---

APIサーバーを設定するには、以下の手順が必要です：

- 
1. サーバーにOpenNebulaパッケージをインストールする
  2. oned.confファイルを更新し、マスターonedおよびデータベースを指すように設定します：

```

DB = [
    BACKEND = "mysql",
        SERVER = "MySQLサーバーのIPを設定", PORT = 0,
        USER = "oneadmin", PASSWD =
        "oneadmin", DB_NAME =
        "opennebula", CONNECTIONS =
        50

    FEDERATION = [
        MODE             = "CACHE",
        ゾーンID         = 0,
        サーバーID       = -1,
        MASTER_ONED     = "マスターノードのXML-RPCエンドポイントを設定します"
    ]
]

```

また、データベースへの接続数を調整する必要がある点にもご注意ください。MySQLサーバーの場合は接続数を増やし、キャッシュサーバーの数を調整してください。ただし、接続数の合計はすべてのサーバーで共有される点にご留意ください。

## ロードバランシング

あるいは、クライアントのリクエストをAPIサーバー間で分散するロードバランサーを設定することも可能です。このタスクにはHAProxyが適しています。このシナリオでは、OpenNebulaサーバー1台とOpenNebulaキャッシュサーバー2台を想定しております。ロードバランサーは別のサーバーのポート2633でリスニングし、クラスターを構成する3台のOpenNebulaサーバーへ接続を転送します。以下は、上記のようなシナリオに必要なHAProxy設定の関連部分です：

```

frontend OpenNebula bind
0.0.0.0:2633
stats enable mode
tcp
default_backend one_nodes

backend one_nodes mode
tcp
stats enable balance
roundrobin
サーバー opennebula1 10.134.236.10:2633 チェックサーバー
opennebula2 10.134.236.11:2633 チェックサーバー
opennebula3 10.134.236.12:2633 チェック

```

サーバーエントリは変更が必要であり、statsセクションはオプションです。

必要に応じて、別のサーバーに2つ目のロードバランサーを追加し、両方のロードバランサーノード間でVRRPなどのアクティブ-パッシブ冗長化プロトコルを設定することで、高可用性を実現できます。

別のサーバーからクラスターに接続するには、以下の2つの方法のいずれか、または両方をご利用いただけます：

- CLIを使用する場合：新しいエンドポイントでONE\_XMLRPC変数を作成します。例：

```
export ONE_XMLRPC=http://ENDPOINT_IP:2633/RPC2
```

- Sunstoneを使用する場合：/etc/one/sunstone-server.conf内のone\_xmlrpcを変更します。

新しいエンドポイントはロードバランサーのアドレスとなります。

## ドライバの調整

OpenNebula ドライバーはデフォルトで 15 スレッドを使用します。これはドライバーが同時に実行できるアクションの最大数であり、それ以降のアクションはキューに追加されます。この値は *oned.conf* で設定可能です。ドライバーパラメータは -t です。

## データベースの調整

非テスト環境でのインストールには、MySQLデータベースをご利用ください。SQLiteは、数台のホストと少数の仮想マシンを超える環境では速度が不足します。MySQLデータベースバックエンドの[推奨されるメンテナンス手順](#)を必ずご確認ください。

## Sunstoneのチューニング

[大規模展開向けのSunstone設定](#)に関するガイドをご参照ください。