



Red Hat OpenStack Platform 14

director を使ったベアメタルへの OpenShift Container Platform のインストール

director を使ってベアメタルに OpenShift Container Platform (OCP) をインストール
するためのガイド

Red Hat OpenStack Platform 14 director を使ったベアメタルへの OpenShift Container Platform のインストール

director を使ってベアメタルに OpenShift Container Platform (OCP) をインストールするためのガイド

OpenStack Team
rhos-docs@redhat.com

法律上の通知

Copyright © 2019 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, 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 Software Collections 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

本書では、CPUのパフォーマンスを向上させディスクおよびネットワークのレイテンシーを抑えるために、OpenShift Container Platform (OCP) クラスターをベアメタルにデプロイする方法について説明します。また、ベアメタルノードごとにホストされるコンテナアプリケーションの数を増やす方法についても言及します。

目次

第1章 DIRECTOR を使ったベアメタルへの OPENSIFT CONTAINER PLATFORM のインストール	3
1.1. 前提条件	3
1.2. DIRECTOR を使った OCP ノードのデプロイ	3
1.2.1. OCP ロール	3
1.2.2. OCP ロールの定義	4
1.2.3. コンテナレジストリーの設定	4
1.2.4. OCP プロファイルの作成	6
1.2.5. OpenShift 環境の定義	7
1.2.6. OpenShift リポジトリーへのオーバークラウドノードの登録	9
1.2.7. OCP ノードのデプロイ	10
1.2.8. OCP デプロイメントの確認	11
1.2.9. OCP を使ったテストアプリケーションのデプロイ	12
その他のリソース	12
第2章 仮想環境での OPENSIFT ON OPENSTACK	13
2.1. 前提条件	13
Red Hat OpenStack director からの検証の実行	13
コマンドラインからの検証の実行	13

第1章 DIRECTOR を使ったベアメタルへの OPENSIFT CONTAINER PLATFORM のインストール

本ガイドには、Red Hat OpenStack director を使用してベアメタルノードに Red Hat OpenShift Container Platform (OCP) クラスターをデプロイするための情報を記載します。

デプロイしたオーバークラウド内の仮想ノード上に OpenShift をインストールすることもできます。仮想環境内に OpenShift on OpenStack をインストールするには、「[仮想環境での OpenShift on OpenStack](#)」を参照してください。

1.1. 前提条件

- Red Hat OpenStack director がインストールされていること。

1.2. DIRECTOR を使った OCP ノードのデプロイ

director を使用してベアメタルノードに Red Hat OpenShift Container Platform (OCP) クラスターをデプロイすることができます。director はノードにオペレーティングシステムをデプロイし、続いて **openshift-ansible** を使用して OCP を設定します。director を使用してベアメタルノードを管理することもできます。

director は、**OpenShiftMaster**、**OpenShiftWorker**、および **OpenShiftInfra** のコンポーザブルロールを使用して OCP サービスをインストールします。**instackenv.json** を使用してベアメタルノードをインポートする場合には、タグ付けて特定のコンポーザブルロールを使用することができます。コンポーザブルロール使用の詳細な情報は、『[オーバークラウドの高度なカスタマイズ](#)』の「[コンポーザブルサービスとカスタムロール](#)」を参照してください。

1.2.1. OCP ロール

OpenShiftMaster ロールは以下のサービスで構成されます。

```
ServicesDefault:
- OS::Triple0::Services::ContainerImagePrepare
- OS::Triple0::Services::Docker
- OS::Triple0::Services::Haproxy
- OS::Triple0::Services::Keepalived
- OS::Triple0::Services::Ntp
- OS::Triple0::Services::OpenShift::Master
- OS::Triple0::Services::Rhsm
- OS::Triple0::Services::Sshd
- OS::Triple0::Services::TripleoFirewall
- OS::Triple0::Services::TripleoPackages
```

OpenShiftWorker ロールは以下のサービスで構成されます。

```
ServicesDefault:
- OS::Triple0::Services::Docker
- OS::Triple0::Services::Ntp
- OS::Triple0::Services::OpenShift::GlusterFS
- OS::Triple0::Services::OpenShift::Worker
- OS::Triple0::Services::Rhsm
- OS::Triple0::Services::Sshd
- OS::Triple0::Services::TripleoFirewall
```

OpenShiftInfra ロールはワーカーロールの種別で、インフラストラクチャーポッドだけを実行します。このロールは以下のサービスで構成されます。

```
ServicesDefault:
- OS::Triple0::Services::Docker
- OS::Triple0::Services::Ntp
- OS::Triple0::Services::OpenShift::GlusterFS
- OS::Triple0::Services::OpenShift::Infra
- OS::Triple0::Services::Rhsm
- OS::Triple0::Services::Sshd
- OS::Triple0::Services::TripleoFirewall
```

1.2.2. OCP ロールの定義

以下の手順により OCP ロールを生成します。

1. OCP ロールを生成します。

```
$ openstack overcloud roles generate -o
/home/stack/openshift_roles_data.yaml OpenShiftMaster
OpenShiftWorker OpenShiftInfra
```

2. OCP ロールを表示します。

```
$ openstack overcloud role list
```

出力には、**OpenShiftMaster**、**OpenShiftWorker**、および **OpenShiftInfra** のエントリが含まれているはずです。

3. **OpenShiftMaster** ロールについて、詳細な情報を表示します。

```
$ openstack overcloud role show OpenShiftMaster
```

1.2.3. コンテナレジストリーの設定

アンダークラウドをデプロイしたら、**director** を設定してコンテナレジストリーを特定する必要があります。

1. **/home/stack/containers-prepare-parameter.yaml** ファイルを生成します。

```
$ openstack tripleo container image prepare default \
--local-push-destination \
--output-env-file containers-prepare-parameter.yaml
```

たとえば、**/home/stack/containers-prepare-parameter.yaml** を編集して以下の設定を追加します。ご自分のデプロイメントに合わせて、これらの設定を変更してください。

```
parameter_defaults:
  ContainerImagePrepare:
    - push_destination: true
  set:
    ceph_image: rhceph-3-rhel7
    ceph_namespace: registry.access.redhat.com/rhceph
```



```
ceph_tag: latest
name_prefix: openstack-
name_suffix: ''
namespace: registry.access.redhat.com/rhosp15
neutron_driver: null
openshift_cluster_monitoring_image: ose-cluster-monitoring-
operator
openshift_cluster_monitoring_namespace:
registry.access.redhat.com/openshift3
openshift_cluster_monitoring_tag: v3.11
openshift_cockpit_image: registry-console
openshift_cockpit_namespace:
registry.access.redhat.com/openshift3
openshift_cockpit_tag: v3.11
openshift_configmap_reload_image: ose-configmap-reloader
openshift_configmap_reload_namespace:
registry.access.redhat.com/openshift3
openshift_configmap_reload_tag: v3.11
openshift_etcd_image: etcd
openshift_etcd_namespace: registry.access.redhat.com/rhel7
openshift_etcd_tag: latest
openshift_gluster_block_image: rhgs-gluster-block-prov-rhel7
openshift_gluster_image: rhgs-server-rhel7
openshift_gluster_namespace: registry.access.redhat.com/rhgs3
openshift_gluster_tag: latest
openshift_grafana_namespace:
registry.access.redhat.com/openshift3
openshift_grafana_tag: v3.11
openshift_heketi_image: rhgs-volmanager-rhel7
openshift_heketi_namespace: registry.access.redhat.com/rhgs3
openshift_kube_rbac_proxy_image: ose-kube-rbac-proxy
openshift_kube_rbac_proxy_namespace:
registry.access.redhat.com/openshift3
openshift_kube_rbac_proxy_tag: v3.11
openshift_kube_state_metrics_image: ose-kube-state-metrics
openshift_kube_state_metrics_namespace:
registry.access.redhat.com/openshift3
openshift_kube_state_metrics_tag: v3.11
openshift_namespace: registry.access.redhat.com/openshift3
openshift_oauth_proxy_tag: v3.11
openshift_prefix: ose
openshift_prometheus_alertmanager_tag: v3.11
openshift_prometheus_config_reload_image: ose-prometheus-
config-reloader
openshift_prometheus_config_reload_namespace:
registry.access.redhat.com/openshift3
openshift_prometheus_config_reload_tag: v3.11
openshift_prometheus_node_exporter_tag: v3.11
openshift_prometheus_operator_image: ose-prometheus-operator
openshift_prometheus_operator_namespace:
registry.access.redhat.com/openshift3
openshift_prometheus_operator_tag: v3.11
openshift_prometheus_tag: v3.11
openshift_tag: v3.11
tag: latest
tag_from_label: '{version}-{release}'
```

1.2.4. OCP プロファイルの作成

以下の手順では、物理ノードを OpenShift ノードとして登録する方法について説明します。

1. 各 OCP ロールのフレーバーを作成します。それぞれの要求に合わせて、これらの値を調整してください。

```
$ openstack flavor create --id auto --ram 4096 --disk 40 --vcpus 1 -
-swap 500 m1.OpenShiftMaster
$ openstack flavor create --id auto --ram 4096 --disk 40 --vcpus 1 -
-swap 500 m1.OpenShiftWorker
$ openstack flavor create --id auto --ram 4096 --disk 40 --vcpus 1 -
-swap 500 m1.OpenShiftInfra
```

2. フレーバーを必要なプロファイルにマッピングします。

```
$ openstack flavor set --property
"capabilities:profile"="OpenShiftMaster" --property
"capabilities:boot_option"="local" m1.OpenShiftMaster
$ openstack flavor set --property
"capabilities:profile"="OpenShiftWorker" --property
"capabilities:boot_option"="local" m1.OpenShiftWorker
$ openstack flavor set --property
"capabilities:profile"="OpenShiftInfra" --property
"capabilities:boot_option"="local" m1.OpenShiftInfra
```

3. ノードを `instackenv.json` に追加します。`capabilities` フィールドを使用するようにノードを定義する必要があります。以下に例を示します。

```
{
  "arch": "x86_64",
  "cpu": "4",
  "disk": "60",
  "mac": [
    "00:0c:29:9f:5f:05"
  ],
  "memory": "16384",
  "pm_type": "ipmi",
  "capabilities": "profile:OpenShiftMaster",
  "name": "OpenShiftMaster_1"
},
{
  "arch": "x86_64",
  "cpu": "4",
  "disk": "60",
  "mac": [
    "00:0c:29:91:b9:2d"
  ],
  "memory": "16384",
  "pm_type": "ipmi",
  "capabilities": "profile:OpenShiftWorker",
  "name": "OpenShiftWorker_1"
}
{
  "arch": "x86_64",
```

```

"cpu": "4",
"disk": "60",
"mac": [
    "00:0c:29:91:b9:6a"
],
"memory": "16384",
"pm_type": "ipmi",
"capabilities": "profile:OpenShiftInfra",
"name": "OpenShiftInfra_1"
}

```

4. 通常のデプロイメントと同様に、OCP ノードをインポートしてイントロスペクションを行います。以下に例を示します。

```

$ openstack overcloud node import ~/instackenv.json
$ openstack overcloud node introspect --all-manageable --provide

```

5. オーバークラウドノードに正しいプロファイルが割り当てられていることを確認します。

```

$ openstack overcloud profiles list
+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+
| Node UUID | Node Name | Provision |
State | Current Profile | Possible Profiles |
+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+
| 72b2b1fc-6ba4-4779-aac8-cc47f126424d | openshift-worker01 | available |
| OpenShiftWorker | | |
| d64dc690-a84d-42dd-a88d-2c588d2ee67f | openshift-worker02 | available |
| OpenShiftWorker | | |
| 74d2fd8b-a336-40bb-97a1-adda531286d9 | openshift-worker03 | available |
| OpenShiftWorker | | |
| 0eb17ec6-4e5d-4776-a080-ca2fdcd38e37 | openshift-infra02 | available |
| OpenShiftInfra | | |
| 92603094-ba7c-4294-a6ac-81f8271ce83e | openshift-infra03 | available |
| OpenShiftInfra | | |
| b925469f-72ec-45fb-a403-b7debfcf59d3 | openshift-master01 | available |
| OpenShiftMaster | | |
| 7e9e80f4-ad65-46e1-b6b4-4cbfa2eb7ea7 | openshift-master02 | available |
| OpenShiftMaster | | |
| c2bcdd3f-38c3-491b-b971-134cab9c4171 | openshift-master03 | available |
| OpenShiftMaster | | |
| ece0ef2f-6cc8-4912-bc00-ffb3561e0e00 | openshift-infra01 | available |
| OpenShiftInfra | | |
| d3a17110-88cf-4930-ad9a-2b955477aa6c | openshift-custom01 | available |
| None | | |
| 07041e7f-a101-4edb-bae1-06d9964fc215 | openshift-custom02 | available |
| None | | |
+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+

```

1.2.5. OpenShift 環境の定義

`openshift_env.yaml` ファイルを作成します。このファイルを使用して OpenShift に関する設定を定義し、その後 director が **openstack overcloud deploy** 操作の一部としてこの設定を適用します。ご自分のデプロイメントに合わせて、これらの値を更新してください。

```
parameter_defaults:
# by default Director assigns the VIP random from the allocation pool
# by using the FixedIPs we can set the VIPs to predictable IPs before
starting the deployment

CloudName: openshift.localdomain
PublicVirtualFixedIPs: [{'ip_address': '10.0.0.200'}]

CloudNameInternal: internal.openshift.localdomain
InternalApiVirtualFixedIPs: [{'ip_address': '172.17.1.200'}]

CloudDomain: openshift.localdomain

## Required for CNS deployments only
OpenShiftInfraParameters:
  OpenShiftGlusterDisks:
    - /dev/vdb

## Required for CNS deployments only
OpenShiftWorkerParameters:
  OpenShiftGlusterDisks:
    - /dev/vdb
    - /dev/vdc

NtpServer: ["clock.redhat.com", "clock2.redhat.com"]

ControlPlaneDefaultRoute: 192.168.24.1
EC2MetadataIp: 192.168.24.1
ControlPlaneSubnetCidr: 24

# The DNS server below should have entries for resolving
{internal,public,apps}.openshift.localdomain names
DnsServers:
  - 10.0.0.90

OpenShiftGlobalVariables:

  openshift_master_identity_providers:
  - name: 'htpasswd_auth'
    login: 'true'
    challenge: 'true'
    kind: 'HTPasswdPasswordIdentityProvider'
  openshift_master_htpasswd_users:
    sysadmin: '$apr1$jpb0UqeU$X4jUsMyCH00p8TFYtPq0v1'

  #openshift_master_cluster_hostname should match the CloudNameInternal
parameter
  openshift_master_cluster_hostname: internal.openshift.localdomain

  #openshift_master_cluster_public_hostname should match the CloudName
parameter
```

```

openshift_master_cluster_public_hostname: public.openshift.localdomain

openshift_master_default_subdomain: apps.openshift.localdomain

```

カスタムネットワークまたはカスタムインターフェースを使用する場合には、カスタムネットワークインターフェーステンプレートを使用する必要があります。

```

resource_registry:
OS::Triple0::OpenShiftMaster::Net::SoftwareConfig: /home/stack/master-
nic.yaml
OS::Triple0::OpenShiftWorker::Net::SoftwareConfig: /home/stack/worker-
nic.yaml
OS::Triple0::OpenShiftInfra::Net::SoftwareConfig: /home/stack/infra-
nic.yaml

```

1.2.6. OpenShift リポジトリへのオーバークラウドノードの登録

OCP パッケージをインストールするためには、オーバークラウドノードから OpenShift リポジトリにアクセスする必要があります。director ベースのデプロイメントで RHSM を設定する方法については、『オーバークラウドの高度なカスタマイズ』の「[Ansible ベースのオーバークラウド登録](#)」を参照してください。ご自分のノードで OpenShift パッケージを利用可能にするには、**rhsm.yaml** ファイルに **rhel-7-server-ose-3.11-rpms** のエントリを追加します。

```

resource_registry:
OS::Triple0::Services::Rhsm: /usr/share/openstack-tripleo-heat-
templates/extraconfig/services/rhsm.yaml
parameter_defaults:
RhsmVars:
rhsm_repos:
- rhel-7-server-rpms
- rhel-7-server-extras-rpms
- rhel-7-server-ose-3.11-rpms
rhsm_pool_ids: "8a85f37c63842fef0166949e5f9c4be0"
rhsm_method: "portal"
rhsm_username: yourusername
rhsm_password: yourpassword
rhsm_autosubscribe: true

```

あるいは、リポジトリを有効にするのに十分なサブスクリプションが含まれたアクティベーションキーを使用します。

```

resource_registry:
OS::Triple0::Services::Rhsm: /usr/share/openstack-tripleo-heat-
templates/extraconfig/services/rhsm.yaml
parameter_defaults:
RhsmVars:
rhsm_repos:
- rhel-7-server-rpms
- rhel-7-server-extras-rpms
- rhel-7-server-ose-3.11-rpms
rhsm_activation_key: "activation-key"
rhsm_org_id: "1234567"
rhsm_pool_ids: "8a85f9833e1404a6023e4cddf95a0599"
rhsm_method: "portal"

```

1.2.7. OCP ノードのデプロイ

以下のファイルを使用して OCP ノードをデプロイします。これには、前項の手順により作成した 2 つの新たな YAML ファイルが含まれます。

- `openshift_env.yaml`
- `openshift_roles_data.yaml`
- `containers-default-parameters.yaml`

カスタムネットワークのデプロイメントの場合には、以下のような NIC およびネットワークテンプレートのファイルが必要な場合があります。

- `master-nic.yaml`
- `infra-nic.yaml`
- `worker-nic.yaml`
- `network_data_openshift.yaml`

これらの YAML ファイルを `openstack overcloud deploy` コマンドに追加します。CNS のデプロイメントの例を以下に示します。

```
$ openstack overcloud deploy \
--stack openshift \
--templates \
-r /home/stack/openshift_roles_data.yaml \
-n /usr/share/openstack-tripleo-heat-templates/network_data_openshift.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/openshift.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/openshift-cns.yaml \
-e /home/stack/openshift_env.yaml \
-e /home/stack/containers-prepare-parameter.yaml \
-e /home/stack/rhsm.yaml
```

CNS 以外のデプロイメントの例を以下に示します。

```
$ openstack overcloud deploy \
--stack openshift \
--templates \
-r /home/stack/openshift_roles_data.yaml \
-n /usr/share/openstack-tripleo-heat-templates/network_data_openshift.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml \
-e /usr/share/openstack-tripleo-heat-templates/environments/openshift.yaml \
-e /home/stack/openshift_env.yaml \
-e /home/stack/containers-prepare-parameter.yaml \
-e /home/stack/rhsm.yaml
```

カスタムネットワークまたはカスタムインターフェースのデプロイメントの場合には、それらを指定する必要があります。以下に例を示します。

```
$ openstack overcloud deploy \  
--stack openshift \  
--templates \  
-r /home/stack/openshift_roles_data.yaml \  
-n /home/stack/network_data_openshift.yaml \  
-e /usr/share/openstack-tripleo-heat-templates/environments/network-  
isolation.yaml \  
-e /usr/share/openstack-tripleo-heat-templates/environments/openshift.  
yaml \  
-e /usr/share/openstack-tripleo-heat-templates/environments/openshift-  
cns.yaml \  
-e /home/stack/openshift_env.yaml \  
-e /home/stack/containers-prepare-parameter.yaml \  
-e /home/stack/custom-nics.yaml
```

1.2.8. OCP デプロイメントの確認

オーバークラウドのデプロイ手順が完了したら、OCP ノードの状態を確認します。

1. すべてのベアメタルノードの一覧を表示します。マスターノードおよびワーカーノードが表示されるはずですが。

```
$ openstack baremetal node list
```

2. マスターノードの1つを探します。

```
$ openstack server list
```

3. OpenShift マスターノードの1つに SSH 接続します。以下に例を示します。

```
$ ssh heat-admin@192.168.122.43
```

4. root ユーザーに切り替えます。

```
$ sudo -i
```

5. コンテナオーケストレーション設定を確認します。

```
$ cat .kube/config
```

6. OCP にログインします。

```
$ oc login -u admin
```

7. 既存のプロジェクトを確認します。

```
$ oc get projects
```

8. OCP の状態を確認します。

```
$ oc status
```

9. OCP からログアウトします。

```
$ oc logout
```

1.2.9. OCP を使ったテストアプリケーションのデプロイ

以下の手順では、新しい OCP デプロイメントでテストアプリケーションを作成する方法について説明します。

1. developer としてログインします。

```
$ oc login -u developer
Logged into "https://192.168.64.3:8443" as "developer" using
existing credentials.

You have one project on this server: "myproject"

Using project "myproject".
```

2. 新規プロジェクトを作成します。

```
$ oc new-project test-project
```

その他のリソース

- OpenShift Container Platform クラスターのインストールに関する詳細な情報は、[『クラスターのインストール』](#)を参照してください。
- OpenShift Container Platform クラスターの設定に関する詳細な情報は、[『クラスターの設定』](#)を参照してください。

第2章 仮想環境での OPENSIFT ON OPENSTACK

2.1. 前提条件

- Red Hat OpenStack director がインストールされていること。
- ハードウェアおよびネットワーク要件の検証が実行されていること。

以下の2つの方法のいずれかで、ハードウェアおよびネットワーク要件の検証を実行することができます。

- Red Hat OpenStack director から
- コマンドラインから

Red Hat OpenStack director からの検証の実行

director から検証を実行するには、以下の手順を実施します。

1. Red Hat OpenStack director で 検証 パネルを開くには、ウィンドウ右上の 新規検証 アイコン をクリックします。
2. OpenShift 用の検証項目を探すには、検証の検索フィールドに「OpenShift」の文字を入力します。OpenShift 用には、2つの検証項目があります。
 - ネットワーク要件
 - ハードウェア要件
3. OpenShift 用の検証を実行するには、一覧から希望の検証項目を選択し、再生アイコンをクリックします。

コマンドラインからの検証の実行

ハードウェア要件の検証を実行するには、以下のコマンドを実行します。

```
$ openstack action execution run tripleo.validations.run_validation
'{"validation": "openshift-hw-requirements", "plan": "overcloud"}'
```

ネットワーク要件の検証を実行するには、以下のコマンドを実行します。

```
$ openstack action execution run tripleo.validations.run_validation
'{"validation": "openshift-nw-requirements", "plan":
"overcloud"}'openstack workflow execution create
tripleo.validations.v1.run_validation '{"plan": "overcloud",
"validation_name": "openshift-nw-requirements"}'
```



注記

検証に失敗しても、OpenShift のインストールを試みることはできます。ただし、OpenShift をインストールする前に検証での要求事項を満たしておくことを推奨します。



注記

両方のコマンドに関して、プラン名「overcloud」は director のインストールで使用されるデフォルトのプランです。専用の heat テンプレートセットを使用している場合には、カスタムプランの作成時に選択した名前を使用してください。

OpenShift on OpenStack のインストールおよび設定については、[『Configuring for OpenStack』](#) を参照してください。