



Red Hat OpenStack Platform 17.1

オーバークラウドのパラメーター

Red Hat OpenStack Platform オーバークラウドのコアテンプレートコレクションを
カスタマイズするためのパラメーター

Red Hat OpenStack Platform 17.1 オーバークラウドのパラメーター

Red Hat OpenStack Platform オーバークラウドのコアテンプレートコレクションをカスタマイズするためのパラメーター

OpenStack Team
rhos-docs@redhat.com

法律上の通知

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

このガイドには、Orchestration サービス (heat) を使用した OpenStack のデプロイメントで使用される可能性のあるパラメーターがリストされています。パラメーターと定義は上アップストリームのソースコードから抽出されており、リストされているすべてのパラメーターがサポートされる構成で使用できるわけではありません。

目次

本ガイドについて	3
多様性を受け入れるオープンソースの強化	4
RED HAT ドキュメントへのフィードバック (英語のみ)	5
第1章 オーバークラウドのパラメーター	6
第2章 オーバークラウドのコアパラメーター	7
第3章 ロールベースのパラメーター	11
第4章 デバッグパラメーター	13
第5章 カーネルパラメーター	15
第6章 DNS (DESIGNATE) パラメーター	17
第7章 BARE METAL (IRONIC) パラメーター	19
第8章 BLOCK STORAGE (CINDER) パラメーター	26
第9章 CEPH STORAGE パラメーター	34
第10章 COMPUTE (NOVA) パラメーター	41
第11章 DASHBOARD (HORIZON) パラメーター	63
第12章 IDENTITY (KEYSTONE) パラメーター	65
第13章 IMAGE STORAGE (GLANCE) パラメーター	70
第14章 KEY MANAGER (BARBICAN) パラメーター	76
第15章 ロードバランサー (OCTAVIA) パラメーター	79
第16章 MESSAGING パラメーター	83
第17章 NETWORKING (NEUTRON) パラメーター	84
第18章 OBJECT STORAGE (SWIFT) パラメーター	97
第19章 ORCHESTRATION (HEAT) パラメーター	100
第20章 SHARED FILE SYSTEM (MANILA) パラメーター	104
第21章 時刻に関するパラメーター	106
第22章 アップグレードのパラメーター	108

本ガイドについて



警告

このガイドは、アップストリームの OpenStack ソースコードに埋め込まれたコメントから自動的に生成されています。したがって、このガイドに記載されているすべてのパラメーターが実稼働環境でサポートされているわけではありません。実際にサポートされるパラメーターに関する情報を見つけるには、サポート対象のユースケースについて説明する関連ガイドを参照してください。

多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、今後の複数のリリースで段階的に用語の置き換えを実施して参ります。詳細は、[Red Hat CTO である Chris Wright のメッセージ](#) をご覧ください。

RED HAT ドキュメントへのフィードバック (英語のみ)

Red Hat ドキュメントに対するご意見をお聞かせください。ドキュメントの改善点があればお知らせください。

Jira でドキュメントのフィードバックを提供する

ドキュメントに関するフィードバックを提供するには、[Create Issue](#) フォームを使用します。Red Hat OpenStack Platform Jira プロジェクトで Jira Issue が作成され、フィードバックの進行状況を追跡できます。

1. Jira にログインしていることを確認してください。Jira アカウントをお持ちでない場合は、アカウントを作成してフィードバックを送信してください。
2. [Create Issue](#) をクリックして、**Create Issue** ページを開きます。
3. **Summary** フィールドと **Description** フィールドに入力します。**Description** フィールドに、ドキュメントの URL、章またはセクション番号、および問題の詳しい説明を入力します。フォーム内の他のフィールドは変更しないでください。
4. **Create** をクリックします。

第1章 オーバークラウドのパラメーター

オーバークラウドパラメーターを使用してオーバークラウド機能を変更できます。パラメーターを設定するには、選択したパラメーターとその値を環境ファイルの **parameter_defaults** セクションに追加し、**openstack overcloud deploy** コマンドでその環境ファイルを含めます。

第2章 オーバークラウドのコアパラメーター

オーバークラウドのコアパラメーターを使用して、一般的なオーバークラウド設定を変更できます。

パラメーター	説明
AddVipsToEtcHosts	ネットワーク別の仮想 IP を各ノードの <code>/etc/hosts</code> に追加するには、 <code>True</code> に設定します。デフォルト値は True です。
AdminPassword	OpenStack Networking (neutron) の監視、クエリー等に使用する keystone 管理アカウントのパスワード
CloudDomain	ホストに使用する DNS ドメイン。これは、アンダークラウドで設定されている <code>overcloud_domain_name</code> と一致する必要があります。デフォルト値は localdomain です。
CloudName	そのクラウドの DNS 名。デフォルト値は overcloud.localdomain です。
CloudNameCtlplane	そのクラウドのコントロールプレーンエンドポイントの DNS 名。デフォルト値は overcloud.ctlplane.localdomain です。
CloudNameInternal	そのクラウドの内部 API エンドポイントの DNS 名。デフォルト値は overcloud.internalapi.localdomain です。
CloudNameStorage	そのクラウドのストレージエンドポイントの DNS 名 (例: <code>ci-overcloud.storage.tripleo.org</code>)。デフォルト値は overcloud.storage.localdomain です。
CloudNameStorageManagement	そのクラウドのストレージ管理エンドポイントの DNS 名。デフォルト値は overcloud.storagegmt.localdomain です。
ControlFixedIPs	コントロールプレーンの固定仮想 IP を定義します。値には <code>[[ip_address:'1.2.3.4']]</code> の形式を使用します。
ControlPlaneSubnet	アンダークラウドの OpenStack Networking (neutron) コントロールプレーンのサブネットの名前。デフォルト値は ctlplane-subnet です。
ControlPlaneSubnetCidr	コントロールプレーンネットワークのサブネット CIDR。このパラメーターは、 ctlplane サブネットの cidr 属性から自動的に解決されます。

パラメーター	説明
DeployedServerPortMap	ctlplane ネットワークまたはオープンクラウドの IP アドレス範囲から固定 IP を割り当てるために使用されます。
DeployIdentifier	このパラメーターを一意的な値に設定すると、OpenStack Orchestration (heat) stack-update で設定を行うデプロイメントタスクが再度実行されません。
DeploymentServerBlacklist	トリガーされたデプロイメントからブロックリストに登録するサーバーのホスト名のリスト。
EndpointMapOverride	計算された EndpointMap をオーバーライドすることができます。
ExternalVirtualFixedIPs	ExternalVirtualInterface ポートの IP 割り当てを制御します (例: <code>[{ip_address:'1.2.3.4'}]</code>)。
ExtraConfig	クラスターに挿入する追加の hiera 設定
ExtraHostFileEntries	/etc/hosts に追記する追加のホストエントリーのリスト
GlobalConfigExtraMapData	各ノードで設定する追加の global_config_settings データのマッピング
HypervisorNeutronPhysicalBridge	各ハイパーバイザーで作成する Open vSwitch ブリッジ。このパラメーターのデフォルトは br-ex で、コントロールプレーンノードと同じです。これにより、Open vSwitch エージェントの設定が統一されます。通常、この値は変更する必要はありません。デフォルト値は br-ex です。
HypervisorNeutronPublicInterface	HypervisorNeutronPhysicalBridge に追加するインターフェイス。デフォルト値は nic1 です。
InternalApiVirtualFixedIPs	InternalApiVirtualInterface ポートの IP 割り当てを制御します。値には <code>[{ip_address:'1.2.3.4'}]</code> の形式を使用します。
KeystoneRegion	エンドポイント用の OpenStack Identity (keystone) リージョン。デフォルト値は regionOne です。

パラメーター	説明
NetConfigDataLookup	事前にプロビジョニングされたノードを使用している場合は、特定のノードの os-net-config マッピングを設定します。 NetConfigDataLookup heat パラメーターの設定は、ノード定義ファイル overcloud-baremetal-deploy.yaml の net_config_data_lookup プロパティと同等です。事前にプロビジョニングされたノードを使用していない場合は、ノード定義ファイルで NIC マッピングを設定する必要があります。
NetworkConfigWithAnsible	ansible フラグを含む NetworkConfig。デフォルト値は True です。
NeutronControlPlaneID	コントロールプレーンネットワークの ID または名前。デフォルト値は ctlplane です。
NeutronPhysicalBridge	外部ネットワークにアクセスするために作成する OVS ブリッジ。デフォルト値は br-ex です。
NeutronPublicInterface	外部ブリッジに接続するインターフェイス。デフォルト値は nic1 です。
NodeCreateBatchSize	ノード作成の最大バッチサイズ。バッチサイズは、32 ノードを超過しないことを推奨します。デフォルト値は 30 です。
NovaAdditionalCell	これがデフォルトのセルに追加するセルかどうかを定義します。デフォルト値は false です。
NovaLocalMetadataPerCell	nova-metadata API サービスがセルごとにデプロイされていることを示します。この場合、マルチセルデプロイメントでのパフォーマンスとデータの分離性が向上します。OpenStack Networking (neutron) の設定方法に応じて、この設定の使用を検討する必要があります。ネットワークが複数セルにまたがる場合は、nova-metadata API サービスをグローバルに実行しなければならない場合があります。ネットワークがセル境界でセグメント化されている場合は、セルごとに nova-metadata API サービスを実行できます。セルごとに nova-metadata API サービスを実行する際には、対応する nova-metadata API サービスを参照するように各 OpenStack Networking (neutron) metadata-agent を設定する必要があります。デフォルト値は false です。
PublicVirtualFixedIPs	PublicVirtualInterface ポートの IP 割り当てを制御します。値には [{ip_address:'1.2.3.4'}] の形式を使用します。

パラメーター	説明
RabbitCookieSalt	RabbitMQ クッキーのソルト。無作為に生成される RabbitMQ クッキーを強制的に変更するには、この値を変更します。デフォルト値は unset です。
RootStackName	スタック/プランの名前
ServerMetadata	オーバークラウドでノードを作成するために OpenStack Compute (nova) に渡される追加のプロパティまたはメタデータ。OpenStack Compute (nova) メタデータ API 経由でアクセスできます。
StorageMgmtVirtualFixedIPs	StorageMgmtVirtualInterface ポートの IP 割り当てを制御します。値には [[ip_address:'1.2.3.4']] の形式を使用します。
StorageVirtualFixedIPs	StorageVirtualInterface ポートの IP 割り当てを制御します。値には [[ip_address:'1.2.3.4']] の形式を使用します。
UndercloudHostsEntries	/etc/hosts に追記するアンダークラウドホストエントリーのリスト。この値には、デプロイの実行中に tripleoclient によって、アンダークラウドの HEAT_HOSTS エントリーの値が入力されます。
UpdateIdentifier	stack-update の実行中に前回使用されなかった値を設定すると、全ノードでパッケージの更新がトリガーされます。

第3章 ロールベースのパラメーター

オーバークラウドのロールベースのパラメーターを使用して、特定のオーバークラウドコンポーザブルロールの動作を変更できます。`_ROLE_` はロール名に置き換えます。たとえば、`_ROLE_Count` には `ControllerCount` を使用します。

パラメーター	説明
<code>_ROLE_AnyErrorsFatal</code>	config-download Ansible Playbook を実行する際に any_errors_fatal の値を設定します。デフォルト値は yes です。
<code>_ROLE_ControlPlaneSubnet</code>	このロールの <code>ctlplane</code> ネットワーク上のサブネット名デフォルト値は ctlplane-subnet です。
<code>_ROLE_Count</code>	1つのロールにデプロイするノード数。デフォルト値は 1 です。
<code>_ROLE_ExtraConfig</code>	クラスターに挿入する、ロール固有の追加の <code>hiera</code> 設定
<code>_ROLE_ExtraGroupVars</code>	オプションの追加 Ansible グループ変数
<code>_ROLE_HostnameFormat</code>	ノードのホスト名の形式。 %index% はノードのインデックス (例: 0/1/2) に、 %stackname% はスタック名 (例: overcloud) に置き換えられる点に注意してください。デフォルト値は %stackname%_-_role_-_%index% です。
<code>_ROLE_LocalMtu</code>	アンダークラウドの <code>local_interface</code> に使用する MTU。デフォルト値は 1500 です。
<code>_ROLE_MaxFailPercentage</code>	config-download Ansible Playbook を実行する際に max_fail_percentage の値を設定します。デフォルト値は 0 です。
<code>_ROLE_NetConfigOverride</code>	<code>os-net-config</code> 設定のオーバーライドに使用されるカスタム JSON データ。アンダークラウドのカスタム ネット設定を簡単に渡す手段を提供するために、 <code>tripleoclient</code> の <code>net_config_override</code> パラメーターが使用するためのものです。
<code>_ROLE_NetworkConfigTemplate</code>	ROLE NetworkConfig テンプレート。
<code>_ROLE_NetworkConfigUpdate</code>	True に設定すると、オーバークラウドで既存のネットワークが更新されます。このパラメーターは、 <code>NetworkDeploymentActions</code> で以前提供されていた機能を置き換えます。デフォルトでは False に設定されており、新規ノードのみネットワークが設定されます。これはロールベースのパラメーターです。デフォルト値は False です。

パラメーター	説明
_ROLE_Parameters	サービスに提供される任意のロール固有のパラメーター
_ROLE_Re movalPolicies	特定リソースの削除が必要な更新の実行時に、そのロールの ResourceGroup から削除されるリソースのリスト
_ROLE_Re movalPoliciesMode	更新を実行する際の、 ROLE ResourceGroup の RemovalPolicies に対する変更の処理方法。デフォルトモードの append の場合には、既存のブラックリストに追加します。 update の場合には、ブラックリストを置き換えます。デフォルト値は append です。
_ROLE_SchedulerHints	OpenStack Compute (nova) に渡すオプションのスケジューラーヒント
_ROLE_ServiceNetMap	ロール固有の ServiceNetMap オーバーライド。提供されるマップは、ServiceNetMap を ROLE_ServiceChain リソースおよび _ROLE リソースグループに渡す際にグローバルの ServiceNetMap とマージされます。例: <code>_ROLE_ServiceNetMap: NovaLibvirtNetwork: internal_api_leaf2</code>
_ROLE_Services	ROLE のロール上にインストールする必要のある各サービスのネストされたスタックを表すサービスリソースのリスト (OpenStack Orchestration (heat) <code>resource_registry</code> で設定される)

第4章 デバッグパラメーター

以下のパラメーターにより、サービス単位でデバッグモードを設定することができます。**Debug** パラメーターは、全サービスのグローバルパラメーターとして機能し、サービスごとのパラメーターは個別のサービス上のグローバルパラメーターの効果を上書きすることができます。

パラメーター	説明
BarbicanDebug	OpenStack Key Manager (barbican) サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
CinderDebug	OpenStack Block Storage (cinder) サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
ConfigDebug	設定管理 (Puppet 等) をデバッグモードで実行するかどうかを定義します。デフォルト値は false です。
デバッグ	全サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
DesignateDebug	Designate サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
GlanceDebug	OpenStack Image Storage (glance) サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
HeatDebug	OpenStack Orchestration (heat) サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
HorizonDebug	OpenStack Dashboard (horizon) サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
IronicDebug	OpenStack Bare Metal (ironic) サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
KeystoneDebug	OpenStack Identity (keystone) サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
ManilaDebug	OpenStack Shared File Systems (manila) サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。

パラメーター	説明
MemcachedDebug	Memcached サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
NeutronDebug	OpenStack Networking (neutron) サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
NovaDebug	OpenStack Compute (nova) サービスのデバッグを有効にするには、True に設定します。デフォルト値は false です。
OctaviaDebug	True に設定すると、OpenStack Load Balancing-as-a-Service (octavia) サービスのデバッグが有効になります。デフォルト値は false です。

第5章 カーネルパラメーター

カーネルパラメーターを使用して、カーネルの動作を変更できます。

パラメーター	説明
BridgeNfCallArpTables	sysctl net.bridge.bridge-nf-call-arpables キーを設定します。デフォルト値は 1 です。
BridgeNfCallIp6Tables	sysctl net.bridge.bridge-nf-call-ip6tables キーを設定します。デフォルト値は 1 です。
BridgeNfCallIpTables	sysctl net.bridge.bridge-nf-call-iptables キーを設定します。デフォルト値は 1 です。
ExtraKernelModules	読み込む追加のカーネルモジュールのハッシュ
ExtraKernelPackages	インストールする追加のカーネル関連パッケージのリスト
ExtraSysctlSettings	適用する追加の sysctl 設定のハッシュ
FsAioMaxNumber	カーネルは aio メモリーをオンデマンドで割り当て、この数により並列 aio リクエストの数が制限されます。制限が大きい場合の唯一の欠点は、悪意のあるゲストが並列リクエストを発行し、カーネルにメモリーを確保させる可能性があることです。この数を少なくとも $128 * (\text{ホスト上の仮想ディスクの数})$ に設定します。Libvirt はデフォルトの 1M リクエストを使用して 8k ディスクを許可し、すべてのディスクが同時に aio リクエストにヒットした場合、最大で 64M のカーネルメモリーを使用します。デフォルト値は 0 です。
InotifyInstancesMax	sysctl fs.inotify.max_user_instances キーを設定します。デフォルト値は 1024 です。
InotifyIntancesMax	sysctl fs.inotify.max_user_instances キーを設定します。デフォルト値は 1024 です。
KernelDisableIPv6	sysctl net.ipv6.{default/all}.disable_ipv6 キーを設定します。デフォルト値は 0 です。
KernelIpForward	net.ipv4.ip_forward キーを設定します。デフォルト値は 1 です。
KernelIpNonLocalBind	net.ipv{4,6}.ip_nonlocal_bind キーを設定します。デフォルト値は 1 です。

パラメーター	説明
KernelIpv4ConfAllRpFilter	net.ipv4.conf.all.rp_filter キーを設定します。デフォルト値は 1 です。
KernelIpv6ConfAllForwarding	net.ipv6.conf.all.forwarding キーを設定します。デフォルト値は 0 です。
KernelPidMax	sysctl kernel.pid_max キーを設定します。デフォルト値は 1048576 です。
NeighbourGcThreshold1	sysctl net.ipv4.neigh.default.gc_thresh1 の値を設定します。これは ARP キャッシュに保持されるエントリーの数の最小値です。キャッシュ内にあるエントリーがこの数より少ない場合、ガベージコレクターは実行されません。デフォルト値は 1024 です。
NeighbourGcThreshold2	sysctl net.ipv4.neigh.default.gc_thresh2 の値を設定します。これは ARP キャッシュに保持されるエントリーの数のソフト最大値です。エントリー数がこの値を超えても、5 秒間ガベージコレクターは実行されません。デフォルト値は 2048 です。
NeighbourGcThreshold3	sysctl net.ipv4.neigh.default.gc_thresh3 の値を設定します。これは ARP キャッシュに保持されるエントリーの数のハード最大値です。キャッシュ内にこの数を超えるエントリーがある場合、必ずガベージコレクターが実行されます。デフォルト値は 4096 です。

第6章 DNS (DESIGNATE) パラメーター

DNS パラメーターを使用して、designate サービスを変更できます。

パラメーター	説明
DesignateBindQueryLogging	BIND でのクエリーのロギングを有効にするには true に設定します。デフォルト値は false です。
DesignateManagedResourceEmail	ゾーン SOA に設定するメールアドレスを設定します。設定しない場合、サービスのデフォルトが使用されます。
DesignateMdnsProxyBasePort	外部/パブリックアクセスネットワーク上の MiniDNS プロキシエンドポイントのベースポートを設定します。デフォルト値は 16000 です。
DesignateMinTTL	許可される最小の TTL を秒単位で設定します。デフォルト値は 0 で、パラメーターは未設定のままになります。デフォルト値は 0 です。
DesignateWorkers	Designate サービスのワーカー数。デフォルト値は 0 です。
UnboundAllowedCIDRs	Unbound 経由でクエリーを実行できる CIDR のリスト。例: [192.0.2.0/24, 198.51.100.0/24]。
UnboundAllowRecursion	false の場合、Unbound は要求を再帰的に解決しようとしません。ローカル情報を使用したクエリーのみ応答します。デフォルト値は true です。
UnboundDesignateIntegration	デプロイ済みのバインドされていないサーバーをデフォルトリゾルバーとして使用して neutron の設定を無効にするには、false に設定します。デフォルト値は true です。
UnboundForwardFallback	true の場合、転送されたクエリーが SERVFAIL を受信すると、Unbound は標準の再帰的解決としてリクエストを処理します。デフォルト値は true です。
UnboundForwardResolvers	Unbound に応答がない場合に Unbound が解決リクエストを転送する DNS リゾルバー IP アドレスのリスト (オプションのポートを含む)。例: [192.0.2.10, 192.0.2.20@53]。
UnboundLogQueries	true の場合、Unbound はクエリーリクエストをログに記録します。デフォルト値は false です。

パラメーター	説明
UnboundSecurityHarden	true の場合、Unbound は Unbound サービスのセキュリティに影響を与える可能性がある特定のクエリーをブロックします。デフォルト値は true です。

第7章 BARE METAL (IRONIC) パラメーター

ベアメタルパラメーターを使用して ironic サービスを変更できます。

パラメーター	説明
AdditionalArchitectures	有効にする追加のアーキテクチャーのリスト
ApacheCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
ApacheTimeout	Apache のタイムアウト (秒単位)。Apache が I/O 操作を待機する時間を定義します。デフォルト値は 90 です。
CertificateKeySize	証明書の作成時に使用される秘密鍵のサイズを指定します。デフォルト値は 2048 です。
EnableSQLAlchemyCollectd	SQLAlchemy-collectd サーバープラグインを有効にするには true に設定します。デフォルト値は false です。
IPALmageURLs	IPA イメージの URL。形式は ["http://path/to/kernel", "http://path/to/ramdisk"] とする必要があります。
IronicAutomatedClean	自動消去を有効または無効にします。自動消去を無効にすると、セキュリティ上の問題が生じ、再ビルド時にデプロイメントに失敗する場合があります。この機能の無効化がもたらす影響を理解した上で、このパラメーターを False に設定してください。デフォルト値は true です。
IronicCleaningDiskErase	初回のデプロイメントおよび再デプロイメントの前に行うディスク消去の種類。完全に消去するには full を指定します。ディスクの metadata (パーティションテーブル) のみをクリーンアップするメタデータ。デフォルト値は full です。
IronicCleaningNetwork	ベアメタルノードの消去に使用するオーバークラウドネットワークの名前または UUID。初期のデプロイメント時 (ネットワークがまだ作成されていない時点) には provisioning に設定し、デプロイ後のスタック更新時に実際の UUID に変更します。デフォルト値は provisioning です。
IronicConductorGroup	OpenStack Bare Metal (ironic) Conductor グループの名前

パラメーター	説明
IronicConfigureSwiftTempUrlKey	direct および ansible デプロイインターフェイスで使用する Swift の一時 URL を設定するかどうかを定義します。デフォルト値は true です。
IronicCorsAllowedOrigin	このリソースがリクエストの元ヘッダーで受け取るドメインと共有されるかどうかを示します。
IronicDefaultBootInterface	デフォルトで使用するブートインターフェイスの実装。何も設定しない場合は空のままにします。ハードウェアタイプが設定されたブートインターフェイスに対応していない場合は、機能しない可能性があります。これは、作成時のデフォルトをオーバーライドします。通常の場合では、有効なブートインターフェイスとハードウェアタイプの順序付けられた結合によりデフォルトが決まります。
IronicDefaultBootMode	ノードの driver_info、ケイパビリティ、または "instance_info" 設定でブートモードが明示的に要求されない場合に使用するデフォルトのブートモード。bios または uefi のいずれか。デフォルト値は uefi です。
IronicDefaultBootOption	ベアメタルインスタンスのブート方法。ローカルのブートローダーを使用するには local に設定します (パーティションイメージには grub2 が必要です)。PXE/iPXE を使用してコントローラーからインスタンスをブートするようにするには、 netboot に設定します。デフォルト値は local です。
IronicDefaultDeployInterface	デフォルトで使用するデプロイインターフェイスの実装。ハードウェア種別のデフォルトを使用する場合には、空白のままにします。
IronicDefaultInspectInterface	デフォルトで使用する検査インターフェイスの実装。ハードウェア種別のデフォルトを使用する場合には、空白のままにします。
IronicDefaultNetworkInterface	デフォルトで使用されるネットワークインターフェイスの実装。フラットなプロバイダーネットワークを使用するには、 flat に設定します。他のネットワーク種別と特定の高度なネットワーク機能を有効にするために OpenStack Bare Metal (ironic) が OpenStack Networking (neutron) ML2 ドライバーと対話するようにするには、 neutron に設定します。 IronicProvisioningNetwork を正しく設定する必要があります。デフォルト値は flat です。

パラメーター	説明
IronicDefaultRescueInterface	使用するデフォルトレスキューの実装。agent レスキューには互換性のある RAM ディスクを使用する必要があります。デフォルト値は agent です。
IronicDefaultResourceClass	新しいノードで使用するデフォルトのリソースクラス
IronicDeployLogsStorageBackend	RAM ディスクログの保管に使用するバックエンドで、local または swift のいずれかです。デフォルト値は local です。
IronicDhcpv6StatefulAddressCount	DHCPv6 ステートフルネットワークでのプロビジョニング、クリーニング、レスキュー、または検査用に作成されるポートに割り当てる IPv6 アドレスの数。チェーンローディングプロセスのステージごとに、異なる CLID/IAID のアドレスが必要です。同一ではない識別子のために、ブートプロセスの各ステップが正常にアドレスをリースできるようにするために、ホスト用に複数のアドレスを確保する必要があります。デフォルト値は 4 です。
IronicEnabledBiosInterfaces	有効にする BIOS インターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['no-bios'] です。
IronicEnabledBootInterfaces	有効にするブートインターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['ipxe', 'pxe'] です。
IronicEnabledConsoleInterfaces	有効にするコンソールインターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['ipmitool-socat', 'no-console'] です。
IronicEnabledDeployInterfaces	有効にするデプロイインターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['direct'] です。
IronicEnabledHardwareTypes	有効にする OpenStack Bare Metal (ironic) のハードウェア種別。デフォルト値は ['ipmi', 'redfish'] です。

パラメーター	説明
IronicEnabledInspectInterfaces	有効にする検査インターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['no-inspect'] です。
IronicEnabledManagementInterfaces	有効にする管理インターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['ipmitool', 'noop', 'redfish'] です。
IronicEnabledNetworkInterfaces	有効にするネットワークインターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['flat', 'neutron'] です。
IronicEnabledPowerInterfaces	有効にする電源インターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['ipmitool', 'redfish'] です。
IronicEnabledRaidInterfaces	有効にする RAID インターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['no-raid', 'agent'] です。
IronicEnabledRescueInterfaces	有効にするレスキューインターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['no-rescue', 'agent'] です。
IronicEnabledStorageInterfaces	有効にするストレージインターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['cinder', 'noop'] です。
IronicEnabledVendorInterfaces	有効にするベンダーインターフェイスの実装。各ハードウェア種別に対して、少なくとも1つ適切な実装を有効にする必要があります。デフォルト値は ['ipmitool', 'no-vendor'] です。
IronicEnableNovaPowerNotifications	nova への電源状態変更コールバックのサポートを有効化/無効化します。デフォルト値は true です。
IronicEnableStagingDrivers	ステージングドライバーの使用を有効にするかどうか。デフォルト値は false です。
IronicForcePowerStateDuringSync	同期中、電源状態を強制的に変更するかどうか。デフォルト値は true です。

パラメーター	説明
IronicImageDownloadSource	direct デプロイインターフェイスのイメージ配信メソッド。Object Storage の一時 URL の場合には swift を使用し、ローカルの HTTP サーバー (iPXE 用と同じ) の場合には http を使用します。デフォルト値は http です。
IronicInspectorCollectors	IPA 検査コレクターのコンマ区切りリスト。デフォルト値は default,logs です。
IronicInspectorDiscoveryDefaultDriver	新規検出されたノードに使用するデフォルトのドライバー (IronicInspectorEnableNodeDiscovery が True に設定されている必要があります)。このドライバーは、enabled_drivers に自動的に追加されます。デフォルト値は ipmi です。
IronicInspectorEnableNodeDiscovery	OpenStack Bare Metal (ironic) でイントロスペクションの RAM ディスクを PXE ブートする未知のノードを ironic-inspector が登録するようにします。新規ノードに使用するデフォルトのドライバーは IronicInspectorDiscoveryDefaultDriver パラメーターによって指定されます。このドライバーは、イントロスペクションのルールを使用して指定することもできます。デフォルト値は false です。
IronicInspectorExtraProcessingHooks	デフォルトリストに追記するプロセスフックのコンマ区切りリスト。デフォルト値は extra_hardware,lldp_basic,local_link_connection,physnet_cidr_map です。
IronicInspectorInterface	検査の dnsmasq がリスンするネットワークインターフェイス。デフォルト値は br-ex です。
IronicInspectorIpRange	検査プロセス中にノードに提供される一時的な IP 範囲。これは、OpenStack Networking (neutron) DHCP によって割り当てられる範囲と重なってはいけません。が、 ironic-inspector に戻るようルーティングする必要があります。このオプションには有意なデフォルトがないため、設定は必須です。
IronicInspectorIPXEEnabled	検査に iPXE を使用するかどうか。デフォルト値は true です。
IronicInspectorKernelArgs	OpenStack Bare Metal (ironic) インспекターのカーネル引数。デフォルト値は ipa-inspection-dhcp-all-interfaces=1 ipa-collect-lldp=1 ipa-debug=1 です。
IronicInspectorStorageBackend	イントロスペクションデータを保存するためのストレージバックエンド。デフォルト値は none です。

パラメーター	説明
IronicInspectorSubnets	検査プロセス中にノードに提供される一時的な IP 範囲。これらの範囲は、OpenStack Networking (neutron) DHCP によって提供される範囲と重なってはなりません、 ironic-inspector API に戻るようにルーティングする必要があります。このオプションには有意なデフォルトがなく、設定は必須です。
IronicInspectorUseSwift	非推奨: 代わりに IronicInspectorStorageBackend を使用してください。イントロスペクションデータの保管に Swift を使用するかどうか。デフォルト値は true です。
IronicIpVersion	PXE ブートに使用する IP バージョン。デフォルト値は 4 です。
IronicIPXEEnabled	デプロイメントに PXE の代わりに iPXE を使用するかどうか。デフォルト値は true です。
IronicIPXEPort	iPXE の使用時にイメージの提供に使用するポート。デフォルト値は 8088 です。
IronicIPXETimeout	IPXE のタイムアウト (秒単位)。タイムアウトを無効にするには 0 に設定します。デフォルト値は 60 です。
IronicIPXEUefiSnponly	SNP (Simple Network Protocol) iPXE EFI を使用するかどうか。true に設定すると ipxe-snponly EFI が使用されます。デフォルト値は true です。
IronicPassword	Bare Metal サービスおよびデータベースアカウントのパスワード
IronicPowerStateChangeTimeout	電源操作が完了するのを待機する秒数。つまり、ベアメタルノードが目的の電源状態になるまで待機します。タイムアウトとなると、電源操作は障害と見なされます。デフォルト値は 60 です。

パラメーター	説明
IronicProvisioningNetwork	IronicDefaultNetworkInterface が neutron に設定されている場合に、ベアメタルノードのプロビジョニングに使用されるオーバークラウドネットワークの名前または UUID。初期のデプロイメント時(ネットワークがまだ作成されていない時点)には provisioning に設定し、デプロイ後のスタック更新時に実際の UUID に変更します。デフォルト値は provisioning です。
IronicRescuingNetwork	IronicDefaultRescueInterface が no-rescue に設定されていない場合に、ベアメタルノードのレスキューに使用される オーバークラウド ネットワークの名前または UUID。デフォルト値の provisioning は、初期のデプロイメント中(ネットワークがまだ作成されていない間)はそのまま使用することができますが、デプロイメント後のスタックの更新時には実際の UUID に変更する必要があります。デフォルト値は provisioning です。
IronicRpcTransport	メッセージングブローカーや JSON RPC など、コンダクターと API プロセス間のリモートプロシージャコールのトランスポート
MemcacheUseAdvancedPool	高度な (eventlet に対して安全な) memcached クライアントプールを使用します。デフォルト値は true です。
PortPhysnetCidrMap	IP サブネット CIDR から物理ネットワークへのマッピング。physnet_cidr_map 処理フックが有効になっている場合、ベアメタルポートの physical_network プロパティは、このマッピングに基づいて入力されます。例: PortPhysnetCidrMap: 10.10.10.0/24: physnet_a 2001:db8::/64: physnet_b.

第8章 BLOCK STORAGE (CINDER) パラメーター

ブロックストレージパラメーターを使用して cinder サービスを変更できます。

パラメーター	説明
ApacheCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
ApacheTimeout	Apache のタイムアウト (秒単位)。Apache が I/O 操作を待機する時間を定義します。デフォルト値は 90 です。
AuthCloudName	認証に使用する clouds.yaml のエントリ。
CephClusterFSID	Ceph クラスタ FSID。UUID でなければなりません。
CephClusterName	Ceph クラスタ名。デフォルト値は ceph です。
CephConfigPath	Ceph クラスタ設定ファイルがホストに保存されているパス。デフォルト値は /var/lib/tripleo-config/ceph です。

パラメーター	説明
CephExternalMultiConfig	<p>追加の外部 Ceph クラスターを設定する際に適用される追加のオーバーライドを記述するマッピングのリスト。このリストが空でない場合には、初回の実行と同じパラメーターを使用して count(list) の回数 ceph-ansible が追加を実行します。ただし、各マッピング内のそれぞれのパラメーターがデフォルト値を上書きします。以下の設定を使用した場合、2 回目の実行でも以前とすべて同じパラメーターで ceph2 クラスターを使用するようにオーバークラウドが設定されます。ただし、/etc/ceph/ceph2.conf の mon_host エントリーには、デフォルトの CephExternalMonHost ではなく以下の external_cluster_mon_ips の値が含まれます。以降の ceph-ansible の実行は、Ceph クライアントだけに制限されます。CephExternalMultiConfig を使用して、OpenStack Orchestration (heat) スタック内に追加の内部 Ceph クラスターをデプロイすることはできません。各リストのマッピングには、tripleo-heat-template パラメーターではなく ceph-ansible パラメーターが含まれている必要があります。 - cluster: ceph2 fsid: e2cba068-5f14-4b0f-b047-acf375c0004a external_cluster_mon_ips: 172.18.0.5,172.18.0.6,172.18.0.7 keys: - name: "client.openstack" caps: mgr: "allow *" mon: "profile rbd" osd: "osd: profile rbd pool=volumes, profile rbd pool=backups, profile rbd pool=vms, profile rbd pool=images" key: "AQCwmeRcAAAAABAA6SQU/bGqFjlfLro5KxrB1Q=" mode: "0600" dashboard_enabled: false</p>
CertificateKeySize	<p>証明書の作成時に使用される秘密鍵のサイズを指定します。デフォルト値は 2048 です。</p>
CinderApiWsgiTimeout	<p>OpenStack Block Storage (cinder) API WSGI の接続がタイムアウトするまでの秒数。デフォルト値は 60 です。</p>
CinderBackupOptVolumes	<p>マウントされるオプションのボリュームのリスト</p>
CinderCronDbPurgeAge	<p>cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをページする (Age)。デフォルト値は 30 です。</p>
CinderCronDbPurgeDestination	<p>cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをページする (Log destination)。デフォルト値は /var/log/cinder/cinder-rowsflush.log です。</p>

パラメーター	説明
CinderCronDbPurgeHour	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Hour)。デフォルト値は 0 です。
CinderCronDbPurgeMaxDelay	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Max Delay)。デフォルト値は 3600 です。
CinderCronDbPurgeMinute	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Minute)。デフォルト値は 1 です。
CinderCronDbPurgeMonth	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Month)。デフォルト値は * です。
CinderCronDbPurgeMonthday	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Month Day)。デフォルト値は * です。
CinderCronDbPurgeUser	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (User)。デフォルト値は cinder です。
CinderCronDbPurgeWeekday	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Week Day)。デフォルト値は * です。
CinderDefaultVolumeType	OpenStack Block Storage (cinder) のデフォルトボリュームタイプの名前。デフォルト値は tripleo です。
CinderEnableDBPurge	OpenStack Block Storage (cinder) データベースでソフト削除された行をパージする cron ジョブを作成するかどうか。デフォルト値は true です。
CinderEnableiscsiBackend	OpenStack Block Storage (cinder) の iSCSI バックエンドを有効にするかどうか。デフォルト値は true です。
CinderEnableNfsBackend	OpenStack Block Storage (cinder) の NFS バックエンドを有効にするかどうか。デフォルト値は false です。
CinderEnableNVMeOFBackend	OpenStack Block Storage (cinder) の NVMeOF バックエンドを有効にするかどうか。デフォルト値は false です。

パラメーター	説明
CinderEnableRbdBackend	OpenStack Block Storage (cinder) の Rbd バックエンドを有効にするかどうか。デフォルト値は false です。
CinderEtcdLocalConnect	OpenStack Block Storage (cinder) を A/A で実行する際に、Etcd ネットワークのローカル IP を使用して Etcd に接続するかどうかを定義します。true に設定すると、ローカルノード上の IP が使用されます。false に設定すると、Etcd ネットワークの仮想 IP が使用されます。デフォルトは false です。デフォルト値は false です。
CinderImageConversionNfsOptions	OpenStack Block Storage (cinder) イメージ変換ディレクトリー用に NFS 共有を使用する場合の NFS マウントオプション。デフォルト値は _netdev,bg,intr,context=system_u:object_r:ccontainer_file_t:s0 です。
CinderImageConversionNfsShare	設定すると、OpenStack Block Storage (cinder) イメージ変換ディレクトリーに NFS 共有が使用されます。
CinderISCSIAvailabilityZone	OpenStack Block Storage (cinder) iSCSI バックエンドのアベイラビリティゾーン。このパラメーターを設定すると、デフォルトの CinderStorageAvailabilityZone がオーバーライドされます。
CinderISCSIHelper	cinder で使用する iSCSI ヘルパー。デフォルト値は lioadm です。
CinderISCSIProtocol	iSCSI に TCP (iscsi) または iSER RDMA (iser) を使用するかどうかを定義します。デフォルト値は iscsi です。
CinderLVMConfigureVG	LVM ドライバーが使用する cinder-volumes VG を設定するかどうか。無効にすると (False に設定すると)、 cinder-volumes VG が他の方法で作成されていれば、LVM ドライバーは機能します。デフォルト値は true です。
CinderLVMLoopDeviceSize	cinder LVM ドライバーが使用するループバックファイルのサイズ。デフォルト値は 10280 です。

パラメーター	説明
CinderNasSecureFileOperations	セキュリティが強化された NFS ファイルの操作を有効にするかどうかを制御します。有効な値は auto 、 true 、または false です。CinderEnableNfsBackend が true の場合に有効です。デフォルト値は false です。
CinderNasSecureFilePermissions	セキュリティが強化された NFS ファイルへのアクセス権限を有効にするかどうかを制御します。有効な値は auto 、 true 、または false です。CinderEnableNfsBackend が true の場合に有効です。デフォルト値は false です。
CinderNfsAvailabilityZone	OpenStack Block Storage (cinder) NFS バックエンドのアベイラビリティゾーン。このパラメーターを設定すると、デフォルトの CinderStorageAvailabilityZone がオーバーライドされます。
CinderNfsMountOptions	OpenStack Block Storage (cinder) NFS バックエンドが使用する NFS マウント用のマウントオプション。CinderEnableNfsBackend が true の場合に有効です。デフォルト値は context=system_u:object_r:container_file_t:s0 です。
CinderNfsServers	OpenStack Block Storage (cinder) NFS バックエンドが使用する NFS サーバー。CinderEnableNfsBackend が true の場合に有効です。
CinderNfsSnapshotSupport	NFS ドライバーのスナップショットのサポートを有効にするかどうかを定義します。CinderEnableNfsBackend が true の場合に有効です。デフォルト値は false です。
CinderNVMeOFAvailabilityZone	NVMeOF OpenStack Block Storage (cinder) NFS バックエンドのアベイラビリティゾーン。このパラメーターを設定すると、デフォルトの CinderStorageAvailabilityZone がオーバーライドされます。
CinderNVMeOFTargetProtocol	ターゲットプロトコル。サポートされる値は nvmet_rdma および nvmet_tcp です。デフォルト値は nvmet_rdma です。
CinderPassword	cinder サービスおよびデータベースアカウントのパスワード

パラメーター	説明
CinderRbdAvailabilityZone	OpenStack Block Storage (cinder) RBD バックエンドのアベイラビリティゾーン。このパラメーターを設定すると、デフォルトの CinderStorageAvailabilityZone がオーバーライドされます。
CinderRbdBackendName	OpenStack Block Storage (cinder) RBD バックエンド名のリスト。デフォルト値は tripleo_ceph です。
CinderRbdExtraPools	OpenStack Block Storage (cinder) の RBD バックエンドに使用する追加の Ceph プールのリスト。追加の OpenStack Block Storage (cinder) RBD バックエンドドライバーは、このリストの各プールに作成されます。これは、CinderRbdPoolName に関連付けられた標準の RBD バックエンドドライバーに追加されます。
CinderRbdFlattenVolumeFromSnapshot	スナップショットの依存関係を削除するために、スナップショットから作成された RBD ボリュームをフラット化する必要があるかどうかを定義します。デフォルト値は false です。
CinderRbdMultiConfig	複数の RBD バックエンドを設定する際の設定のディクショナリー。ハッシュキーはバックエンドの名前で、値はそのバックエンドに固有のパラメーター値のディクショナリーです。以下のパラメーターが必須であり、CephExternalMultiConfig で定義された対応する値と一致する必要があります。 CephClusterName (CephExternalMultiConfig エントリーの cluster と一致する必要があります) CephClusterFSID (CephExternalMultiConfig エントリーの fsid と一致する必要があります)。以下のパラメーターはオプションであり、対応するパラメーターのデフォルト値をオーバーライドします。 CephClientUserName CinderRbdPoolName CinderRbdExtraPools CinderRbdAvailabilityZone CinderRbdFlattenVolumeFromSnapshot.
CinderRbdPoolName	cinder ボリュームに使用する Ceph プール。デフォルト値は volumes です。
CinderRpcResponseTimeout	cinder の RPC 応答のタイムアウト (秒単位)。デフォルト値は 60 です。
CinderStorageAvailabilityZone	OpenStack Block Storage (cinder) サービスのストレージアベイラビリティゾーン。デフォルト値は nova です。

パラメーター	説明
CinderVolumeCluster	cinder-volume サービスを active-active (A/A) 設定でデプロイするために使用するクラスター名。この設定は、OpenStack Block Storage (cinder) バックエンドドライバーが A/A 設定をサポートすること、および cinder-volume サービスが Pacemaker により管理されないことが条件です。これらの条件が満たされない場合には、クラスター名を空白のままにする必要があります。
CinderVolumeOptEnvVars	オプションの環境変数のハッシュ
CinderVolumeOptVolumes	マウントされるオプションのボリュームのリスト
CinderWorkers	Block Storage サービスのワーカー数を設定します。ワーカー数が多いと、システムのプロセス数も増え、メモリーを過剰消費してしまう点に注意してください。CPU コア数が多いシステムでは、デフォルト以外の適切な値を選択することを推奨します。 0 の場合は、OpenStack の内部デフォルト値 (ノードにある CPU コア数) に設定されます。デフォルト値は、物理ノードにある仮想 CPU コア数と同じ数値です。
ClusterCommonTag	false に設定すると、Pacemaker サービスはコンテナイメージ名にフローティングタグを使用するように設定されます (例: REGISTRY/NAMESPACE/IMAGENAME:pcmklates)。true に設定すると、サービスは、 cluster.common.tag/IMAGENAME:pcmklatest などのフローティング接頭辞も使用します。デフォルト値は false です。
ClusterFullTag	true に設定すると、Pacemaker サービスはコンテナイメージ名に完全定数タグを使用します (例: cluster.common.tag/SERVICENAME:pcmklatest)。デフォルト値は true です。
DockerCinderVolumeUlimit	OpenStack Block Storage (cinder) ボリュームコンテナのユーザーリミット。デフォルト値は ['nofile=131072'] です。
EnableEtcdInternalTLS	残りの内部 API ネットワークが TLS を使用している場合でも etcd および cinder-volume サービスが cinder のロックマネージャーに TLS を使用するかどうかを制御します。デフォルト値は true です。
EnableSQLAlchemyCollectd	SQLAlchemy-collectd サーバープラグインを有効にするには true に設定します。デフォルト値は false です。

パラメーター	説明
MemcacheUseAdvancedPool	高度な (eventlet に対して安全な) memcached クライアントプールを使用します。デフォルト値は true です。
MultipathdEnable	マルチパスデーモンを有効にするかどうかを定義します。デフォルト値は false です。
NotificationDriver	通知の送信を処理する単一または複数のドライバー。デフォルト値は noop です。

第9章 CEPH STORAGE パラメーター

Ceph Storage パラメーターを使用して Ceph Storage クラスターを変更できます。

パラメーター	説明
CephAnsibleDisksConfig	ディスクの設定。デフォルト値は <code>{'devices': [], 'osd_scenario': 'lvm', 'osd_objectstore': 'bluestore'}</code> です。
CephAnsibleEnvironmentVariables	デフォルトをオーバーライドする Ansible 環境変数のマッピング
CephAnsibleExtraConfig	ceph-ansible Playbook の追加の変数
CephAnsiblePlaybook	実行する ceph-ansible Playbook へのパスのリスト。指定しない場合には、実施中の操作のタイプ (デプロイ/更新/アップグレード) に応じて Playbook が自動的に決定されます。デフォルト値は <code>['default']</code> です。
CephAnsiblePlaybookVerbosity	ansible-playbook コマンドに渡される、 <code>-v</code> 、 <code>-vv</code> 等に対応する数値。デフォルト値は 1 です。
CephAnsibleRepo	正しい ceph-ansible パッケージをインストールするのに使用すべきリポジトリ。この値を <code>tripleo-validations</code> で使用して、正しい ceph-ansible バージョンがインストールされていることを再確認することができます。デフォルト値は centos-ceph-nautilus です。
CephAnsibleSkipClient	このブール値 (<code>true</code> の場合) は、 ceph-ansible タグの <code>ceph_client</code> を <code>--skip-tags</code> リストに追加することで、 ceph-ansible クライアントロールの実行を防ぎます。デフォルト値は true です。
CephAnsibleSkipTags	スキップする ceph-ansible タグのリスト。デフォルト値は package-install,with_pkg です。
CephAnsibleWarning	特定のシナリオでは、警告は表示するがエラーにはしないためにこの検証パラメーターが必要です。パッケージはシステムにインストールされているがレポジトリが無効なためです。デフォルト値は true です。
CephCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。

パラメーター	説明
CephClientKey	Ceph クライアントのキー。現在は、外部の Ceph デプロイメントでの OpenStack ユーザーキーリング作成のみに使用されています。 ceph-authtool --gen-print-key で作成することができます。
CephClusterFSID	Ceph クラスタ FSID。UUID でなければなりません。
CephClusterName	Ceph クラスタ名。デフォルト値は ceph です。
CephConfigOverrides	ceph.conf にダンプする追加の設定
CephConfigPath	Ceph クラスタ設定ファイルがホストに保存されているパス。デフォルト値は /var/lib/tripleo-config/ceph です。
CephDashboardAdminPassword	ダッシュボードコンポーネントの管理者パスワード
CephDashboardAdminRO	読み取り専用の管理ユーザーを設定するのに使用されるパラメーター。デフォルト値は true です。
CephDashboardAdminUser	ダッシュボードコンポーネントの管理ユーザー。デフォルト値は admin です。
CephDashboardPort	ceph ダッシュボードポートを定義するパラメーター。デフォルト値は 8444 です。
CephEnableDashboard	ダッシュボードのデプロイメントのトリガーに使用されるパラメーター。デフォルト値は false です。
CephExternalMonHost	外部で管理される Ceph Mon Host IP のリスト。外部の Ceph デプロイメントにのみ使用されます。

パラメーター	説明
CephExternalMultiConfig	<p>追加の外部 Ceph クラスターを設定する際に適用される追加のオーバーライドを記述するマッピングのリスト。このリストが空でない場合には、初回の実行と同じパラメーターを使用して count(list) の回数 ceph-ansible が追加を実行します。ただし、各マッピング内のそれぞれのパラメーターがデフォルト値を上書きします。以下の設定を使用した場合、2 回目の実行でも以前とすべて同じパラメーターで ceph2 クラスターを使用するようにオーバークラウドが設定されます。ただし、/etc/ceph/ceph2.conf の mon_host エントリーには、デフォルトの CephExternalMonHost ではなく以下の external_cluster_mon_ips の値が含まれます。以降の ceph-ansible の実行は、Ceph クライアントだけに制限されます。CephExternalMultiConfig を使用して、OpenStack Orchestration (heat) スタック内に追加の内部 Ceph クラスターをデプロイすることはできません。各リストのマッピングには、tripleo-heat-template パラメーターではなく ceph-ansible パラメーターが含まれている必要があります。-</p> <pre>cluster: ceph2 fsid: e2cba068-5f14-4b0f-b047-acf375c0004a external_cluster_mon_ips: 172.18.0.5,172.18.0.6,172.18.0.7 keys: - name: "client.openstack" caps: mgr: "allow *" mon: "profile rbd" osd: "osd: profile rbd pool=volumes, profile rbd pool=backups, profile rbd pool=vms, profile rbd pool=images" key: "AQCwmeRcAAAAABAA6SQU/bGqFjlfLro5KxrB1Q==" mode: "0600" dashboard_enabled: false</pre>
CephExtraKeys	<p>デプロイされた Ceph クラスター上に作成される追加の鍵を記述するマッピングのリスト。ceph-ansible/library/ceph_key.py Ansible モジュールを使用します。リストの各項目は次に例示する形式でなければなりません (例: name: "client.glance" caps: mgr: "allow *" mon: "profile rbd" osd: "profile rbd pool=images" key: "AQBRgQ9eAAAAABAav84zEilJYZPNuJ0lwn9Ndg==" mode: "0600")。</p>
CephGrafanaAdminPassword	grafana コンポーネントの管理者パスワード
CephIPv6	IPv6 アドレスにバインドする Ceph デーモンを有効にします。デフォルト値は False です。
CephManilaClientKey	Ceph クライアントのキー。 ceph-authtool --gen-print-key で作成することができます。

パラメーター	説明
CephMsgrSecureMode	Ceph msgr2 セキュアモードを有効にし、Ceph デーモン間および Ceph クライアントとデーモン間の伝送時暗号化を可能にします。デフォルト値は false です。
CephOsdPercentageMin	デプロイメントの失敗を防ぐために、Ceph クラスター内で実行中でなければならない最低限の Ceph OSD パーセンテージ (ceph osd stat により確認する)。デプロイメントのエラーを早期に把握するために使用されます。この確認を無効にするには、この値を 0 に設定します。 ceph-ansible から cephadm に移行したため、Walaby では非推奨になりました。cephadm は OSD を帯域外で起動するだけで、起動待機中はデプロイメントがブロックされないため、これを行うことはできなくなりました。デフォルト値は 0 です。
CephPoolDefaultPgNum	RBD プールに使用するデフォルトの配置グループサイズ。デフォルトは 16 です。
CephPoolDefaultSize	デフォルトで適用される RBD コピーの最小レプリカ数。デフォルト値は 3 です。
CephPools	事前定義されたプールの 1 つの設定をオーバーライドするか、追加の設定を作成します。(例: { "volumes": { "size": 5, "pg_num": 128, "pgp_num": 128 } })}。
CephRbdMirrorConfigure	ローカルプールとリモートプール間で設定のミラーリングを実施します。デフォルト値は true です。
CephRbdMirrorCopyAdminKey	すべてのノードに管理者の鍵をコピーします。デフォルト値は false です。
CephRbdMirrorPool	リモートクラスターにミラーリングするローカルプールの名前
CephRbdMirrorRemoteCluster	ローカルクラスターからリモート Ceph クラスターに与えられる名前。鍵は /etc/ceph ディレクトリーにあります。デフォルト値は not-ceph です。
CephRbdMirrorRemoteUser	rbd-mirror デーモンでは、ユーザーはリモートクラスターとの認証が必要です。デフォルトでは、この鍵は /etc/ceph/<remote_cluster>.client.<remote_user>.keyring にあります。
CephRgwCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。

パラメーター	説明
CephRgwClientName	RADOSGW サービスのクライアント名。デフォルト値は radosgw です。
CephRgwKey	RADOSGW クライアントの cephx キー。ceph-authtool --gen-print-key で作成することができます。
CephValidationDelay	検証チェックの間隔 (秒単位)。デフォルト値は 30 です。
CephValidationRetries	Ceph の検証の再試行回数。デフォルト値は 40 です。
CertificateKeySize	証明書の作成時に使用される秘密鍵のサイズを指定します。デフォルト値は 2048 です。
CinderBackupBackend	使用する OpenStack Block Storage (cinder) バックアップバックエンドの省略名。デフォルト値は swift です。
CinderBackupRbdPoolName	Block Storage (cinder) のバックアップが有効な場合に使用するプール。デフォルト値は backups です。
CinderEnableRbdBackend	OpenStack Block Storage (cinder) の Rbd バックエンドを有効にするかどうか。デフォルト値は false です。
CinderRbdExtraPools	OpenStack Block Storage (cinder) の RBD バックエンドに使用する追加の Ceph プールのリスト。追加の OpenStack Block Storage (cinder) RBD バックエンドドライバーは、このリストの各プールに作成されます。これは、CinderRbdPoolName に関連付けられた標準の RBD バックエンドドライバーに追加されます。
CinderRbdPoolName	Block Storage (cinder) サービスに使用するプール。デフォルト値は volumes です。
DeploymentServerBlacklist	トリガーされたデプロイメントからブロックリスト登録するサーバーのホスト名のリスト。
GlanceBackend	使用する OpenStack Image Storage (glance) バックエンドの省略名。Ceph Storage を使用するには rbd に設定します。デフォルト値は swift です。

パラメーター	説明
GlanceMultistoreConfig	追加の glance バックエンドを設定する際の設定のディクショナリー。ハッシュキーはバックエンドの ID で、値はそのバックエンドに固有のパラメーター値のディクショナリーです。複数の rbd および cinder バックエンドを設定することが可能ですが、file および swift バックエンドはそれぞれ1つに制限されます。(例: # Default glance store is rbd.GlanceBackend: rbd GlanceStoreDescription: デフォルト rbd ストア # GlanceMultistoreConfig は 2 番目の rbd バックエンドおよび cinder # バックエンドを指定します。GlanceMultistoreConfig: rbd2_store: GlanceBackend: rbd GlanceStoreDescription: 2 番目の rbd ストア CephClusterName: ceph2 # オーバーライド CephClientUserName (このクラスターが別の # クライアント名を使用する場合)。CephClientUserName: client2 cinder1_store: GlanceBackend: cinder GlanceCinderVolumeType: volume-type-1 GlanceStoreDescription: First cinder store cinder2_store: GlanceBackend: cinder GlanceCinderVolumeType: volume-type-2 GlanceStoreDescription: Seconde cinder store.
GlanceRbdPoolName	Image Storage (glance) サービスに使用するプール。デフォルト値は images です。
GnocchiBackend	使用する OpenStack Telemetry Metrics (gnocchi) バックエンドの省略名。swift、rbd、file、または s3 のいずれかにする必要があります。デフォルト値は swift です。
GnocchiRbdPoolName	Telemetry ストレージに使用するプール。デフォルト値は metrics です。
LocalCephAnsibleFetchDirectoryBackup	ceph-ansible フェッチディレクトリーからのデータのコピーを保持するアンダークラウド上のファイルシステムへのパス。Swift の fetch_directory バックアップの代替として使用します。パスは、config-download から Ansible を実行するユーザーが読み取り/書き込み可能でなければなりません。たとえば、mistral-executor コンテナの mistral ユーザーは、/var/lib/mistral/ceph_fetch への読み取り/書き込みが可能です。
ManilaCephFSCephFSAuthId	Shared File System サービス (manila) の CephFS ユーザー ID。デフォルト値は manila です。
ManilaCephFSDataPoolName	ファイル共有のストレージに使用するプール。デフォルト値は manila_data です。

パラメーター	説明
ManilaCephFSMetadataPoolName	ファイル共有のメタデータストレージに使用するプール。デフォルト値は manila_metadata です。
ManilaCephFSShareBackendName	ファイル共有ストレージの CephFS ファイル共有のバックエンド名デフォルト値は cephfs です。
NodeExporterContainerImage	Ceph NodeExporter コンテナイメージ
NovaEnableRbdBackend	Compute (nova) 用の Ceph バックエンドを有効にするかどうかを定義します。デフォルト値は false です。
NovaRbdPoolName	Compute ストレージに使用するプール。デフォルト値は vms です。

第10章 COMPUTE (NOVA) パラメーター

Compute パラメーターを使用して nova サービスを変更できます。

パラメーター	説明
ApacheCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
ApacheTimeout	Apache のタイムアウト (秒単位)。Apache が I/O 操作を待機する時間を定義します。デフォルト値は 90 です。
AuthCloudName	認証に使用する clouds.yaml のエントリー。
CertificateKeySize	証明書の作成時に使用される秘密鍵のサイズを指定します。デフォルト値は 2048 です。
ContainerNovaLibvirtPidsLimit	nova_libvirt コンテナ PID のリミットを調整します (リミットを解除するには 0 に設定します) (デフォルトは 65536)。デフォルト値は 65536 です。
ContainerNovaLibvirtUlimit	OpenStack Compute (nova) Libvirt コンテナのユーザーリミット。デフォルト値は ['nofile=131072', 'nproc=126960'] です。
CustomProviderInventories	<p>Compute ロールのカスタムプロバイダーを記述するハッシュの配列。Format: 名前/uuid - 対象となるリソースプロバイダーは、UUID または名前で識別できます。さらに、値 \$COMPUTE_NODE を UUID フィールドで使用して、サービスによって管理されるすべてのノードを識別できます。uuid または名前のいずれかのみ必須です。uuid または名前の両方を指定すると、特別な uuid \$COMPUTE_NODE がテンプレートに設定されます。</p> <p>inventory - (オプション) カスタムプロバイダーインベントリーのハッシュ。total は必須プロパティです。設定されていないその他のオプションのプロパティには、デフォルト値が配置されます。既存のプロバイダーをオーバーライドすると、値は既存のインベントリーから保持されません。</p> <p>traits - (オプション) 追加の traits の配列。例: ComputeParameters: CustomProviderInventories: - uuid: \$COMPUTE_NODE inventories: CUSTOM_EXAMPLE_RESOURCE_CLASS: total: 100 reserved: 0 min_unit: 1 max_unit: 10 step_size: 1 allocation_ratio: 1.0 CUSTOM_ANOTHER_EXAMPLE_RESOURCE_CLASS: total: 100 traits: - CUSTOM_P_STATE_ENABLED - CUSTOM_C_STATE_ENABLED.</p>

パラメーター	説明
DockerNovaComputeUlimit	OpenStack Compute (nova) コンピュートコンテナのユーザーリミット。デフォルト値は ['nofile=131072', 'memlock=67108864'] です。
DockerNovaMigrationSshdPort	ドッカー nova 移行対象 sshd サービスがバインドするポート。デフォルト値は 2022 です。
EnableCache	memcached でキャッシュを有効にします。デフォルト値は true です。
EnableConfigPurge	director で生成されていない設定を削除します。アップグレード後に以前の設定が残らないようにするために使用します。デフォルト値は false です。
EnableInstanceHA	インスタンス HA の設定を有効にするかどうかを定義します。この設定には、Compute ロールへの PacemakerRemote サービスの追加が必要です。デフォルト値は false です。
EnableSQLAlchemyCollectd	SQLAlchemy-collectd サーバープラグインを有効にするには true に設定します。デフォルト値は false です。
EnforceSecureRbac	このオプションを True に設定すると、 oslo_policy enforce_new_defaults と oslo_policy enforce_scope を True に設定して、各 OpenStack サービスが Secure RBAC を適用するように設定されます。これにより、システムおよびプロジェクトスコープのサポートや keystone のデフォルトロール、管理者、メンバー、リーダーなど、OpenStack サービス全体で一貫性のある一連の RBAC ペルソナが導入されます。デプロイメント内のすべてのサービスがセキュアな RBAC を実際にサポートするまで、この機能は有効にしないでください。デフォルト値は false です。
GlanceBackendID	デフォルトのバックエンドの識別子。デフォルト値は default_backend です。

パラメーター	説明
GlanceMultistoreConfig	<p>追加の glance バックエンドを設定する際の設定のディクショナリー。ハッシュキーはバックエンドの ID で、値はそのバックエンドに固有のパラメーター値のディクショナリーです。複数の rbd および cinder バックエンドを設定することが可能ですが、file および swift バックエンドはそれぞれ1つに制限されます。(例: # Default glance store is rbd.GlanceBackend: rbd GlanceStoreDescription: デフォルト rbd ストア # GlanceMultistoreConfig は 2 番目の rbd バックエンドおよび cinder # バックエンドを指定します。GlanceMultistoreConfig: rbd2_store: GlanceBackend: rbd GlanceStoreDescription: 2 番目の rbd ストア CephClusterName: ceph2 # オーバーライド CephClientUserName (このクラスターが別の # クライアント名を使用する場合)。CephClientUserName: client2 cinder1_store: GlanceBackend: cinder GlanceCinderVolumeType: volume-type-1 GlanceStoreDescription: First cinder store cinder2_store: GlanceBackend: cinder GlanceCinderVolumeType: volume-type-2 GlanceStoreDescription: Seconde cinder store.</p>
InstanceNameTemplate	<p>インスタンス名の生成に使用するテンプレートの文字列。デフォルト値は instance-%08x です。</p>
InternalTLSVncProxyCAFile	<p>VNC TLS に使用する CA 証明書を指定します。デフォルト値は /etc/ipa/ca.crt です。</p>
KernelArgs	<p>ホストに適用するカーネル引数</p>
LibvirtCACert	<p>これは、libvirt 内の TLS に使用する CA 証明書を指定します。このファイルは、libvirt 内のデフォルト CA のパス (/etc/pki/CA/cacert.pem) へのシンボリックリンクです。libvirt のバックエンドである GNU TLS の制限により、ファイルは 65 K 以下にする必要がある点に注意してください(そのため、システムの CA バンドルは使用できません)。このパラメーターは、デフォルト (内部 TLSCAFile パラメーターがベース) が望ましくない場合に使用すべきです。現在のデフォルトは、TripleO のデフォルト CA である FreeIPA を反映しています。これは、内部の TLS が有効な場合にのみ使用されます。</p>
LibvirtCertificateKeySize	<p>このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。</p>

パラメーター	説明
LibvirtEnabledPerfEvents	これは、モニターとして使用することのできるパフォーマンスイベントリストです(例: cmt,mbml,mbmt)。ベースには Red Hat Enterprise Linux 7.4 を使用し、 libvirt バージョンは 1.3.3 以上であることを確認します。また、通知を有効にし、 cmt フラグをサポートする CPU を搭載したハードウェアを使用するようにしてください。
LibvirtLogFilters	libvirt デーモンでフィルターを定義して、カテゴリログ出力ごとに異なるログレベルを選択します(詳しくは https://libvirt.org/logging.html を参照)。デフォルト値は 1:libvirt 1:qemu 1:conf 1:security 3:event 3:json 3:file 3:object 1:util です。
LibvirtTLSPassword	TLS が有効な場合の libvirt サービスのパスワード
LibvirtTLSPriority	コンパイル時のデフォルト TLS 優先度文字列をオーバーライドします。デフォルト値は NORMAL:-VERS-SSL3.0:-VERS-TLS-ALL:+VERS-TLS1.2 です。
LibvirtVirtlogdLogFilters	virtlogd でフィルターを定義して、カテゴリログ出力ごとに異なるログレベルを選択します(詳しくは https://libvirt.org/logging.html を参照)。デフォルト値は 1:logging 4:object 4:json 4:event 1:util です。
LibvirtVirtnodedevdLogFilters	virtnodedevd でフィルターを定義して、カテゴリログ出力ごとに異なるログレベルを選択します(詳しくは https://libvirt.org/logging.html を参照)。デフォルト値は 1:qemu 1:libvirt 4:object 4:json 4:event 1:util です。
LibvirtVirtproxydLogFilters	virtproxyd でフィルターを定義して、カテゴリログ出力ごとに異なるログレベルを選択します(詳しくは https://libvirt.org/logging.html を参照)。デフォルト値は 1:qemu 1:libvirt 4:object 4:json 4:event 1:util です。
LibvirtVirtqemudLogFilters	virtqemud でフィルターを定義して、カテゴリログ出力ごとに異なるログレベルを選択します(詳しくは https://libvirt.org/logging.html を参照)。デフォルト値は 1:qemu 1:libvirt 4:object 4:json 4:event 1:util です。
LibvirtVirtsecretLogFilters	virtsecret でフィルターを定義して、カテゴリログ出力ごとに異なるログレベルを選択します(詳しくは https://libvirt.org/logging.html を参照)。デフォルト値は 1:qemu 1:libvirt 4:object 4:json 4:event 1:util です。

パラメーター	説明
LibvirtVirtstoredLogFilters	virtstored でフィルターを定義して、カテゴリログ出力ごとに異なるログレベルを選択します (詳しくは https://libvirt.org/logging.html を参照)。デフォルト値は 1:qemu 1:libvirt 4:object 4:json 4:event 1:util です。
LibvirtVncCACert	これは、VNC TLS に使用する CA 証明書を指定します。このファイルは、デフォルト CA パス (/etc/pki/CA/certs/vnc.crt) へのシンボリックリンクです。このパラメーターは、デフォルト (InternalTLSVncProxyCAFile パラメーターで指定されている) が望ましくない場合に使用すべきです。現在のデフォルトは、TripleO のデフォルト CA である FreeIPA を反映しています。これは、内部の TLS が有効な場合にのみ使用されます。
LibvirtVNCClientCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
MemcachedTLS	Memcached サービスで TLS を有効にするには、True に設定します。すべてのサービスが Memcached TLS をサポートする訳ではないので、移行期間中、Memcached は 2 つのポート (MemcachedPort パラメーター (上記) で設定したポート、および TLS を使用しない 11211) をリスンします。デフォルト値は false です。
MemcacheUseAdvancedPool	高度な (eventlet に対して安全な) memcached クライアントプールを使用します。デフォルト値は true です。
MigrationSshKey	移行用の SSH 鍵。public_key および private_key の鍵を持つディクショナリーが必要です。値は SSH 公開/秘密鍵ファイルと全く同じでなければなりません。デフォルト値は {'public_key': '', 'private_key': ''} です。
MigrationSshPort	ssh を介したマイグレーションのターゲットポート。デフォルト値は 2022 です。
MultipathdEnable	マルチパスデーモンを有効にするかどうかを定義します。デフォルト値は false です。
MysqllIPv6	MySQL で IPv6 を有効にします。デフォルト値は false です。
NeutronMetadataProxySharedSecret	スプーフィングを防ぐための共有シークレット

パラメーター	説明
NeutronPhysnetNUMANodesMapping	物理ネットワーク名(キー)と NUMA ノード(値)のマッピング(例: NeutronPhysnetNUMANodesMapping: {'foo': [0, 1], 'bar': [1]})。ここで、 foo および bar は物理ネットワークの名前で、これに対応する値は関連する numa_nodes のリストです。
NeutronTunnelNUMANodes	すべてのトンネル化ネットワークの NUMA アフィニティーを設定するのに使用します。
NotificationDriver	通知の送信を処理する単一または複数のドライバー。デフォルト値は noop です。
NovaAdditionalCell	これがデフォルトのセルに追加するセルかどうかを定義します。デフォルト値は false です。
NovaAllowResizeToSameHost	サイズ変更のために、宛先マシンがソースに一致することを許可します。デフォルト値は false です。
NovaApiHostStatusPolicy	NovaShowHostStatus パラメーターの設定値に対してカスタム API ポリシールールまたはロールを設定します。クラウドユーザーが Compute ホストステータスの設定されたレベルにアクセスするには、このロールが付与されている必要があり、そのためにはロールが RHOSP デプロイメントで作成されていなければなりません。デフォルト値は role:reader です。 NovaShowHostStatus パラメーターを unknown-only に設定すると、指定された NovaApiHostStatusPolicy ロールを持つユーザーは、設定されたしきい値内で heartbeat が受信されない場合に、制限付きホストステータス UNKNOWN を表示できます。 NovaShowHostStatus パラメーターを all に設定すると、指定されたロールを持つユーザーは、インスタンスの Compute ホストの UP 、 DOWN 、 MAINTENANCE 、 UNKNOWN 、および NONE ステータス表示できます。 NovaShowHostStatus を hidden に設定すると、指定されたロールを持つユーザーはインスタンスの Compute ホストのステータスを表示できません。RHOSP 管理者のみが Compute ホストのステータスを表示できます。この機能を使用するには、使用している RHOSP 環境で SRBAC が有効になっていることを確認してください。 openstack server list --long コマンドを使用する際にホストステータスを表示するには、python-openstackclient-6.0.0以降がインストールされている必要があります。デフォルト値は role:reader です。

パラメーター	説明
NovaApiMaxLimit	API クエリーごとに返されるオブジェクトの最大数。デフォルト値は 1000 です。
NovaAutoDisabling	nova-compute が自分自身を無効にするまでに許容される、連続したビルドエラー回数の最大値。デフォルト値は 10 です。
NovaComputeCpuDedicatedSet	<p>ピンニングされたインスタンス CPU のプロセスをスケジューリングできる物理ホスト CPU 番号のコンマ区切りリストまたは範囲。たとえば、NovaComputeCpuDedicatedSet: [4-12,^8,15] は、コア 4-12 の範囲 (ただし 8 を除く) および 15 を確保します。このオプションを設定する場合は、非推奨の NovaVcpuPinSet パラメーターを設定しないでください。</p>
NovaComputeCpuSharedSet	<p>非推奨の NovaVcpuPinSet オプションが設定されていない場合には、ピンニングされていないインスタンスをスケジューリングできるホスト CPU および共有エミュレータースレッドポリシー hw:emulator_threads_policy=share が設定されたインスタンスのエミュレータースレッドをオフロードすべきホスト CPU を決定できるように、NovaComputeCpuSharedSet に物理ホスト CPU 番号のコンマ区切りリストまたは範囲を設定して仮想 CPU インベントリを提供します。非推奨の NovaVcpuPinSet が設定されている場合には、共有エミュレータースレッドポリシー (hw:emulator_threads_policy=share) が設定されたインスタンスのエミュレータースレッドをオフロードすべきホスト CPU を決定できるように、NovaComputeCpuSharedSet にホスト CPU コアのリストまたは範囲を設定します。この場合、NovaVcpuPinSet は、ピンニングされたインスタンスとピンニングされていないインスタンスの両方をスケジューリングできるホスト CPU を決定できるように、仮想 CPU インベントリを提供するのに使用されます。たとえば、NovaComputeCpuSharedSet: [4-12,^8,15] は、コア 4-12 の範囲 (ただし 8 を除く) および 15 を確保します。</p>
NovaComputeDisableIrqBalance	コンピュートノードで irqbalance を無効にするかどうかを定義します。特に、リアルタイム Compute ロールでは無効のままにします。デフォルト値は false です。
NovaComputeEnableKsm	コンピュートノードで KSM を有効にするかどうかを定義します。特に NFV のユースケースでは、無効のままにします。デフォルト値は false です。

パラメーター	説明
NovaComputeForceRawImages	<p>raw 以外の形式でキャッシュされたベースイメージを raw 形式に変換するには、"True" に設定します。I/O が遅いシステムや空き容量が少ないシステムを使用している場合は、"False" に指定して、圧縮の際の CPU 要件を軽減することで入力の変換幅を最小限に抑えます。注記: CPU のボトルネックを防ぐために、Compute サービスは圧縮時にベースイメージからすべての圧縮を削除します。変換されたイメージには、セキュリティ上の問題となる可能性のあるバックアップファイルを含めることはできません。raw イメージ形式は、qcow2 などの他のイメージ形式よりも多くの領域を使用します。Raw ベースイメージは常に libvirt_images_type=lvms と併せて使用されます。デフォルト値は true です。</p>
NovaComputeImageCacheManagerInterval	<p>コンピュータノードでのベースイメージのキャッシュに影響を与えるイメージキャッシュマネージャーの実行間で待機する時間を秒単位で指定します。この期間は、remove_unused_base_images および remove_unused_original_minimum_age_seconds で設定された未使用のキャッシュイメージの自動削除に使用されます。</p> <p>デフォルトの間隔である 60 秒で実行するには、0 に設定します (推奨されません)。デフォルト値は 2400 です。</p>
NovaComputeImageCachePrecacheConcurrency	<p>イメージの事前キャッシュを同時にトリガーするコンピュータノードの最大数。イメージの事前キャッシュが要求されると、コンピュータノードはダウンロードの開始を指示されます。</p> <p>数値が大きいと、compute が並行して機能し、操作完了までの時間が短縮される可能性があります。Image サービスにも DDoS が実行される可能性があります。数値が小さいと、順次操作が実行され、イメージサービスの負荷は低くなりますが、完了までの実行時間が長くなる可能性が高くなります。デフォルト値は 1 です。</p>
NovaComputeImageCacheRemoveUnusedBasesImages	<p>image_cache_manager_interval を使用して設定された間隔で使用されていないベースイメージをキャッシュから自動的に削除するには、True に設定します。イメージは、remove_unused_original_minimum_age_seconds を使用して指定された期間アクセスされなかった場合に unused と定義されます。デフォルト値は true です。</p>

パラメーター	説明
NovaComputeImageCacheRemoveUnusedResizedMinimumAge	未使用のサイズ変更されたベースイメージをキャッシュから削除する最短の期間を指定します (秒単位)。未使用のサイズ変更されたベースイメージは、この期間削除されません。デフォルト値は 3600 です。
NovaComputeLibvirtPreAllocateImages	<p>libvirt インスタンスディスクに対する事前割り当てモードを指定します。次の有効な値のいずれかに設定します: - none - インスタンスの開始時にストレージはプロビジョニングされません。 - space - ストレージはインスタンスの開始時に <code>fallocate</code> を使用して完全に割り当てられます。これによりスペースが確保され、I/O パフォーマンスも向上します。</p> <p>CoW インスタンスディスクを使用しない場合でも、各インスタンスが取得するコピーはスパースであるため、インスタンスが ENOSPC でランタイムに予期せず失敗する可能性があります。インスタンスディスクに <code>fallocate(1)</code> を実行すると、Compute サービスは即時に、ファイルシステム内でイメージに領域を効率的に割り当てます (サポートされている場合)。ファイルシステムではランタイム時にブロックを動的に割り当てる必要がないため、ランタイムのパフォーマンスも向上し、CPU オーバーヘッドとファイルの断片化が軽減されます。デフォルト値は none です。</p>
NovaComputeLibvirtType	Libvirt ドメインの種別。デフォルトは kvm です。デフォルト値は kvm です。
NovaComputeOptEnvVars	オプションの環境変数のリスト
NovaComputeOptVolumes	オプションのボリュームのリスト
NovaComputeStartupDelay	コンピュータノードがブートした後に <code>nova-compute</code> サービスの起動を遅延します。これは、インスタンス起動前およびオーバークラウドの再起動後に、 <code>ceph</code> が正常に戻る機会を与えるためです。デフォルト値は 0 です。
NovaComputeUseCowImages	libvirt インスタンスディスクに <code>cqow2</code> 形式の CoW (Copy on Write) イメージを使用するには、 <code>True</code> に設定します。CoW を使用する場合には、バッキングストアとホストのキャッシュによっては、各インスタンスが独自のコピー上で稼働することで、並行処理が改善される場合があります。raw 形式を使用するには、 <code>False</code> に設定します。raw 形式は、ディスクイメージの共通の部分により多くの領域を使用します。デフォルト値は true です。

パラメーター	説明
NovaCPUAllocationRatio	物理 CPU に対する仮想 CPU の割当て比率。デフォルト値は 0.0 です。
NovaCronArchiveDeleteAllCells	すべてのセルから削除されたインスタンスをアーカイブします。デフォルト値は true です。
NovaCronArchiveDeleteRowsAge	cron ジョブが削除されたインスタンスをアーカイブする (Age)。削除されたインスタンスエントリをアーカイブする際の保持ポリシーを日数単位で定義します。0 は、シャドウテーブル内のその日以前のデータがアーカイブされることを意味します。デフォルト値は 90 です。デフォルト値は 90 です。
NovaCronArchiveDeleteRowsDestination	cron ジョブが削除されたインスタンスを別のテーブルに移動する (Log destination)。デフォルト値は /var/log/nova/nova-rowsflush.log です。
NovaCronArchiveDeleteRowsHour	cron ジョブが削除されたインスタンスを別のテーブルに移動する (Hour)。デフォルト値は 0 です。
NovaCronArchiveDeleteRowsMaxDelay	cron ジョブが削除されたインスタンスを別のテーブルに移動する (Max Delay)。デフォルト値は 3600 です。
NovaCronArchiveDeleteRowsMaxRows	cron ジョブが削除されたインスタンスを別のテーブルに移動する (Max Rows)。デフォルト値は 1000 です。
NovaCronArchiveDeleteRowsMinute	cron ジョブが削除されたインスタンスを別のテーブルに移動する (Minute)。デフォルト値は 1 です。
NovaCronArchiveDeleteRowsMonth	cron ジョブが削除されたインスタンスを別のテーブルに移動する (Month)。デフォルト値は * です。
NovaCronArchiveDeleteRowsMonthday	cron ジョブが削除されたインスタンスを別のテーブルに移動する (Month Day)。デフォルト値は * です。
NovaCronArchiveDeleteRowsPurge	スケジュールされたアーカイブの直後にシャドウテーブルをパージします。デフォルト値は false です。
NovaCronArchiveDeleteRowsUntilComplete	cron ジョブが削除されたインスタンスを別のテーブルに移動する (Until complete)。デフォルト値は true です。
NovaCronArchiveDeleteRowsUser	cron ジョブが削除されたインスタンスを別のテーブルに移動する (User)。デフォルト値は nova です。

パラメーター	説明
NovaCronArchiveDeleteRowsWeekday	cron ジョブが削除されたインスタンスを別のテーブルに移動する (Week Day)。デフォルト値は * です。
NovaCronPurgeShadowTablesAge	cron ジョブがシャドウテーブルをパージする (Age)。シャドウテーブルをパージする際の保持ポリシーを、日数単位で定義します。0 は、シャドウテーブル内のその日以前のデータがパージされることを意味します。デフォルト値は 14 です。
NovaCronPurgeShadowTablesAllCells	cron ジョブがシャドウテーブルをパージする (All cells)。デフォルト値は true です。
NovaCronPurgeShadowTablesDestination	cron ジョブがシャドウテーブルをパージする (Log destination)。デフォルト値は /var/log/nova/nova-rowspurge.log です。
NovaCronPurgeShadowTablesHour	cron ジョブがシャドウテーブルをパージする (Hour)。デフォルト値は 5 です。
NovaCronPurgeShadowTablesMaxDelay	cron ジョブがシャドウテーブルをパージする (Max Delay)。デフォルト値は 3600 です。
NovaCronPurgeShadowTablesMinute	cron ジョブがシャドウテーブルをパージする (Minute)。デフォルト値は 0 です。
NovaCronPurgeShadowTablesMonth	cron ジョブがシャドウテーブルをパージする (Month)。デフォルト値は * です。
NovaCronPurgeShadowTablesMonthday	cron ジョブがシャドウテーブルをパージする (Month Day)。デフォルト値は * です。
NovaCronPurgeShadowTablesUser	cron ジョブがシャドウテーブルをパージする (User)。デフォルト値は nova です。
NovaCronPurgeShadowTablesVerbose	cron ジョブがシャドウテーブルをパージする (Verbose)。デフォルト値は false です。
NovaCronPurgeShadowTablesWeekday	cron ジョブがシャドウテーブルをパージする (Week Day)。デフォルト値は * です。
NovaCrossAZAttach	インスタンスに異なるアベイラビリティゾーンからの cinder ボリュームをアタッチできるかどうかを定義します。デフォルト値は true です。
NovaDefaultFloatingPool	Floating IP アドレスのデフォルトプール。デフォルト値は public です。

パラメーター	説明
NovaDisableComputeServiceCheckForFfu	コンピュータノードがサービスレコードを更新する前に、新しいコントロールサービスが開始される Fast-Forward アップグレードを促進します。デフォルト値は false です。
NovaDisableImageDownloadToRbd	glance からのダウンロードおよび COW クローンの代わりに ceph へのアップロードが必要な場合は、インスタンスのブートを拒否します。デフォルト値は false です。
NovaDiskAllocationRatio	物理ディスクに対する仮想ディスクの割当て比率。デフォルト値は 0.0 です。
NovaEnableDBArchive	OpenStack Compute (nova) データベースでソフト削除された行をアーカイブする cron ジョブを作成するかどうか。デフォルト値は true です。
NovaEnableDBPurge	OpenStack Compute (nova) データベースでソフト削除された行をパージする cron ジョブを作成するかどうか。デフォルト値は true です。
NovaEnableVirtlogdContainerWrapper	virtlogd が別のコンテナで起動し、マイナー更新などの再起動を行わないように virtlogd ラッパースクリプトを生成します。デフォルト値は true です。
NovaEnableVTPM	列挙されたトラステッドプラットフォームモジュール (TPM) デバイスのサポートを有効にするかどうか。デフォルト値は false です。
NovaGlanceEnableRbdDownload	直接 RBD 経由で OpenStack Image Storage (glance) イメージをダウンロードするのを有効にします。デフォルト値は false です。
NovaGlanceRbdCopyPollInterval	イメージをローカル rbd ストアにコピーするよう要求した後、OpenStack Image Storage (glance) をポーリングする間隔 (秒単位)。デフォルト値は 15 です。
NovaGlanceRbdCopyTimeout	OpenStack Image Storage (glance) がローカル rbd ストアへのイメージコピーを完了するまで待機する全体の最大時間。デフォルト値は 600 です。

パラメーター	説明
NovaGlanceRbdDownloadMultistoreID	NovaGlanceEnableRbdDownload が有効でデフォルトはオーバーライドする必要がある場合に、ロールに使用される GlanceMultistoreConfig のバックエンド ID であるハッシュキー。CephClientUserName または GlanceRbdPoolName が GlanceMultistoreConfig に設定されていない場合、それらのパラメーターのグローバル値が使用されます。
NovaHWMachineType	各ホストアーキテクチャーのデフォルトのマシントイプを指定します。Red Hat は、ライブマイグレーション中の下位互換性のために、デフォルトをご使用の環境で最も低い RHEL マイナーリリースに設定することを推奨します。デフォルト値は x86_64=pc-q35-rhel9.0.0 です。
NovalmageCacheTTL	ホスト上のどのインスタンスもイメージを使用しなくなった後、nova Compute がキャッシュし続ける時間 (秒単位)。デフォルト値は 86400 です。
NovalmageTypeExcludeList	Compute サービスでサポートされるとは公開すべきでないイメージ形式のリスト。
NovaLibvirtCPUMode	設定する libvirt CPU モード。virt_type が kvm に設定されている場合、デフォルトでは host-model に設定されます。それ以外の場合は、デフォルトでは none に設定されます。デフォルト値は host-model です。
NovaLibvirtCPUModelExtraFlags	このパラメーターにより、CPU モデルを指定する際に、CPU 機能フラグを細かく指定することができます。cpu_mode を none に設定していない場合に限り有効です。
NovaLibvirtCPUModels	名前付きの libvirt CPU モデル (/usr/share/libvirt/cpu_map.xml に記載されている名前を参照)。cpu_mode="custom" および virt_type="kvm qemu" の場合に限り有効です。
NovaLibvirtFileBackedMemory	ファイルベースのメモリーに利用可能な容量 (MiB 単位)。このパラメーターを設定する場合、 NovaRAMAllocationRatio パラメーターおよび NovaReservedHostMemory パラメーターを 0 に設定する必要があります。デフォルト値は 0 です。

パラメーター	説明
NovaLibvirtMaxQueues	libvirt max_queues を設定するためのパラメーターを追加します。マルチキューゲストの作成時に有効にできる virtio キューペアの最大数。割り当てられる virtio キューの数は、ゲストによって要求される CPU と定義される最大値のどちらか小さい方です。デフォルトの 0 は設定なしに対応します。デフォルト値は 0 です。
NovaLibvirtMemStatsPeriodSeconds	メモリー使用量の統計値表示期間 (秒単位)。ゼロまたは負の値の場合には、メモリー使用量の統計値表示が無効になります。デフォルト値は 10 です。
NovaLibvirtNumPciePorts	num_pcie_ports を設定して、インスタンスが取得する PCIe ポートの数を指定します。libvirt では、対象のインスタンスが取得する PCIe ポートの数をカスタマイズすることができます (pcie-root-port コントローラー)。一部はデフォルトで使用されますが、残りはホットプラグの使用に利用されます。デフォルトは 16 です。
NovaLibvirtOptVolumes	マウントされるオプションのボリュームのリスト
NovaLibvirtRxQueueSize	Virtio-net RX キューのサイズ。有効な値は、256、512、1024 です。デフォルト値は 512 です。
NovaLibvirtTxQueueSize	Virtio-net TX キューのサイズ。有効な値は、256、512、1024 です。デフォルト値は 512 です。
NovaLibvirtVolumeUseMultipath	ボリュームのマルチパス接続を有効にするかどうか。デフォルト値は false です。
NovaLiveMigrationPermitAutoConverge	デフォルトは True で、移行のパフォーマンスが遅く完了しない可能性がある場合に、メモリーコピープロセスがインスタンスのメモリー書き込みよりも速くなるまでインスタンスの CPU の速度を低下させます。このフラグが True に設定され、ポストコピーが禁止または libvirt および QEMU のバージョンによりポストコピーが利用できない場合に限り、自動収束が使用されます。デフォルト値は true です。
NovaLiveMigrationPermitPostCopy	True の場合、移行の完了前に移行先ノード上でインスタンスがアクティブになり、転送する必要のあるメモリーの上限を設定します。Compute ロールがリアルタイムロールではない場合や、このパラメーターが無効になっている場合は、デフォルト設定によりポストコピーが有効になります。デフォルト値は true です。

パラメーター	説明
NovaLiveMigrationWaitForVIFPlug	network-vif-plugged イベントを待機してからゲストの転送を開始するかどうかを定義します。デフォルト値は true です。
NovaLocalMetadataPerCell	nova-metadata API サービスがセルごとにデプロイされていることを示します。この場合、マルチセルデプロイメントでのパフォーマンスとデータの分離性が向上します。OpenStack Networking (neutron) の設定方法に応じて、この設定の使用を検討する必要があります。ネットワークが複数セルにまたがる場合は、nova-metadata API サービスをグローバルに実行しなければならない場合があります。ネットワークがセル境界でセグメント化されている場合は、セルごとに nova-metadata API サービスを実行できます。セルごとに nova-metadata API サービスを実行する際には、対応する nova-metadata API サービスを参照するように各 OpenStack Networking (neutron) metadata-agent を設定する必要もあります。デフォルト値は false です。
NovaMaxDiskDevicesToAttach	1つのサーバーに接続できる最大ディスクデバイス数。サーバーがサポートするディスク数は、使用するバスによって異なることに注意してください。たとえば、ide ディスクバスは、アタッチされている4つのデバイスに制限されます。設定された最大値は、サーバーの作成、再構築、退避、取り消し、ライブマイグレーション、および接続時に実施されません。Operator は、サーバーをホストする Compute サービスのこのパラメーターを変更すると、サーバーにすでにアタッチされているデバイスの数より小さいと、リビルドが失敗する可能性があることを認識する必要があります。Operator は、コールドマイグレーション時にも、設定された最大値がインプレースのみ適用され、移動前に宛先がチェックされないことに注意してください。-1は無制限を意味します。デフォルト値は -1 です。
NovaMkisofsCmd	ISO イメージの作成に使用するツールの名前またはパス。デフォルト値は mkisofs です。
NovaNfsEnabled	OpenStack Compute (nova) の NFS バックエンドを有効にするかどうか。デフォルト値は false です。
NovaNfsOptions	nova ストレージの NFS マウントオプション (NovaNfsEnabled が true の場合)。デフォルト値は context=system_u:object_r:nfs_t:s0 です。
NovaNfsShare	nova ストレージをマウントするための NFS 共有 (NovaNfsEnabled が true の場合)

パラメーター	説明
NovaNfsVersion	nova ストレージに使用する NFS バージョン (NovaNfsEnabled が true の場合)。NFSv3 は完全なロックをサポートしないため、NFSv4 バージョンを使用する必要があります。デフォルト値は 4.2 です。
NovaOVSBridge	Open vSwitch で使用する統合ブリッジ名。デフォルト値は br-int です。
NovaOVSDBConnection	OpenStack Compute (nova) が使用する OVS DB 接続文字列
NovaPassword	OpenStack Compute (nova) サービスおよびデータベースアカウントのパスワード
NovaPCIPassthrough	PCI パススルーのホワイトリストパラメーターの YAML リスト
NovaPMEMappings	仮想 PMEM 機能のバックエンドとしての PMEM 名前空間のマッピング。このパラメーターにより、Nova の pmem_namespaces 設定オプションが定義されます。PMEM 名前空間は、手動で作成するか、 NovaPMEMNamespaces パラメーターと共に作成する必要があります。 \$LABEL:\$NSNAME[\$NSNAME] [\$LABEL:\$NSNAME[\$NSNAME]] の形式で指定する必要があります。
NovaPMEMNamespaces	Ansible を介して ndctl ツールを使用して、ホストサーバーに PMEM 名前空間を作成します。 \$SIZE:\$NSNAME[\$SIZE:\$NSNAME...] の形式で指定する必要があります。\$SIZE は、k または K(KiB)、m または M(MiB)、g または G(GiB)、および t または T(TiB) の接尾辞に対応しています。注記: NVDIMM リージョンを適切に設定し、要求される名前空間用に十分な領域を確保する必要があります。
NovaRAMAllocationRatio	物理メモリーに対する仮想メモリーの割当て比率。デフォルト値は 1.0 です。
NovaReservedHostMemory	ホストプロセス用に確保されるメモリー。デフォルト値は 4096 です。

パラメーター	説明
NovaReservedHugePages	NUMA ノード ID、ページサイズ (デフォルトの単位は KiB)、および確保されるページ数を表す有効なキーと値のリスト。例: NovaReservedHugePages: ["node:0,size:2048,count:64","node:1,size:1GB,count:1"] は、NUMA ノード 0 に 2 MiB のページを 64 ページ、NUMA ノード 1 に 1 GiB のページを 1 ページ、それぞれ確保することを意味します。
NovaResumeGuestsShutdownTimeout	ゲストがシャットダウンするのを待機する秒数。このパラメーターを 0 に設定すると、タイムアウトは発生しません (ゲストがシャットダウン要求にตอบสนองしない場合があるため、注意して使用してください)。デフォルト値は 300 秒 (5 分) です。デフォルト値は 300 です。
NovaResumeGuestsStateOnHostBoot	コンピュータホストのリポート時に実行中のインスタンスを起動するかどうか。デフォルト値は false です。
NovaSchedulerAvailableFilters	OpenStack Compute (nova) がノードのフィルタリングに使用する利用可能なフィルターのリスト
NovaSchedulerDefaultFilters	ノードをフィルタリングするために OpenStack Compute (nova) が使用するフィルター配列。これらのフィルターはリスト順に適用されるため、最も制限の厳しいフィルターを最初に配置して、フィルタリングプロセスを効率化します。
NovaSchedulerDiscoverHostsInCellsInterval	この値は、セルに追加された新規ホストの検出をスケジューラーが試みる頻度 (秒単位) です。デフォルト値の -1 は、定期タスクを完全に無効化します。OpenStack Bare Metal (ironic) を使用するデプロイメントには、このパラメーターを設定することを推奨します。デフォルト値は -1 です。
NovaSchedulerEnabledFilters	ノードをフィルタリングするために OpenStack Compute (nova) が使用するフィルターアレイ。OpenStack Compute は、これらのフィルターをリスト順に適用します。フィルタリングのプロセスがより効率的になるように、最も制限の厳しいフィルターを最初に配置します。

パラメーター	説明
NovaSchedulerEnableIsolatedAggregateFiltering	この設定により、スケジューラーは、アグリゲートメタデータおよびインスタンスのフレーバー/イメージの要求される特性のマッチングに基づき、アグリゲートのホストを制限することができます。アグリゲートにキー <code>trait:\$TRAIT_NAME</code> と値が必要な属性が設定されている場合、そのアグリゲートのホストにスケジューリングされるためには、インスタンスのフレーバー <code>extra_specs</code> またはイメージのメタデータにも <code>trait:\$TRAIT_NAME=required</code> が含まれている必要があります。デフォルト値は false です。
NovaSchedulerHostSubsetSize	スケジューラーが選択する最適なホストのサブセットのサイズ。デフォルト値は 1 です。
NovaSchedulerLimitTenantsToPlacementAggregate	この値により、placement を使用したテナントの分離が可能になります。これにより、ホストはテナントを分離したホストアグリゲートに属し、アベイラビリティゾーンは特定のテナントセットでのみ利用できるようになります。デフォルト値は false です。
NovaSchedulerMaxAttempts	インスタンスをデプロイする際のスケジューラーの最大試行回数。スケジューリング時に競合状態が発生しないように、一度にデプロイする予定のベアメタルノードの数以上に維持する必要があります。デフォルト値は 3 です。
NovaSchedulerPlacementAggregateRequiredForTenants	NovaSchedulerLimitTenantsToPlacementAggregate が true の場合、この設定でアグリゲートアフィニティが設定されていないテナントを任意のホストにスケジューリング可能とすることがかを制御します。アグリゲートを使用して一部のテナントだけを制限する場合は、このパラメーターを False に設定する必要があります。アグリゲートによりすべてのテナントを制限する場合は、このパラメーターを True に設定する必要があります。デフォルト値は false です。
NovaSchedulerQueryImageType	この設定により、スケジューラーは、要求で 사용되는イメージの <code>disk_format</code> をサポートするコンピュートホストの配置だけを要求するようになります。デフォルト値は true です。
NovaSchedulerQueryPlacementForAvailabilityZone	この設定により、スケジューラーは、アベイラビリティゾーンのメタデータキーが受信したリクエストの提供する値に設定されたホストアグリゲートを検索し、配置結果をそのアグリゲートに制限するように要求することができます。デフォルト値は false です。

パラメーター	説明
NovaSchedulerQueryPlacementForRoutedNetworkAggregates	この設定により、スケジューラーは、要求されたネットワークまたはポートが OpenStack Networking (neutron) ルーティングネットワークに関連しているかどうかを確認できます。これには、関連するアグリゲートを配置で報告する必要があるため、要求されたアグリゲート内のホストのみが受け入れられます。デフォルト値は false です。
NovaSchedulerShuffleBestSameWeighedHosts	同じ最善の重みを持つホスト間でインスタンスを分散できるようにします。デフォルト値は false です。
NovaSchedulerWorkers	OpenStack Compute (nova) スケジューラーサービスのワーカー数。デフォルト値は 0 です。
NovaShowHostStatus	NovaApiHostStatusPolicy パラメーターで設定されたロールが、インスタンスの Compute ホストのステータスにアクセスできるようにします。 NovaShowHostStatus パラメーターを次の有効な値のいずれかに設定できます。 hidden (デフォルト) を設定すると、RHOSP 管理者のみが Compute ホストのステータスにアクセスできます。 all の場合、 NovaApiHostStatusPolicy で設定されたロールに UP、DOWN、MAINTENANCE、UNKNOWN、および NONE の Compute ホストステータスが表示されます。 unknown-only の場合、 NovaApiHostStatusPolicy パラメーターで設定されたロールに UNKNOWN の Compute ホストステータスが表示されます。この機能を使用するには、使用している RHOSP 環境で SRBAC が有効になっていることを確認してください。 openstack server list --long コマンドを使用する際にホストステータスを表示するには、python-openstackclient-6.0.0 以降がインストールされている必要があります。デフォルト値は hidden です。
NovaStatedirOwnershipSkip	所有者および selinux コンテキストを再帰的に設定する際に無視する nova_statedir からの相対パスのリスト。デフォルト値は ['triliovault-mounts'] です。
NovaSyncPowerStateInterval	データベースとハイパーバイザー間で電源の状態を同期する間隔。無効にするには -1 に設定します。これを 0 に設定すると、oslo.service で定義されたデフォルトのレート (60) で実行されます。デフォルト値は 600 です。

パラメーター	説明
NovaVcpuPinSet	<p>(非推奨) 仮想マシンプロセス用に確保する物理 CPU コアのリストまたは範囲。たとえば、NovaVcpuPinSet: [4-12,^8] は、コア 4-12 の範囲 (ただし 8 を除く) を確保します。このパラメーターは非推奨となっています。代わりに NovaComputeCpuDedicatedSet および NovaComputeCpuSharedSet を使用してください。 NovaComputeCpuDedicatedSet および NovaComputeCpuSharedSet の代わりに NovaVcpuPinSet を引き続き使用する RHOSP デプロイメントのサポートが必要な場合は、 Red Hat Customer Experience and Engagement チーム に連絡してサポート例外を取得してください。 NovaVcpuPinSet をサポートされているパラメーターの NovaComputeCpuDedicatedSet および NovaComputeCpuSharedSet に更新する方法について、詳細は アップグレード用フレームワーク (16.2 から 17.1) ガイドの CPU ピニングパラメーターの更新 を参照してください。</p>
NovaVGPUTypesDeviceAddressesMapping	<p>インスタンスがキーとして取得できる vgpu タイプのマッピングと、対応するデバイスアドレスの値のリスト。たとえば、 NovaVGPUTypesDeviceAddressesMapping: {nvidia-35: [0000:84:00.0, 0000:85:00.0], nvidia-36: [0000:86:00.0]} では、 nvidia-35 および nvidia-36 が vgpu タイプと対応する値で、関連するデバイスアドレスの値のリストです。</p>
NovaVNCCertificateKeySize	<p>このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。</p>
NovaVNCProxySSLCiphers	<p>クライアントからの TLS 接続に許可する暗号を指定する OpenSSL 暗号設定文字列。暗号設定文字列の形式および許可される値の詳細は、 OpenSSL ciphers コマンドの man ページを参照してください。</p>
NovaVNCProxySSLMinimumVersion	<p>許可される SSL/TLS プロトコルの最低バージョン。有効な値は default、 tlsv1_1、 tlsv1_2、 および tlsv1_3 です。値が default の場合、基礎となるシステム OpenSSL のデフォルト値が使用されます。デフォルト値は default です。</p>
NovaWorkers	<p>Compute の Conductor サービスのワーカー数。ワーカー数が多いと、システムのプロセス数も増え、メモリーを過剰消費してしまう点に注意してください。 0 の場合は、OpenStack の内部デフォルト値 (ノードにある CPU コア数) に設定されます。デフォルト値は 0 です。</p>

パラメーター	説明
OvsDpdkSocketMemory	NUMA ノードごとに割り当てるヒュージページメモリの容量を設定します。目的の DPDK NIC に使用する PCIe スロットに最も近いソケットの使用が推奨されます。形式は "<socket 0 mem>, <socket 1 mem>, <socket n mem>" で、値は MB 単位で指定します。たとえば、"1024,0" となります。
PlacementAPIInterface	placement API に使用するエンドポイントインターフェイス。デフォルト値は internal です。
PlacementPassword	Placement サービスおよびデータベースアカウントのパスワード
QemuCACert	このパラメーターは、qemu に使用する CA 証明書を指定します。このファイルは、デフォルト CA のパス (/etc/pki/qemu/ca-cert.pem) へのシンボリックリンクです。このパラメーターは、デフォルト (内部 TLSCAFile パラメーターがベース) が望ましくない場合に使用すべきです。現在のデフォルトは、TripleO のデフォルト CA である FreeIPA を反映しています。これは、内部の TLS が有効な場合にのみ使用されます。
QemuClientCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
QemuDefaultTLSVerify	TLS クライアント証明書の検証を有効/無効にするかどうか。このオプションを有効にすると、/etc/pki/qemu/ca-cert.pem に CA が署名した証明書がないクライアントは拒否されます。デフォルト値は true です。
QemuMemoryBackingDir	memoryBacking ソースがファイルとして設定されている場合に、そのソースに使用するディレクトリ。注記: 大容量のファイルがここに保存されません。
QemuServerCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
RbdDiskCachemodes	RBD バックエンドのディスクキャッシュモード。デフォルト値は ['network=writeback'] です。
UpgradeLevelNovaCompute	OpenStack Compute アップグレードレベル

パラメーター	説明
UseTLSTransportForNbd	true に設定して、EnableInternalTLS が有効な場合には、libvirt NBD の TLS トランスポートが有効となり、libvirt の関連するキーが設定されます。デフォルト値は true です。
UseTLSTransportForVnc	true に設定して、EnableInternalTLS が有効な場合には、libvirt VNC の TLS トランスポートが有効となり、libvirt の関連するキーが設定されます。デフォルト値は true です。
VerifyGlanceSignatures	イメージの署名を検証するかどうか。デフォルト値は False です。
VhostuserSocketGroup	vhost-user ソケットディレクトリーのグループ名。デフォルトは qemu です。vhostuser モードが dpdkvhostuserclient (デフォルトモード) の場合には、qemu により vhost ソケットが作成されます。デフォルト値は qemu です。

第11章 DASHBOARD (HORIZON) パラメーター

Dashboard パラメーターを使用して horizon サービスを変更できます。

パラメーター	説明
HorizonAllowedHosts	OpenStack Dashboard (horizon) が実行中のサーバーの IP/ホスト名のリスト。ヘッダーチェックに使用します。デフォルト値は * です。
HorizonCustomizationModule	OpenStack Dashboard (horizon) には、カスタマイズを行うためのグローバルオーバーライドメカニズムがあります。
HorizonDomainChoices	選択可能なドメインを指定します。ハッシュ配列が必要であり、ハッシュにはそれぞれ OpenStack Identity (keystone) ドメイン名と人間が読めるドメインの記述を含む 2 つの項目 (名前、表示) が含まれている必要があります。
HorizonHelpURL	Dashboard 上部には Help ボタンがあります。このボタンを使用して、ベンダーのドキュメントや専用のヘルプポータルにユーザーをリダイレクトすることができます。デフォルト値は http://docs.openstack.org です。
HorizonHstsHeaderValue	応答で HTTP Strict-Transport-Security ヘッダーを有効にします。
HorizonPasswordValidator	パスワード検証のための正規表現
HorizonPasswordValidatorHelp	パスワード検証のためのヘルプテキスト
HorizonSecret	Web サーバーの秘密鍵
HorizonSecureCookies	OpenStack Dashboard (horizon) で CSRF_COOKIE_SECURE / SESSION_COOKIE_SECURE を設定します。デフォルト値は false です。
HorizonSessionTimeout	horizon のセッションタイムアウトを秒単位で設定します。デフォルト値は 1800 です。
HorizonVhostExtraParams	OpenStack Dashboard (horizon) vhost 設定の追加パラメーター。デフォルト値: <code>{'add_listen': 'true', 'priority': '10', 'access_log_format': '%a %l %u %t \\\r\\' %>s %b \\\r%{Referer}i\\' \\\r%{User-Agent}i\\', 'options': ['FollowSymLinks', 'MultiViews']}</code>

パラメーター	説明
MemcachedIPv6	Memcached の IPv6 機能を有効にします。デフォルト値は false です。
TimeZone	オーバークラウド上で設定するタイムゾーン。デフォルト値は UTC です。
WebSSOChoices	表示する SSO 認証の選択肢リストを指定します。各項目は、SSO の選択肢の ID および表示メッセージのリストです。デフォルト値は [['OIDC', 'OpenID Connect']] です。
WebSSOEnable	Web シングルサインオンのサポートを有効にします。デフォルト値は false です。
WebSSOIDPMapping	SSO 認証の選択肢から認証プロバイダーおよびプロトコルへのマッピングを指定します。認証プロバイダーおよびプロトコル名は、keystone で定義したリソースと一致している必要があります。デフォルト値は {'OIDC': ['myidp', 'openid']} です。
WebSSOInitialChoice	デフォルトで選択する認証の選択肢の初期値。デフォルト値は OIDC です。

第12章 IDENTITY (KEYSTONE) パラメーター

ID パラメーターを使用して keystone サービスを変更できます。

パラメーター	説明
AdminToken	OpenStack Identity (keystone) のシークレットおよびデータベースのパスワード
ApacheCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
ApacheTimeout	Apache のタイムアウト (秒単位)。Apache が I/O 操作を待機する時間を定義します。デフォルト値は 90 です。
CertificateKeySize	証明書の作成時に使用される秘密鍵のサイズを指定します。デフォルト値は 2048 です。
EnableCache	memcached でキャッシュを有効にします。デフォルト値は true です。
EnablePublicTLS	パブリックインターフェイスで TLS を有効にするかどうかを定義します。デフォルト値は true です。
EnableSQLAlchemyCollectd	SQLAlchemy-collectd サーバープラグインを有効にするには true に設定します。デフォルト値は false です。
EnforceSecureRbac	このオプションを True に設定すると、 oslo_policy enforce_new_defaults と oslo_policy enforce_scope を True に設定して、各 OpenStack サービスが Secure RBAC を適用するように設定されます。これにより、システムおよびプロジェクトスコープのサポートや keystone のデフォルトロール、管理者、メンバー、リーダーなど、OpenStack サービス全体で一貫性のある一連の RBAC ペルソナが導入されます。デプロイメント内のすべてのサービスがセキュアな RBAC を実際にサポートするまで、この機能は有効にしないでください。デフォルト値は false です。
KeystoneAuthMethods	認証に使用するメソッドのリスト
KeystoneChangePasswordUponFirstUse	このオプションを有効にすると、ユーザーの作成時や、管理者がパスワードをリセットした場合に、ユーザーによるパスワードの変更が必要となります。
KeystoneCorsAllowedOrigin	このリソースがリクエストの元ヘッダーで受け取るドメインと共有されるかどうかを示します。

パラメーター	説明
KeystoneCredential0	最初の OpenStack Identity (keystone) の認証情報キー。有効なキーでなければなりません。
KeystoneCredential1	2 番目の OpenStack Identity (keystone) の認証情報キー。有効なキーでなければなりません。
KeystoneCronTrustFlushDestination	期限切れまたはソフト削除された信頼を消去するための cron - ログの宛先。デフォルト値は /var/log/keystone/keystone-trustflush.log です。
KeystoneCronTrustFlushEnsure	期限切れまたはソフト削除された信頼を消去するための cron - Ensure。デフォルト値は present です。
KeystoneCronTrustFlushHour	期限切れまたはソフト削除された信頼を消去するための cron - Hour。デフォルト値は * です。
KeystoneCronTrustFlushMaxDelay	期限切れまたはソフト削除された信頼を消去するための cron - Max Delay。デフォルト値は 0 です。
KeystoneCronTrustFlushMinute	期限切れまたはソフト削除された信頼を消去するための cron - Minute。デフォルト値は 1 です。
KeystoneCronTrustFlushMonth	期限切れまたはソフト削除された信頼を消去するための cron - Month。デフォルト値は * です。
KeystoneCronTrustFlushMonthday	期限切れまたはソフト削除された信頼を消去するための cron - Month Day。デフォルト値は * です。
KeystoneCronTrustFlushUser	期限切れまたはソフト削除された信頼を消去するための cron - User。デフォルト値は keystone です。
KeystoneCronTrustFlushWeekday	期限切れまたはソフト削除された信頼を消去するための cron - Week Day。デフォルト値は * です。
KeystoneDisableUserAccountDaysInactive	ユーザーが認証なしでアカウントを使用し続けることのできる最大日数。この期間が過ぎるとアカウントは非アクティブと見なされて自動的に無効 (ロック状態) になります。
KeystoneEnableDBPurge	OpenStack Identity (keystone) データベースでソフト削除された行をパージする cron ジョブを作成するかどうか。デフォルト値は true です。
KeystoneEnableMember	アンダークラウドのデプロイメントに有用な member ロールを作成します。デフォルト値は False です。

パラメーター	説明
KeystoneFederationEnable	フェデレーション認証のサポートを有効にします。デフォルト値は false です。
KeystoneFernetKeys	OpenStack Identity (keystone) Fernet 鍵とそれらのパスが含まれたマッピング
KeystoneFernetMaxActiveKeys	OpenStack Identity (keystone) Fernet 鍵リポジトリ内のアクティブなキーの最大数。デフォルト値は 5 です。
KeystoneLDAPBackendConfigs	keystone に設定した LDAP バックエンドの設定を含むハッシュ
KeystoneLDAPDomainEnable	ldap_backend の puppet keystone 定義を呼び出すトリガー。デフォルト値は False です。
KeystoneLockoutDuration	認証の最大試行回数 (KeystoneLockoutFailureAttempts で指定) を超過した場合にユーザーアカウントがロックされる秒数
KeystoneLockoutFailureAttempts	ユーザーの認証失敗がこの最大回数を超えると、KeystoneLockoutDuration で指定された秒数の期間ユーザーアカウントがロックされます。
KeystoneMinimumPasswordAge	ユーザーがパスワードを変更できるようになるまで、そのパスワードを使用する必要がある日数。これは、パスワードの履歴を消去して古いパスワードを再利用するためにユーザーがパスワードを直ちに変更するのを防ぎます。
KeystoneNotificationDriver	OpenStack Identity (keystone) で使用される Oslo 通知ドライバーのコンマ区切りリスト。
KeystoneNotificationFormat	OpenStack Identity (keystone) の通知形式。デフォルト値は basic です。
KeystoneNotificationTopics	有効にする OpenStack Identity (keystone) の通知トピック
KeystoneOpenIdcClientId	OpenID Connect プロバイダーとハンドシェイクする際に使用するクライアント ID
KeystoneOpenIdcClientSecret	OpenID Connect プロバイダーとハンドシェイクする際に使用するクライアントシークレット
KeystoneOpenIdcCryptoPassphrase	OpenID Connect ハンドシェイクのデータを暗号化する際に使用するパスフレーズ。デフォルト値は openstack です。

パラメーター	説明
KeystoneOpenIdcEnable	OpenIDC フェデレーションのサポートを有効にします。デフォルト値は false です。
KeystoneOpenIdcEnableOAuth	OAuth 2.0 統合を有効にします。デフォルト値は false です。
KeystoneOpenIdcIdpName	OpenStack Identity (keystone) の IdP に関連付けられた名前
KeystoneOpenIdcIntrospectionEndpoint	mod_auth_openidc 用の OAuth 2.0 イントロスペクションエンドポイント
KeystoneOpenIdcProviderMetadataUrl	OpenID Connect プロバイダーのメタデータをポイントする URL
KeystoneOpenIdcRemoteIdAttribute	環境から認証プロバイダーのエンティティ ID を取得するのに使用される属性。デフォルト値は HTTP_OIDC_ISS です。
KeystoneOpenIdcResponseType	想定される OpenID Connect プロバイダーからの応答種別。デフォルト値は id_token です。
KeystonePasswordExpiresDays	パスワードが有効と見なされる日数。この期間が過ぎるとパスワードの変更が要求されます。
KeystonePasswordRegex	パスワードの強度要件の検証に使用される正規表現
KeystonePasswordRegexDescription	人が判読できる言語でパスワードの正規表現を記述してください。
KeystoneSSLCertificate	トークンの有効性を検証するための OpenStack Identity (keystone) 証明書
KeystoneSSLCertificateKey	トークンに署名するための OpenStack Identity (keystone) キー
KeystoneTokenProvider	OpenStack Identity (keystone) のトークン形式。デフォルト値は fernet です。
KeystoneTrustedDashboards	シングルサインオン用に信頼されるダッシュボード URL のリスト
KeystoneUniqueLastPasswordCount	これにより、新たに作成されたパスワードが一意であることを強制するために、履歴に保管する以前のユーザーパスワードのイテレーション数が制御されます。

パラメーター	説明
KeystoneWorkers	OpenStack Identity (keystone) サービスのワーカー数を設定します。ワーカー数が多いと、システムのプロセス数も増え、メモリーを過剰消費してしまう点に注意してください。CPU コア数が多いシステムでは、デフォルト以外の適切な値を選択することを推奨します。 0 の場合は、OpenStack の内部デフォルト値 (ノードにある CPU コア数) に設定されます。デフォルト値は、物理ノードにある仮想 CPU コア数と同じ数値です。
ManageKeystoneFernetKeys	director が OpenStack Identity (keystone) fernet キーを管理するかどうか。True に設定した場合には、fernet キーは OpenStack Workflow (mistrall) 内の保存済みのキーレポジトリの KeystoneFernetKeys 変数から値を取得します。False に設定されている場合には、スタックを作成した場合にのみキーが初期化され、その後の更新時にはキーには触れません。デフォルト値は true です。
MemcachedTLS	Memcached サービスで TLS を有効にするには、True に設定します。すべてのサービスが Memcached TLS をサポートする訳ではないので、移行期間中、Memcached は 2 つのポート (MemcachedPort パラメーター (上記) で設定したポート、および TLS を使用しない 11211) をリスンします。デフォルト値は false です。
NotificationDriver	通知の送信を処理する単一または複数のドライバー。デフォルト値は noop です。
PublicSSLCertificateAutogenerated	公開 SSL 証明書を自動生成するかどうかを定義します。デフォルト値は false です。
PublicTLSCAFile	TLS がパブリックネットワーク内のサービスに使用される場合に使用するデフォルトの CA 証明書を指定します。
SSLCertificate	PEM 形式の SSL 証明書 (キーなし) の内容
TokenExpiration	トークンの有効期限を秒単位で設定します。デフォルト値は 3600 です。

第13章 IMAGE STORAGE (GLANCE) パラメーター

Image サービスのパラメーターを使用して glance サービスを変更できます。

パラメーター	説明
CephClusterName	Ceph クラスター名。デフォルト値は ceph です。
CephConfigPath	Ceph クラスター設定ファイルがホストに保存されているパス。デフォルト値は /var/lib/tripleo-config/ceph です。
CinderEnableNVMeOFBackend	OpenStack Block Storage (cinder) の NVMeOF バックエンドを有効にするかどうか。デフォルト値は false です。
CinderNVMeOFAvailabilityZone	NVMeOF OpenStack Block Storage (cinder) NFS バックエンドのアベイラビリティゾーン。このパラメーターを設定すると、デフォルトの CinderStorageAvailabilityZone がオーバーライドされます。
CinderNVMeOFTargetProtocol	ターゲットプロトコル。サポートされる値は nvmet_rdma および nvmet_tcp です。デフォルト値は nvmet_rdma です。
EnableSQLAlchemyCollectd	SQLAlchemy-collectd サーバープラグインを有効にするには true に設定します。デフォルト値は false です。
EnforceSecureRbac	このオプションを True に設定すると、 oslo_policy enforce_new_defaults と oslo_policy enforce_scope を True に設定して、各 OpenStack サービスが Secure RBAC を適用するように設定されます。これにより、システムおよびプロジェクトスコープのサポートや keystone のデフォルトロール、管理者、メンバー、リーダーなど、OpenStack サービス全体で一貫性のある一連の RBAC ペルソナが導入されます。デプロイメント内のすべてのサービスがセキュアな RBAC を実際にサポートするまで、この機能は有効にしないでください。デフォルト値は false です。
GlanceApiOptVolumes	マウントされるオプションのボリュームのリスト
GlanceBackend	使用する OpenStack Image Storage (glance) バックエンドの省略名。swift、rbd、cinder、または file のいずれかにする必要があります。デフォルト値は swift です。

パラメーター	説明
GlanceBackendID	デフォルトのバックエンドの識別子。デフォルト値は default_backend です。
GlanceCacheEnabled	OpenStack Image Storage (glance) のイメージキャッシュを有効にします。デフォルト値は False です。
GlanceCinderMountPointBase	glance が cinder をストアとして使用し、cinder バックエンドが NFS の場合のマウントポイントベース。このマウントポイントは、glance ノードで NFS ボリュームがマウントされる場所です。デフォルト値は /var/lib/glance/mnt です。
GlanceCinderVolumeType	複数の Cinder ストアを Glance バックエンドとして設定する際に、各 Cinder ストアに必要な固有のボリュームタイプ。OpenStack Block Storage (cinder) でも同じボリュームタイプを設定する必要があります。関連する cinder ストアにイメージを追加しようとする前に、ボリュームタイプが cinder に存在している必要があります。ボリュームタイプが指定されていない場合、cinder のデフォルトのボリュームタイプが使用されます。
GlanceCronDbPurgeAge	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Age)。デフォルト値は 30 です。
GlanceCronDbPurgeDestination	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Log destination)。デフォルト値は /var/log/glance/glance-rowsflush.log です。
GlanceCronDbPurgeHour	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Hour)。デフォルト値は 0 です。
GlanceCronDbPurgeMaxDelay	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Max Delay)。デフォルト値は 3600 です。
GlanceCronDbPurgeMaxRows	削除済みとマークされかつ \$age よりも古いデータベースのエントリーを消去する cron - Max Rows。デフォルト値は 100 です。
GlanceCronDbPurgeMinute	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Minute)。デフォルト値は 1 です。

パラメーター	説明
GlanceCronDbPurgeMonth	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Month)。デフォルト値は * です。
GlanceCronDbPurgeMonthday	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Month Day)。デフォルト値は * です。
GlanceCronDbPurgeUser	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (User)。デフォルト値は glance です。
GlanceCronDbPurgeWeekday	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Week Day)。デフォルト値は * です。
GlanceDiskFormats	Glance で許可されるディスク形式のリスト。未設定のままにすると、すべての形式が許可されます。
GlanceEnabledImportMethods	有効なイメージのインポートメソッドのリスト。リスト内での有効な値は glance-direct 、 web-download 、または copy-image です。デフォルト値は web-download です。
GlanceIgnoreUserRoles	イメージメタデータのプロパティを挿入する際に無視されるユーザーロールのリスト。デフォルト値は admin です。
GlanceImageCacheDir	イメージのキャッシュが使用するベースディレクトリー。デフォルト値は /var/lib/glance/image-cache です。
GlanceImageCacheMaxSize	キャッシュサイズの上限 (バイト単位)。これを超えると、 cache-pruner がイメージのキャッシュをクリーンアップします。デフォルト値は 10737418240 です。
GlanceImageCacheStallTime	イメージがアクセスされないままキャッシュ内に留まることのできる時間 (秒単位)。デフォルト値は 86400 です。
GlanceImageConversionOutputFormat	イメージ変換プラグインの希望する出力フォーマット。デフォルト値は raw です。

パラメーター	説明
GlanceImageImportPlugins	有効なイメージのインポートプラグインのリスト。このリスト内での有効な値は image_conversion 、 inject_metadata 、 no_op です。デフォルト値は ['no_op'] です。
GlanceImageMemberQuota	1イメージあたりのイメージメンバーの最大数。マイナスの値を指定すると無制限に設定されます。デフォルト値は 128 です。
GlanceImagePrefetcherInterval	定期的なジョブ <code>cache_images</code> を実行する間隔 (秒単位)。デフォルト値は 300 です。
GlanceInjectMetadataProperties	イメージに挿入されるメタデータのプロパティー
GlanceLogFile	OpenStack Image (glance) のストレージからのメッセージのログ記録に使用するファイルのパス
GlanceMultistoreConfig	追加の glance バックエンドを設定する際の設定のディクショナリー。ハッシュキーはバックエンドの ID で、値はそのバックエンドに固有のパラメーター値のディクショナリーです。複数の rbd および cinder バックエンドを設定することが可能ですが、file および swift バックエンドはそれぞれ1つに制限されます。(例: # Default glance store is rbd.GlanceBackend: rbd GlanceStoreDescription: デフォルト rbd ストア # GlanceMultistoreConfig は 2 番目の rbd バックエンドおよび cinder # バックエンドを指定します。GlanceMultistoreConfig: rbd2_store: GlanceBackend: rbd GlanceStoreDescription: 2 番目の rbd ストア CephClusterName: ceph2 # オーバーライド CephClientUserName (このクラスターが別の # クライアント名を使用する場合)。CephClientUserName: client2 cinder1_store: GlanceBackend: cinder GlanceCinderVolumeType: volume-type-1 GlanceStoreDescription: First cinder store cinder2_store: GlanceBackend: cinder GlanceCinderVolumeType: volume-type-2 GlanceStoreDescription: Seconde cinder store.
GlanceNetappNfsEnabled	GlanceBackend: file を使用する場合は、Netapp がイメージストレージ用に NFS 共有をマウントします。デフォルト値は false です。
GlanceNfsEnabled	GlanceBackend: file を使用する場合は、イメージストレージ用に NFS 共有をマウントします。デフォルト値は false です。

パラメーター	説明
GlanceNfsOptions	GlanceNfsEnabled が true の場合のイメージストレージの NFS マウントオプション。デフォルト値は _netdev,bg,intr,context=system_u:object_r:container_file_t:s0 です。
GlanceNfsShare	GlanceNfsEnabled が true の場合にイメージストレージ用にマウントする NFS 共有
GlanceNodeStagingUri	イメージのインポート時に使用するステージングロケーションを指定する URI。デフォルト値は file:///var/lib/glance/staging です。
GlanceNotifierStrategy	OpenStack Image Storage (glance) の通知キューに使用するストラテジー。デフォルト値は noop です。
GlancePassword	Image Storage サービスおよびデータベースアカウントのパスワード
GlanceShowMultipleLocations	複数のイメージの保管場所を表示するかどうかを定義します (RBD または Netapp バックエンドでの Copy-on-Write のサポート)。セキュリティリスクを伴う可能性があります。詳しくは、 <code>glance.conf</code> を参照してください。デフォルト値は false です。
GlanceSparseUploadEnabled	GlanceBackend ファイル および rbd を使用する場合に、スパースアップロードを有効または無効にします。デフォルト値は false です。
GlanceStagingNfsOptions	NFS イメージインポートのステージングの NFS マウントオプション。デフォルト値は _netdev,bg,intr,context=system_u:object_r:container_file_t:s0 です。
GlanceStagingNfsShare	イメージインポートのステージングをマウントするための NFS 共有
GlanceStoreDescription	OpenStack Image Storage (glance) バックエンドのユーザー向け説明。デフォルト値は Default glance store backend. です。
GlanceWorkers	Image Storage サービスのワーカー数を設定します。ワーカー数が多いと、システムのプロセス数も増え、メモリーを過剰消費してしまう点に注意してください。CPU コア数が多いシステムでは、デフォルト以外の適切な値を選択することを推奨します。0 の場合は、OpenStack の内部デフォルト値 (ノードにある CPU コア数) に設定されます。

パラメーター	説明
MemcacheUseAdvancedPool	高度な (eventlet に対して安全な) memcached クライアントプールを使用します。デフォルト値は true です。
MultipathdEnable	マルチパスデーモンを有効にするかどうかを定義します。デフォルト値は false です。
NetappShareLocation	Image Storage 用にマウントする Netapp 共有 (GlanceNetappNfsEnabled が true の場合)
NotificationDriver	通知の送信を処理する単一または複数のドライバー。デフォルト値は noop です。

第14章 KEY MANAGER (BARBICAN) パラメーター

barbican サービスは、キーマネージャーのパラメーターを使用して変更できます。

パラメーター	説明
ApacheCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
ApacheTimeout	Apache のタイムアウト (秒単位)。Apache が I/O 操作を待機する時間を定義します。デフォルト値は 90 です。
ATOSVars	ATOS クライアントソフトウェアのインストールに使用する atos-hsm ロール変数のハッシュ
BarbicanDogtagStoreGlobalDefault	このプラグインがグローバルのデフォルトプラグインかどうか。デフォルト値は false です。
BarbicanDogtagStoreHost	Dogtag サーバーのホスト名
BarbicanDogtagStoreNSSPassword	NSS DB のパスワード
BarbicanDogtagStorePEMPath	リクエストの認証に使用する PEM ファイルのパス。デフォルト値は /etc/barbican/kra_admin_cert.pem です。
BarbicanDogtagStorePort	Dogtag サーバーのポート。デフォルト値は 8443 です。
BarbicanKmpStoreGlobalDefault	このプラグインがグローバルのデフォルトプラグインかどうか。デフォルト値は false です。
BarbicanKmpStoreHost	KMIP デバイスのホスト
BarbicanKmpStorePassword	KMIP デバイスに接続するためのパスワード
BarbicanKmpStorePort	KMIP デバイスのポート
BarbicanKmpStoreUsername	KMIP デバイスに接続するためのユーザー名
BarbicanPassword	OpenStack Key Manager (barbican) サービスアカウントのパスワード
BarbicanPkcs11AlwaysSetCkaSensitive	常に CKA_SENSITIVE=CK_TRUE と設定します。デフォルト値は true です。
BarbicanPkcs11CryptoAESGCMGenerateIV	CKM_AES_GCM 暗号化メカニズムの IV を生成します。デフォルト値は true です。

パラメーター	説明
BarbicanPkcs11CryptoATOSEnabled	PKCS11 の ATOS を有効にします。デフォルト値は false です。
BarbicanPkcs11CryptoEnabled	PKCS11 を有効にします。デフォルト値は false です。
BarbicanPkcs11CryptoEncryptionMechanism	暗号化に使用される Cryptoki メカニズム。デフォルト値は CKM_AES_CBC です。
BarbicanPkcs11CryptoGlobalDefault	このプラグインがグローバルのデフォルトプラグインかどうか。デフォルト値は false です。
BarbicanPkcs11CryptoHMACKeygenMechanism	マスター HMAC 鍵を生成するのに使用される Cryptoki メカニズム。デフォルト値は CKM_AES_KEY_GEN です。
BarbicanPkcs11CryptoHMACKeyType	マスター HMAC 鍵の Cryptoki 鍵種別。デフォルト値は CKK_AES です。
BarbicanPkcs11CryptoHMACLabel	HMAC 鍵のラベル
BarbicanPkcs11CryptoLibraryPath	ベンダー PKCS11 ライブラリーへのパス
BarbicanPkcs11CryptoLogin	PKCS#11 セッションにログインするためのパスワード (PIN)
BarbicanPkcs11CryptoLunasaEnabled	PKCS11 の Luna SA HSM を有効にします。デフォルト値は false です。
BarbicanPkcs11CryptoMKEKLabel	マスター KEK のラベル
BarbicanPkcs11CryptoMKEKLength	マスター KEK の長さ (バイト単位)。デフォルト値は 256 です。
BarbicanPkcs11CryptoOsLockingOk	クライアントライブラリーを初期化する際に CKF_OS_LOCKING_OK フラグを設定します。デフォルト値は false です。
BarbicanPkcs11CryptoRewrapKeys	マスター HMAC 鍵を生成するのに使用される Cryptoki メカニズム。デフォルト値は false です。
BarbicanPkcs11CryptoSlotId	使用する PKCS#11 トークンのスロット ID。デフォルト値は 0 です。
BarbicanPkcs11CryptoThalesEnabled	PKCS11 の Thales を有効にします。デフォルト値は false です。

パラメーター	説明
BarbicanPkcs11CryptoTokenLabel	(非推奨) 代わりに BarbicanPkcs11CryptoTokenLabels を使用してください。
BarbicanPkcs11CryptoTokenLabels	使用するトークンのラベルのコンマ区切りリスト。これは通常1つのラベルですが、一部のデバイスには負荷分散および高可用性設定用に複数のラベルが必要な場合があります。
BarbicanPkcs11CryptoTokenSerialNumber	使用する PKCS#11 トークンのシリアル番号
BarbicanSimpleCryptoGlobalDefault	このプラグインがグローバルのデフォルトプラグインかどうか。デフォルト値は false です。
BarbicanSimpleCryptoKek	シークレットの暗号化に使用する KEK
BarbicanWorkers	barbican::wsgi::apache のワーカー数を設定します。デフォルト値は %{::processorcount} です。
CertificateKeySize	証明書の作成時に使用される秘密鍵のサイズを指定します。デフォルト値は 2048 です。
EnableSQLAlchemyCollectd	SQLAlchemy-collectd サーバープラグインを有効にするには true に設定します。デフォルト値は false です。
LunasaClientIPNetwork	(オプション) OpenStack Key Manager (barbican) を設定する際に、ノードは FQDN ではなくこのネットワークからの IP を使用して HSM で登録されます。
LunasaVars	Lunasa クライアントソフトウェアのインストールに使用する lunasa-hsm ロール変数のハッシュ
MemcacheUseAdvancedPool	高度な (eventlet に対して安全な) memcached クライアントプールを使用します。デフォルト値は true です。
NotificationDriver	通知の送信を処理する単一または複数のドライバー。デフォルト値は noop です。
ThalesHSMNetworkName	HSM がリッスンしているネットワーク。デフォルト値は internal_api です。
ThalesVars	Thales クライアントソフトウェアのインストールに使用する thales_hsm ロール変数のハッシュ。

第15章 ロードバランサー (OCTAVIA) パラメーター

パラメーター	説明
OctaviaAdminLogFacility	管理ログメッセージに使用する syslog "LOG_LOCAL" ファシリティー。デフォルト値は 1 です。
OctaviaAdminLogTargets	管理ログメッセージを受信するための syslog エンドポイントのリスト (host:port コンマ区切りリスト)。
OctaviaAmphoraExpiryAge	未使用の Amphora が期限切れと見なされ、クリーンアップされるまでの秒単位の間隔。0 のままにすると、設定は設定されず、システムはサービスのデフォルトを使用します。デフォルト値は 0 です。
OctaviaAmphoraSshKeyDir	OpenStack Load Balancing-as-a-Service (octavia) が生成した SSH キーディレクトリー。デフォルト値は /etc/octavia/ssh です。
OctaviaAmphoraSshKeyFile	公開鍵ファイルのパス。ユーザーは、提供されたキーを使用して amphorae に SSH で接続できます。ほとんどの場合、ユーザーはユーザー centos (CentOS)、 ubuntu (Ubuntu)、または cloud-user (RHEL) から root に昇格できます (amphora イメージの作成方法によって異なります)。amphorae にログインすると、負荷分散サービスをデバッグするなどの便利な方法が提供されます。
OctaviaAmphoraSshKeyName	SSH キーの名前。デフォルト値は octavia-ssh-key です。
OctaviaAntiAffinity	アンチアフィニティー機能がオンになっているかどうかを示すフラグ。デフォルト値は true です。
OctaviaCaCert	OpenStack Load Balancing-as-a-Service (octavia) CA 証明書データ。指定されている場合は、これにより、OctaviaCaCertFile で指定されたパスを使用して、ホスト上に証明書データを使用してファイルが作成または更新されます。
OctaviaCaKey	OctaviaCaCert で提供される証明書の秘密鍵。指定されている場合、これにより、OctaviaCaKeyFile で指定されたパスを使用して、ホスト上にキーデータを使用してファイルが作成または更新されます。
OctaviaCaKeyPassphrase	CA 秘密鍵のパスフレーズ

パラメーター	説明
OctaviaClientCert	OpenStack Load Balancing-as-a-Service (octavia) クライアント証明書データ。指定されている場合は、これにより、OctaviaClientCertFile で指定されたパスを使用して、ホスト上に証明書データを使用してファイルが作成または更新されます。
OctaviaConnectionLogging	false の場合、テナント接続フローはログに記録されません。デフォルト値は true です。
OctaviaDefaultListenerCiphers	新しい TLS 対応リスナーの OpenSSL 暗号のデフォルトリスト。
OctaviaDefaultPoolCiphers	新しい TLS 対応プールの OpenSSL 暗号のデフォルトリスト。
OctaviaDisableLocalLogStorage	true の場合、ログは amphora ファイルシステムに保存されません。これには、すべてのカーネル、システム、およびセキュリティーログが含まれます。デフォルト値は false です。
OctaviaEnableDriverAgent	何らかの理由でドライバーエージェントを無効にする必要がある場合は、false に設定します。デフォルト値は true です。
OctaviaEnableJobboard	amphorav2 ドライバーの jobboard を有効にすると、amphora ドライバーのフロー再開が有効になります。デフォルト値は false です。
OctaviaFlavorId	flexible の nova フレーバーを作成するときに使用される OpenStack Compute (nova) フレーバー ID。デフォルト値は 65 です。
OctaviaForwardAllLogs	true の場合、amphora からのすべてのログメッセージは、非負荷分散関連のログを含め、管理ログエンドポイントに転送されます。デフォルト値は false です。
OctaviaGenerateCerts	セキュリティーが問題とされない孤立したプライベートクラウドまたはシステムの amphora との安全な通信のための証明書の内部生成を有効にします。それ以外の場合は、OctaviaCaCert、OctaviaCaKey、OctaviaCaKeyPassphrase、OctaviaClientCert、および OctaviaServerCertsKeyPassphrase を使用して、OpenStack Load Balancing-as-a-Service (octavia) を設定します。デフォルト値は false です。

パラメーター	説明
OctaviaJobboardExpirationTime	jobboard で要求したジョブの有効期限。デフォルト値は 30 です。
OctaviaListenerTlsVersions	新しい TLS 対応リスナーに使用する TLS バージョンの OpenSSL 暗号文字列のリスト。デフォルト値は ['TLSv1.2', 'TLSv1.3'] です。
OctaviaLoadBalancerTopology	ロードバランサーのトポロジー設定。
OctaviaLogOffload	true の場合、amphora からのログメッセージは管理ログエンドポイントに転送され、コントローラーログとともに保存されます。デフォルト値は false です。
OctaviaLogOffloadProtocol	RSyslog ログのオフロード機能に使用するプロトコル。デフォルト値は udp です。
OctaviaMinimumTlsVersion	リスナーとプールに許可される最小の TLS バージョン。
OctaviaMultiVcpuFlavorId	複数の vCPU を備えた amphora の nova フレーバーの名前。デフォルト値は amphora-mvcpu-ha です。
OctaviaMultiVcpuFlavorProperties	active-standby トポロジーと垂直スケーリングのための複数の vCPU を備えた amphora の nova フレーバーを記述するディクショナリー。デフォルト値は {'ram': '4096', 'disk': '3', 'vcpus': '4'} です。
OctaviaPoolTlsVersions	新しい TLS 対応プールに使用する TLS バージョンのリスト。デフォルト値は ['TLSv1.2', 'TLSv1.3'] です。
OctaviaTenantLogFacility	テナントトラフィックフローログメッセージに使用する syslog "LOG_LOCAL" 機能。デフォルト値は 0 です。
OctaviaTenantLogTargets	テナントトラフィックフローログメッセージを受信するための syslog エンドポイントのリスト (host:port コンマ区切りリスト)。
OctaviaTimeoutClientData	フロントエンドクライアントの停止状態タイムアウト。デフォルト値は 50000 です。
OctaviaTimeoutMemberData	バックエンドメンバーの停止状態タイムアウト。デフォルト値は 50000 です。

パラメーター	説明
OctaviaTlsCiphersProhibitList	OpenSSL 暗号のリスト。これらの暗号の使用はブロックされます。
RedisPassword	redis サービスアカウントのパスワード。

第16章 MESSAGING パラメーター

メッセージングパラメーターを使用してメッセージキューサービスを変更できます。

パラメーター	説明
RpcCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
RpcPassword	メッセージングバックエンドのパスワード
RpcPort	メッセージングバックエンドのネットワークポート。デフォルト値は 5672 です。
RpcUserName	メッセージングバックエンドのユーザー名。デフォルト値は guest です。
RpcUseSSL	メッセージング用ホストへの SSL 接続を指定するためのメッセージングクライアントのサブスクリバードパラメーター。デフォルト値は false です。

第17章 NETWORKING (NEUTRON) パラメーター

ネットワークパラメーターを使用して neutron サービスを変更できます。

パラメーター	説明
CertificateKeySize	証明書の作成時に使用される秘密鍵のサイズを指定します。デフォルト値は 2048 です。
ContainerOvnCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
DerivePciWhitelistEnabled	pci パススルーホワイトリストの自動化を有効にするかどうか。デフォルト値は true です。
DhcpAgentNotification	DHCP エージェントの通知を有効にします。デフォルト値は false です。
DockerAdditionalSockets	Docker デーモンがバインドする追加のドメインソケット (他のコンテナを起動するコンテナにマウントするのに役立つ)。デフォルト値は ['/var/lib/openstack/docker.sock'] です。
DockerNeutronDHCPAgentUlimit	OpenStack Networking (neutron) DHCP エージェントコンテナのユーザーリミット。デフォルト値は ['nofile=16384'] です。
DockerNeutronL3AgentUlimit	OpenStack Networking (neutron) L3 エージェントコンテナのユーザーリミット。デフォルト値は ['nofile=16384'] です。
DockerOpenvswitchUlimit	Openvswitch コンテナの Ulimit。デフォルト値は ['nofile=16384'] です。
DockerPuppetMountHostPuppet	コンテナ化された puppet の実行で、ベアメタルホストのモジュールを使用するかどうか。デフォルトは true です。直接コンテナからの puppet モジュールを使用するには、false に設定します。デフォルト値は true です。
DockerSRIOVUlimit	SR-IOV コンテナの Ulimit。デフォルト値は ['nofile=16384'] です。
EnableSQLAlchemyCollectd	SQLAlchemy-collectd サーバープラグインを有効にするには true に設定します。デフォルト値は false です。

パラメーター	説明
EnableVLANTransparency	True の場合には、プラグインが VLAN トランスペアレントネットワークの作成に対応していれば、そのタイプのネットワークの作成を許可します。デフォルト値は false です。
EnforceSecureRbac	このオプションを True に設定すると、 oslo_policy enforce_new_defaults と oslo_policy enforce_scope を True に設定して、各 OpenStack サービスが Secure RBAC を適用するように設定されます。これにより、システムおよびプロジェクトスコープのサポートや keystone のデフォルトロール、管理者、メンバー、リーダーなど、OpenStack サービス全体で一貫性のある一連の RBAC ペルソナが導入されます。デプロイメント内のすべてのサービスがセキュアな RBAC を実際にサポートするまで、この機能は有効にしないでください。デフォルト値は false です。
MemcacheUseAdvancedPool	高度な (eventlet に対して安全な) memcached クライアントプールを使用します。デフォルト値は true です。
NeutronAgentDownTime	エージェントをダウンと見なす秒数。エージェントが完全にダウンしていることを確認するために、少なくとも NeutronGlobalReportInterval の 2 倍にする必要があります。デフォルト値は 600 です。
NeutronAgentExtensions	OpenStack Networking (neutron) エージェント用に有効化されている拡張機能のコンマ区切りリスト。デフォルト値は qos です。
NeutronAllowL3AgentFailover	自動の l3-agent フェイルオーバーを許可します。デフォルト値は True です。
NeutronApiOptEnvVars	オプションの環境変数のハッシュ
NeutronApiOptVolumes	マウントされるオプションのボリュームのリスト
NeutronBridgeMappings	使用する論理ブリッジから物理ブリッジへのマッピング。デフォルト設定 (datacentre:br-ex) では、 br-ex (ホスト上の外部ブリッジ) をプロバイダーネットワークが使用可能な物理名 datacentre にマッピングします (例: デフォルトのフローティングネットワーク)。これを変更する場合には、別のインストール後のネットワークスクリプトを使用するか、マッピングネットワーク名として datacentre を保持するようにしてください。デフォルト値は datacentre:br-ex です。

パラメーター	説明
NeutronCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
NeutronCorePlugin	ネットワークのコアプラグイン。この値は、 neutron.core_plugins の名前空間から読み込まれるエントリーポイントでなければなりません。デフォルト値は ml2 です。
NeutronDBSyncExtraParams	neutron-db-manage upgrade head コマンドに追記する追加のコマンドラインパラメーターの文字列
NeutronDefaultAvailabilityZones	リソースがアベイラビリティゾーンヒントを設定せずに作成されている場合に、OpenStack Networking (neutron) が使用するデフォルトのネットワークアベイラビリティゾーンのコンマ区切りリスト。設定されていない場合、OpenStack Networking (neutron) ネットワークサービスに対して AZ は設定されません。
NeutronDhcpAgentAvailabilityZone	OpenStack Networking (neutron) DHCP エージェントのアベイラビリティゾーン。設定されていない場合、OpenStack Networking (neutron) ネットワークサービスに対して AZ は設定されません。
NeutronDhcpAgentDnsmasqDnsServers	dnsmasq フォワーダーとして使用するサーバーのリスト
NeutronDhcpAgentDnsmasqEnableAddr6List	ポートに同じサブネットに複数の IPv6 アドレスがある場合に、dhcp-host エントリーをアドレスのリストで有効にします。デフォルト値は true です。
NeutronDhcpAgentsPerNetwork	ネットワークごとにスケジュールする DHCP エージェントの数。デフォルト値は 0 です。
NeutronDhcpCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
NeutronDhcpLoadType	アベイラビリティゾーン対応ネットワークスケジューラーへの追加。デフォルト値は networks です。
NeutronDhcpOvsIntegrationBridge	使用する Open vSwitch ブリッジの名前
NeutronDhcpServerBroadcastReply	DHCP 応答でブロードキャストを使用するための OpenStack Networking (neutron) DHCP エージェント。デフォルト値は false です。

パラメーター	説明
NeutronDnsDomain	ホスト名の構築に使用するドメイン。デフォルト値は openstacklocal です。
NeutronEnableARPResponder	OVS エージェントで ARP レスポンダー機能を有効にします。デフォルト値は false です。
NeutronEnableDnsmasqDockerWrapper	OpenStack Networking (neutron) が別のコンテナで dnsmasq を起動するように、dnsmasq ラッパースクリプトを生成します。デフォルト値は true です。
NeutronEnableDVR	分散仮想ルーター (DVR: Distributed Virtual Router) を有効にします。
NeutronEnableForceMetadata	True の場合には、DHCP は常に仮想マシンにメタデータをルーティングします。デフォルト値は false です。
NeutronEnableHaproxyDockerWrapper	OpenStack Networking (neutron) が別のコンテナで haproxy を起動するように、ラッパースクリプトを生成します。デフォルト値は true です。
NeutronEnableIcmpSnooping	IGMP スヌーピングを有効にします。デフォルト値は false です。
NeutronEnableInternalDNS	True の場合、仮想マシン間の名前解決を提供する内部 OpenStack Networking (neutron) DNS サーバーを有効にします。 NeutronDhcpAgentDnsmasqDnsServers が設定されている場合、このパラメーターは機能しません。デフォルト値は false です。
NeutronEnableIsolatedMetadata	True の場合には、DHCP は常に分離ネット上でメタデータのサポートを許可します。デフォルト値は false です。
NeutronEnableKeepalivedWrapper	OpenStack Networking (neutron) が別のコンテナで keepalived プロセスを起動するように、ラッパースクリプトを生成します。デフォルト値は true です。
NeutronEnableL2Pop	OpenStack Networking (neutron) エージェントの L2 Population 機能を有効化/無効化します。デフォルト値は False です。

パラメーター	説明
NeutronEnableMetadataNetwork	True の場合には、DHCP はメタデータネットワークを提供します。このパラメーターを True に指定するには、 NeutronEnableIsolatedMetadata または NeutronEnableForceMetadata のいずれかのパラメーターも True でなければなりません。デフォルト値は false です。
NeutronExcludeDevices	network_device を、仮想ネットワークに使用してはならないエージェントのノード固有の仮想機能のリストにマッピングする <network_device>: <excluded_devices> のリスト。excluded_devices は、network_device から除外する仮想機能のセミコロン区切りリストです。マッピングの network_device が physical_device_mappings リストに表示されます。
NeutronFirewallDriver	OpenStack Networking (neutron) のセキュリティーグループ機能を有効にするためのファイアウォールドライバー。
NeutronFlatNetworks	フラットネットワーク名がプラグインで設定されるように指定します。デフォルト値は datacentre です。
NeutronGeneveMaxHeaderSize	Geneve カプセル化ヘッダーのサイズ。デフォルト値は 38 です。
NeutronGlobalPhysnetMtu	下層の物理ネットワークの MTU。OpenStack Networking (neutron) はこの値を使用して、全仮想ネットワークのコンポーネントの MTU を算出します。フラットネットワークおよび VLAN ネットワークの場合は、OpenStack Networking はこの値を変更せずに使用します。VXLAN などのオーバーレイネットワークの場合は、OpenStack Networking は自動的にこの値からオーバーレイプロトコルのオーバーヘッドを除算します。デフォルト値は 0 です。
NeutronGlobalReportInterval	サーバーに状態を報告するノード間の秒数。NeutronAgentDownTime 未満である必要があります。NeutronAgentDownTime の半分以下にすると最適です。デフォルト値は 300 です。
NeutronInterfaceDriver	OpenStack Networking (neutron) DHCP エージェントインターフェイスドライバー。デフォルト値は neutron.agent.linux.interface.OVSInterfaceDriver です。

パラメーター	説明
NeutronL3AgentAvailabilityZone	OpenStack Networking (neutron) L3 エージェントのアベイラビリティゾーン。設定されていない場合、OpenStack Networking (neutron) ネットワークサービスに対して AZ は設定されません。
NeutronL3AgentExtensions	OpenStack Networking (neutron) L3 エージェント用に有効化されている拡張機能のコンマ区切りリスト。
NeutronL3AgentLoggingBurstLimit	rate_limit あたりの最大パケット数。デフォルト値は 25 です。
NeutronL3AgentLoggingLocalOutputLogBase	エージェント側の出力ログファイルパス、デフォルトの syslog ファイル。
NeutronL3AgentLoggingRateLimit	1秒あたりにログに記録されるパケットの最大数。デフォルト値は 100 です。
NeutronL3AgentMode	L3 エージェントのエージェントモード。 legacy または dvr_snat のいずれかを指定する必要があります。デフォルト値は legacy です。
NeutronL3AgentRadvdUser	radvd に渡されるユーザー名。root 権限をドロップし、ユーザー ID を username および group ID を、ユーザー名のプライマリーグループに変更します。ユーザーを指定しないと、L3 エージェントを実行するユーザーが渡されます。root を指定すると、radvd が root として生成されるため、username パラメーターは渡されません。デフォルト値は root です。
NeutronMechanismDrivers	OpenStack Networking (neutron) テナントネットワークのメカニズムドライバー。デフォルト値は ovn です。
NeutronMetadataProxySharedSecret	スプーフィングを防ぐための共有シークレット
NeutronMetadataWorkers	OpenStack Networking (neutron) OVN メタデータエージェントのワーカープロセスの数を設定します。デフォルト値では設定が設定されないままになり、システムに依存するデフォルトが選択されます (通常はプロセッサの数)。これにより、コア数の多いシステムでは多数のプロセスとメモリー消費が発生する可能性があることに注意してください。このようなシステムでは、負荷要件に一致するデフォルト以外の値を選択することを推奨します。

パラメーター	説明
NeutronML2PhysicalNetworkMtus	物理ネットワークと MTU 値のマッピングリスト。マッピングのフォーマットは <physnet>:<mtu val> です。このマッピングにより、ML2 プラグインのデフォルト segment_mtu 値とは異なる物理ネットワーク MTU 値を指定し、特定のネットワークについて global_physnet_mtu からの値を上書きすることができます。
NeutronNetworkSchedulerDriver	アベイラビリティゾーンに使用するネットワークスケジューラドライバー。デフォルト値は neutron.scheduler.dhcp_agent_scheduler.AZ AwareWeightScheduler です。
NeutronNetworkType	OpenStack Networking (neutron) のテナントネットワークタイプ。デフォルト値は geneve です。この値を変更する場合は、新しい値がパラメーター OVNEncapType と一致することを確認してください。たとえば、ML2/OVN 環境で Geneve の代わりに VXLAN を使用する場合は、 NeutronNetworkType と OVNEncapType の両方が vxlan に設定されていることを確認します。デフォルト値は geneve です。
NeutronNetworkVLANRanges	サポートされる OpenStack Networking (neutron) ML2 および Open vSwitch VLAN マッピングの範囲。デフォルトでは、物理ネットワーク datacentre 上の VLAN を許可するように設定されています (NeutronBridgeMappings を参照)。デフォルト値は datacentre:1:1000 です。
NeutronOverlayIPVersion	全オーバーレイネットワークエンドポイントに使用される IP バージョン。デフォルト値は 4 です。
NeutronOVNLoggingBurstLimit	rate_limit あたりの最大パケット数。デフォルト値は 25 です。
NeutronOVNLoggingLocalOutputLogBase	エージェント側の出力ログファイルパス、デフォルトの syslog ファイル。
NeutronOVNLoggingRateLimit	1秒あたりにログに記録されるパケットの最大数。デフォルト値は 100 です。
NeutronOVSAgentLoggingBurstLimit	rate_limit あたりの最大パケット数。デフォルト値は 25 です。
NeutronOVSAgentLoggingLocalOutputLogBase	エージェント側の出力ログファイルパス、デフォルトの syslog ファイル。

パラメーター	説明
NeutronOVSAgentLoggingRateLimit	1秒あたりにログに記録されるパケットの最大数。デフォルト値は 100 です。
NeutronOVSAgentLoggingRateLimit	セキュリティグループの実装に使用するファイアウォールドライバーのクラス名を設定します。指定可能な値は、システム設定により異なります。たとえば、 noop 、 openvswitch 、 iptables_hybrid です。デフォルト値である空の文字列を指定すると、デフォルトのサポート対象設定となります。
NeutronOvsIntegrationBridge	使用する Open vSwitch ブリッジの名前
NeutronOvsResourceProviderBandwidths	<bridge>:<egress_bw>:<ingress_bw> タプルのコンマ区切りリスト。指定のブリッジで利用可能な帯域幅が指定の方向に表示されます。方向は、VM の視点から行われます。帯域幅は、1秒あたりのキロバイト単位 (kbps) で測定されます。ブリッジは bridge_mappings に値として表示されるはずですが、
NeutronOVSTunnelCsum	発信 IP パケットを転送する GRE/VXLAN トンネルのトンネルヘッダーチェックサムを設定または設定します。デフォルト値は false です。
NeutronOvsVnicTypeBlacklist	管理上の理由で OVS メカニズムドライバーにより OpenStack Networking (neutron) でのサポートが禁止されている仮想 NIC 種別のコンマ区切りリスト
NeutronPassword	OpenStack Networking (neutron) サービスおよびデータベースアカウントのパスワード
NeutronPermittedEthertypes	OpenStack Networking (neutron) ファイアウォールで設定する追加の ethertype を設定します。
NeutronPhysicalDevMappings	<physical_network>:<physical device> のリスト。サーバーの network_vlan_ranges にリスト表示されるすべての物理ネットワークには、各エージェントの適切なインターフェイスへのマッピングが必要です。例: "tenant0:ens2f0,tenant1:ens2f1"。
NeutronPluginExtensions	有効な拡張プラグインのコンマ区切りリスト。デフォルト値は qos,port_security,dns_domain_ports です。
NeutronPluginMI2PuppetTags	puppet での設定ファイルを生成するのに使用される puppet リソースタグ名。デフォルト値は neutron_plugin_mi2 です。
NeutronPortQuota	テナントごとに許容できるポート数。負の値は無制限を意味します。デフォルト値は 500 です。

パラメーター	説明
NeutronRouterSchedulerDriver	アベイラビリティゾーンに使用するルータースケジューラードライバー。デフォルト値は neutron.scheduler.l3_agent_scheduler.AZLeastRoutersScheduler です。
NeutronRpcWorkers	OpenStack Networking (neutron) サービスの RPC ワーカー数を設定します。このパラメーターを指定しないと、NeutronWorkers の値が用いられます。それも指定されていない場合には、デフォルト値は未設定のままとなり、システムに依存したデフォルトが選択されます (通常は 1)。
NeutronSecurityGroupQuota	テナントごとに許容できるセキュリティグループ数。負の数は無制限を意味します。デフォルト値は 10 です。
NeutronServicePlugins	サービスプラグインのエントリーポイントのコンマ区切りリスト。デフォルト値は qos,ovn-router,trunk,segments,port_forwarding,log です。
NeutronSriovAgentExtensions	OpenStack Networking (neutron) SR-IOV エージェント用に有効化されている拡張機能のコンマ区切りリスト。
NeutronSriovResourceProviderBandwidths	<network_device>:<egress_bw>:<ingress_bw>:<ingress_bw> タプルのコンマ区切りリスト。指定のデバイスの利用可能な帯域幅が指定の方向に表示されます。方向は、VM の視点から行われます。帯域幅は、1秒あたりのキロバイト単位 (kbps) で測定されます。デバイスは、physical_device_mappings に値として表示されるはずですが、ここでは省略されています。
NeutronSriovVnicTypeBlacklist	管理上の理由で SR-IOV メカニズムドライバーにより OpenStack Networking (neutron) でのサポートが禁止されている仮想 NIC 種別のコンマ区切りリスト
NeutronTunnelIdRanges	テナントネットワークの割り当てに使用できる GRE トンネリング ID の範囲を列挙した <tun_min>:<tun_max> タプルのコンマ区切りリスト。デフォルト値は ['1:4094'] です。
NeutronTunnelTypes	OpenStack Networking (neutron) テナントネットワークのトンネルタイプ。デフォルト値は vxlan です。
NeutronTypeDrivers	読み込むネットワーク種別ドライバーのエントリーポイントのコンマ区切りリスト。デフォルト値は geneve,vxlan,vlan,flat です。

パラメーター	説明
NeutronVhostuserSocketDir	OVS 向けの vhost-user ソケットディレクトリー
NeutronVniRanges	テナントネットワークの割り当てに使用できる VXLAN VNI ID の範囲を列挙した <vni_min>: <vni_max> タプルのコンマ区切りリスト。デフォルト値は ['1:65536'] です。
NeutronWorkers	OpenStack Networking サービスの API および RPC ワーカー数を設定します。ワーカー数が多いと、システムのプロセス数も増え、メモリーを過剰消費してしまう点に注意してください。CPU コア数が多いシステムでは、デフォルト以外の適切な値を選択することを推奨します。 0 の場合は、OpenStack の内部デフォルト値 (ノードにある CPU コア数) に設定されます。
NotificationDriver	通知の送信を処理する単一または複数のドライバー。デフォルト値は noop です。
OVNAvailabilityZone	ovs db で設定する az オプション (例: [az-0, az-1, az-2])。
OVNCMSOptions	OVS データベースで設定する CMS オプション
OVNContainerCpusetCpus	コンテナが使用することのできる CPU またはコアを、特定の CPU/コアに制限します。シングルコア (例: 0)、コンマ区切りのリスト (例: 0,1)、範囲 (例: 0-3)、またはこれらを組み合わせて (例: 0-3,7,11-15) 指定できます。可能な限り最高のパフォーマンスを得るには、選択したコアをゲストおよびハイパーバイザーから分離する必要があります。
OVNControllerImageUpdateTimeout	更新中、コンテナイメージが更新されるまで待機する時間 (秒単位)。デフォルト値は 600 です。
OVNControllerUpdateTimeout	更新中、コンテナが更新されるまで待機する時間 (秒単位)。デフォルト値は 600 です。
OVNDbConnectionTimeout	OVSDb 接続トランザクションのタイムアウト (秒単位)。デフォルト値は 180 です。
OvnDBSCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
OVNDnsServers	DNS フォワーダーとして使用するサーバーのリスト

パラメーター	説明
OVNEmitNeedToFrag	MTU の不一致の場合に need to frag パケットを発行するように OVN を設定します。この設定を有効にする前に、ホストカーネル (バージョン ≥ 5.2) でサポートされていることを確認するか、コマンド <code>ovs-appctl -t ovs-vswitchd dpif/show-dp-features br-int grep "Check pkt length action"</code> の出力を確認します。デフォルト値は false です。
OVNEnableHaproxyDockerWrapper	haproxy が別のコンテナで起動するように、ラップスクリプトを生成します。デフォルト値は true です。
OVNEncapTos	Open_vSwitch データベースインターフェイステーブルで指定されたとおり、OVN トンネルインターフェイスの option:tos に適用される値。この機能は OVN v21.12.0 からサポートされています。デフォルト値は 0 です。
OVNEncapType	OVN で使用されるカプセル化のタイプ。 geneve または vxlan が可能です。デフォルト値は geneve です。この値を変更する場合は、新しい値がパラメーター NeutronNetworkType にもリストされていることを確認してください。たとえば、 OVNEncapType を vxlan に変更する場合は、 NeutronNetworkType のリストに vxlan が含まれていることを確認してください。デフォルト値は geneve です。
OvnHardwareOffloadedQos	ハードウェアオフロードポートの QoS サポートを有効にします。デフォルト値は false です。
OVNIntegrationBridge	OVN Controller が統合ブリッジとして使用する OVS ブリッジ名。デフォルト値は br-int です。
OvnMetadataCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
OVNMetadataEnabled	Metadata サービスを有効にする必要があるかどうかを定義します。デフォルト値は true です。
OVNNeutronSyncMode	OpenStack Networking (neutron) データベースと OVN データベースの同期モード。デフォルト値は log です。
OVNNorthboundClusterPort	OVN ノースバウンド DB サーバーのクラスターポート。デフォルト値は 6643 です。
OVNNorthboundServerPort	OVN Northbound DB サーバーのポート。デフォルト値は 6641 です。

パラメーター	説明
OVNOfctrlWaitBeforeClear	すべてのオープンフロールールをクリアし、新しいルールをインストールする前に ovn-controller が起動時に待機する時間 (ミリ秒単位) を設定します。デフォルト値は 8000 です。
OVNOpenflowProbeInterval	OpenvSwitch 統合ブリッジへの OpenFlow 接続の停止状態検出間隔 (秒単位)。デフォルト値は 60 です。
OVNOvsdbProbeInterval	OVSDB セッションのプロブ間隔 (ミリ秒)。デフォルト値は 60000 です。
OVNQosDriver	OpenStack Networking (neutron) QOS サービスプラグイン用 OVN 通知ドライバー。デフォルト値は ovn-qos です。
OVNRemoteProbeInterval	検出間隔 (ミリ秒単位)。デフォルト値は 60000 です。
OVNSouthboundClusterPort	OVN サウスバウンド DB サーバーのクラスターポート。デフォルト値は 6644 です。
OVNSouthboundServerPort	OVN Southbound DB サーバーのポート。デフォルト値は 6642 です。
OVNStaticBridgeMacMappings	静的 OVN ブリッジの MAC アドレスマッピング。OpenStack Networking (neutron) ポートを作成することにより、一意の OVN ブリッジ MAC アドレスが動的に割り当てられます。OpenStack Networking (neutron) が使用できない場合 (スタンドアロンデプロイメントなど)、このパラメーターを使用して静的な OVN ブリッジの MAC アドレスを指定します。例: controller-0: datacenter: 00:00:5E:00:53:00 provider: 00:00:5E:00:53:01 compute-0: datacenter: 00:00:5E:00:54:00 provider: 00:00:5E:00:54:01.
OvsDisableEMC	OVS 完全一致キャッシュを無効にします。デフォルト値は false です。
OvsHwOffload	OVS ハードウェアオフロードを有効にします。この機能は、OVS 2.8.0 からサポートされています。デフォルト値は false です。
PythonInterpreter	python および ansible の動作に使用する python インタープリター。デフォルト値は、`\$(command -v python3` `command -v python)` です。

パラメーター	説明
TenantNetPhysnetMtu	<p>下層の物理ネットワークの MTU。OpenStack Networking (neutron) はこの値を使用して、全仮想ネットワークのコンポーネントの MTU を算出します。フラットネットワークおよび VLAN ネットワークの場合は、OpenStack Networking (neutron) はこの値を変更せずに使用します。VXLAN などのオーバーレイネットワークの場合は、OpenStack Networking (neutron) は自動的にこの値からオーバーレイプロトコルのオーバーヘッドを除算します。(network_data.yaml の Tenant ネットワークの MTU 設定で、このパラメーターを制御します)デフォルト値は 1500 です。</p>

第18章 OBJECT STORAGE (SWIFT) パラメーター

オブジェクトストレージパラメーターを使用して swift サービスを変更できます。

パラメーター	説明
EnforceSecureRbac	このオプションを True に設定すると、 oslo_policy enforce_new_defaults と oslo_policy enforce_scope を True に設定して、各 OpenStack サービスが Secure RBAC を適用するように設定されます。これにより、システムおよびプロジェクトスコープのサポートや keystone のデフォルトロール、管理者、メンバー、リーダーなど、OpenStack サービス全体で一貫性のある一連の RBAC ペルソナが導入されます。デプロイメント内のすべてのサービスがセキュアな RBAC を実際にサポートするまで、この機能は有効にしないでください。デフォルト値は false です。
MemcachedTLS	Memcached サービスで TLS を有効にするには、True に設定します。すべてのサービスが Memcached TLS をサポートする訳ではないので、移行期間中、Memcached は 2 つのポート (MemcachedPort パラメーター (上記) で設定したポート、および TLS を使用しない 11211) をリスンします。デフォルト値は false です。
SwiftAccountWorkers	Swift アカウントサービスのワーカー数。デフォルト値は 0 です。
SwiftCeilometerIgnoreProjects	無視するプロジェクト名のコンマ区切りのリスト。デフォルト値は ['service'] です。
SwiftCeilometerPipelineEnabled	False に設定してオブジェクトストレージプロキシの ceilometer パイプラインを無効にします。デフォルト値は false です。
SwiftContainerSharderEnabled	Swift コンテナ細分化サービスを有効にするには、True に設定します。デフォルト値は false です。
SwiftContainerWorkers	Swift アカウントサービスのワーカー数。デフォルト値は 0 です。
SwiftCorsAllowedOrigin	このリソースがリクエストの元ヘッダーで受け取るドメインと共有されるかどうかを示します。
SwiftEncryptionEnabled	Swift 内の Data at Rest の暗号化を有効にするには、True に設定します。デフォルト値は false です。

パラメーター	説明
SwiftHashPrefix	リング内でマッピングを決定するためのハッシングを行う際に追加のソルトとして使用するランダム文字列。
SwiftHashSuffix	リング内でマッピングを決定するためのハッシングを行う際にソルトとして使用するランダム文字列
SwiftMinPartHours	リバランスの後にリング内のパーティションを移動できるようになるまでの最小時間 (時間単位)。デフォルト値は 1 です。
SwiftMountCheck	ルートデバイスに誤って書き込まれないようにデバイスがマウントされているかどうかを確認します。デフォルト値は false です。
SwiftObjectWorkers	Swift アカウントサービスのワーカー数。デフォルト値は 0 です。
SwiftPartPower	Object Storage リングの構築時の Partition Power。デフォルト値は 10 です。
SwiftPassword	Object Storage サービスアカウントのパスワード
SwiftProxyNodeTimeout	swift-proxy からアカウント、コンテナ、オブジェクトサービスへのリクエストのタイムアウト。デフォルト値は 60 です。
SwiftProxyRecoverableNodeTimeout	swift-proxy から swift a/c/o サービスに移行する GET/HEAD リクエストのタイムアウト。デフォルト値は 30 です。
SwiftRawDisks	Object Storage バックエンドに使用する追加の RAW デバイス(例: {sdb: {}})。
SwiftReplicas	Object Storage リングで使用するレプリカ数。デフォルト値は 3 です。
SwiftRingBuild	Object Storage リングを管理するかどうか。デフォルト値は true です。
SwiftRingGetTempurl	リングのダウンロード元の一時的な Swift URL
SwiftRingPutTempurl	リングのアップロード先の一時的な Swift URL
SwiftUseLocalDir	リングの構築時に Object Storage サービスにローカルディレクトリーを使用します。デフォルト値は true です。

パラメーター	説明
SwiftUseNodeDataLookup	永続的な命名を使用するには、NodeDataLookup をディスクデバイスに使用します。デフォルト値は false です。
SwiftWorkers	Object Storage サービスのワーカー数。ワーカー数が多いと、システムのプロセス数も増え、メモリーを過剰消費してしまう点に注意してください。CPU コア数が多いシステムでは、デフォルト以外の適切な値を選択することを推奨します。 0 の場合は、OpenStack の内部デフォルト値 (ノードにある CPU コア数) に設定されます。デフォルト値は 0 です。

第19章 ORCHESTRATION (HEAT) パラメーター

オーケストレーションパラメーターを使用して Heat サービスを変更できます。

パラメーター	説明
ApacheCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
ApacheTimeout	Apache のタイムアウト (秒単位)。Apache が I/O 操作を待機する時間を定義します。デフォルト値は 90 です。
CertificateKeySize	証明書の作成時に使用される秘密鍵のサイズを指定します。デフォルト値は 2048 です。
ClientRetryLimit	クライアントが一時的なエラーに対して再試行を行います。デフォルト値は 2 です。
EnableCache	memcached でキャッシュを有効にします。デフォルト値は true です。
EnableSQLAlchemyCollectd	SQLAlchemy-collectd サーバープラグインを有効にするには true に設定します。デフォルト値は false です。
EnforceSecureRbac	このオプションを True に設定すると、 oslo_policy enforce_new_defaults と oslo_policy enforce_scope を True に設定して、各 OpenStack サービスが Secure RBAC を適用するように設定されます。これにより、システムおよびプロジェクトスコープのサポートや keystone のデフォルトロール、管理者、メンバー、リーダーなど、OpenStack サービス全体で一貫性のある一連の RBAC ペルソナが導入されます。デプロイメント内のすべてのサービスがセキュアな RBAC を実際にサポートするまで、この機能は有効にしないでください。デフォルト値は false です。
HeatApiOptEnvVars	オプションの環境変数のハッシュ
HeatApiOptVolumes	マウントされるオプションのボリュームのリスト
HeatAuthEncryptionKey	heat-engine の認証暗号鍵
HeatConfigureDelegatedRoles	権限を委譲されたロールを作成します。デフォルト値は false です。
HeatConvergenceEngine	コンバージェンスアーキテクチャーの heat エンジン を有効にします。デフォルト値は true です。

パラメーター	説明
HeatCorsAllowedOrigin	このリソースがリクエストの元ヘッダーで受け取るドメインと共有されるかどうかを示します。
HeatCronPurgeDeletedAge	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Age)。デフォルト値は 30 です。
HeatCronPurgeDeletedAgeType	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Age type)。デフォルト値は days です。
HeatCronPurgeDeletedDestination	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Log destination)。デフォルト値は /dev/null です。
HeatCronPurgeDeletedEnsure	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Ensure)。デフォルト値は present です。
HeatCronPurgeDeletedHour	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Hour)。デフォルト値は 0 です。
HeatCronPurgeDeletedMaxDelay	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Max Delay)。デフォルト値は 3600 です。
HeatCronPurgeDeletedMinute	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Minute)。デフォルト値は 1 です。
HeatCronPurgeDeletedMonth	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Month)。デフォルト値は * です。
HeatCronPurgeDeletedMonthday	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Month Day)。デフォルト値は * です。
HeatCronPurgeDeletedUser	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (User)。デフォルト値は heat です。
HeatCronPurgeDeletedWeekday	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Week Day)。デフォルト値は * です。

パラメーター	説明
HeatEnableDBPurge	OpenStack Orchestration (heat) データベースでソフト削除された行をパージする cron ジョブを作成するかどうか。デフォルト値は true です。
HeatEngineOptEnvVars	オプションの環境変数のハッシュ
HeatEngineOptVolumes	マウントされるオプションのボリュームのリスト
HeatEnginePluginDirs	プラグインを検索するディレクトリーアレイ
HeatMaxJsonBodySize	OpenStack Orchestration (heat) API の JSON リクエスト本文の RAW データ型最大バイトサイズ。デフォルト値は 4194304 です。
HeatMaxNestedStackDepth	スタックのネスト深さの最大値。デフォルト値は 6 です。
HeatMaxResourcesPerStack	最上位のスタックごとに許容できる最大リソース。-1は無制限に設定されます。デフォルト値は 1000 です。
HeatPassword	Orchestration サービスおよびデータベースアカウントのパスワード
HeatReauthenticationAuthMethod	長時間のタスクが完了するように、トークン期限切れ時の再認証を許可します。この設定は指定したあらゆるユーザートークンの有効期限に優先する点に注意してください。
HeatStackDomainAdminPassword	OpenStack Identity (keystone) の OpenStack Orchestration (heat) ドメインの管理者パスワード
HeatWorkers	OpenStack Orchestration (heat) サービスのワーカー数。ワーカー数が多いと、システムのプロセス数も増え、メモリーを過剰消費してしまう点に注意してください。CPU コア数が多いシステムでは、デフォルト以外の適切な値を選択することを推奨します。 0 の場合は、OpenStack の内部デフォルト値 (ノードにある CPU コア数) に設定されます。デフォルト値は 0 です。
HeatYaqlLimitIterators	yaql 式がその評価に使用することのできるコレクションの最大要素数。デフォルト値は 1000 です。
HeatYaqlMemoryQuota	yaql 式がその評価に使用することのできる最大のメモリーサイズ (バイト単位)。デフォルト値は 100000 です。

パラメーター	説明
MemcachedTLS	Memcached サービスで TLS を有効にするには、True に設定します。すべてのサービスが Memcached TLS をサポートする訳ではないので、移行期間中、Memcached は 2 つのポート (MemcachedPort パラメーター (上記) で設定したポート、および TLS を使用しない 11211) をリッスンします。デフォルト値は false です。
MemcacheUseAdvancedPool	高度な (eventlet に対して安全な) memcached クライアントプールを使用します。デフォルト値は true です。
NotificationDriver	通知の送信を処理する単一または複数のドライバー。デフォルト値は noop です。

第20章 SHARED FILE SYSTEM (MANILA) パラメーター

Shared File サービスのパラメーターを使用して manila サービスを変更できます。

パラメーター	説明
ApacheCertificateKeySize	このサービスの証明書の作成時に使用される秘密鍵のサイズを上書きします。
ApacheTimeout	Apache のタイムアウト (秒単位)。Apache が I/O 操作を待機する時間を定義します。デフォルト値は 90 です。
CephClusterName	Ceph クラスタ名。デフォルト値は ceph です。
CephConfigPath	Ceph クラスタ設定ファイルがホストに保存されているパス。デフォルト値は /var/lib/tripleo-config/ceph です。
CertificateKeySize	証明書の作成時に使用される秘密鍵のサイズを指定します。デフォルト値は 2048 です。
EnableSQLAlchemyCollectd	SQLAlchemy-collectd サーバープラグインを有効にするには true に設定します。デフォルト値は false です。
ManilaCephClientUserName	manila 統合用の Ceph クライアントのユーザー名。デフォルト値は manila です。
ManilaCephFSCephFSProtocolHelperType	cephfs バックエンドが有効な場合のプロトコル種別 (CEPHFS または NFS)。manila cephfs 環境ファイルにより設定します。デフォルト値は CEPHFS です。
ManilaCronDbPurgeAge	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Age)。デフォルト値は 30 です。
ManilaCronDbPurgeDestination	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Log destination)。デフォルト値は /var/log/manila/manila-rowsflush.log です。
ManilaCronDbPurgeHour	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Hour)。デフォルト値は 0 です。
ManilaCronDbPurgeMaxDelay	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Max Delay)。デフォルト値は 3600 です。

パラメーター	説明
ManilaCronDbPurgeMinute	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Minute)。デフォルト値は 1 です。
ManilaCronDbPurgeMonth	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Month)。デフォルト値は * です。
ManilaCronDbPurgeMonthday	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Month Day)。デフォルト値は * です。
ManilaCronDbPurgeUser	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (User)。デフォルト値は manila です。
ManilaCronDbPurgeWeekday	cron ジョブが削除済みとマークされかつ \$age よりも古いデータベースのエントリーをパージする (Week Day)。デフォルト値は * です。
ManilaEnabledShareProtocols	manila でのファイル共有の作成に許可されるプロトコルのリスト。これが設定されていない場合、リストは有効なストレージバックエンドにより推測されます。
ManilaIPv6	manila での IPv6 アクセスを有効にするには True に設定します。デフォルト値は False です。
ManilaPassword	Shared File サービスアカウントのパスワード
ManilaStorageAvailabilityZone	OpenStack Shared File Systems (manila) サービスのストレージアベイラビリティゾーン。デフォルト値は nova です。
ManilaWorkers	manila::wsgi::apache のワーカー数を設定します。デフォルト値は、物理ノードにある仮想 CPU コア数と同じ数値です。
MemcacheUseAdvancedPool	高度な (eventlet に対して安全な) memcached クライアントプールを使用します。デフォルト値は true です。
NotificationDriver	通知の送信を処理する単一または複数のドライバー。デフォルト値は noop です。

第21章 時刻に関するパラメーター

time パラメーターを使用して time synchronization サービスを変更できます。

パラメーター	説明
ChronyAclRules	NTP クライアントのアクセス制御リスト。デフォルトでは、どのクライアントも許可されません。デフォルト値は ['deny all'] です。
ChronyGlobalPoolOptions	chrony.conf で設定した NTP プールのデフォルトプールオプション。このパラメーターを指定すると、NtpIburstEnable、MaxPoll、および MinPoll は無視されます。
ChronyGlobalServerOptions	chrony.conf で設定した NTP サーバーのデフォルトサーバーオプション。このパラメーターを指定すると、NtpIburstEnable、MaxPoll、および MinPoll は無視されます。
EnablePackageInstall	デプロイ時にパッケージのインストールを有効にするには true に設定します。デフォルト値は false です。
MaxPoll	アップストリームのサーバーが NTP メッセージをポーリングする最大の間隔を 2 の累乗の秒数で指定します。設定可能な値は 4 から 17 です。デフォルト値は 10 です。
MinPoll	アップストリームのサーバーが NTP メッセージをポーリングする最小の間隔を 2 の累乗の秒数で指定します。最小のポーリング間隔のデフォルトは 6 (64 秒) です。設定可能な値は 4 から 17 です。デフォルト値は 6 です。
NtpIburstEnable	全 NTP ピアに対して iburst オプションを有効にするかどうかを指定します。iburst が有効な場合には、NTP サーバーに到達できなければ、NTP が 1 つではなく 8 つのパッケージのバーストを送信します。これは、初期同期を迅速化するために設計されています。デフォルト値は true です。
NtpPool	NTP プールのリスト。デフォルトは [] です。したがって、デフォルトでは NtpServer だけが使用されます。
NtpServer	NTP サーバーリスト。デフォルト値は ['0.pool.ntp.org', '1.pool.ntp.org', '2.pool.ntp.org', '3.pool.ntp.org'] です。

パラメーター	説明
TimeZone	オーバークラウド上で設定するタイムゾーン。デフォルト値は UTC です。

第22章 アップグレードのパラメーター

アップグレードパラメーターを使用してアップグレードプロセスの動作を変更できます。

パラメーター	説明
UpgradeInitCommand	アップグレードプロセスを初期化するためにすべてのオーバークラウドノード上で実行するコマンドまたはスクリプトのスニペット。たとえば、リポジトリーの切り替えなど。
UpgradeInitCommonCommand	アップグレードプロセスに必要な共通のコマンド。操作者は通常このパラメーターを変更する必要はなく、major-upgrade-composable-steps.yaml および major-upgrade-converge.yaml 環境ファイルで設定および設定解除されます。
UpgradeLeappCommandOptions	Leapp コマンドに追加するその他のコマンドラインオプション
UpgradeLeappDebug	Leapp の実行中にデバッグのアウトプットを出力します。デフォルト値は false です。
UpgradeLeappDevelSkip	開発/テスト環境で Leapp を実行する場合は、環境変数を設定して Leapp の確認を省略します。たとえば、LEAPP_DEVEL_SKIP_RHSM=1 と設定します。
UpgradeLeappEnabled	オペレーティングシステムのアップグレードに Leapp を使用します。デフォルト値は false です。
UpgradeLeappPostRebootDelay	マシンがリブートしてテストコマンドに応答するのを待つ最大の時間 (秒)。デフォルト値は 120 です。
UpgradeLeappRebootTimeout	Leapp による OS アップグレードフェーズのタイムアウト時間 (秒単位)。デフォルト値は 3600 です。
UpgradeLeappToInstall	Leapp によるアップグレード後にインストールするパッケージのリスト
UpgradeLeappToRemove	Leapp によるアップグレード時に削除するパッケージのリスト