Altair PBS Professional 2021.1.2

Cloud Guide

You are reading the Altair PBS Professional 2021.1.2

Cloud Guide (CG)

Updated 9/20/21

Contact Us

For the most recent information, go to the PBS Works website, www.pbsworks.com, select "My PBS", and log in with your site ID and password.

Altair

Altair Engineering, Inc., 1820 E. Big Beaver Road, Troy, MI 48083-2031 USA www.pbsworks.com

Sales

pbssales@altair.com 248.614.2400

Please send any questions or suggestions for improvements to agu@altair.com.

Technical Support

Need technical support? We are available from 8am to 5pm local times:

Location	Telephone	e-mail
Australia	+1 800 174 396	anz-pbssupport@india.altair.com
China	+86 (0)21 6117 1666	pbs@altair.com.cn
France	+33 (0)1 4133 0992	pbssupport@europe.altair.com
Germany	+49 (0)7031 6208 22	pbssupport@europe.altair.com
India	+91 80 66 29 4500 +1 800 208 9234 (Toll Free)	pbs-support@india.altair.com
Italy	+39 800 905595	pbssupport@europe.altair.com
Japan	+81 3 6225 5821	pbs@altairjp.co.jp
Korea	+82 70 4050 9200	support@altair.co.kr
Malaysia	+91 80 66 29 4500 +1 800 425 0234 (Toll Free)	pbs-support@india.altair.com
North America	+1 248 614 2425	pbssupport@altair.com
Russia	+49 7031 6208 22	pbssupport@europe.altair.com
Scandinavia	+46 (0)46 460 2828	pbssupport@europe.altair.com
Singapore	+91 80 66 29 4500 +1 800 425 0234 (Toll Free)	pbs-support@india.altair.com
South Africa	+27 21 831 1500	pbssupport@europe.altair.com
South America	+55 11 3884 0414	br_support@altair.com
UK	+44 (0)1926 468 600	pbssupport@europe.altair.com

Contents

About PBS Documentation

The PBS Professional guides and release notes apply to the *commercial* releases of PBS Professional.

Document Conventions

`Abbreviation`

The shortest acceptable abbreviation of a command or subcommand is underlined

Attribute

Attributes, parameters, objects, variable names, resources, types

`Command`

Commands such as `qmgr` and `scp`

Definition

Terms being defined

`File name`

File and path names

`Input`

Command-line instructions

Method

Method or member of a class

`Output`

Output, example code, or file contents

Syntax

Syntax, template, synopsis

`Utility`

Name of utility, such as a program

Value

Keywords, instances, states, values, labels

Notation

Optional Arguments

Optional arguments are enclosed in square brackets. For example, in the `qstat` man page, the `-E` option is shown this way:

qstat [-E]

To use this option, you would type:

```
qstat -E
```

Variable Arguments

Variable arguments (where you fill in the variable with the actual value) such as a job ID or vnode name are enclosed in angle brackets. Here's an example from the `pbsnodes` man page:

pbsnodes -v <vnode>

To use this command on a vnode named "my_vnode", you'd type:

```
pbsnodes -v my_vnode
```

Optional Variables

Optional variables are enclosed in angle brackets inside square brackets. In this example from the `qstat` man page, the job ID is optional:

qstat [<job ID>]

To query the job named "1234@my_server", you would type this:

```
qstat 1234@my_server
```

Literal Terms

Literal terms appear exactly as they should be used. For example, to get the version for a command, you type the command, then "--version". Here's the syntax:

qstat --version

And here's how you would use it:

```
qstat --version
```

Multiple Alternative Choices

When there are multiple options and you should choose one, the options are enclosed in curly braces. For example, if you can use either "-n" or "--name":

```
{-n | --name}
```

List of PBS Professional Documentation

The PBS Professional guides and release notes apply to the *commercial* releases of PBS Professional.

PBS Professional Release Notes

Supported platforms, what's new and/or unexpected in this release, deprecations and interface changes, open and closed bugs, late-breaking information. For administrators and job submitters.

PBS Professional Big Book

All your favorite PBS guides in one place: *Installation & Upgrade, Administrator's, Hooks, Reference, User's*, *Programmer's, Cloud, Budget,* and *Simulate* guides in a single book.

PBS Professional Installation & Upgrade Guide

How to install and upgrade PBS Professional. For the administrator.

PBS Professional Administrator's Guide

How to configure and manage PBS Professional. For the PBS administrator.

PBS Professional Hooks Guide

How to write and use hooks for PBS Professional. For the PBS administrator.

PBS Professional Reference Guide

Covers PBS reference material: the PBS commands, resource, attributes, configuration files, etc.

PBS Professional User's Guide

How to submit, monitor, track, delete, and manipulate jobs. For the job submitter.

PBS Professional Programmer's Guide

Discusses the PBS application programming interface (API). For integrators.

PBS Professional Manual Pages

PBS commands, resources, attributes, APIs.

PBS Professional Licensing Guide

How to configure licensing for PBS Professional. For the PBS administrator.

PBS Professional Cloud Guide

How to configure and use the PBS Professional Cloud feature.

PBS Professional Budgets Guide

How to configure Budgest and use it to track and manage resource usage by PBS jobs.

PBS Professional Simulate Guide

How to configure and use the PBS Professional Simulate feature.

Where to Keep the Documentation

To make cross-references work, put all of the PBS guides in the same directory.

Ordering Software and Licenses

To purchase software packages or additional software licenses, contact your Altair sales representative at pbssales@altair.com.

1
Introduction to PBS Cloud

1.1 Introduction to Cloud Bursting

PBS Cloud allows PBS Professional to burst nodes in the cloud, so that your site can handle demand peaks. PBS Professional uses a cloud bursting hook called cloud_hook to analyze jobs from the cloud queue(s), estimate the demand, and burst the required cloud nodes having the specified instance type and OS image. The PBS scheduler runs the jobs from the cloud queue in the cloud nodes. The cloud bursting hook dynamically adjusts the number of cloud nodes according to current load and how long you want nodes to wait for jobs to appear.

PBS Cloud provides the framework for the interface to the cloud. PBS Cloud supports multiple cloud vendors as well as private OpenStack clouds. You can use multiple vendors at the same time, and multiple accounts at each vendor. PBS Cloud also supports instance types that are on-demand, preemptable (GCP) and spot (AWS and Azure). PBS Cloud supports Infiniband, MPI, and scaleset jobs.

1.2 Cloud Bursting Terminology

Burst

The action of creating a node in the cloud and adding it to the PBS complex

Instance type

A specification for a machine including characteristics such as CPUs, memory, storage capacity, network technology, etc. The PBS Cloud administrator specifies the instance types that will be available to job submitters. Jobs can request and use only instance types that the administrator has made available.

PBS Cloud supports instance types that are on-demand, preemptable (GCP) and spot (AWS and Azure).

Unburst

The action of removing a node from both the PBS complex and the cloud.

Cloud queue

Each scenario uses its own cloud queue. This is where the jobs for that scenario are enqueued. Cloud jobs must be submitted to the appropriate cloud queue.

Scenario

A bursting scenario encapsulates information needed to burst cloud nodes, such as the default OS image and which cloud-init script should initialize cloud nodes. You create a scenario data structure in PBS Cloud; this is where you specify information about resources provided by the cloud vendor that PBS Cloud uses for bursting. You define other aspects of the same scenario in the cloud bursting hook. A scenario can use one or more instance types to burst cloud nodes, although all the instance types for a scenario must be non-preemptable or be preemptable. You can have as many scenarios as you want.

Cloud Node

A virtual machine that has been created on cloud hardware. Each cloud node is burst using the OS image specified for the job. After the node is burst, it is initialized via cloud-init scripts with everything required to run PBS jobs.

Cloud Bursting Hook

The cloud bursting hook is called cloud_hook, and it is installed when you install PBS Cloud. The PBS cloud bursting hook manages cloud nodes and jobs via PBS Cloud and cloud queues. You specify details for each scenario that you want the hook to handle.

OS Image

A pre-configured OS image in the cloud from which virtual machines can be instantiated. At the vendor, you create an OS image to use as the default for a particular scenario at that vendor. Jobs can request a specific OS image, the cloud bursting hook can specify a default OS image for that scenario, and you can set a default OS image for the cloud queue for that scenario.

1.3 How PBS Cloud Bursting Works

1.3.1 How Node Bursting Works

You create an administrator account with your cloud vendor. PBS Cloud uses this vendor administrator account manage cloud nodes.

You create a cloud queue for each scenario, job submitters request the cloud queue for their jobs, and the cloud bursting hook analyzes the resources needed by those jobs and bursts the required cloud nodes via PBS Cloud.

The PBS scheduler runs the cloud jobs in the cloud nodes.

1.3.1.1 OS Image and Instance Type Assignment to Job

The OS image and instance type used for a particular job depend on whether the job requests it, or whether the job inherits it along the way. Assignment works in this order, with the first one encountered being the one assigned:

- Job request
- Queue default
- Cloud bursting hook default
- PBS Cloud scenario default

1.3.1.2 Typical Cloud Bursting Arrangement

The following figure shows one typical arrangement used for cloud bursting:

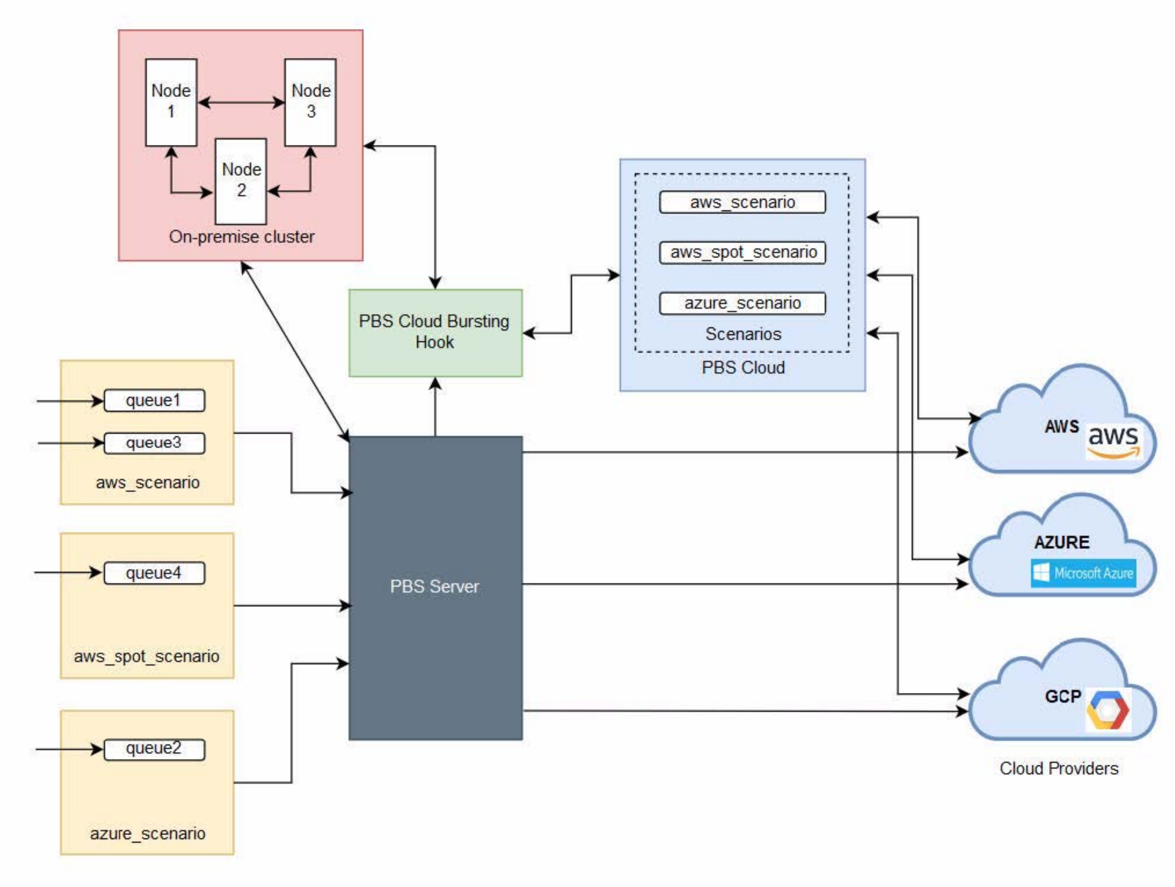


Figure 1-1:Typical Cloud Bursting Setup

1.3.2 Tracking Application Licenses

Jobs that run in the cloud may require application licenses. PBS Cloud bursts nodes for these jobs only when application licenses are available; otherwise the nodes could sit idle.

PBS Cloud uses a custom consumable server-level integer resource to track how many of each kind of application license are available. The cloud bursting hook checks the value of this resource before bursting cloud nodes, so that it only bursts new nodes for jobs requiring application licenses when those licenses are available. The administrator creates a script, typically run as a `cron` job, that keeps this resource as up-to-date as possible.

2
Installing the PBS Cloud Module

2.1 Supported Platforms for PBS Cloud

2.1.1 Supported Platforms for PBS Cloud Head Node

- CentOS 7.2, 8
- RHEL 7, 8
- SLES 12, with restrictions:
  - Each SLES host must be registered with the SUSE Customer Center via SUSEConnect, and have a support contract. This happens automatically for cloud nodes.
  - SLES hosts require Docker Enterprise Edition.

2.1.2 Supported Platforms for Nodes Burst in Cloud

- Linux: any Linux platform that supports both PBS MoM and `cloud-init`
- Windows: 10, Server 2012

2.1.3 Supported Cloud Providers

You must already have an account with one of the supported cloud providers. We support the listed variant for each cloud provider:

Table 2-1: Supported Clouds

Cloud Provider	Variant
Microsoft Azure	Azure Compute
Amazon Web Services (AWS)	Elastic Compute Cloud (EC2)
Google Cloud Platform (GCP)	Compute Engine
Oracle Cloud Platform	Oracle Cloud Infrastructure
Orange Technical Cloud (OTC)	Orange Flexible
Deutsche Telekom	Open Telekom Cloud (OTC)
HUAWEI Cloud	Elastic Cloud Server
OpenStack cloud on premise	Stein

2.2 Prerequisites

2.2.1 Prerequisites for Installing on a Connected Host

- A working PBS complex managed by PBS Professional version 2020.1. The PBS complex can be either of the recommended configurations in Recommended Configurations
- docker-ce v19.x or later for most systems
- docker-ee v19.x or later for SLES
- SELinux must be disabled (reboot the system to make this the active config)
- VPN connection to the cloud you will use
    - Except in the case of a cloud hosted head node.
- Cloud provider account with:
    - Correct authorizations
    - Approved method of payment

2.2.2 Prerequisites for Installing on an Offline Host

- One host that is connected to the Internet
    - An additional 20GB of disk space on the connected host
- An offline host where you will use PBS Cloud
    - An additional 30GB of disk space on the offline host
- A working PBS complex managed by PBS Professional version 2020.1. The PBS complex can be either of the recommended configurations in Recommended Configurations
- docker-ce v19.x or later for most systems
- docker-ee v19.x or later for SLES
- SELinux must be disabled (reboot the system to make this the active config)
- VPN connection to the cloud you will use
    - Except in the case of a cloud hosted head node.
- Cloud provider account with:
    - Correct authorizations
    - Approved method of payment

2.2.3 Licensing

Make sure that the time zone for your on premise hardware and all cloud nodes and your licenses is the same. You will also need the following:

- Altair License Server 14.5.1 or later
- PBSProNodes v20 license

2.2.4 Required Accounts

- To install PBS Cloud, you need to be root.
- To configure PBS Cloud, you must use pbsadmin@altair. This account is created by the PBS Cloud installer software during installation. The default password for pbsadmin@altair is Altair@123. We recommend changing the password to something known only to you.

2.3 Recommended Configurations

Head node and service node can be one of either:

- Both on premises
- Both in cloud

Do not put one on premises and one in the cloud.

See Chapter 10, "Example Azure Head Node", on page 137.

2.3.1 Recommended Configuration for Larger Installations

For larger installations using on premises hosts:

- Hosts on premises for PBS server, scheduler, some MoM daemons, cloud service node
- Head, service, and first N execution nodes are on premises
- Extra execution nodes are burst
- On head node, PBS components running as usual
- On service node
  - PBS Cloud is designed to run in a container
- VPN connection to the cloud you will use

Notes:

- You many not want to run PBS Cloud on the head node, because it runs in a Docker container, which may impose too high a load.
- All components are mix-and-match (with Docker restriction).
- You don't need to configure additional `pbs_comm` daemons for cloud nodes, because cloud can't cause enough throughput to need one.
- For PBS configuration instructions, see the *PBS Professional Administrator's Guide*.

2.3.2 Recommended Configuration for Smaller Installations

For smaller installations cloud-only installations where the workload is low enough:

- All PBS components can be hosted in the cloud
- All components can run on the same node
- You can run Docker on the same node as the PBS components

Notes:

- No VPN is required for this configuration.
- See Chapter 10, "Example Azure Head Node", on page 137.

2.4 Installation Steps

You can install PBS Cloud on a host that is connected to the Internet, or one that is not. For instruction on installing on an offline host, see Installing on an Offline Host.

2.4.1 Installing on a Connected Host

2.4.1.1 Install Docker

1. Log in as root
2. Choose whether your head node will be on premises or in the cloud. If it will be in the cloud, build your cloud head node. For an Azure example, see Chapter 10, "Example Azure Head Node", on page 137.
3. Install, start, and enable `docker-ce`:
   - For CentOS or RedHat:

     Log in to the machine where PBS Cloud is to be installed.

     ```
     yum install -y yum-utils
     yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo
     yum install docker-ce docker-ce-cli containerd.io
     systemctl enable docker
     systemctl start docker
     ```
   - For SLES12 or 15:

     Log in to the machine where PBS Cloud is to be installed.

     For SLES 12:

     ```
         sudo SUSEConnect -p sle-module-containers/12/x86_64 -r ''
     ```

     For SLES 15:

     ```
         sudo SUSEConnect -p sle-module-containers/15.1/x86_64 -r ''
     sudo zypper install docker
     sudo systemctl enable docker.service
     sudo systemctl start docker.service
     ```

     Configure the firewall to allow forwarding of Docker traffic to the external network:

     ```
     Set FW_ROUTE="yes" in /etc/sysconfig/SuSEfirewall2
     ```
   - For Ubuntu:

     Log in to the machine where PBS Cloud is to be installed.

     ```
     sudo apt-get update
     sudo apt-get install apt-transport-https ca-certificates curl gnupg-agent software-properties-common
     curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
     sudo apt-key fingerprint 0EBFCD88
     ```

     The key should match the second line in the output; validate the last 8 characters. Example of second line:

     9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88

     ```
     sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu
        $(lsb_release -cs) stable"
     sudo apt-get update
     sudo apt-get install docker-ce docker-ce-cli containerd.io
     sudo systemctl enable docker.service
     sudo systemctl start docker.service
     ```

2.4.1.2 Install the PBS Cloud Module

1. Log in as root
2. Clean up any previous installations of the PBS Cloud module:
   - If the PBS cloud hook, named "cloud_hook", exists, delete it:
     ```
     qmgr -c 'd h cloud_hook'
     ```
   - If the pclm command exists in `$PBS_EXEC/bin` or `/opt/pbs/bin`, delete it:
     ```
     source /etc/pbs.conf
     rm $PBS_EXEC/bin/pclm
     ```
3. Extract the installer:
   ```
   tar xvfz PBSPro-cloud_2021.1-<OS>_x86_64.tar.gz
   ```
4. Install PBS cloud feature, PBS cloud hook, and PBS Cloud command layer. Execute the installation script:

   a. Change directory to `pbspro-cloud-installer` directory:
   ```
   cd pbspro-cloud-installer
   ```
   b. Run the installer:
   ```
   ./install.sh
   ```

2.4.1.3 Allow Easy PBS Cloud Status Check

1. Log in as root
2. Create an alias to easily use `pkr`. Type the following all one line:
   ```
   alias pkr="docker run -ti --network host --rm -e PBSCLOUD_VERSION=pbspro-cloud-2021.1 -e
       PKR_VERSION=pbspro-cloud-2021.1 -v /run/docker.sock:/run/docker.sock -v /root/kard:/pkr/kard
       pbscloudio.azurecr.io/pkr:pbspro-cloud-2021.1 pkr"
   ```

2.4.1.4 Test the Installation

1. Log in as root
2. Make sure that relevant services are up and running. Each should have an IP address. See section 11.2.3, “Sample pkr Output while Running”, on page 151 for sample healthy output reference data for `pkr`.
   ```
   pkr ps
   ```
3. If you are running the PBS Cloud module on a head node in the cloud, use the vendor tools to open access through the firewall to port 9980 so that you can use the vendor web interface.
4. Log into PBS Cloud from your web browser:
   ```
   http://<PBS Cloud host name or IP address>:<port>/pbspro-cloud/#/login
   ```
   - Default port: 9980
   - Username: pbsadmin@altair
   - Password: Altair@123

2.4.2 Installing on an Offline Host

In order to install PBS Cloud on an offline host, you download the standard installation package on a connected host, extract the package, download Altair files to the package, tar it back up, copy it over to the offline host, untar it, and run the offline installer. We detail the steps below.

2.4.2.1 Install Docker on Connected Host

1. Log in as root

- Choose whether your head node will be on premises or in the cloud. If it will be in the cloud, build your cloud head node. For an Azure example, see Chapter 10, "Example Azure Head Node", on page 137.

2. Install, start, and enable `docker-ce`:

   - For CentOS or RedHat:

     Log in to the machine where PBS Cloud is to be installed.

     ```
     yum install -y yum-utils
     yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo
     yum install docker-ce docker-ce-cli containerd.io
     systemctl enable docker
     systemctl start docker
     ```

   - For SLES12 or 15:

     Log in to the machine where PBS Cloud is to be installed.

     For SLES 12:

     ```
     sudo SUSEConnect -p sle-module-containers/12/x86_64 -r ''
     ```

     For SLES 15:

     ```
     sudo SUSEConnect -p sle-module-containers/15.1/x86_64 -r ''
     ```

     ```
     sudo zypper install docker
     sudo systemctl enable docker.service
     sudo systemctl start docker.service
     ```

     Configure the firewall to allow forwarding of Docker traffic to the external network:

     ```
     Set FW_ROUTE="yes" in /etc/sysconfig/SuSEfirewall2
     ```

   - For Ubuntu:

     Log in to the machine where PBS Cloud is to be installed.

     ```
     sudo apt-get update
     sudo apt-get install apt-transport-https ca-certificates curl gnupg-agent software-properties-common
     curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
     sudo apt-key fingerprint 0EBFCD88
     ```

     The key should match the second line in the output; validate the last 8 characters. Example of second line:

     9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88

     ```
     sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu
        $(lsb_release -cs) stable"
     sudo apt-get update
     sudo apt-get install docker-ce docker-ce-cli containerd.io
     sudo systemctl enable docker.service
     sudo systemctl start docker.service
     ```

2.4.2.2 Download Installation Tarball to Connected Host

On the connected host:

1. Log in as root
2. Extract the contents of the installer package:

   ```
   tar xvfz PBSPro-cloud_2021.1-<OS>_x86_64.tar.gz
   ```

   This creates the directory named "pbspro-cloud-installer"
3. Run the script that uses Docker to pull images from the Altair site and adds them to installer:
   a. Change directory to `pbspro-cloud-installer` directory:

      ```
      cd pbspro-cloud-installer
      ```

   b. Make the scripts executable:

      ```
      chmod 0755 *.sh
      ```

   c. Run the download script:

      ```
      ./offline_download.sh
      ```
4. Tar up the modified installer:

   ```
   tar cvfz PBSPro-cloud_2021.1.0-CentOS7_x86_64_offline.tar.gz pbspro-cloud-installer/
   ```

2.4.2.3 Copy Tarball to Offline Host

Copy the tarball of the modified installation script onto the offline host.

2.4.2.4 Install Docker on Offline Host

On the offline host:

1. Log in as root
2. Install Docker and its dependencies
   - For CentOS, RedHat, and Ubuntu, install `docker-ce` (Docker Community Edition) and its dependencies:

     ```
     docker-ce
     docker-ce-cli
     containerd.io
     ```
   - For SLES 12 and 15, install `docker-ee` (Docker Enterprise Edition). You may need to use SUSEConnect.
3. Enable Docker
4. Start Docker

2.4.2.5 Extract Tarball to Offline Host

On the offline host:

1. Untar the package for the modified installer:

   ```
   tar xvfz PBSPro-cloud_2021.1.0-CentOS7_x86_64_offline.tar.gz
   ```

This creates the directory named "pbspro-cloud-installer"

2. Run the script for installing PBS Cloud on an offline host:
    a. Change directory to `pbspro-cloud-installer` directory:

       ```
       cd pbspro-cloud-installer
       ```

    b. Run the offline installation script:

       ```
       ./offline_install.sh
       ```

2.4.2.6 Allow Easy PBS Cloud Status Check

On the offline host:

1. Log in as root
2. Create an alias to easily use `pkr`. Type the following all one line:

```
alias pkr="docker run -ti --network host --rm -e PBSCLOUD_VERSION=pbspro-cloud-2021.1 -e
    PKR_VERSION=pbspro-cloud-2021.1 -v /run/docker.sock:/run/docker.sock -v /root/kard:/pkr/kard
    pbscloudio.azurecr.io/pkr:pbspro-cloud-2021.1 pkr"
```

2.4.2.7 Test the Installation

On the offline host:

1. Log in as root
2. Make sure that relevant services are up and running. Each should have an IP address. See section 11.2.3, “Sample pkr Output while Running”, on page 151 for sample healthy output reference data for `pkr`.

   ```
   pkr ps
   ```

3. If you are running the PBS Cloud module on a head node in the cloud, use the vendor tools to open access through the firewall to port 9980 so that you can use the vendor web interface.
4. Log into PBS Cloud from your web browser:

   ```
   http://<PBS Cloud host name or IP address>:<port>/pbspro-cloud/#/login
   ```

   - Default port: 9980
   - Username: pbsadmin@altair
   - Password: Altair@123

3 Configuring PBS Cloud

3.1 Overview of Configuring PBS Cloud

Much of the information required to configure PBS Cloud and PBS Professional is generated or chosen while you are logged into your cloud provider, building your cloud components. We provide a list of what to collect while you are doing this so that you will have the information you need later. Alternatively, you may want to have one window open for each purpose simultaneously, rather than performing these steps in sequence. We recommend reading through the instructions once before starting so that you can see where information is transferred from one tool to another.

1. If you have not done so already, install PBS Professional; see the PBS Professional Installation & Upgrade Guide
2. If you have not done so already, install the PBS Cloud module; see Chapter 2, "Installing the PBS Cloud Module", on page 5
3. Configure PBS Professional for cloud bursting. This step is covered in section 3.2, “Configuring PBS Professional for Cloud Bursting”, on page 16, but we outline it here for clarity:
   a. Log into the PBS server host as administrator
   b. Configure PBS Professional for bursting to cloud nodes
      1. Create and configure resources
      1. Create and configure cloud queue(s)
      2. Configure scheduling
4. Log into your cloud provider
5. PBS Cloud will use an account at the cloud provider to manage cloud nodes. Create a cloud provider account for this purpose; see section 3.3.2, “Create Your Cloud Provider Account”, on page 21.
6. Build and configure your cloud components. During this step, capture the information listed; while you are generating or selecting each item, we’ll remind you to collect it. You will use this information in the following steps here. Building and configuring cloud components is different for each cloud provider; use the provider-specific instructions in Chapter 5, "Using Cloud Provider Services", on page 49. We show an outline here so you can see what is covered:
   a. Log into your cloud provider
   b. Create the necessary cloud provider components, such as a virtual network
   c. Create a virtual machine
   d. Install the PBS Professional MoM, and other required software in the VM, including cloud-init
   e. If you will use cloud-init to configure freshly burst nodes, create a startup script for configuring burst nodes; see Chapter 6, "The Cloud Node Startup Script", on page 113
   f. Create the OS image to be used for bursting

7. Configure PBS Cloud. This step is described in section 3.3, "Configuring PBS Cloud", on page 21, but we give you an idea of what's involved here:
    a. Log into PBS Cloud; see section 3.3.1, "Log Into PBS Cloud", on page 21
    b. Add your cloud provider account to PBS Cloud; see section 3.3.3, "Add Your Provider Account to PBS Cloud", on page 22
    c. Create a bursting scenario; see section 3.3.4, "Create a Bursting Scenario", on page 25
    d. Manually test bursting; see section 6.2.5, "Developing the Startup Script", on page 117
    e. For each remaining scenario, create and test the scenario
    f. Disable public IP addresses for the scenario
8. Configure the PBS cloud bursting hook; see section 4.1, "The Cloud Bursting Hook", on page 39
    a. Log into the PBS server host as administrator
    b. Configure and enable the cloud hook
    c. For each scenario, test it with the cloud hook by using it to automatically burst scenario nodes; see section 4.3, "Testing Automated Cloud Bursting", on page 47
9. To prevent running out of PBSProNodes licenses, set a limit on the number of cloud nodes that can exist simultaneously. See section 3.3.6, "Manage Licenses", on page 36

3.2 Configuring PBS Professional for Cloud Bursting

Each PBS Cloud scenario requires its own cloud queue, and vice versa. You associate a cloud queue with a scenario by using the same name for both; at the queue you set the cloud_scenario resource to be identical to the name set for the PBS Cloud scenario.

3.2.1 Create and Configure Custom Resources for Cloud Bursting

3.2.1.1 List of PBS Professional Custom Resources for Cloud Bursting

PBS Professional uses the following custom resources to manage cloud jobs:

burst_by_hook

Used by cloud bursting hook and PCLM to distinguish manually burst cloud nodes from automatically burst cloud nodes. Do not set.

cloud_instance_type

Queue-level string

Default cloud provider instance type (machine, shape type or flavor) associated with this queue.

cloud_min_instances

Server-level integer

Minimum number of cloud nodes (instances) to be present in the cloud at any time.

cloud_max_instances

Server-level integer

Maximum number of cloud nodes (instances) that can be present in the cloud at any time.

cloud_max_jobs_check_per_queue

Queue-level integer

Maximum number of jobs in the cloud queue to be checked to determine the number of cloud nodes that must be burst based on the requested instance type. The cloud bursting hook computes the number of nodes it must burst in order to run the checked cloud jobs. For example, if the instance type has 2 CPUs, and the first 3 jobs in the queue need a total of 10 CPUs, the hook bursts 5 nodes.

Must be greater than zero. Setting this to zero results in no jobs from this queue being considered for cloud bursting.

Default: 64

cloud_network

Host-level string

For requesting InfiniBand-enabled nodes.

Nodes with the same network name are grouped.

cloud_node_image

Host-level string

Default OS image to use when bursting a cloud node. Overridden if job specifies an image.

cloud_node_instance_type

Host-level string

Set this at the cloud queue to the default instance type for cloud nodes burst for this queue. Make sure you choose an instance type that is enabled in PBS Cloud.

This is set at the cloud node by the cloud bursting hook when that node is burst.

If a job requests an instance type, that overrides the default set at the queue.

cloud_provisioned_time

Host-level integer

Time in seconds from since the cloud bursting hook started bursting cycle; if node doesn't successfully come up in that time, the cloud bursting hook tries again

cloud_queue

Queue-level Boolean

Indicates whether the queue is a cloud queue. When True, the queue is a cloud queue. The cloud bursting hook monitors only cloud queues. Cloud jobs must be submitted to a cloud queue.

cloud_scenario

Node-level string.

Indicates the associated scenario type for the queue and the node. Functions as placement tool.

lic_signature

Host-level string

Contains licensing information.

node_location

Host-level string

Used to differentiate on-premise nodes from cloud nodes. Lets you keep a job either all in the cloud or all on premise. If you have multiple clouds, name this for the cloud.

(static resource to represent each application license)

Consumable server-level integer

Each application license is represented by one static and one dynamic resource. Used by cloud bursting hook to track license availability.

(dynamic resource to represent each application license)

Consumable server-level integer

Each application license is represented by one static and one dynamic resource. Used by cloud bursting hook to track license availability.

3.2.1.2 Create Custom Resources for Cloud Bursting

1. Log in to the PBS server host as the PBS administrator
2. Create the custom resources required for cloud bursting:

```
qmgr -c 'create resource cloud_queue type=boolean'
qmgr -c 'create resource cloud_instance_type type=string'
qmgr -c 'create resource cloud_node_instance_type type=string,flag=h'
qmgr -c 'create resource cloud_min_instances type=long'
qmgr -c 'create resource cloud_max_instances type=long'
qmgr -c 'create resource cloud_provisioned_time type=long,flag=h'
qmgr -c 'create resource lic_signature type=string,flag=h'
qmgr -c 'create resource cloud_node_image type=string,flag=h'
qmgr -c 'create resource cloud_network type=string,flag=h'
qmgr -c 'create resource node_location type=string,flag=h'
qmgr -c 'create resource cloud_max_jobs_check_per_queue type=long'
qmgr -c 'create resource cloud_scenario type=string,flag=h'
qmgr -c 'create resource burst_by_hook type=boolean'
```

3.2.2 Configure PBS Server and Scheduler for Cloud Bursting

1. Log in to the PBS server host as the PBS administrator
2. Change directory to `$PBS_HOME/sched_priv`
3. Edit the `sched_config` file. Add cloud_scenario, cloud_node_image, and cloud_node_instance_type to the `resources:` line:

```
resources: "ncpus, mem, arch, host, vnode, netwins, aoe, cloud_scenario, cloud_node_image,
    cloud_node_instance_type"
```

4. Make the scheduler reread its configuration file; HUP the scheduler:

```
kill -HUP <scheduler PID>
```

5. Set server limits:

```
qmgr -c "set server resources_available.cloud_min_instances = 0"
qmgr -c "set server resources_available.cloud_max_instances = <max nodes>"
```

3.2.3 Create Resources and Scripts to Manage Application Licenses

If the cloud jobs at your site are not using externally-managed application licenses, you can skip this step.

PBS Cloud needs two custom server-level consumable integer resources to represent each kind of application license.

- A dynamic resource updated by a server_dyn_res script: The scheduler uses the dynamic resource to check license availability for jobs. The scheduler needs to use a dynamic resource because it needs the resource to be up-to-date for each scheduling cycle.
- A static resource updated via a `cron` script: The cloud bursting hook uses the static resource to check license availability for jobs. The cloud bursting hook cannot use the dynamic resource because that resource is not available to the hook.

3.2.3.1 Create cron Script and Static Resource

1. For each kind of application license, create one custom server-level static consumable long resource. The command looks like this:

   ```
   qmgr -c 'create resource <application license static resource> type=long,flag=q'
   ```

2. Create a `cron` script that updates the value of each static consumable license-tracking resource. The script can update the resource value by calling `qmgr`:

   ```
   qmgr –c 'set server resources_available.<resource name>=<updated value>'
   ```

3. The script has to run as manager or root. If you run it as manager, add the script owner to the server's list of managers:

   ```
   qmgr -c 'set server managers += <script owner>'
   ```

4. Set the permissions of the script to *0700*.
5. The script should run at least as frequently as the cloud bursting hook. The script does not need to run more than twice as often as the cloud bursting hook.

If you have any questions, contact Altair support; we will be happy to work with you.

3.2.3.2 Create Dynamic Server-level Resource for Each Application License

If you have already created a dynamic server-level resource for each application license, and it is already updated via a server_dyn_res script, you can skip this step.

1. For each kind of application license, create one custom server-level dynamic long resource. The command looks like this:

   ```
   qmgr -c 'create resource <application license dynamic resource> type=long'
   ```

2. Write a server_dyn_res script that returns the number of available licenses via `stdout`.

   The format of a dynamic server-level resource query is a shell escape:

   server_dyn_res: "<resource name> !<path to command>"

   In this query, <*resource name*> is the name of the dynamic resource, and <*path to command*> is typically the full path to the script or program that performs the query in order to determine the status and/or availability of the new resource you have added. This usually means querying a license server.

3. Name the script to indicate what it does, for example, "serverdyn.pl".
4. Place the script on the server host. For example, it could be placed in `/usr/local/bin/serverdyn.pl`

5. Make sure the server_dyn_res script meets the following requirements:
    - The script:
        - Owned by root
        - Has permissions of 0755
        - Returns its output via `stdout`, and the output must be in a single line ending with a newline
        - The scheduler has access to the script, and can run it
        - If you have set up peer scheduling, make sure that the script is available to any scheduler that needs to run it
    - The directory containing the script:
        - Owned by root
        - Accessible only by root (must not give write permission to *group* or *others*)
        - Has permissions 0550
6. Configure the scheduler to use the server_dyn_res script by adding the resource and the path to the script in the `server_dyn_res` line of `<sched_priv directory>/sched_config`. For example:

    ```
    server_dyn_res: "floatlicense !/usr/local/bin/serverdyn.pl"
    ```

7. Optionally give the scheduler a time limit for the script by setting its server_dyn_res_alarm attribute:

    ```
    qmgr -c 'set sched <scheduler name> server_dyn_res_alarm=<new value>'
    ```

8. For each application license, add its custom dynamic resource (the one tracked by the server_dyn_res script, **not** the one tracked by the `cron` job) to the `resources:` line in `<sched_priv directory>/sched_config`. For example, if your dynamic resource that tracks App1 is app1_dyn:

    ```
    resources: "ncpus, mem, arch, host, vnode, netwins, aoe, cloud_scenario, cloud_node_image,
        cloud_node_instance_type, app1_dyn"
    ```

9. Restart the scheduler. See "Restarting and Reinitializing Scheduler or Multisched" on page 166 in the PBS Professional Installation & Upgrade Guide.

3.2.4 Create and Configure Cloud Queues

Each PBS Cloud scenario requires its own cloud queue, and vice versa. You associate a cloud queue with a scenario by using the same name for both; at the queue you set the cloud_scenario resource to be identical to the name set for the PBS Cloud scenario. You may find it helpful to name each cloud queue with information that makes it easy to match the queue with its purpose.

The instance type that is burst is defined at the queue level, unless overridden at job submission time.

1. Log in to the PBS server host as root.
2. Create a queue:

    ```
    qmgr -c "create queue <queue name> queue_type=execution,enabled=true,started=true"
    ```

3. Make this queue into a cloud queue:

    ```
    qmgr -c "set queue <queue name> resources_available.cloud_queue = True"
    ```

4. Optionally, set the maximum number of jobs to check in the queue to determine the number of nodes to burst:

    ```
    qmgr -c "set queue <queue name> resources_available.cloud_max_jobs_check_per_queue = <max jobs to
        check>"
    ```

    Must be greater than 0; setting to 0 results in no bursting. Default is 64.

5. Assign a bursting scenario to the queue:

   ```
   qmgr -c "set queue <queue name> resources_available.cloud_scenario = <scenario name>"
   ```

   The queue scenario name must be identical to the scenario name in the cloud bursting hook configuration file. See Chapter 4, "Configuring the Cloud Bursting Hook", on page 39.

   Here is an example scenario named "azure_scenario_1" as it would appear in the cloud bursting hook configuration file:

   ```
   "scenario": {
       "azure_scenario_1": {
           "api_key": "<API key>",
           "cloud_default_image": "azure_bursting_image1",
           "cloud_max_instances": 10,
           "max_vms_for_infiniband_scaleset" : 100,
           "max_nodes_per_burst":50,
           "cloud_node_wait_timeout":180,
           "check_resources":["hw_units"],
           "preemptable": false
       }
   }
   ```

6. Assign a default instance type to the queue:

   ```
   qmgr -c "set queue <queue name> resources_available.cloud_instance_type = <instance type>"
   ```

   This is the default instance type to be burst for this queue. It must match one of the instance types enabled for the bursting scenario that you assigned to the queue. For example:

   ```
   qmgr -c "set queue <queue name> resources_available.cloud_instance_type = Standard_DS2_v2"
   ```

   See section 3.3.4.5.i, “Cloud Provider Instance Types”, on page 30.

3.3 Configuring PBS Cloud

3.3.1 Log Into PBS Cloud

You must be logged into PBS Cloud as pbsadmin@altair account to configure PBS Cloud.

Log into PBS Cloud from your web browser:

```
http://<PBS Cloud host name or IP address>:<port>/pbspro-cloud/#/login
```

- Default port: 9980
- Username: pbsadmin@altair
- Password: Altair@123

We recommend changing the password immediately; use the UX to do this.

3.3.2 Create Your Cloud Provider Account

PBS Cloud uses an administrator account at the cloud provider to manage cloud nodes. Create your cloud provider account; follow the steps for your cloud provider in section , “Using Cloud Provider Services”, on page 49. While you are creating the account, capture the information we list at the start of the provider-specific instructions; you will use that information to add your cloud provider account to PBS Cloud.

3.3.3 Add Your Provider Account to PBS Cloud

PBS Cloud can use multiple cloud providers, and can use multiple accounts at each provider.

Tell PBS Cloud about the cloud account you have created at your cloud provider by filling in the parameters.

The Name parameter is common to all accounts, meaning that all accounts have a name, but the name can be different for each provider. The other parameters for an account are different for each vendor; for example, only Oracle uses a fingerprint ID. Follow the steps for your vendor to capture the information we list so that you can use it to fill in the account parameters. For the vendor-specific steps, see Chapter 5, "Using Cloud Provider Services", on page 49.

The information required varies by cloud provider.

1. Click the Cloud tab.
2. Under Infrastructure, click Cloud.
3. Select your cloud provider
4. Fill in the account name. This is an arbitrary string, and it is not the administrator account name created at the vendor. We recommend making it informative.

5. Add the information for the parameters that are vendor-specific. We list the parameter information by vendor:
   - For an Amazon Web Services (AWS) account:

Table 3-1: Account Parameters for Amazon Web Services (AWS)

Account Parameter	Collected During Configuration at Vendor	Format
Access Key ID	Access Key ID from vendor `.csv` file	String
Secret Access Key	Secret Access Key from vendor `.csv` file	String

   - For an Azure account:

Table 3-2: Account Parameters for Microsoft Azure

Account Parameter	Collected During Configuration at Vendor	Format
Client ID	Application ID generated when registering PBS Cloud with the Azure Active Directory	String
Secret Key	Secret Key generated during account creation at vendor	String
AD Tenant ID	Azure tenant ID generated during account creation at vendor	String
Subscription ID	Subscription ID generated during account creation at vendor	String

   - For a Google Cloud Platform (GCP) account:

Table 3-3: Account Parameters for Google Cloud Platform (GCP)

Account Parameter	Collected During Configuration at Vendor	Format
Project ID	Value of project_id in JSON file created at vendor	String
Client ID	Value of client_id in JSON file created at vendor	String
Client Mail	Value of client_email in JSON file created at vendor	String
Private Key ID	Value of private_key_id in JSON file created at vendor	String
Private Key	Value of private_key in JSON file created at vendor	String

   - For an Oracle account:

Table 3-4: Account Parameters for Oracle

Account Parameter	Collected During Configuration at Vendor	Format
User OCID	User OCID generated when creating Oracle cloud user account at vendor	String
Tenant OCID	Tenancy OCID generated at vendor	String

Table 3-4: Account Parameters for Oracle

Account Parameter	Collected During Configuration at Vendor	Format
Compartment OCID	Root compartment OCID generated at vendor	String
Fingerprint OCID	Fingerprint generated when adding the public SSH key for Oracle user at vendor	String
Private Key	RSA private key generated at vendor	String

- For a Huawei, Deutsche Telekom, Orange, or OpenStack account:

Table 3-5: Account Parameters for Huawei, Deutsche Telekom, Orange, OpenStack

Account Parameter	Collected During Configuration at Vendor	Format
Auth URL	Orange: *https://iam.<orange region>.<console link>* Huawei: *https://iam.ap-southeast-1.myhwclouds.com* Deutsche Telekom: *https://iam.eu-de.otc.t-systems.com/v3* OpenStack: contact Altair support	String
User Domain Name	Orange: Customer ID used to log in to Orange account. Same as domain name Deutsche Telekom: OTC domain name used to log in to OTC console at vendor Huawei: Domain Name provided when your subscription to HUAWEI Cloud was created OpenStack: Domain name for cloud account in private cloud	String
Username	Administrator username created at vendor	String
Password	Huawei & Deutsche Telekom: Administrator password created at vendor Orange: API password generated at vendor OpenStack: Password for administrator account	String

6. Click Create account.

3.3.3.1 Example of Adding Azure Account to PBS Cloud

1. Log in to PBS Cloud.
2. Click the Cloud tab.
3. Under Infrastructure, click Cloud.
4. Click Azure

5. Enter the following to add a cloud account:
   a. For Account name, enter any name for the cloud account.

      The name can be anything meaningful to your organization, e.g., azure_cloudaccount.
   b. For Client ID, enter the Application ID generated when PBS Cloud was registered with the Azure Active Directory.
   c. For Secret Key, enter the client secret key generated when you register PBS Cloud.
   d. For AD tenant ID, enter your Azure tenant ID.
   e. For Subscription ID, enter your Azure subscription ID.
6. Click Create account.

3.3.4 Create a Bursting Scenario

A bursting scenario encapsulates information needed to burst cloud nodes. Some scenario parameters are common to all scenarios, meaning that all scenarios have an associated hostname prefix, but the hostname prefix can be different for each scenario. Some parameters for a bursting scenario are different for each vendor; for example, only Azure uses a resource group. Follow the steps for your vendor to capture the information we list so that you can use it to fill in the scenario parameters. For the vendor-specific steps, see Chapter 5, "Using Cloud Provider Services", on page 49.

You can create an unlimited number of bursting scenarios. We recommend creating and testing them one at a time.

1. Go to the bursting scenario interface in PBS Cloud:
   a. Open a browser window and log in to PBS Cloud.
   b. Click on Cloud.
   c. Under Infrastructure, click on Bursting.
   d. Click Add Bursting Scenario.
2. Add the information for the following parameters common to all scenarios:

Table 3-6: Common Scenario Parameters

Scenario Parameter	Description	Format
Name	Arbitrary friendly scenario name	String
Description	Arbitrary scenario description	String
Cloud account	Name of your account at cloud vendor	String
Region	Region selected during configuration at cloud vendor	Drop-down list
Domain name	Domain used by PBS Cloud module head node	String
Hostname prefix	Optional prefix for burst node names; default is "node"; chosen during configuration at vendor; see section 3.3.4.1, “Adding Hostname Prefix”, on page 29	String
Add public IP to VMs	Optional public IP address to VMs for initial troubleshooting; see section 3.3.4.4, “Temporarily Adding Public IP for Debugging”, on page 29	Checkbox

Table 3-6: Common Scenario Parameters

Scenario Parameter	Description	Format
cloud-init	Name of startup script launched by `cloud-init` for configuring freshly burst nodes. Browse to a file, and optionally edit the file; see section 3.3.4.9, "Specifying the Cloud Node Startup Script", on page 32	String
SSH keys	Administrator SSH key to give you access for initial debugging if `/home` won't mount in the node; see section 3.3.4.8, "Adding SSH Key for Access to Burst Nodes", on page 32	String
Idle time before unbursting	Time for burst node to sit idle before unbursting. Default: 180 seconds; see section 3.3.4.2, "Setting Idle Time", on page 29	Integer seconds
Tag	Optional labeling system for convenience. You can apply multiple tags. See section 3.3.4.7, "Adding Tags (Labels) to Scenario", on page 31	Key:value pair as *<string>:<string>*
Instances	Instances that will be available for job submitters to select for their jobs. Instances must be all non-preemptable or all preemptable; see section 3.3.4.5, "Managing Instances", on page 30	Checkboxes

3. Add the information for the parameters that are vendor-specific:

   - For an AWS scenario:

Table 3-7: Scenario Parameters for Amazon Web Services (AWS)

Scenario Parameter	Collected During Configuration at Vendor	Format
AMI ID	Name of image to be burst; chosen during configuration at vendor	String
Security Group ID	Name of security group associated with VPC and VM created at vendor	String
Subnet ID	Name of security group subnet for bursting VPC created at vendor. To burst nodes in multiple Availability Zones, use a comma-separated list of subnet IDs	String

   - For a GCP scenario:

Table 3-8: Scenario Parameters for Google Cloud Platform (GCP)

Scenario Parameter	Collected During Configuration at Vendor	Format
Network name	Name of VPC for cloud bursting created at vendor	String
Subnetwork name	Name of VPC network subnet created at vendor	String
OS Image URI	Value for entry called selfLink of REST equivalent of OS image created for cloud bursting at vendor	String

   - For an Azure scenario:

Table 3-9: Scenario Parameters for Microsoft Azure

Scenario Parameter	Collected During Configuration at Vendor	Format
Resource group name	Name of resource group (virtual network, virtual machine, OS image) created at vendor	String
Network name	Name of virtual network created at vendor If the network is in a different resource group from the one specified, enter it as Resource Group Name/Virtual Network Name	String
Subnetwork name	Name of virtual subnet created at vendor If the subnet is in a different resource group from the one specified, enter it as Resource Group Name/Subnet Name	String
Network security group name	Name of network security group for resource group If the network security group is contained in a resource group that is different from the one entered for the bursting scenario, enter it as Resource Group Name/Network Security Group Name.	String

Table 3-9: Scenario Parameters for Microsoft Azure

Scenario Parameter	Collected During Configuration at Vendor	Format
Managed Storage	Managed disk feature selected at vendor	Boolean
OS Image	If using managed disks, name of the image. If not using managed disks, Linux Source BLOB URI. If the OS image is contained in a Resource group that is different from the one entered for the bursting scenario, enter it as Resource Group Name/OS Image Name or Resource Group Name/URI.	String
Maximum number of VMs inside a ScaleSet with Managed Storage and a single Placement Group	Limit selected during configuration at vendor. Default: 100 If you use InfiniBand, Azure limits the number of VMs on a scaleset to 100. Without InfiniBand, for a scaleset with managed disk and custom image, you can specify a higher limit.	Integer

- For an Oracle scenario:

Table 3-10: Scenario Parameters for Oracle

Scenario Parameter	Collected During Configuration at Vendor	Format
Subnet ID	OCID of subnet associated with data center where cloud bursting virtual machine is hosted	String
OS Image URI	Vendor link to bursting image OCID	String

- For a Huawei, Deutsche Telekom, Orange, or OpenStack scenario:

Table 3-11: Scenario Parameters for Huawei, Deutsche Telekom, Orange, OpenStack

Scenario Parameter	Collected During Configuration at Vendor	Format
Subnet ID	ID of subnet for VPC created at cloud vendor	String
Security Group ID	ID of security group created at cloud vendor	String
OS Image URI	ID of OS image created at cloud vendor	String

4. Create the scenario: click on Instantiate scenario.
5. Create an API key and add it to the scenario; see section 3.3.4.3, “Creating API Key for Cloud Hook to Use”, on page 29.
6. **Save the API key** to use when configuring the cloud bursting hook
7. We recommend adding quotas on resource usage, and alerts when those quotas are reached; see section 3.3.4.6, “Adding Quotas and Alerts”, on page 31.

3.3.4.1 Adding Hostname Prefix

The hostname prefix is the base name of each burst node. For example, if you choose "cloudnode", burst nodes are named "cloudnode1", "cloudnode2", "cloudnode3", etc.

Format: string

3.3.4.2 Setting Idle Time

The Idle before unburst parameter specifies the minimum time that a cloud node can be idle before it is unburst.

Default idle time is 180 seconds. We recommend making the idle time more than double the PBS scheduler cycle time.

Format: integer seconds

3.3.4.3 Creating API Key for Cloud Hook to Use

The cloud hook uses an API as a unique identifier for each scenario. You cannot use the same API key for more than one scenario. If you lose an API key, you can generate a new one. Later, you will put this API key in the cloud bursting hook configuration file so that the hook can identify the correct scenario.

The default lifetime of an API key is one year. You can have multiple keys for a scenario; this is to allow overlap near the expiration date. You can only list one API key per scenario in the cloud hook.

You can create an API key only for an existing scenario.

Generate and save the key using the following steps:

1. Log in to PBS Cloud.
2. Click Cloud.
3. Under Infrastructure, click Bursting.
4. Click the name of the bursting scenario.
5. Click *Add token* located at the bottom of the web page.
6. For Name, enter a name for the API key. Format: lowercase alphabetic + numeric
7. For Expiration date, specify the expiration date. Format: MM/DD/YYYY
8. Generate the API key by clicking Add Token.

   PBS Cloud generates the API key, and displays it only once.
9. **Copy and save the API key** so that you can paste it into the cloud hook configuration file later.
10. Click Close.

3.3.4.4 Temporarily Adding Public IP for Debugging

You can add a public IP address to the bursting scenario to make it easier to debug, then remove the IP address when you have your scenario working. To add a public IP address to the scenario:

1. Log in to PBS Cloud.
2. Click on Cloud.
3. Under Infrastructure, click on Bursting.
4. Click the name of the bursting scenario.
5. Click on the Customization tab
6. Click Edit

7. Check the checkbox next to Enable Public IP Address.
8. Click Save.

3.3.4.5 Managing Instances

When you create a scenario, choose the instance types that you want job submitters to be able to use with that scenario. The instance type determines which hardware is used to burst the cloud node. The available instance types depend on which cloud provider you use.

For each scenario, you can use either all non-preemptable instance types, or all preemptable instance types. Do not mix the two. See section 3.3.5, "Using Spot or Preemptable Pricing", on page 34.

For information about spot and preemptable instance types, see section 3.3.5, "Using Spot or Preemptable Pricing", on page 34.

3.3.4.5.i Cloud Provider Instance Types

The instance type (also called shape, machine type or flavor) determines the hardware of the host computer used for your cloud nodes. Each instance type offers different compute, memory, and storage capabilities. The following table lists some instance types by cloud provider:

Table 3-12: Cloud Provider Instance Types

Provider	Classification System	Example Instance Types
Azure	Sizes for virtual machines in Azure and High performance compute VM sizes	Standard_DS1_v2, Standard_D2s_v3, Standard_NC6 Standard_H16r, Standard_H16mr (InfiniBand)
AWS	Amazon EC2 Instance Types	t2.medium, r4.large, p3.2xlarge
GCP	Machine Types	n1-standard-8, n1-highmem-2, n1-highcpu-64
Oracle	VM Shapes and GPU Instances	VM.Standard1.1, VM.DenseIO1.16, VM.GPU3.1
Orange Cloud	Instance Family	s1.medium, s3.large.4, cc3.large.4
HUAWEI Cloud	ECS Types	s2.small.1, s2.medium.4, s2.xlarge.2
OTC	ECS Types	s1.medium, c1.large, m1.xlarge

3.3.4.5.ii Steps to Choose Instance Types

1. Log in to to PBS Cloud.
2. Click Cloud.
3. Under Infrastructure, click Bursting.
4. Click the name of the bursting scenario.
5. Click *Edit Instances*.
6. Choose the instance types you want available for the bursting scenario by clicking the Enabled checkbox located to the far right of the instance type name.
7. Click Save. You will see a list of the enabled instance types.

3.3.4.5.iii Managing Hyperthreading for an Instance Type

You can choose whether or not to enable hyperthreading via a checkbox next to the instance type. By default, PBS cloud enables hyperthreading. The vendor reports hyperthreading to PBS Cloud, and PBS Cloud reports hyperthreading to the cloud bursting hook.

If you turn off hyperthreading for an instance type, PBS Cloud disables hyperthreading in the cloud node, and reports the the number of available cores to the hook, so that job requests are aligned with core availability.

If you leave hyperthreading on, hyperthreading is enabled, and PBS Cloud reports the number of threads to the cloud bursting hook. In this case, job requests are aligned with thread availability.

3.3.4.6 Adding Quotas and Alerts

For each scenario, you can optionally specify quotas on resource usage at any point in time, so that when PBS Cloud hits a quota, it stops bursting nodes until usage drops back down. Use quotas to prevent huge jumps in expenditure.

For each quota, you can set an associated alert so that the web interface displays a notification when the quota is reached.

You can add quotas and alerts only to an existing scenario.

You can set quotas on the following:

- Number of CPUs in use
- Amount of memory in use
- Number of burst nodes

To set each quota and associated alert:

1. Log in to PBS Cloud.
2. Click Cloud.
3. Under Infrastructure, click Bursting.
4. Click the name of the bursting scenario.
5. Click Add Quota.
6. For Resource, choose a resource from the menu.
7. Click Add Quota.
8. For Limit, provide a limit for the resource.
9. To add an alert for this quota, click Add.
10. Provide an alert value.

3.3.4.7 Adding Tags (Labels) to Scenario

You can optionally add tags (labels) to burst nodes in order to categorize them, for example by purpose, owner, or environment. You can add multiple tags.

1. Log in to PBS Cloud.
2. Click Cloud.
3. Under Infrastructure, click Bursting.
4. Click the name of the bursting scenario.
5. Click the Cloud tab, then the global edit pen.
6. The Tags box appears.

1. To add a label to that will be applied to burst nodes, enter a key-value pair, followed by <return>. Format:

   <key>:<value><return>

   The key and the value can contain alphanumeric, dash (-) and an underscore (_). The maximum length for the key is 35 characters, and the maximum length for the value is 42 characters.

The <return> is required.

When you add the tag, it appears within its own bubble:

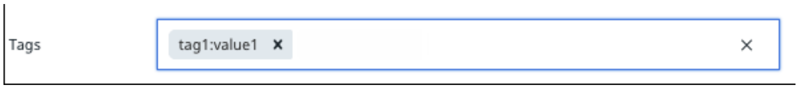


Figure 3-1:Key-Value Tag

Repeat the previous step to add more tags:

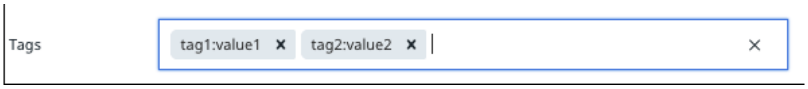


Figure 3-2:Additional Tags

2. Click Save.

3.3.4.8 Adding SSH Key for Access to Burst Nodes

You typically add the administrator SSH key so that if `/home` fails to mount during node bursting, you can log in and debug the problem. For example you need an SSH key when, `yum update` turns SELinux back on, and /home won't mount. Public SSH keys in a scenario are copied to each burst node for secure connectivity.

You can add SSH keys for only those users allowed to submit jobs to these burst nodes, although you have to make those users' home directories available on the burst nodes so that PBS Professional can use them for jobs.

To add a public SSH key for access to burst nodes:

1. Log in to the PBS Professional server host.
2. Copy the public SSH key for the user; public key files are usually stored in `/.ssh` in the user's home directory.
3. In the PBS Cloud interface:
   a. Click Add; PBS Cloud displays an editable box
   b. In Public SSH keys, paste the public SSH key.

To remove public SSH keys so that users do not have access to burst nodes, click the "x" located next to the SSH key box.

3.3.4.9 Specifying the Cloud Node Startup Script

You can optionally use the cloud-init tool to run a cloud node startup script when each node is burst, to automate node configuration tasks; see Chapter 6, "The Cloud Node Startup Script", on page 113. The startup script can be located anywhere that your PBS Cloud web interface can browse to. Once a script has been added to a scenario, the script is stored in the PBS Cloud database.

Note that you may not need a startup script. If you configure the VM that you use to create your OS image to have everything you need to run jobs, you do not need a startup script.

3.3.4.9.i Startup Script Prerequisites

- The startup script must run using a shell or language available in the freshly burst node. For example, if you have `bash` and Python available, your script can use `bash`, or it could use a `bash` script to launch a Python script.
- On Windows cloud nodes, use a PowerShell startup script. Enclose the content of the PowerShell script in <powershell> and </powershell>. Refer to Microsoft documentation for more information about PowerShell.
- The startup script can have any name.

3.3.4.9.ii Steps to Add Startup Script to Scenario

You can add and edit the cloud node startup script for this scenario from the web interface:

1. Log in to PBS Cloud.
2. Click Cloud.
3. Under Infrastructure, click Bursting.
4. Click the name of the bursting scenario.
5. Click the edit pen.
6. Click the Browse button, and browse to the file you want
7. Optionally edit the file as needed
8. Click Save.

3.3.4.10 Example of Creating a Scenario

Example 3-1: Create an Azure cloud bursting scenario; see section 5.2.8, “Collect Information for an Azure Cloud Bursting Scenario”, on page 69

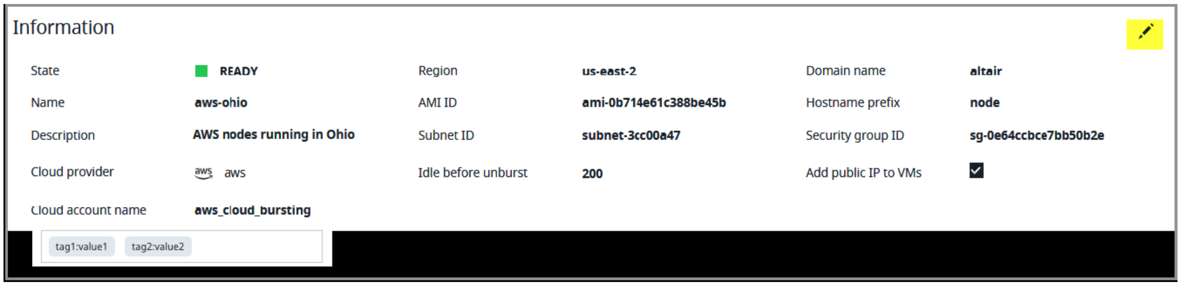
We show some recommended options:

- For Name, give the scenario a short name, all in lowercase, e.g. hyperburst
- Select Managed Disks
- For node debugging, select Public IP; you can disable this once your scenario is working properly
- We recommend using the Standard_A4 instance type for minimal cost
- Optionally provide a startup script; for a sample script, see section 6.2.4, “Example cloud-init Startup Script for Linux”, on page 116
- Create an API key and store it safely. It is lost once you close the window, although you can simply create another if you lose it

3.3.4.11 Editing a Bursting Scenario

You can use the web interface to PBS Cloud to edit a scenario, and specify bursting scenario elements, including scenario name, description, domain name, OS image, and VPC details.

1. Before you edit a bursting scenario, stop the bursting process and make sure no jobs are running on the burst nodes; see section 7.1.4, “Disabling Bursting for a Scenario and Queue”, on page 122.
2. Edit the bursting scenario:
   a. Log in to PBS Cloud.
   b. Click Cloud.
   c. Under Infrastructure, click Bursting.
   d. Select the name of the bursting scenario.
   e. To modify the scenario, click the edit pen.



Information

State	🟩 READY	Region	us-east-2	Domain name	altair
Name	aws-ohio	AMI ID	ami-0b714e61c388be45b	Hostname prefix	node
Description	AWS nodes running in Ohio	Subnet ID	subnet-3cc00a47	Security group ID	sg-0e64ccbce7bb50b2e
Cloud provider	aws	Idle before unburst	200	Add public IP to VMs	☑
Cloud account name	aws_cloud_bursting				

tag1:value1 tag2:value2

Figure 3-3:Edit Scenario Details

3. After you edit the scenario, enable it; see section 7.1.3, “Enabling or Disabling a Bursting Scenario in PBS Cloud”, on page 122.

3.3.5 Using Spot or Preemptable Pricing

3.3.5.1 When to Use Spot or Preemptable Instances

Preemptable instances offer spare compute capacity available in the cloud at steep discounts compared to on-demand instances. Spot instances are preemptable. Drawbacks:

- Spot and preemptable instances can be interrupted with two minutes' notice when the provider needs the capacity back. AWS, Azure, and GCP can preempt your spot instance when the vendor needs the resource.
- AWS and Azure can interrupt your spot instance when the spot price exceeds your maximum price, when the demand for spot instances rises, or when the supply of spot instances decreases.
- Google Cloud Platform (GCP) will kill your instance if it has been running for 24 hours.

We do not recommend running critical or long-running jobs in spot instances, as jobs may be killed when spot instances are preempted.

You can take advantage of preemptable and spot instances by using them for shorter jobs, and jobs that can be preempted.

3.3.5.2 How Spot and Preemptable Instances Work

- Dedicated on-demand: the instance is guaranteed to be available, but does not need longer-term commitments or up-front payments. You can increase or decrease your compute capacity depending on the demands of your application. You pay only for what you use. Fixed per hour or per second price depending on the instance type.
- Preemptable: vendor can preempt instance with about 2 minutes' notice; You pay a fixed discount. Instance is killed after 24 hours. GCP uses this model.
- Spot: price is set by vendor, based on demand, varies constantly. You bid a max. You pay actual spot price. If current price goes over your bid price, all your instances are preempted. If the vendor needs them, they are preempted. Cost savings of up to 90%, but most volatile. AWS and Azure use this model.

3.3.5.3 Bidding Spot Pricing

For AWS and Azure, you can bid a spot price. Specify the maximum price that you are willing to pay to continue using the spot instance. Your bid should be greater than 0 and less than the current on-demand price.

3.3.5.4 Preemptable Pricing

For GCP, the price for preemptable instances is fixed.

3.3.5.5 Specifying Spot Pricing

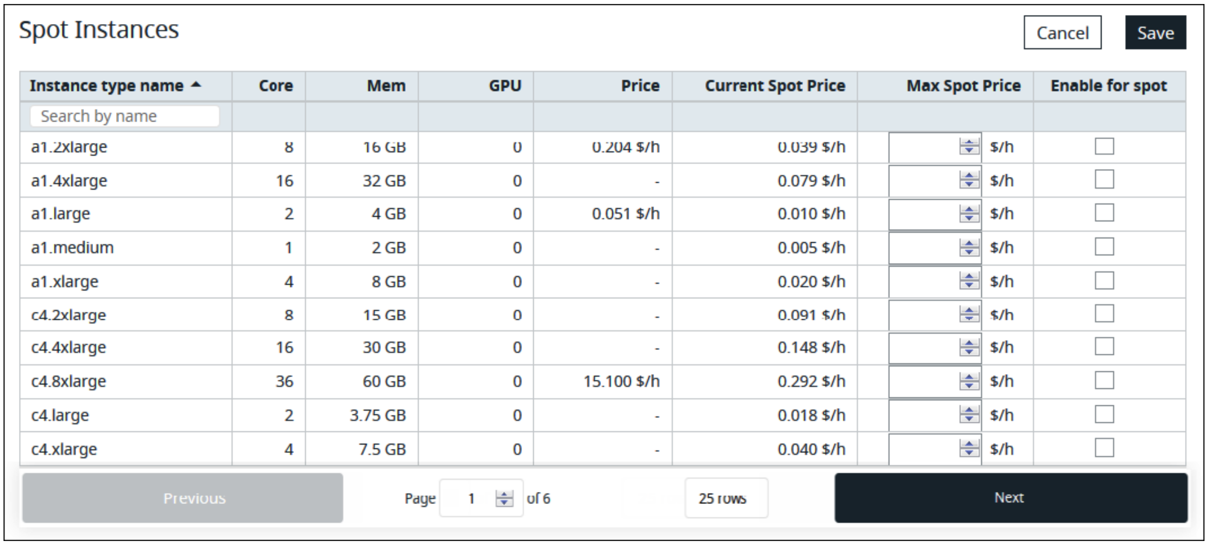
The scenario you use for spot pricing must already exist, must contain only preemptable instances, and must have preemptable set to *true* in the cloud hook.

1. Configure the cloud bursting hook with a scenario for this instance type. Make sure that the scenario contains only preemptable instance types. Make sure the instance type is preemptable. In the cloud bursting hook, the scenario must include:

   ```
   "preemptable": true
   ```

   See section 4.2.7, “Creating a Scenario for a Preemptable or Spot Instance”, on page 45.
2. Log in to PBS Cloud.
3. Click the Cloud tab.
4. Under Infrastructure, click Bursting.
5. Select a bursting scenario by clicking on its name.
6. Go to the instance table.

PBS Cloud displays a table of instance types you can enable for spot pricing:



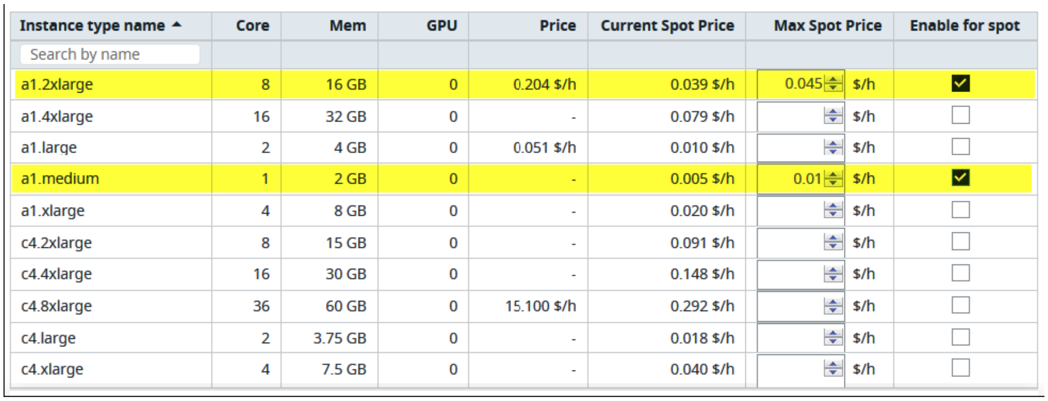
Instance type name	Core	Mem	GPU	Price	Current Spot Price	Max Spot Price	Enable for spot
a1.2xlarge	8	16 GB	0	0.204 $/h	0.039 $/h	$/h	□
a1.4xlarge	16	32 GB	0	-	0.079 $/h	$/h	□
a1.large	2	4 GB	0	0.051 $/h	0.010 $/h	$/h	□
a1.medium	1	2 GB	0	-	0.005 $/h	$/h	□
a1.xlarge	4	8 GB	0	-	0.020 $/h	$/h	□
c4.2xlarge	8	15 GB	0	-	0.091 $/h	$/h	□
c4.4xlarge	16	30 GB	0	-	0.148 $/h	$/h	□
c4.8xlarge	36	60 GB	0	15.100 $/h	0.292 $/h	$/h	□
c4.large	2	3.75 GB	0	-	0.018 $/h	$/h	□
c4.xlarge	4	7.5 GB	0	-	0.040 $/h	$/h	□

Figure 3-4:Spot Instances

7. Enable spot pricing for an instance type by checking the *Enable for spot* box next to the instance name.
8. For AWS and Azure, bid the spot price your site is willing to pay by entering it in the *Max Spot Price* box.
9. Disable spot pricing for an instance type by un-checking the *Enable for spot* box.
10. For AWS and Azure, enter a maximum spot price that you are willing to pay for the instance type.

3.3.5.6 Example of Choosing Instances for Spot Pricing

Two instance types are selected for spot pricing:



Instance type name	Core	Mem	GPU	Price	Current Spot Price	Max Spot Price	Enable for spot
a1.2xlarge	8	16 GB	0	0.204 $/h	0.039 $/h	0.045 $/h	☑
a1.4xlarge	16	32 GB	0	-	0.079 $/h	$/h	□
a1.large	2	4 GB	0	0.051 $/h	0.010 $/h	$/h	□
a1.medium	1	2 GB	0	-	0.005 $/h	0.01 $/h	☑
a1.xlarge	4	8 GB	0	-	0.020 $/h	$/h	□
c4.2xlarge	8	15 GB	0	-	0.091 $/h	$/h	□
c4.4xlarge	16	30 GB	0	-	0.148 $/h	$/h	□
c4.8xlarge	36	60 GB	0	15.100 $/h	0.292 $/h	$/h	□
c4.large	2	3.75 GB	0	-	0.018 $/h	$/h	□
c4.xlarge	4	7.5 GB	0	-	0.040 $/h	$/h	□

Figure 3-5:Spot Instances

3.3.6 Manage Licenses

Make sure that the number of cloud nodes plus the number of on premise nodes does not exceed the number of PBSProNodes licenses for your PBS complex. Set the value of resources_available.cloud_max_instances at the PBS server to be the number of PBSProNodes licenses minus the number of on premise nodes.

3.4 Testing Cloud Bursting

We recommend testing each scenario before using it in the cloud bursting hook. After you can manually burst a working cloud node for that scenario, add it to the cloud bursting hook and test that the hook can run jobs on the cloud nodes.

3.4.1 Test Each Scenario using Manual Bursting

3.4.1.1 Troubleshooting Prerequisites

To be able to troubleshoot cloud nodes, make sure the scenario has the following:

- The Add Public IP to VMs scenario option is enabled; see section 3.3.4.4, “Temporarily Adding Public IP for Debugging”, on page 29
- The SSH keys parameter has an administrator SSH key; see section 3.3.4.8, “Adding SSH Key for Access to Burst Nodes”, on page 32
- You have the corresponding private key
- Port 22 in the vendor firewall has to be open (already covered when you were configuring vendor components)

3.4.1.2 Testing and Refining a Scenario

Manually burst a single cloud node for that scenario and test its initial configuration, by following the steps in section 6.2.5, “Developing the Startup Script”, on page 117.

The typical testing cycle is burst a node, check it, unburst it, modify the cloud-init script, and repeat.

3.4.1.3 Disabling Public IP Address

Once your scenario is working, you can disable its public IP address:

1. Log in to PBS Cloud.
2. Click Cloud.
3. Under Infrastructure, click Bursting.
4. Click the name of the bursting scenario.
5. Disable the public IP address.
6. Click Save.

4
Configuring the Cloud Bursting Hook

4.1 The Cloud Bursting Hook

When you install the PBS Cloud module, the installer creates and imports the cloud bursting hook, which is named cloud_hook. PBS Cloud uses a single cloud bursting hook to handle all cloud bursting. The hook comes with a default configuration file. You configure the cloud bursting hook via its configuration file.

The cloud bursting hook runs at queuejob and periodic events.

4.1.1 Default Cloud Bursting Hook Configuration File

```
{
    "pclm_server": "https://<hostname or IP address of PBS Cloud module>:9980/pbspro-cloud/",
    "_comment_pclm_server_example": "e.g. http://control_server.mydomain:9980/pbspro-cloud/",
    "use_node_hour_license": false,
    "_comment_node_hour": "Node Hour License: True for Control, False for PBS Pro",
    "pclm_no_check_ssl_certificate": false,
    "cloud_min_instances": 1,
    "resources":["ncpus", "mem", "ngpus"],
    "cloud_driver": "PclmDriver",
    "scenario":{
        "<scenario name>":{
            "api_key": "<API key>",
            "cloud_default_image": "<default cloud image for this scenario>",
            "cloud_max_instances": 10,
            "max_vms_for_infiniband_scaleset" : 100,
            "__comment__infiniband": "Use the above option for Azure infiniband scenarios only.",
            "max_nodes_per_burst":50,
            "cloud_node_wait_timeout":180,
            "check_resources":["<resource name>", "<resource name>"],
            "__comment_check_resources":"List of static server-level license tracking resources",
            "preemptable": false,
            "__comment__preemptable": "Scenarios are preemptable or on-demand, not mixed.",
            "__comment__preemptable_example": "Only set to true for supported clouds and scenarios
    with spot/preemptable instances selected."
        },
        "<additional scenario name>":{
            "api_key": "<API key>",
            "cloud_default_image": "<default cloud image for this scenario>",
            "cloud_max_instances": 20,
            "max_nodes_per_burst":50,
            "cloud_node_wait_timeout":180,
            "check_resources":["<resource name>", "<resource name>"],
            "preemptable": false
        }
    }
}
```

4.2 Configuring the Cloud Bursting Hook

4.2.1 Main Configuration Parameters in Cloud Hook

The main section of the cloud bursting hook configuration file contains the following parameters:

pclm_server

Endpoint for accessing PBS Cloud.

Format: either of these:

http://<PBS Cloud hostname>:<port>/pbspro-cloud/

http://<PBS Cloud IP address>:<port>/pbspro-cloud/

Default port: *9980*

cloud_min_instances

Required. Minimum number of instances to be present in the cloud at any time. Does not apply during startup; cloud nodes are not immediately burst on startup. This is the minimum number that are maintained after they are initially burst on demand.

This value is overridden by the value of the cloud_min_instances resource set at the PBS server.

resources

Resources to be considered for calculating the number of nodes to burst. Resource names must be in quotes. Use one of the following strings:

- ["ncpus", "mem", "ngpus"]
- ["ncpus", "mem"]

cloud_driver

Cloud driver used by the cloud bursting hook.

Currently, the only value supported is "PclmDriver". DO NOT change this value.

scenario

Container for bursting scenarios. You can have one scenario for each cloud provider or multiple scenarios for a cloud provider or both. A scenario can contain either all non-preemptable instances or all preemptable instances.

Each bursting scenario must have its own API key.

4.2.2 Scenario Configuration Parameters in Cloud Hook

api_key

API key you generated for a bursting scenario via the PBS Cloud user interface. Each scenario definition in the cloud bursting hook configuration file can have only one API key. You can use the API key for only one scenario; each API key can appear only once in the cloud hook configuration file, and in only one PBS Cloud scenario

cloud_default_image

Default OS image to use when bursting a cloud node. This is the OS image you create via the vendor interface. Overridden when the OS image is provided at job submission via the `qsub` command.

cloud_max_instances

Maximum number of instances that can be made available in the cloud for this scenario. Required.

Must be greater than 0.

Format: Integer

max_vms_for_infiniband_scaleset (Optional)

Limit on number of nodes on a single InfiniBand switch. Set by the vendor. You can negotiate with the vendor to increase this limit; if you do, update this parameter.

Supported on Azure only.

Default: 100

max_nodes_per_burst

Maximum number of nodes allowed to burst in a single hook cycle. Maximum number of cloud node licenses to renew per hook cycle.

cloud_node_wait_timeout

Maximum time to wait for freshly burst node to become usable. Minimum value: 180 seconds. You can set this to a higher value, but not lower.

Default: 180 seconds.

pclm_no_check_ssl_certificate

Specifies whether or not the cloud bursting hook checks that the PCLM platform has an SSL certificate.

If you set this to True, you need to set up an SSL certificate for the PCLM platform. Then you can use *https://* for the endpoint for PBS Cloud instead of *http://*.

Default: False; the hook does not check for an SSL certificate

check_resources

Specifies list of static consumable server-level license-tracking resources to check for license availability.

Format: comma-separated list of quoted resources, enclosed in square brackets

Examples:

```
"check_resources": [],
```

or

```
"check_resources": ["App1", "App2"],
```

Default: no default

preemptable

Specifies whether this scenario supports preemptable or spot instances.

Default: False

use_node_hour_license

This parameter must be set to *false* for PBS Cloud.

Default: False

4.2.3 Steps to Configure Cloud Bursting Hook

To summarize the process, you export the cloud bursting hook configuration file, edit it for your site and add scenarios, and import it into the cloud bursting hook. Then you set the hook's frequency and alarm timeout, and enable the hook. You can create as many scenarios as you need.

1. Log in to the PBS server host as root
2. Export the cloud bursting hook configuration to a file:

```
qmgr -c "export hook cloud_hook application/x-config default" > cloud_config.json
```

3. Edit the cloud hook configuration file.
4. Set pclm_server to the endpoint for PBS Cloud:

   Format is either of these:

   http://<PBS Cloud hostname>:<port>/pbspro-cloud/
   http://<PBS Cloud IP address>:<port>/pbspro-cloud/

   Default port: *9980*

5. Set the value of cloud_min_instances to the minimum number of instances to be present in the cloud at any time. Required for bursting.
6. Set resources to a comma-separated list of resources that are to be considered for calculating the number of nodes to burst. Resource names must be in quotes. Use one of the following strings:
   - ["ncpus", "mem", "ngpus"]
   - ["ncpus", "mem"]
7. Create each scenario; see section 4.2.4, “Defining a Scenario in the Cloud Bursting Hook Configuration File”, on page 43.
8. Re-import the hook configuration file:

   ```
   qmgr -c "import hook cloud_hook application/x-config default cloud_config.json"
   ```
9. The default cloud bursting hook frequency is 2 minutes (120 seconds). Optionally set the frequency; the format is integer seconds:

   ```
   qmgr -c "set hook cloud_hook freq=<number of seconds>"
   ```
10. The default cloud bursting hook alarm timeout is 10 minutes (600 seconds). We recommend setting this to less than 20 minutes (1200 seconds). Consider the following factors:
    - Time required to burst nodes in the cloud
    - Time required to unburst nodes in the cloud
    - Number of cloud queues

    Set the alarm timeout in integer seconds:

    ```
    qmgr -c "set hook cloud_hook alarm=<number of seconds>"
    ```
11. The cloud bursting hook is disabled by default. Enable the cloud bursting hook:

    ```
    qmgr -c "set hook cloud_hook enabled=True"
    ```

4.2.4 Defining a Scenario in the Cloud Bursting Hook Configuration File

When you define a scenario in the cloud bursting hook configuration file, you are telling the cloud bursting hook about a scenario that you have already created using the PBS Cloud interface. The cloud bursting hook uses the API key that you generated using the PBS Cloud interface to identify the correct scenario to burst.

4.2.4.1 Prerequisites for Defining a Scenario in Hook Configuration File

- The cloud queue for this scenario must exist and be configured for this scenario. See section 6.1.3, “Create and Configure Cloud Queues”, on page 141.
- The PBS Cloud scenario must exist and have an API key. See section 3.3.4, “Create a Bursting Scenario”, on page 25.

4.2.4.2 Steps to Define a Scenario in Hook Configuration File

Define each scenario in the "scenario" section, under its own name. You can add as many scenarios as you want.

- Set the value of api_key to the API key you generated for this scenario using the PBS Cloud interface. You can use the API key for only one scenario; each API key can appear only once in the cloud hook configuration file, and in only one PBS Cloud scenario.
- Set the value of cloud_default_image to the OS image identifier (name or ID; see vendor instructions) that should be used for bursting. This is the OS image you created using the vendor interface. If a job submitter specifies an OS image, that overrides the default.
- Set the value of cloud_max_instances to the maximum number of instances that can be made available in the cloud. Required. Must be greater than 0.

  Format: integer seconds
- Optionally, set the value of max_vms_for_infiniband_scaleset to define the maximum number of nodes allowed on a single InfiniBand switch.

  This value should match the value of Maximum number of VMs inside a ScaleSet as specified in the Azure bursting scenario. If you are not using InfiniBand, then you can eliminate this parameter. Supported by Azure only.
- Set the value of max_nodes_per_burst to the maximum number of nodes allowed to burst in a single hook cycle.
- Set the value of cloud_node_wait_timeout to the maximum time to wait for freshly burst node to become usable.

  Minimum value: 180 seconds. You can set this to a higher value, but not lower.

  Default: 180 seconds
- If this scenario requires application licenses, set check_resources to the static consumable server-level resources that track those licenses (these are updated via `cron` scripts, and are **not** the same as the dynamic resources updated by server_dyn_res scripts). For example, if this scenario needs two kinds of application license App1 and App2, and you track them with resources app1_static and app2_static, set check_resources like this:

  ```
  "check_resources": ["app1_static","app2_static"],
  ```

  The cloud bursting hook bursts this scenario only when the listed resources indicate that licenses are available.
- Set preemptable to:
  - True: the bursting scenario supports preemptable or spot instances and cloud jobs can be preempted.
  - False: jobs that are run in the cloud should not be preempted.

  Default: *False*

4.2.5 Modifying the Cloud Bursting Hook Configuration File

When you modify the cloud bursting hook configuration file, the hook uses the new configuration information the next time it runs.

1. If you are modifying a scenario, disable bursting for the scenario. See section 7.1.4, “Disabling Bursting for a Scenario and Queue”, on page 122.
2. Log in to the PBS server host as root
3. Export the cloud bursting hook configuration to a file:

   ```
   qmgr -c "export hook cloud_hook application/x-config default" > cloud_config.json
   ```
4. Edit the cloud hook configuration file as needed.
5. To add or modify a scenario, follow the steps in section 4.2.4, “Defining a Scenario in the Cloud Bursting Hook Configuration File”, on page 43.

6. Re-import the hook configuration file:

   ```
   qmgr -c "import hook cloud_hook application/x-config default cloud_config.json"
   ```

7. If you modified a scenario, re-enable bursting for the scenario; see section 7.1.5, “Re-enabling Bursting for a Scenario and Queue”, on page 123.

4.2.6 Adding or Changing a Scenario in the Cloud Bursting Hook Configuration File

1. Disable bursting for the scenario. See section 7.1.4, “Disabling Bursting for a Scenario and Queue”, on page 122.
2. Log in to the PBS server host as root
3. Export the cloud bursting hook configuration to a file:

   ```
   qmgr -c "export hook cloud_hook application/x-config default" > cloud_config.json
   ```

4. To add or modify a scenario, follow the steps in section 4.2.4, “Defining a Scenario in the Cloud Bursting Hook Configuration File”, on page 43.
5. Re-import the hook configuration file:

   ```
   qmgr -c "import hook cloud_hook application/x-config default cloud_config.json"
   ```

6. Re-enable bursting for the scenario; see section 7.1.5, “Re-enabling Bursting for a Scenario and Queue”, on page 123.

4.2.7 Creating a Scenario for a Preemptable or Spot Instance

- If you are changing an existing scenario:
  - Disable bursting for the scenario. See section 7.1.4, “Disabling Bursting for a Scenario and Queue”, on page 122.
  - Make sure the PBS Cloud scenario exists and has only preemptable instance types. If the scenario exists but has non-preemptable instance types:
    1. Create a new PBS Cloud scenario for the instance type
    2. Create and configure a cloud queue for the new scenario
- If the PBS Cloud scenario doesn't exist, create it; see section 3.3.4, “Create a Bursting Scenario”, on page 25

1. Log in to the PBS server host as root
2. Export the cloud bursting hook configuration to a file:

   ```
   qmgr -c "export hook cloud_hook application/x-config default" > cloud_config.json
   ```

3. To add or modify a scenario, follow the steps in section 4.2.4, “Defining a Scenario in the Cloud Bursting Hook Configuration File”, on page 43.
4. Make sure that the preemptable parameter is set to *true* for the scenario:

   ```
   "preemptable": true
   ```

5. Re-import the hook configuration file:

   ```
   qmgr -c "import hook cloud_hook application/x-config default cloud_config.json"
   ```

6. If you disabled bursting for the scenario, re-enable bursting; see section 7.1.5, “Re-enabling Bursting for a Scenario and Queue”, on page 123.

4.2.8 Deleting a Scenario from the Cloud Bursting Hook Configuration File

1. Disable bursting for the scenario. See section 7.1.4, “Disabling Bursting for a Scenario and Queue”, on page 122.
2. Log in to the PBS server host as root
3. Export the cloud bursting hook configuration to a file:

   ```
   qmgr -c "export hook cloud_hook application/x-config default" > cloud_config.json
   ```
4. Edit the configuration file: delete the scenario from the "scenario" section.
5. Re-import the hook configuration file:

   ```
   qmgr -c "import hook cloud_hook application/x-config default cloud_config.json"
   ```
6. Re-enable bursting for the scenario; see section 7.1.5, “Re-enabling Bursting for a Scenario and Queue”, on page 123.

4.2.9 Changing PBS Cloud Host or Port

If you will move the PBS Cloud module to a different host or port:

1. Disable bursting for each scenario. See section 7.1.4, “Disabling Bursting for a Scenario and Queue”, on page 122.
2. Log in to the PBS server host as root
3. Export the cloud bursting hook configuration to a file:

   ```
   qmgr -c "export hook cloud_hook application/x-config default" > cloud_config.json
   ```
4. Edit the cloud hook configuration file as needed.
5. Set pclm_server to the new endpoint for PBS Cloud:

   ```
   http://<IP address or hostname of the PBS Cloud host>:<port number>/pbspro-cloud/
   ```
6. Re-import the hook configuration file:

   ```
   qmgr -c "import hook cloud_hook application/x-config default cloud_config.json"
   ```
7. Re-enable bursting for each scenario; see section 7.1.5, “Re-enabling Bursting for a Scenario and Queue”, on page 123.

4.3 Testing Automated Cloud Bursting

4.3.1 Prerequisites for Testing Cloud Bursting Hook

- You have installed PBS Professional and PBS Cloud, configured PBS Professional, created a cloud administrator account at your cloud provider and added that account to PBS Cloud, created and configured your cloud provider components including an OS image, and configured PBS Cloud. These steps are outlined in section 3.1, "Overview of Configuring PBS Cloud", on page 15.
- Make sure the cloud bursting hook is enabled.
- You should have at least one PBS Cloud scenario to test.
- This scenario must be enabled in PBS Cloud; see section 7.1.3, "Enabling or Disabling a Bursting Scenario in PBS Cloud", on page 122
- You have created a cloud queue for this scenario; see section 6.1.3, "Create and Configure Cloud Queues", on page 141
- Make sure that the scenarios in the hook configuration file match the scenarios in PBS Cloud, especially whether or not a scenario is preemptable

4.3.2 Steps to Test Automated Cloud Bursting

1. Log into the PBS server host as root

- Add the scenario you are testing to the cloud bursting hook; see section 4.2.6, "Adding or Changing a Scenario in the Cloud Bursting Hook Configuration File", on page 45

2. Enable the cloud bursting hook:

```
qmgr -c "set hook cloud_hook enabled=True"
```

3. Specify that all log events should be captured in the PBS server logs:

```
qmgr -c "set server log_events=2047"
```

1. Submit jobs to the cloud queue for this scenario:

```
qsub -l select=1:ncpus=4 -q <scenario queue> TestJobScript.sh
qsub -l select=1:ncpus=4 -q <scenario queue> TestJobScript.sh
```

2. Check the status of the jobs:

```
qstat -s
```

3. Tail the PBS Professional server logs:

```
tail -f PBS_HOME/server_logs/<current PBS server log file>
```

4. Check the current log file to verify that the cloud bursting hook is started. Search for the name of the cloud bursting hook:

```
<PBS server>@<PBS server host>;Hook;<cloud_hook>;started
```

5. Log into PBS Cloud and go to your burst scenario. You should see the initiation of the workflow that is triggered by the bursting hook. The workflow should automatically start within a couple of minutes.

6. List the nodes known to PBS Professional in order to verify that the cloud bursting hook has burst cloud nodes:

```
pbsnodes -av
```

7. Once a node is burst, jobs should start running. Check the status of the jobs:

   ```
   qstat -s
   ```

8. You should see that any nodes that were burst for the test are unburst after the Idle Before Unburst period has elapsed.

9. Go back to your normal server log levels. Reset the server log_events attribute to its previous value:

   ```
   qmgr -c "set server log_events=<previous value>"
   ```

5
Using Cloud Provider Services

5.1 Configuring Amazon Web Service Cloud Bursting

5.1.1 Types of Amazon Accounts

Amazon has two kinds of accounts: owner (root user), and AWS Identity and Access Management (IAM) users. Here we outline the steps you will follow below:

1. Create and activate an AWS root user account
2. Use the root user account to create an AWS IAM account and give the account administrator permissions.
3. Use the AWS IAM administrator account to do all administrative tasks. This is the account that PBS Cloud will use to manage cloud nodes.
4. Use the AWS IAM administrator account to create the AWS components required for cloud bursting.

5.1.2 Creating and Activating AWS Owner Account

Create and activate your AWS owner account. See How do I create and activate a new Amazon Web Services account?

5.1.3 Creating an AWS IAM User Account

Follow these steps to create an AWS user account and give this account administrative permissions. See Creating an IAM User in Your AWS Account. During this process, make sure you download a CSV file containing the following:

- Access key ID
- Secret access key

We will remind you of this step.

1. Log in to the AWS console.
2. Using the search box located under AWS services, enter IAM.
3. Click the IAM search result to open the Identity and Access Management dashboard.
4. In the navigation pane on the left-hand side of the web page, click Users.
5. Click Add user.
6. Enter the following information for this user:
    a. For User name, enter a name for the user.

       The name can be anything meaningful to your organization, e.g., pc_clouduser.

    b. For Access type, enable Programmatic access.

       The user requires this type of access because PBS Cloud needs to make API calls or use the AWS CLI. The AWS interface generates an access key ID and a secret access key for the user.

7. Click Next:Permissions.
8. Optional: Click Add user to group. This button may already be selected.
9. Click Create group.
10. Enter the following information to create a group, add the user to the group, and choose a permission policy for the group:
    a. For Group name, enter a group name.

       The name can be anything meaningful to your organization, e.g., pc_cloudgroup.

    b. For Policy type, enable Administrator Access.

       This policy provides full access to AWS services and resources.
11. Click Create group.

    This returns you to the Add user page and enables the new group, indicating that the user is added to the new group.
12. Click Next: Tags.
13. Click Next: Review.
14. Click Create user.
15. Click Download.csv.
16. **Download and sav**e this file in a secure location.

    PBS Cloud will use the access key ID and secret access key in this file to manage cloud nodes.
17. Click Close.

    The new user account is displayed.

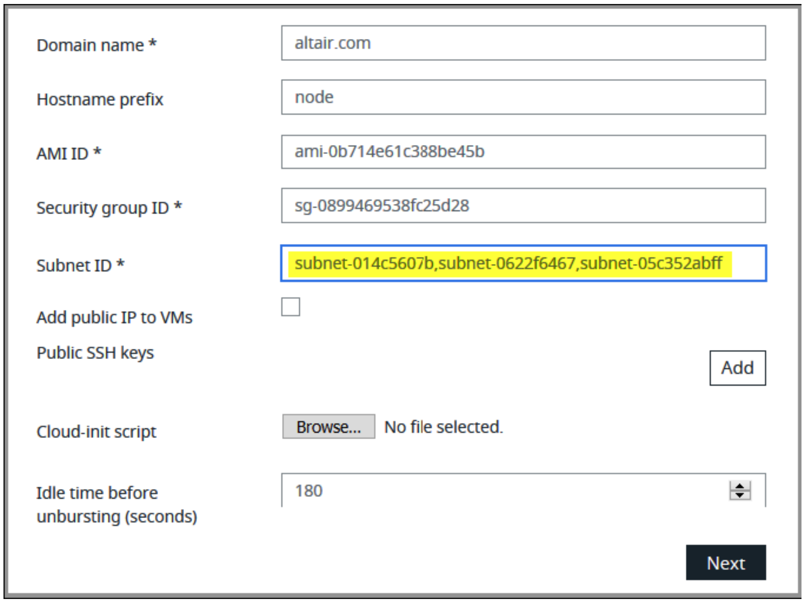
5.1.4 Multi-Availability Zone Management on AWS

If you are not familiar with AWS regions, Availability Zones, VPCs or subnets, see the following AWS documentation:

- Regions, Availability Zones, and Local Zones
- VPCs and Subnets

Bursting cloud nodes in multiple Availability Zones allows an HPC complex to distribute the load across a region and take advantage of AWS Spot Instances. To use multiple Availability Zones, your virtual private cloud must have a subnet for each Availability Zone; all these subnets must belong to the same VPC.

Once these prerequisites are met, then it is as simple as providing a comma-separated list of subnets when the bursting scenario is created.



Domain name *	altair.com
Hostname prefix	node
AMI ID *	ami-0b714e61c388be45b
Security group ID *	sg-0899469538fc25d28
Subnet ID *	subnet-014c5607b,subnet-0622f6467,subnet-05c352abff
Add public IP to VMs	□
Public SSH keys	Add
Cloud-init script	Browse... No file selected.
Idle time before unbursting (seconds)	180

Figure 5-1:List of Subnets

PBS Cloud attempts to burst cloud nodes in the first subnet in the list. If there is no availability in that subnet, then it attempts to burst cloud nodes in the next subnet in the list and will continue until it finds a subnet where it can burst the cloud nodes or until bursting fails because no subnets have availability. The cloud bursting hook attempts to burst all requested cloud nodes in a single subnet. It does not burst cloud nodes across subnets. The cloud bursting hook follows this process each bursting cycle until it finds availability to burst the cloud nodes.

Example 5-1: 10 cloud nodes are requested for bursting.

a. The cloud bursting hook attempts to burst all 10 nodes in subnet-014c5607b.

b. If there is no availability in subnet-014c5607b, the hook attempts to burst all 10 cloud nodes in subnet-0622f6467.

c. If there is no availability in subnet-0622f6467, the hook attempts to burst all 10 cloud nodes in subnet-05c352abff.

d. If there is no availability in subnet-05c352abff, cloud bursting fails for this cycle.

5.1.5 Create a Virtual Private Cloud Network

5.1.5.1 Choose a Region

Log in to your AWS Management Console and choose a region based on the geographical location of your users. All cloud resources that are created are placed in this region.

For more information see Regions and Availability Zones. The menu for selecting a region is located at the upper right-hand corner of the AWS Console menu bar.

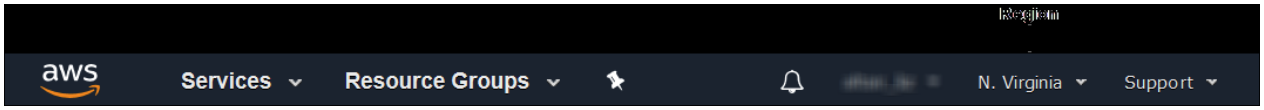


Figure 5-2:AWS Region

AWS documentation can be found at Getting Started with IPv4 for Amazon VPC and Working with VPCs and Subnets.

Record the region(s) you selected; you will use this later in the Region parameter in the bursting scenario.

5.1.5.2 Create a VPC

1. Log in to the AWS console.
2. Click aws located in the upper left-hand corner of the web page.
3. Using the search box located under AWS services, enter VPC.
4. Click the VPC search result. The VPC dashboard is opened.
5. In the menu located on the left-hand side of the web page, click Your VPCs.
6. To create a virtual private cloud, click Create VPC.
7. Enter the following to create a VPC:
    a. For Name, enter any name for the VPC.
       The name can be anything meaningful to your organization, e.g., bursting_vpc.
    b. For IPv4 CIDR block, provide an address range in CIDR notation.
    c. For IPv6 CIDR block, enable No IPv6 CIDR Block.
    d. For Tenancy, choose Default.
    e. Click Yes, Create.

5.1.5.3 Create Subnets for the VPC

Create at least one subnet for the VPC. To allow node bursting in several Availability Zones, create a subnet for each Availability Zone that you want to burst in. For more information see Multi-Availability Zone Management on AWS.

1. In the menu located on the left-hand side of the web page, click Subnets.
2. Click Create Subnet.
    a. For Name tag, enter a name for the subnet.
       The name can be anything meaningful to your organization, e.g., bursting_subnet.
    b. For VPC, choose the VPC that was previously created (e.g. bursting_vpc).
    c. For Availability Zone, choose one of the following options:
        - Choose a unique availability zone for each subnet.
        - Choose No Preference to let Amazon choose an Availability Zone for you.
    d. For IPv4 CIDR block, provide an address range in CIDR notation.
    e. Click Create.
3. Click Close.

5.1.6 Create an Internet Gateway

You can SSH into a virtual machine that is used for cloud bursting via an internet gateway. You create an internet gateway and attach it to the bursting VPC. See AWS documentation at Internet Gateways.

1. Log in to the AWS console.
2. Click aws located in the upper left-hand corner of the web page.
3. Using the search box located under AWS services, enter VPC.

4. Click the VPC search result to open the VPC dashboard.
5. In the menu located on the left-hand side of the web page, click Internet Gateways.
6. Click Create internet gateway.
7. Enter a value for Name tag, to be a name for the internet gateway.

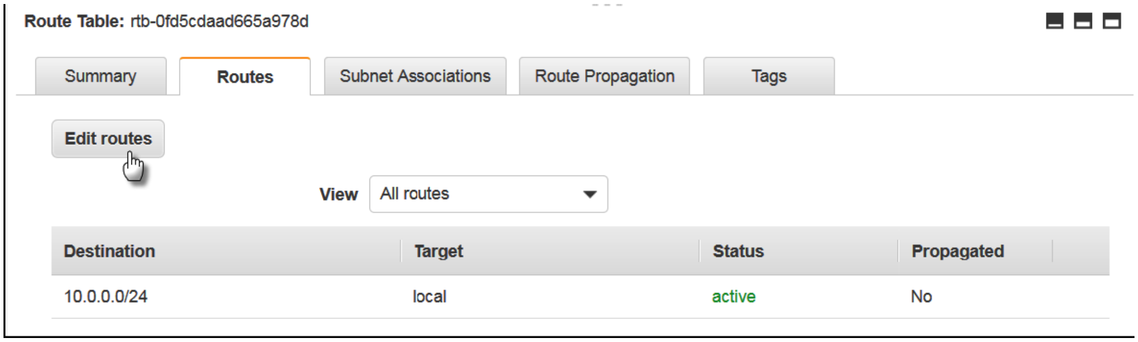
   The name can be anything meaningful to your organization, e.g., bursting_gateway
8. Click Create.
9. Click Close.
10. Select the internet gateway that you just created by enabling the check box next to the name of the gateway.
11. You may need to deselect any other internet gateways that are displayed in the list. Amazon creates default resources for your selected region so a default internet gateway may already exist.
12. Click Actions > Attach to VPC.
13. Select the VPC that you created previously (e.g. bursting_vpc).
14. Click Attach.

5.1.7 Update the VPC Route Table

You add a rule to the VPC route table that allows all internet access, and associate the route table with the bursting subnet. You can find AWS documentation at Route Tables.

1. Log in to the AWS console.
2. Click aws located in the upper left-hand corner of the web page.
3. Using the search box located under AWS services, enter VPC.
4. Click the VPC search result to open the VPC dashboard.
5. In the menu located on the left-hand side of the web page, click Route Tables to display a list of route tables.
6. Select the route table attached to your VPC (e.g. bursting_vpc) by enabling the check box next to the name of the route table.

   The VPC column in the route table list specifies the VPC to which the route table is attached.
7. Click the Routes tab at the bottom of the web page.
8. Click Edit routes.



Route Table: rtb-0fd5cdaad665a978d

Destination	Target	Status	Propagated
10.0.0.0/24	local	active	No

Figure 5-3:Add a Route

9. Click Add route.

10. Enter the following to add a rule that allows all traffic access to the internet gateway:
    - For Destination enter the PBS Cloud firewall IP address.
    - For Target, select Internet Gateway, then the internet gateway that you created previously (e.g. bursting_gateway).
11. Click Save routes.
12. Click Close.
13. Associate the route table to the bursting subnet:
    a. Click the Subnet Associations tab.
    b. Click Edit subnet associations.
    c. Select the subnet you created for cloud bursting from the list.
14. Click Save.

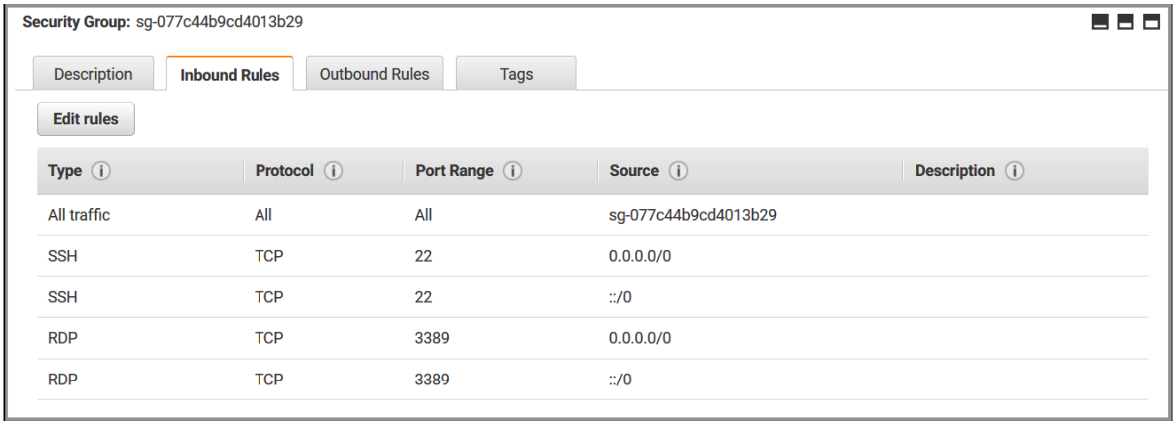
5.1.8 Add Inbound Rules to the VPC Security Group

Add inbound rules to the VPC security group so that a connection can be established with an AWS VM using SSH or RDP.

1. Log in to the AWS console.
2. Click aws located in the upper left-hand corner of the web page.
3. Using the search box located under AWS services, enter VPC.
4. Click the VPC search result to open the VPC dashboard.
5. In the menu located on the left-hand side of the web page, under Security, click Security Groups.
6. Select the security group associated with the VPC you created for cloud bursting by enabling the check box next to its name.

   When you created the VPC, the vendor system created a default VPC security group.
7. Click the Inbound Rules tab at the bottom of the web page.
8. Click Edit rules.
9. Click Add Rule.

10. Add security rules based on your site's requirements. If you enable a public IP address for the associated scenario, there is access to these ports, but the rules here filter who is allowed that access.
    - On Linux platforms, add an inbound rule to allow SSH traffic on port 22.
    - On Windows platforms, add an inbound rule to allow RDP traffic on port 3389.
    - Add the IP address of the PBS Cloud firewall (replace what is here):



Security Group: sg-077c44b9cd4013b29

Type	Protocol	Port Range	Source	Description
All traffic	All	All	sg-077c44b9cd4013b29	
SSH	TCP	22	0.0.0.0/0	
SSH	TCP	22	::/0	
RDP	TCP	3389	0.0.0.0/0	
RDP	TCP	3389	::/0	

Figure 5-4:Security Rules

Warning: 0.0.0.0/0 enables all IPv4 addresses to access your instance. ::/0 enables all IPv6 address to access your instance. This is acceptable for a short time in a test environment, but it's unsafe for production environments. In production, authorize only a specific IP address or range of addresses to access your instance.

11. Click Save rules.

5.1.9 Create a Virtual Machine

In this section you create a virtual machine in AWS Elastic Compute Cloud (EC2).

For AWS documentation, see Launch a Linux Virtual Machine and Launching a Virtual Machine with Amazon EC2.

1. Log in to the AWS console.
2. Click aws located in the upper left-hand corner of the web page.
3. Using the search box located under AWS services, enter EC2.
4. Click the EC2 search result to open the EC2 dashboard.
5. In the menu located on the left-hand side of the web page, click Instances.
6. Click Launch Instance.
7. In the menu located on the left-hand side of the web page, click AWS Marketplace.
8. Using the search box:
    - On Linux platforms, choose a Linux platform that is supported for the PBS MoM and press ENTER.
    - On Windows platforms, choose a Windows platform that is supported for the PBS MoM and press ENTER.
9. Locate the appropriate Amazon Machine Image (AMI) and click Select.
10. Click Continue.

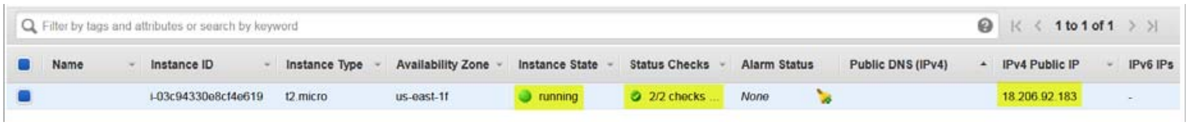
11. Select an Instance Type appropriate for your site's workload, based on these criteria:
    - Number of cores
    - Amount of memory
    - Storage
    - Network performance

    Consider the nature of the applications that you plan to deploy on the instance, the number of users that you expect to use the applications, and also how you expect the load to scale in the future. Remember to also factor in the CPU and memory resources that are necessary for the operating system.
12. Click Next: Configure Instance Details.
13. Enter the following to configure instance details:
    a. For Number of instances, specify 1.
    b. For Network, choose the VPC that you previously created (e.g. bursting_vpc). The bursting subnet is populated automatically.
    c. For Auto-assign Public IP, select Enable.
14. Click Next: Add Storage.
15. Specify the storage options your site needs. We recommend enabling Delete on Termination to delete EBS volumes (attached disks) when the virtual machine is terminated.
16. Click Next: Add Tags.
17. Optional: You can add tags for the VM in key-value pairs.
18. Click Next: Configure Security Group.
19. Assign a security group to the VM. Enter the following:
    a. For Assign a security group, enable Select an existing security group.
    b. Select the security group that was automatically created for the cloud bursting VPC by enabling the check box next to its name.
20. Click Review and Launch.
21. Review the information about the VM and click Launch.
22. Create a new public/private key pair for the VM. Enter the following:
    a. Select Create a new key pair.
    b. Provide a name for the key pair.
    c. Click Download Key Pair.
    d. Download and save this file in a secure location.

       PBS Cloud will use the information in this .pem file later to SSH into the cloud node.
23. Click Launch Instances.
24. At the bottom of the web page, click View Instances.

    This displays all virtual machines that have been created.

Your virtual machine is ready when the Instance State is "running" and Status Checks are complete. The virtual machine can be accessed via its IPv4 Public IP.



Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs
	i-03c94330e8cf4e619	t2.micro	us-east-1f	● running	✓ 2/2 checks ...	None		18.206.92.183	-

Figure 5-5:Bursting Virtual Machine

5.1.10 Install a PBS MoM on the VM

5.1.10.1 Installing a PBS MoM on a Linux VM

On Linux platforms, the username for logging into the virtual machine is dependent on the Amazon Machine Image (AMI) that you used to create the virtual machine. For example, the username for a CentOS AMI is "centos". Typically you log in as "centos", using your SSH key, then switch to root:

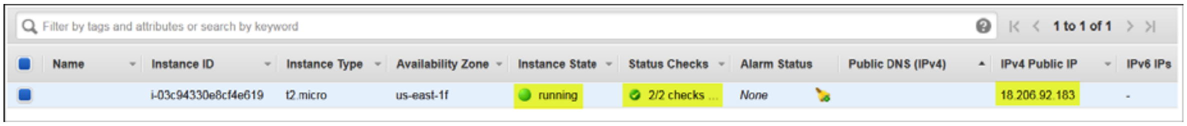
```
sudo -
```

For information about usernames and instructions for connecting and copying files to the Linux virtual machine see Connecting to Your Linux Instance Using SSH.

To establish a connection to the VM, you need the .pem file you downloaded while creating the VM.

1. Copy the PBS Professional installer package to the virtual machine. Use scp to copy the tarball file from the PBS server host to the virtual machine. For more information, see Transferring Files to Linux Instances from Linux Using SCP.
2. Log in to your site's PBS Professional server host.
3. SSH into the virtual machine as the user "centos" using the .pem file and the IPv4 Public IP assigned to the VM:

   ```
   ssh -i /<path to .pem file>/<.pem file> centos@<public IP address of virtual machine>
   ```

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs
	i-03c94330e8cf4e619	t2.micro	us-east-1f	● running	✓ 2/2 checks ...	None		18.206.92.183	-

   Figure 5-6:Bursting Virtual Machine

4. Switch to root:

   ```
   sudo -i
   ```

5. Copy the PBS Professional installation package to the VM.
6. Using the PBS Professional Installation and Upgrade Guide, install and configure the PBS Professional MoM.
7. Configure the VM to work with your site environment, for example mounting file systems, connecting to the authentication service, installing any applications you need, etc. We recommend that you include the following:
   - Mount /home in the VM
   - Either install applications in the VM or cross-mount them from the head node
   - Either add users to the password file or connect the VM to a service such as NIS
8. If cloud-init is not installed, install it.

5.1.10.2 Installing the PBS MoM on the Windows VM

Use an RDP client to access the virtual machine. You can establish a connection to the Windows virtual machine through the AWS EC2 console. See Connect to Your Windows Instance for more information.

You will need the .pem file downloaded while creating the VM to establish a connection.

You will copy the PBS Professional installer package to the virtual machine, and use RDP to map a local drive to get access to the installer package. For more information, see Transfer Files to Windows Instances.

1. Log in to the AWS console.
2. Click aws located in the upper left-hand corner of the web page.
3. Using the search box located under AWS services, enter EC2.
4. Click the EC2 search result.
5. In the menu located on the left-hand side of the web page, click Instances.
6. Select the Windows virtual machine created for cloud bursting by enabling the check box next to its name.
7. At the top, click Connect.
8. Click Get Password.
9. Browse to the .pem file downloaded while creating the VM.
10. Open the .pem file.
11. Click Decrypt Password.
12. Hover over the decrypted password. You will see a copy to clipboard icon.
13. Click the copy to clipboard icon.
14. Click Download Remote Desktop File.
15. Open the file.
16. Click Connect.
17. For Password, paste the password copied to the clipboard.
18. Click OK.
19. Click Yes to connect, even if there are certificate errors.

    A connection is established with the Windows virtual machine.
20. Copy the PBS Professional installation package to the VM.
21. Using the PBS Professional Installation and Upgrade Guide, install and configure the PBS Professional MoM.
22. Configure the VM for your site's environment, for example mounting file systems, connecting to the authentication service, installing any applications, etc.

5.1.11 Create an OS Image

In the following steps, you will create an image of the virtual machine you have configured.

You can find AWS documentation at Create an AMI from an Amazon EC2 Instance.

1. Log in to the AWS console.
2. Click aws located in the upper left-hand corner of the web page.

3. Using the search box located under AWS services, enter EC2.
4. Click the EC2 search result to open the EC2 dashboard.
5. In the menu located on the left-hand side of the web page, click Instances.
6. Select the virtual machine created for cloud bursting by enabling the check box next to its name.
7. At the top, click Actions > Instance State > Stop.
8. Click Yes, Stop.

   It may take some time for the virtual machine to be stopped.

   Do not proceed until the Instance State is "Stopped".
9. Click Actions > Image > Create Image.
10. For Image name, enter a name for the image.

    The name can be anything meaningful to your organization, e.g., bursting_image.

    On Windows platforms, the name of the image must contain the string "windows" (case-insensitive). For example, Windows_Server-2012-R2__RTM-English-64Bit-Base-2019.11.13.
11. Click Create Image.
12. Click the View pending image ami-xxxxxxxxxx link. The image is complete when its Status is "available".
13. You can delete the virtual machine now to avoid storage costs, or keep the virtual machine and update it over time to create updated OS images for bursting. You will incur storage costs, but this is an effective way to keep your OS images up to date when there are changes in packages, patches, or applications.

5.1.12 Collect Information for an AWS Cloud Bursting Scenario

5.1.12.1 Scenario Parameters to Collect at Vendor Interface

Make sure that you capture the following scenario parameters. We will remind you about them:

Table 5-1: Scenario Parameters for Amazon Web Services (AWS)

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Cloud account	Name of your account at cloud vendor	String
Region	Region selected during configuration at cloud vendor	Drop-down list
Domain name	Domain used by PBS Cloud module head node	String
Hostname prefix	Optional prefix for burst node names; default is "node"; chosen during configuration at vendor	String
AMI ID	Name of image to be burst; chosen during configuration at vendor	String
Security Group ID	Name of security group associated with VPC and VM created at vendor	String
Subnet ID	Name of security group subnet for bursting VPC created at vendor. To burst nodes in multiple Availability Zones, save a comma-separated list of subnet IDs	String

5.1.12.2 Steps to Collect Information

1. Open a browser window and log in to your AWS Management Console.
2. Click aws located in the upper left-hand corner of the web page.
3. Using the search box located under AWS services, enter EC2.
4. Click the EC2 search result to open the EC2 dashboard.
5. In the menu located on the left-hand side of the web page, click AMIs.
6. Select the Amazon Machine Image (AMI) you created for cloud bursting by enabling the check box next to its name.
7. In the Details tab located at the bottom of the web page, hover over the AMI ID so that the interface displays a copy to clipboard icon.
8. Click the copy to clipboard icon.
9. **Save the image name** to use for the AMI ID scenario parameter.
10. In the menu located on the left-hand side of the web page, under NETWORK & SECURITY, click Security Groups.
11. Select the Security Group associated with the VPC and the VM by enabling the check box next to its Group ID.
12. In the Description tab located at the bottom of the web page, hover over the Group ID so that the interface displays a copy to clipboard icon.
13. Click on the copy to clipboard icon.
14. **Save the security group name** to use for the Security Group ID scenario parameter.
15. Click aws located in the upper left-hand corner of the web page.
16. Using the search box located under AWS services, enter VPC.
17. Click the VPC search result to open the VPC dashboard.
18. In the menu located on the left-hand side of the web page, click Subnets.
19. Select subnets for the bursting VPC by checking the box next to their names. Subnets are required in order to make multiple availability zones work.
20. In the Description tab located at the bottom of the web page, hover over the Subnet ID so that the interface displays a copy to clipboard icon.
21. Click the copy to clipboard icon.
22. **Save the security group subnet name** to use for the Subnet ID scenario parameter. To use cloud nodes in several Availability Zones, save a comma-separated list of subnet IDs.

5.2 Configuring Microsoft Azure Cloud Bursting

5.2.1 Prerequisites

Purchase an Azure subscription, get a tenant ID, and get an Azure user account. For more information about subscriptions see What is an Azure subscription. For more information about tenants see How to get an Azure Active Directory tenant.

5.2.2 Register PBS Cloud with Azure

You can find Azure documentation at Quickstart: Register an application with the Microsoft identity platform.

As you work through this section, save the following information in a file. We will remind you about these:

Table 5-2: Account Parameters for Microsoft Azure

Account Parameter	What to Collect During Configuration at Vendor	Format
Client ID	Application ID generated when registering PBS Cloud with the Azure Active Directory	String
Secret Key	Secret Key generated during account creation at vendor	String
AD Tenant ID	Azure tenant ID generated during account creation at vendor	String
Subscription ID	Subscription ID generated during account creation at vendor	String

1. Register PBS Cloud with the Azure Active Directory and create a client secret key:
   a. Log in to to your Microsoft Azure account.
   b. Using the search box, enter app reg. You will see a list of search results.
   c. Under Services, click App registrations.
   d. Click New registration.
   e. Enter the following to register PBS Cloud with the Azure Active Directory:
      1. For Name, enter the name you will use for PBS Cloud at the vendor.

         The name can be anything meaningful to your organization, e.g., pbs_cloud
      2. For Supported account types, choose the option that best suits your organization. Click the Help me choose link for additional information about the available options.
      3. For Redirect URI, select Web and enter the URL https://<PBS Cloud host name or IP address>:<PBS Cloud port>/pc. The default PBS Cloud port is 9980.

where hostname is the hostname of the machine where the PBS Cloud web interface is installed. This is the URL that is used to log in to PBS Cloud.

f. Register the application: click Register.

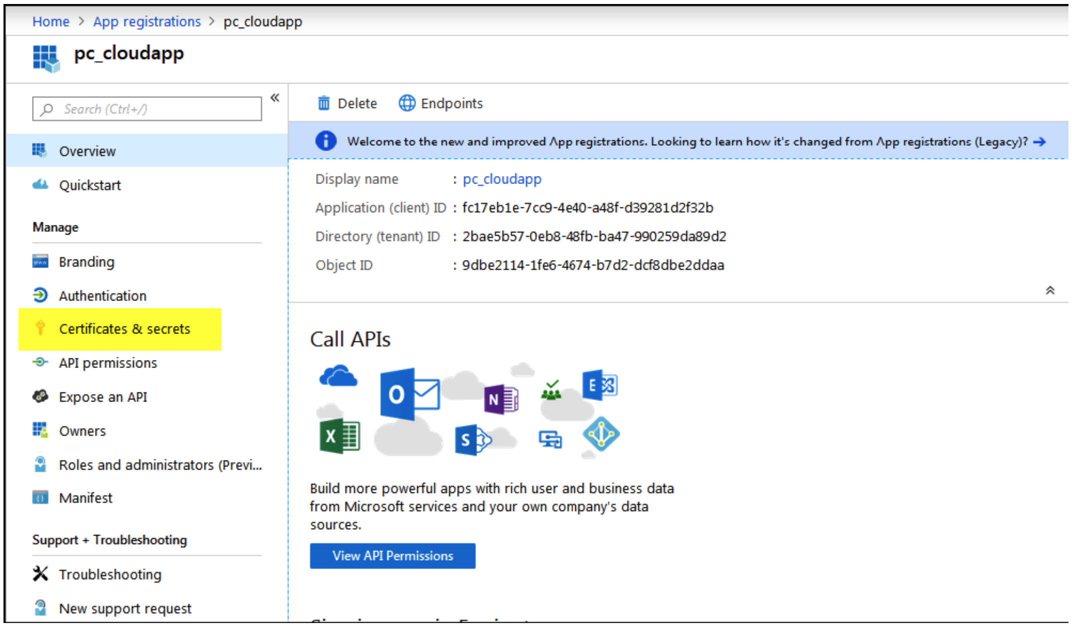
   Once the application registration is complete, its details are displayed, including an Application ID.

g. To get the application ID for PBS Cloud, hover over the Application (client) ID and click the copy-to-clipboard icon when it appears.

h. **Store the application ID to a file**. You will need this later when you add the vendor cloud account to PBS Cloud.

1. Create a client secret key for PBS Cloud:

a. Under, Manage, click Certificates and secrets.



Welcome to the new and improved App registrations. Looking to learn how it's changed from App registrations (Legacy)? →

Display name : pc_cloudapp
Application (client) ID : fc17eb1e-7cc9-4e40-a48f-d39281d2f32b
Directory (tenant) ID : 2bae5b57-0eb8-48fb-ba47-990259da89d2
Object ID : 9dbe2114-1fe6-4674-b7d2-dcf8dbe2ddaa

Call APIs

Build more powerful apps with rich user and business data from Microsoft services and your own company's data sources.

Figure 5-7:Certificates and Secrets

b. Under Client secrets, click New client secret.

c. Enter the following to add a client secret:

   1. For DESCRIPTION, enter pc_client_secret.
   2. For EXPIRES, select Never.

d. Click Add to generate the client secret key.

   You will see the client secret key under the heading VALUE.

e. Click the copy icon next to the client secret key.

f. **Store the client secret key to a file**. The client secret key is used later to create a cloud account in PBS Cloud.

2. Get your Azure subscription ID:
    a. Using the search box, enter subscription. A list of search results is listed.
    b. Under Services, click Subscriptions.
    c. Locate and click your subscription to see details about the subscription, including its Subscription ID.
    d. Hover over the Subscription ID and click on the copy icon when it appears.
    e. **Store the Subscription ID value to a file**. You will use the Subscription ID later when you add the vendor cloud account to PBS Cloud.
3. Assign an access control role to PBS Cloud.
    a. Click Access control (IAM).

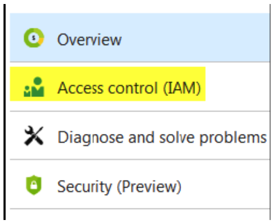


Figure 5-8:Add Access Controls

    b. Click Add.
    c. Click Add role assignment.
    d. In the Add role assignment panel, enter the following to assign a role to PBS Cloud.
        1. For Role, select Contributor.
        2. For Assign access to, select Azure AD user, group, or service principal.
        3. For Select, search for the newly registered application by entering its name, e.g., pbs_cloud.
        4. Select the application by clicking on it.
    e. Click Save.
4. Obtain your Azure tenant ID:
    a. At the top of the web page, click ?.
    b. Click Show diagnostics.

       A dialog box is displayed allowing a file called PortalDiagnostics.json to be saved.
    c. Open the file using any text editor.
    d. Search for tenantId.
    e. **Store the value of tenantId to a file**. You will use the Tenant ID later to add the vendor cloud account to PBS Cloud.

5.2.3 Create a Resource Group

Azure documentation can be found at [Manage Azure resources through portal]().

A resource group is container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization. Once the resource group is created, resources that are placed into the resource group are a virtual network, a virtual machine, and an image of the virtual machine.

1. Log in to your Microsoft Azure account.
2. Using the search box, enter resource groups. A list of search results is listed.
3. Under Services, click Resource Groups.
4. Click Add.
5. Enter the following to configure the basic settings for the resource group:
    a. For Project Details enter the following:
        - For Subscription, choose the subscription to be billed for the use of the VM.
        - For Resource group, enter a name for the resource group.

          The name can be anything meaningful to your organization, e.g., bursting_resource_group.
    b. For Resource Details enter the following:
        - For Region, select a location based on the geographical location of users.
6. Click Review + create.
7. Click Create.

It may take a moment to create the resource group. All resources (networks, virtual machines, etc.) that are created are placed within this resource group. The name of the resource group is required for creating a bursting scenario in PBS Cloud.

5.2.4 Create a Virtual Network

Azure documentation can be found at Virtual Network Documentation.

1. Log in to your Microsoft Azure account.
2. Using the search box, enter virtual networks. A list of search results is listed.
3. Under Services, click Virtual networks.
4. Click Add.
5. For Name, enter a name for the virtual network.

    The name can be anything meaningful to your organization, e.g., bursting_virtual_network
6. For Address space, enter an address range for the network using CIDR notation.
7. For Subscription, select the same subscription as was selected for the previously created resource group.
8. For Resource group, select the previously created resource group.
9. For Location, select the same geographical location as was selected for the previously created resource group.
10. For Subnet, enter the following:
    a. For Name, enter a name for the virtual machine's subnet.

       The name can be anything meaningful to your organization, e.g., bursting_subnet
    b. For Address range, enter an address range for the subnet in CIDR notation.
11. Click Create.

It may take a moment to create the virtual network. The name of the virtual network is required for creating a bursting scenario in PBS Cloud.

5.2.5 Create a Virtual Machine

You may want to view the following web page to learn about Azure Managed Disks before creating a VM. Additionally, a video is available from Microsoft that shows how to create a virtual machine: Create a Linux Virtual Machine.

1. Log in to your Microsoft Azure account.
2. Using the search box, enter virtual machines. A list of search results is listed.
3. Under Services, click Virtual machines.
4. Click Add.

Enter the following to configure the basic settings for the virtual machine:

5. For Project Details enter the following:
   a. For Subscription, choose the subscription to be billed for the use of the VM.
   b. For Resource group, choose the previously created resource group.
   c. For Virtual machine name, enter a name for the virtual machine.
      The name can be anything meaningful to your organization, e.g., bursting-vm.
   d. For Region, select the same geographical location as was selected for the previously created resource group.
   e. For Availability options, choose No infrastructure redundancy required.
   f. For Image, click the Browse all public and private images link.
   g. Using the search box:
      - On Linux platforms, enter CentOS 7 or RHEL 7 and press ENTER.
      - On Windows platforms, enter Windows and press ENTER.
   h. Locate the appropriate image and select it.
      - On Linux platforms, cloud bursting has been tested on on CentOS 7.2 - 7.6.
      - On Windows platforms, cloud bursting has been tested on Windows 10 and Windows Server 2012.
   i. For Size, click the Change size link and select a machine size appropriate for your site's workload based on:
      - the number of cores
      - the amount of memory
      - storage
      - network performance

      Consider the nature of the applications that you plan to deploy on the instance, the number of users that you expect to use the applications, and also how you expect the load to scale in the future. Remember to also factor in the CPU and memory resources that are necessary for the operating system.
   j. Click Select.
6. For Administrator Account, enter a user account :

This user will have sudo rights and will be able to connect to the VM to install the PBS MoM.

- On Linux platforms:
    - For Authentication type, enable SSH public key.
    - For Username, enter a username of a user account that exists on your site's PBS Server.
    - For SSH public key, copy the SSH public key (i.e., id_rsa.pub) of the user account and paste it.
- On Windows platforms:
    - For Username, enter a username.
    - For Password, enter a password.

7. For Inbound Port Rules, enter the following:
    a. For Public inbound ports, enable Allow selected ports.
    b. For Select inbound ports:
        - For Linux platforms, enable SSH (22).
        - For Windows platforms, enable RDH (3389).
8. Click Next.

Enter the following to configure the storage settings for the virtual machine:

9. For Disk Options, enter the following:
    a. For OS disk size, choose an appropriate disk size based on your site's needs.
    b. For OS disk type, choose one of the following options:
        - Premium SSD
        - Standard SSD
        - Standard HDD

        Choose SSD for I/O-intensive applications, where low latency and high throughput are critical. For testing, consider HDD to keep costs down, as you scale up and down quickly.
1. For Advanced, enter the following:
    a. Click Advanced.
    b. For Use managed disks, choose one of the following options:
        - Yes to use managed disks.
        - No to not use managed disks.

        Enable this feature to have Azure automatically manage the availability of disks to provide data redundancy and fault tolerance, without creating and managing storage accounts on your own. This option is recommended by Azure as it is a lot more scalable.
1. Click Next.

Enter the following to configure the networking settings for the virtual machine:

2. For Network Interface, enter the following:
    - For Virtual network, choose the virtual network previously created.
3. Click Review + create.
4. Click Create.

    It may take a few minutes for the VM to be deployed. You will use this virtual machine to create an OS image.

Once the virtual machine is deployed a message is displayed indicating success, click on Go to resource.

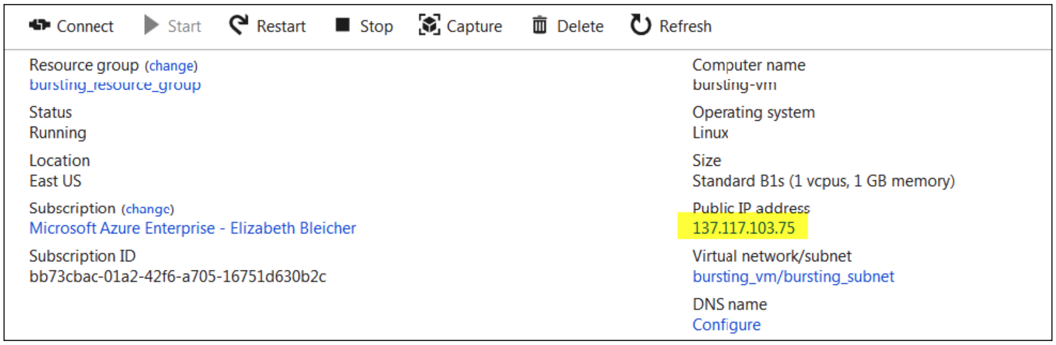
5.2.6 Install a PBS MoM on the VM

5.2.6.1 Install a PBS MoM on a Linux VM

1. Log in to your site's PBS server host as the user account (username and the public SSH key) provided during the creation of the VM.
2. SSH into the virtual machine using the public IP address of the VM:

   ```
   ssh IPV4PublicIP
   ```

   where IPV4PublicIP is the public IP address of the virtual machine.

   Resource group (change)
   bursting_resource_group

   Status
   Running

   Location
   East US

   Subscription (change)
   Microsoft Azure Enterprise - Elizabeth Bleicher

   Subscription ID
   bb73cbac-01a2-42f6-a705-16751d630b2c

   Computer name
   bursting-vm

   Operating system
   Linux

   Size
   Standard B1s (1 vcpus, 1 GB memory)

   Public IP address
   137.117.103.75

   Virtual network/subnet
   bursting_vm/bursting_subnet

   DNS name
   Configure

   Figure 5-9:Bursting Virtual Machine
3. Enter the command:

   ```
   sudo -i
   ```
4. Copy the PBS Professional installation package to the VM. Use SCP to copy the tarball file from the PBS server host to the virtual machine. For more information, see Move files to and from a Linux VM using SCP.
5. Using the PBS Professional Installation and Upgrade Guide, install and configure the PBS MOM.
6. Configure the VM to work with your site environment, for example mounting file systems, connecting to the authentication service, installing any applications you need, etc. We recommend that you include the following:
   - Mount /home in the VM
   - Either install applications in the VM or cross-mount them from the head node
   - Either add users to the password file or connect the VM to a service such as NIS
7. If cloud-init is not installed, install it.

5.2.6.2 Install a PBS MoM on a Windows VM

You will use an RDP client to access the virtual machine. A connection can be established to the Windows virtual machine through the Azure portal. For more information see How to connect and sign on to an Azure virtual machine running Windows.

You will copy the PBS Professional installer package to the virtual machine. Use RDP to map a local drive to gain access to the installer package.

1. Log in to the Azure portal.
2. Using the search box, enter virtual machines.
3. Under Services, click Virtual machines.
4. Select the Windows virtual machine created for cloud bursting by clicking its name.

5. Click Connect.
6. Click the RDP tab.
7. Click Download RDP File.
8. Open the file.
9. Click Connect.
10. Enter the password that was established for the Administrator Account when creating the virtual machine.
11. Click OK.
12. Click Yes to connect even if there are certificate errors.

    A connection is established with the Windows virtual machine.
13. Copy the PBS Professional installation package to the VM.
14. Using the PBS Professional Installation and Upgrade Guide, install and configure the PBS MoM.
15. Configure the VM for your site's environment such as mounting file systems, connecting it to the authentication service, installing any applications, etc.

5.2.7 Create an OS Image

5.2.7.1 Create a Linux OS Image

Creating an OS image requires the Azure CLI. Refer to these instructions for installing the CLI: How to install the Azure CLI. We recommend installing the CLI on a Windows or Mac machine and then using the command prompt to execute the CLI commands.

Before you can create an OS image of the previously created VM, you must first SSH into the VM and deprovision it. Next you will use the Azure CLI to deallocate and generalize the VM and then create the image. Generalizing the virtual machine removes any SSH keys and DNS settings from the VM.

Follow Step 1 and Step 2 as documented in "How to create an image of a virtual machine or VHD" to create an image of the VM.

Before you can deallocate the virtual machine you may have to execute the following commands to set your subscription to be the active subscription:

```
az account list
az account set --subscription <your subscription ID>
```

You can now delete the virtual machine so that you are no longer charged for it.

5.2.7.2 Create a Windows OS Image

You will generalize the VM using Sysprep. For more information see Create a managed image of a generalized VM in Azure.

1. Log in to the Azure portal.
2. Using the search box, enter virtual machines.
3. Under Services, click Virtual machines.
4. Select the Windows virtual machine created for cloud bursting by clicking its name.
5. Click Connect.
6. Click the RDP tab.

7. Click Download RDP File.
8. Open the file.
9. Click Connect.
10. Enter the password that was established for the Administrator Account when creating the virtual machine.
11. Click OK.
12. Click Yes to connect even if there are certificate errors.
13. Open a Command Prompt window as an administrator.
14. Using Windows Explorer, navigate to the directory C:\Windows\System32\Sysprep.
15. Right-click sysprep.exe and select Run as Administrator.
16. For System Cleanup Action, choose Enter System Out-of-Box Experience (OOBE).
17. Enable the Generalize check box.
18. For Shutdown Options, choose Shutdown.
19. Click OK.
20. Once the VM is shut down, close the RDP session.
21. Navigate to the browser window where the Azure portal is open and the VM details are displayed.
22. Click Capture.
23. For name, enter a name for the image.

    The name of the image should contain the string "windows" (case insensitive). For example, Windows Server 2012 R2 Datacenter.
24. For Resource group, choose the previously created resource group.
25. For Type the virtual machine name, enter the name of the VM.
26. Click Create.
27. You can now delete the virtual machine so that you are no longer charged for it.

5.2.8 Collect Information for an Azure Cloud Bursting Scenario

5.2.8.1 Scenario Parameters to Collect at Vendor Interface

Make sure that you capture the following scenario parameters. We will remind you about each one:

Table 5-3: Scenario Parameters for Microsoft Azure

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Cloud account	Name of your account at cloud vendor	String
Region	Region selected during configuration at cloud vendor	Drop-down list
Domain name	Domain used by PBS Cloud module head node	String

Table 5-3: Scenario Parameters for Microsoft Azure

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Hostname prefix	Optional prefix for burst node names; default is "node"; chosen during configuration at vendor	String
Resource group name	Name of resource group (virtual network, virtual machine, OS image) created at vendor	String
Network name	Name of virtual network created at vendor If the network is in a different resource group from the one specified, enter it as Resource Group Name/Virtual Network Name	String
Subnetwork name	Name of virtual subnet created at vendor	String
Network security group name	Name of network security group for resource group	String
Managed Storage	Managed disk feature selected at vendor	Boolean
OS Image	If using managed disks, name of the image. If not using managed disks, Linux Source BLOB URI.	String
Maximum number of VMs inside a ScaleSet with Managed Storage and a single Placement Group	Limit selected during configuration at vendor. Default: 100	Integer

5.2.8.2 Steps to Collect Information

Open a browser window and log in to your Microsoft Azure account.

For information on virtual machine scale sets, see the following Azure article about scale sets: What are virtual machine scale sets.

A bursting scenario requires a resource group, but other scenario resources (network, subnet, network security group and the OS image) can all reside in a different resource group. However, the resource groups must be in the same geographic location for this to work.

28. In the menu located on the left-hand side of the web page, click Resource Groups.
29. Copy the name of the resource group created for cloud bursting.
30. **Save the name of the resource group to a file**. You will use this later when you create the bursting scenario.
31. In the menu located on the left-hand side of the web page, click Virtual Networks.
32. Copy the name of the virtual network created for cloud bursting.
33. **Save the name of the network to a file**. You will use this later when you create the bursting scenario.
34. Select the virtual network created for cloud bursting.

35. Click Subnets.

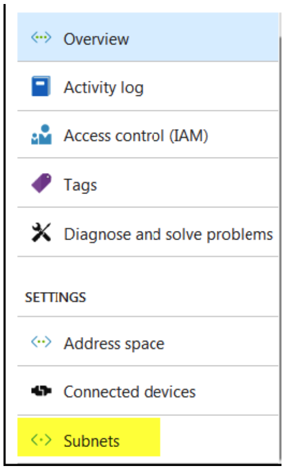


Figure 5-10:Subnet

36. For Subnet name, copy the name of the subnet created for the cloud bursting virtual network.
37. **Save the name of the subnet to a file**. You will use this later when you create the bursting scenario.
38. In the menu located on the left-hand side of the web page, click Resource Groups.
39. Select the Resource Group created for cloud bursting.
40. In the list, locate the Network Security group created for cloud bursting. The Type is Network Security group.
41. Copy the name of the network security group.
42. **Save the name of the network security group to a file**. You will use this later when you create the bursting scenario.
43. In the menu located on the left-hand side of the web page, click Resource Groups.
44. Select the Resource Group created for cloud bursting.
45. In the list, locate the image that was created for cloud bursting. The Type is Image.
46. Select the image.
47. Choose one of the following options:
    - If you chose to use managed disks when you created the VM, copy the name of the image.
    - If you did not choose to use managed disks when you created the VM, copy the Linux Source BLOB URI.
48. **Save the name of the image or the Linux Source BLOB URI to a file**. You will use this later when you create the bursting scenario.

5.3 Configure Google Cloud Platform Cloud Bursting

5.3.1 Sign Up for a GCP Account

Sign up for a GCP user account. This is different from a GCP service account. Go to the Google Account signup page.

5.3.2 Create a Project

Google Cloud Platform projects form the basis for creating, enabling, and using all Cloud Platform services including managing APIs, enabling billing, adding and removing collaborators, and managing permissions for Cloud Platform resources. GCP documentation can be found at Creating and Managing Projects.

1. Log in to the GCP console.
2. Click ☰ located in the upper left-hand corner of the web page.
3. Click Home.
4. Click Create.
5. For Project Name, enter a name for the project.

   The name can be anything meaningful to your organization, e.g., pc_cloudproject.
6. Click Create.

   It may take a few moments to create the project.
7. Using a browser, navigate to the following URL: https://console.developers.google.com/ apis/library/compute.googleapis.com?project=PROJECTNAME

   where PROJECTNAME is the name of the project.
8. Click Enable.

5.3.3 Create a Service Account

GCP documentation can be found at Understanding Service Accounts and Compute Engine IAM Roles .

A service account is a special Google account that is used by applications to use the Google Cloud APIs. PBS Cloud will use a service account to manage cloud nodes.

1. Log in to the GCP console.
2. Click ☰ located in the upper left-hand corner of the web page.
3. Click IAM & admin > Service accounts.
4. Click CREATE SERVICE ACCOUNT.
5. Enter the following to create a service account:
   a. For the Service account name, enter a name for the service account.

      The name can be anything meaningful to your organization, e.g., pc-service-account.

   b. Click CREATE.

   c. For the Project role, click Select a role > Compute Engine > Compute Admin. This role gives full control of all Compute Engine resources.

   d. Click CONTINUE.

   e. Under Create key (optional), click CREATE KEY.

   f. For Key type, enable JSON.

   g. Click CREATE.

   h. **Save the JSON file** in a secure location. Use the dialog box to choose a place to save it. You will need this information later when you add the provider account to PBS Cloud.

6. Make sure you have downloaded a JSON file containing the following:
    - Project ID
    - Client ID
    - Client email
    - Private key ID
    - Private key
7. Click CLOSE.
8. Click DONE.

5.3.4 Create a Virtual Private Cloud Network

GCP documentation can be found at Virtual Private Cloud Documentation and Using VPC Networks .

1. Log in to the GCP console.
2. Click ☰ located in the upper left-hand corner of the web page.
3. Click VPC network > VPC networks.
4. Click CREATE VPC NETWORK.
5. Enter the following to create a VPC:
    a. For the Name, enter a name for the VPC.
    The name can be anything meaningful to your organization, e.g., bursting-vpc.
    b. In the Subnets section, click the Custom tab under Subnet creation mode.
    c. For Name, enter a name for the subnet.
    The name can be anything meaningful to your organization, e.g., bursting-subnet.
    d. For Region, select a Region based on the geographical location of users.
    e. For IP address range, enter an IP address range using CIDR notation
    f. For Private Google access, enable Off.
    g. Click Done.
    h. For Dynamic routing mode, enable Regional.
6. Click Create.

    Creating the VPC network may take some time. Do not proceed until the VPC is created.
7. Select the VPC by clicking on its name.
8. Click the Firewall rules tab.
9. Click CREATE FIREWALL RULE.

10. Enter the following to create a firewall rule:

    a. For Name, enter a name for the firewall rule.

    The name can be anything meaningful to your organization, e.g., ssh-all.

    b. For Direction of Traffic, enable Ingress.

    c. For Action on match, enable Allow.

    d. For Targets, select All instances in the network.

    e. For Source filter, select IP ranges.

    f. For Source IP ranges, enter the IP address of the PBS Cloud firewall

    g. For Protocols and ports, enable Specified protocols and ports.

    h. Enable tcp.

    i. Enter 22.

11. Click Create.

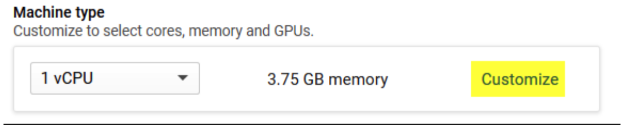
5.3.5 Create a Virtual Machine

GCP documentation can be found at Virtual Machine Instances and Creating and Starting a VM Instance.

1. Log in to the GCP console.
2. Click ☰ located in the upper left-hand corner of the web page.
3. Click Compute Engine > VM instances.
4. Click CREATE INSTANCE.

5. Enter the following to create a virtual machine:
    a. For the Name, enter a name for the virtual machine.

       The name can be anything meaningful to your organization, e.g., bursting-vm.
    b. For Zone, select a zone that is in the same Region as the subnet of the previously created VPC.
    c. In the Machine type, click the Customize link.



Machine type

Customize to select cores, memory and GPUs.

Figure 5-11:Customize the Machine Type

    d. Specify the CPUs, GPUs and RAM.

       Consider the nature of the applications that you plan to deploy on the instance, the number of users that you expect to use the applications, and also how you expect the load to scale in the future. Remember to also factor in the CPU and memory resources that are necessary for the operating system.
    e. For Boot disk, click Change.
    f. Choose CentOS 7.
    g. For Boot disk type, choose one of the following options:
       - Standard persistent disk
       - SSD persistent disk

       Choose SSD for I/O-intensive applications, where low latency and high throughput are critical. For testing, consider Standard persistent disk to keep costs down.
    h. For Size, specify the size of the boot disk.
    i. Click Select.
    j. Under Identity and API access, for Service Account, select No service account.
    k. For Firewall, choose Allow HTTP traffic.
    l. Click Management, disks, networking, SSH Keys.
    m. Click the Networking tab.
    n. Click Add network interface.
    o. For Network, choose the VPC you previously created for bursting.
    p. For Network Service Tier, click Standard.
    q. Click Done.
    r. Delete any default network interfaces that might have been automatically generated.
    s. Click the Security tab.
    t. For SSK Keys, copy the SSH public key (i.e., id_rsa.pub) of the administrator account that exists on your site's PBS server host and paste it.

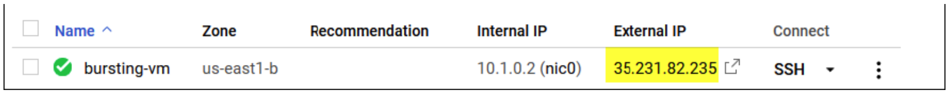
       This user will have sudo rights and will be able to SSH into the VM to install the PBS MoM.
6. Click Create.

   Creating the virtual machine may take some time.

5.3.6 Install and Configure a PBS MoM on the VM

1. Log in to your site's PBS Server as the user account (public SSH key) provided during the creation of the VM.
2. SSH into the virtual machine using the public IP address of the VM:

```
ssh <public IP address of VM>
```



□	Name ^	Zone	Recommendation	Internal IP	External IP	Connect	
□	✓ bursting-vm	us-east1-b		10.1.0.2 (nic0)	35.231.82.235	SSH ▾	⋮

Figure 5-12:Bursting Virtual Machine

3. Switch to root:

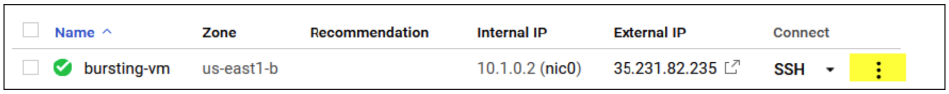
```
sudo -i
```

4. Using the PBS Professional Installation and Upgrade Guide, install and configure the PBS MOM.
5. Configure the VM to work with your site environment, for example mounting file systems, connecting to the authentication service, installing any applications you need, etc. We recommend that you include the following:
   - Mount /home in the VM
   - Either install applications in the VM or cross-mount them from the head node
   - Either add users to the password file or connect the VM to a service such as NIS
6. If cloud-init is not installed, install it.

5.3.7 Create an OS Image

GCP documentation can be found at Creating, Deleting, and Deprecating Custom Images.

1. Log in to the GCP console.
2. Click ☰ located in the upper left-hand corner of the web page.
3. Click Compute Engine > VM instances. A list of virtual machines is displayed.
4. Click the three vertical dots next to the virtual machine that was created for cloud bursting.



□	Name ^	Zone	Recommendation	Internal IP	External IP	Connect	
□	✓ bursting-vm	us-east1-b		10.1.0.2 (nic0)	35.231.82.235	SSH ▾	⋮

Figure 5-13:Bursting Virtual Machine

5. Click Stop.

   It may take some time for the VM to be stopped. Do not proceed until the VM is stopped.
6. In the menu located on the left-hand side of the web page, click Images .
7. Click CREATE IMAGE.
8. Enter the following to create an image:
   a. For Name, enter a name for the image.

   The name can be anything meaningful to your organization, e.g., bursting-image.

   b. For Source select Disk.

   c. For Source disk, select the previously created virtual machine.
9. Click Create.

It may take some time to create the image. Do not proceed until the image is created.

10. You can delete the virtual machine now to avoid storage costs, or keep the virtual machine and update it over time to create updated OS images for bursting. You will incur storage costs, but this is an effective way to keep your OS images up to date when there are changes in packages, patches, or applications.

5.3.8 Collect Information for GCP Cloud Bursting Scenario

5.3.8.1 Scenario Parameters to Collect at Vendor Interface

Make sure that you capture the following scenario parameters. We will remind you about each one:

Table 5-4: Scenario Parameters for Google Cloud Platform (GCP)

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Cloud account	Name of your account at cloud vendor	String
Region	Region selected during configuration at cloud vendor	Drop-down list
Domain name	Domain used by PBS Cloud module head node	String
Hostname prefix	Optional prefix for burst node names; default is "node"; chosen during configuration at vendor	String
Network name	Name of VPC network for cloud bursting created at vendor	String
Subnetwork name	Name of VPC network subnet created at vendor	String
OS Image URI	Value for entry called selfLink of REST equivalent of OS image created for cloud bursting at vendor	String

5.3.8.2 Steps to Collect Information

1. Open a browser window and log in to your GCP console.
2. Click ☰ located in the upper left-hand corner of the web page.
3. Click VPC network > VPC networks.
4. Click on the name of the VPC that was created for cloud bursting. VPC network details are displayed.
5. Copy the name of the VPC network.
6. **Save the name of the VPC network**. You will use this later when you create a bursting scenario.
7. Copy the name of the VPC network subnet.
8. **Save the name of the VPC subnet**. You will use this later when you create a bursting scenario.
9. Click ☰ located in the upper left-hand corner of the web page.
10. Click Compute Engine > Images.
11. Select the image created for cloud bursting.
12. Click Equivalent REST
13. Copy the value for the entry called selfLink.
14. **Save this value**. You will use this later when you create a bursting scenario.

15. Click Next.

5.4 Configure Oracle Cloud Platform Cloud Bursting

While you are working your way through the following sections, make sure you save the items in the following table for later when you add the vendor account to PBS Cloud. We will remind you about them:

Table 5-5: Account Parameters for Oracle

Account Parameter	What to Collect During Configuration at Vendor	Format
User OCID	User OCID generated when creating Oracle cloud user account at vendor	String
Tenant OCID	Tenancy OCID generated at vendor	String
Compartment OCID	Root compartment OCID generated at vendor	String
Fingerprint OCID	Fingerprint generated when adding the public SSH key for Oracle user at vendor	String
Private Key	RSA private key generated at vendor	String

5.4.1 Sign Up for an Oracle Cloud Account

Sign up for an Oracle Cloud account and get an associated tenancy. Oracle documentation can be found at Adding Users and Resource Identifiers.

5.4.2 Create Oracle Cloud User Account

Click ☰ located in the upper left-hand corner of the web page.

1. Log in to the Oracle Cloud Infrastructure console.
2. Click Identity > Users.
3. Click Create User.
4. Enter the following to create the user:
   a. For NAME, enter a name for the user.
      The name can be anything meaningful to your organization, e.g., pc_clouduser.
   b. For DESCRIPTION, enter a description of the user.
5. Click Create.
   The user account is created and displayed in the users list.
6. Click Show located under the name of the user. The user account's OCID is displayed.
7. Click Copy to copy the OCID.
8. **Store the user OCID to a file**. You will need this later when you add the account to PBS Cloud.
9. Click Group from the menu located on the left-hand side of the web page.
10. Click Create Group.

11. Enter the following:

    a. For Name, enter Administrators.

    b. For Description, enter a description for the group.

    c. Click Submit.

    The group is created and is displayed in the Groups list.

12. Click on the name of the group.

13. Click Add User to Group.

    a. For User, select the user that was previously created (e.g., pc_clouduser).

    b. Click Add.

14. Click ≡ located in the upper left-hand corner of the web page.

15. Click Identity > Policies.

16. Click Create Policy.

    a. For Name, enter a name for the policy.

    b. For Policy Versioning, enable Keep Policy Current.

    c. For Policy Statements, enter: ALLOW GROUP Administrators to manage all-resources IN TENANCY

    d. Click Create.

5.4.3 Generating an SSH Public Key for the Oracle Cloud User

You will use OpenSSL to create a private and public key in a PEM format for the previously created Oracle Cloud user.

If you're using Windows, you'll need to install Git Bash for Windows and run the commands with that tool.

1. Generate a private key by executing the following command:

   ```
   openssl genrsa -out oracle_private_key.pem 2048
   ```

   We recommend changing the permissions on this file so that only you have read/write access.

2. **Save the private key file**. You will need this later when you add the vendor to PBS Cloud.

3. Generate the public key by executing the following command:

   ```
   openssl rsa -pubout -in oracle_private_key.pem -out oracle_public_key.pem
   ```

4. Log in to the Oracle Cloud Infrastructure console.

5. Click ≡ located in the upper left-hand corner of the web page.

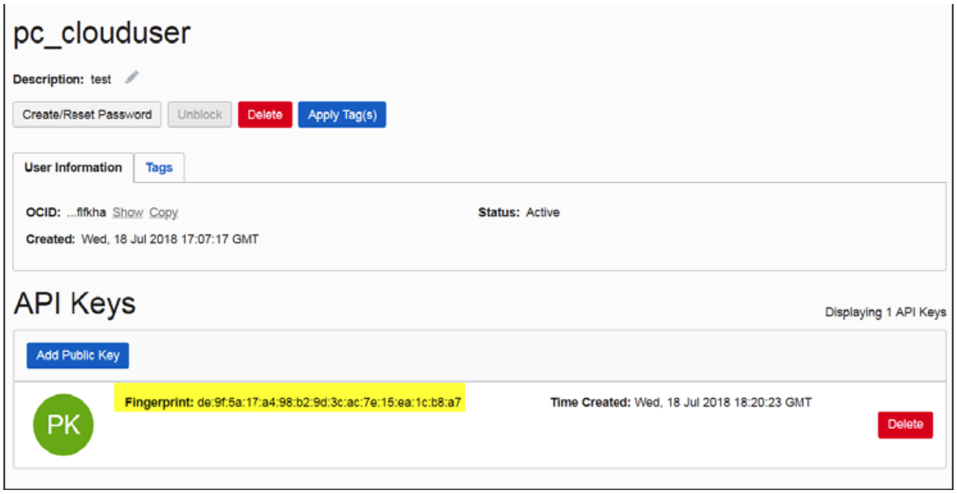
6. Click Identity > Users.

7. Click the name of the previously created user (e.g., pc_clouduser).

8. Click Add Public Key.

9. Copy and paste the contents of the public RSA key file.

10. Click Add to generate a fingerprint:



pc_clouduser

Description: test

User Information

OCID: ...flfkha Show Copy

Status: Active

Created: Wed, 18 Jul 2018 17:07:17 GMT

API Keys

Displaying 1 API Keys

Fingerprint: de:9f:5a:17:a4:98:b2:9d:3c:ac:7e:15:ea:1c:b8:a7

Time Created: Wed, 18 Jul 2018 18:20:23 GMT

Figure 5-14:Public Key Fingerprint

11. Copy the fingerprint.
12. **Store the fingerprint to a file**. You will need the fingerprint later to add the vendor account to PBS Cloud.

5.4.4 Obtain the Root Compartment Identifier

Oracle documentation can be found at Understand Compartments.

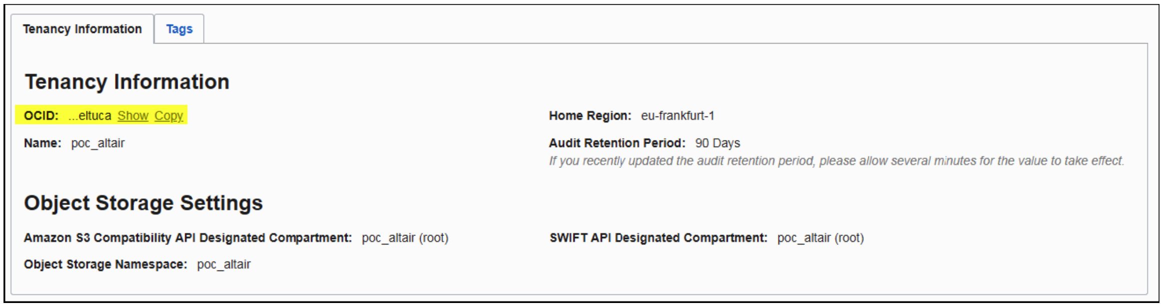
When your tenancy is provisioned, a root compartment is created for you. Compartments can be used to organize and isolate your resources to make it easier to manage and secure access to them. Your root compartment holds all of your cloud resources. You can think of the root compartment like a root folder in a file system. The first time you sign in to the Oracle Cloud Console and select a service, you will see your root compartment. All the resources required for cloud bursting will be contained in this root compartment. You will need the root compartment's resource identifier to add an Oracle cloud account to PBS Cloud.

1. Log in to the Oracle Cloud Infrastructure console.
2. Click ≡ located in the upper left-hand corner of the web page.
3. Click Identity > Compartments.
4. Click Show located under the name of the root compartment. The compartment's OCID is displayed.
5. Click Copy to copy the OCID.
6. **Store the root compartment OCID to a file**. You will need this later when you add the vendor to PBS Cloud.

5.4.5 Obtain the Tenancy Identifier

1. Log in to the Oracle Cloud Infrastructure console.
2. Click ≡ located in the upper left-hand corner of the web page.
3. Click Administration > Tenancy Details.
4. Under Tenancy Information, click Show located to the right of OCID:

The tenancy's OCID is displayed.


Tenancy Information | Tags

Tenancy Information

OCID: ...eltuca Show Copy

Home Region: eu-frankfurt-1

Name: poc_altair

Audit Retention Period: 90 Days

If you recently updated the audit retention period, please allow several minutes for the value to take effect.

Object Storage Settings

Amazon S3 Compatibility API Designated Compartment: poc_altair (root)

SWIFT API Designated Compartment: poc_altair (root)

Object Storage Namespace: poc_altair

Figure 5-15:Tenancy OCID

5. Click Copy to copy the OCID.
6. **Store the tenancy OCID to a file**. You will need this later when you add the vendor cloud account to PBS Cloud

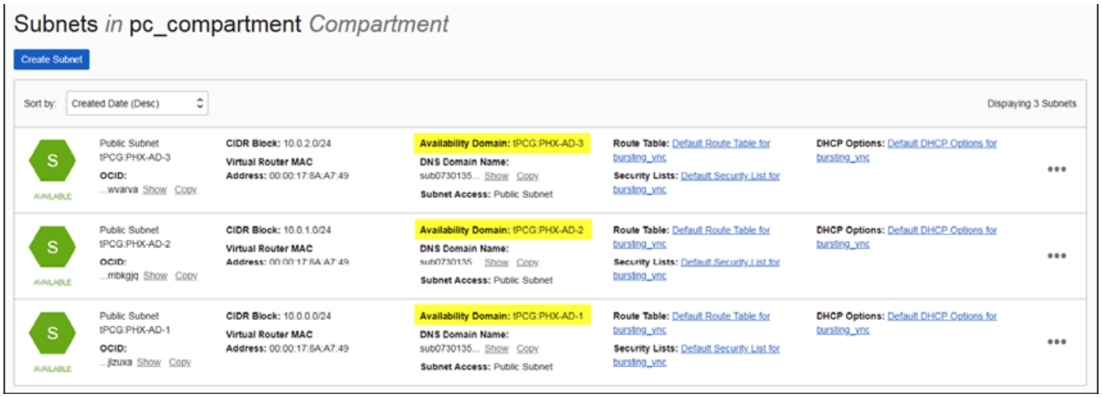
5.4.6 Create a Virtual Cloud Network

Oracle documentation can be found at Overview of Networking and Creating a Virtual Cloud Network.

Make sure that the VCN has a subnet associated with each of the region's availability domains.

1. Log in to the Oracle Cloud Infrastructure console.
2. Click ☰ located in the upper left-hand corner of the web page.
3. Click Networking > Virtual Cloud Networks.
4. Choose a region based on the geographical location of your users. Use the REGION pull-down menu.
5. Click Create Virtual Cloud Network.
6. Enter the following to create a VCN:
   a. For CREATE IN COMPARTMENT, select the root compartment.
   b. For NAME, enter a name for the VCN.
      The name can be anything meaningful to your organization, e.g., bursting_vcn.
   c. Enable CREATE VIRTUAL CLOUD NETWORK PLUS RELATED RESOURCES.
      Choosing this option automatically creates a VCN with a CIDR block 10.0.0.0/16, an internet gateway, a route rule to enable traffic to and from the internet gateway, the default security list, the default set of DHCP options, and one public subnet per availability domain.
7. Click Create Virtual Cloud Network.
   A summary of the VCN, internet gateway, default route table, and subnets is displayed.
8. Click Close.

A list of VCNs is displayed.



Subnets *in* pc_compartment *Compartment*

Sort by: Created Date (Desc)

Dispaying 3 Subnets

S AVAILABLE	Public Subnet tPCG:PHX-AD-3 **OCID:** ...wvarva Show Copy	**CIDR Block:** 10.0.2.0/24 **Virtual Router MAC Address:** 00:00:17:8A:A7:49	**Availability Domain:** tPCG:PHX-AD-3 **DNS Domain Name:** sub0730135... Show Copy **Subnet Access:** Public Subnet	**Route Table:** Default Route Table for bursting_vnc **Security Lists:** Default Security List for bursting_vnc	**DHCP Options:** Default DHCP Options for bursting_vnc	•••
S AVAILABLE	Public Subnet tPCG:PHX-AD-2 **OCID:** ...mbkgjq Show Copy	**CIDR Block:** 10.0.1.0/24 **Virtual Router MAC Address:** 00:00:17:8A:A7:49	**Availability Domain:** tPCG:PHX-AD-2 **DNS Domain Name:** sub0730135 Show Copy **Subnet Access:** Public Subnet	**Route Table:** Default Route Table for bursting_vnc **Security Lists:** Default Security List for bursting_vnc	**DHCP Options:** Default DHCP Options for bursting_vnc	•••
S AVAILABLE	Public Subnet tPCG:PHX-AD-1 **OCID:** ...jlzuxa Show Copy	**CIDR Block:** 10.0.0.0/24 **Virtual Router MAC Address:** 00:00:17:8A:A7:49	**Availability Domain:** tPCG:PHX-AD-1 **DNS Domain Name:** sub0730135... Show Copy **Subnet Access:** Public Subnet	**Route Table:** Default Route Table for bursting_vnc **Security Lists:** Default Security List for bursting_vnc	**DHCP Options:** Default DHCP Options for bursting_vnc	•••

Figure 5-16:Virtual Machine Subnets and Associated Availability Domains

9. Click the name of the VCN.

   The subnets are displayed. A subnet is created for each availability domain (data center) located in the previously selected region.

5.4.7 Check Tenancy Service Limits

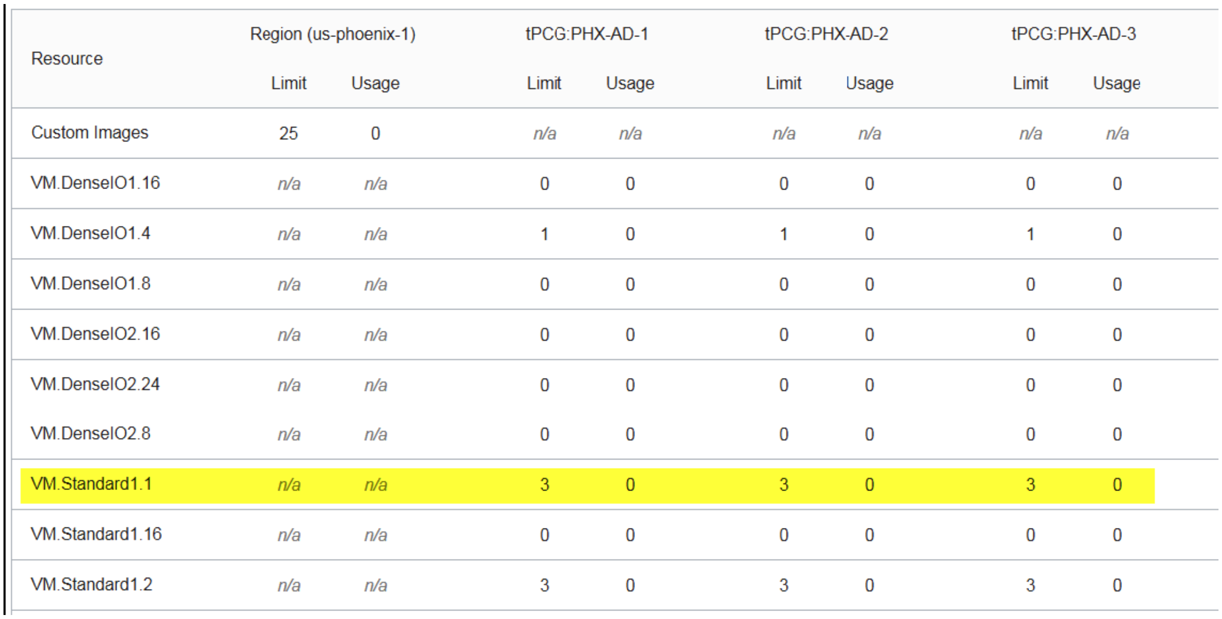
When you sign up for Oracle Cloud Infrastructure, a set of service limits are configured for your tenancy. The service limit is the quota or allowance set on a resource. For example, your tenancy is allowed a maximum number of compute instances (virtual machines) per availability domain. These limits are generally established with your Oracle sales representative when you purchase Oracle Cloud Infrastructure. Oracle documentation can be found at Service Limits and Regions and Availability Domains.

When you reach the service limit for a resource, you receive an error when you try to create a new resource of that type. You cannot create a new resource until you are granted an increase to your service limit or you terminate an existing resource.

View your tenancy's limits to ensure that there are sufficient resources available in a region's availability domains.

1. Log in to the Oracle Cloud Infrastructure console.
2. Choose the region where the previously created VCN is hosted. Use the REGION pull-down menu.
3. Click ☰ located in the upper left-hand corner of the web page.
4. Click Governance > Service Limits.
5. Scroll down to the Service Limits section.
6. Click Compute.

Availability domains (data centers) for the region are displayed. For each resource (VM shape) the number of nodes that can be burst in the corresponding availability domains are displayed. In the below example, three nodes can be burst in each data center in the us-phoenix-1 region for the VM Standard1.1 shape.



Resource	Region (us-phoenix-1)		tPCG:PHX-AD-1		tPCG:PHX-AD-2		tPCG:PHX-AD-3	
	Limit	Usage	Limit	Usage	Limit	Usage	Limit	Usage
Custom Images	25	0	n/a	n/a	n/a	n/a	n/a	n/a
VM.DenseIO1.16	n/a	n/a	0	0	0	0	0	0
VM.DenseIO1.4	n/a	n/a	1	0	1	0	1	0
VM.DenseIO1.8	n/a	n/a	0	0	0	0	0	0
VM.DenseIO2.16	n/a	n/a	0	0	0	0	0	0
VM.DenseIO2.24	n/a	n/a	0	0	0	0	0	0
VM.DenseIO2.8	n/a	n/a	0	0	0	0	0	0
VM.Standard1.1	n/a	n/a	3	0	3	0	3	0
VM.Standard1.16	n/a	n/a	0	0	0	0	0	0
VM.Standard1.2	n/a	n/a	3	0	3	0	3	0

Figure 5-17:Virtual Machine Type Limits

7. Verify that the appropriate service limits are set for your tenancy based on the VM shape chosen for the virtual machine and the region's availability domains.

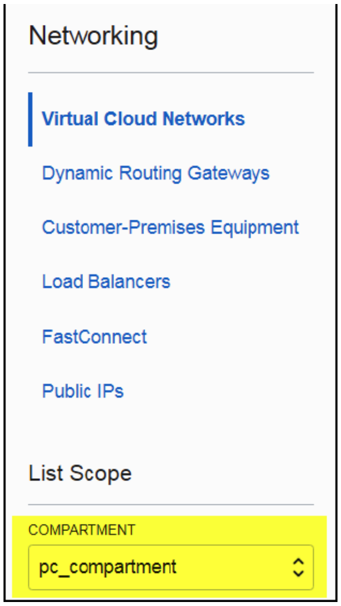
To request an increase a service limits for your tenancy see Requesting a Service Limit Increase.

5.4.8 Creating a Virtual Machine

Virtual machines are hosted in availability domains (data centers) located in a region and are based on predefined VM shapes. Before proceeding, determine the VM shape that your site requires for cloud bursting based on the number of CPUs, memory, disk space, network bandwidth, and virtual network interface cards. While selecting the shape for a VM, consider the nature of the applications that you plan to deploy on the instance, the number of users that you expect to use the applications, and also how you expect the load to scale in the future. Remember to also factor in the CPU and memory resources that are necessary for the operating system.

1. Log in to the Oracle Cloud Infrastructure console.
2. Click ≡ located in the upper left-hand corner of the web page.
3. Click Compute > Instances.
4. Choose the region where the previously created VCN is hosted. Use the REGION pull-down menu.

5. For COMPARTMENT, select the root compartment.



COMPARTMENT
pc_compartment

6. Click Create Instance.

7. Enter the following to create a virtual machine:

   a. For NAME, enter a name for the VM.

      The name can be anything meaningful to your organization, e.g., bursting_vm.

   b. For AVAILABILITY DOMAIN, choose one of the region's availability domains.

      The virtual machine is hosted in the chosen availability domain (data center). Choose the availability domain that best suits your site's cloud bursting requirements based on the machine type of the virtual machine (VM shape) and service limits.

   c. For BOOT VOLUME, enable ORACLE-PROVIDED OS IMAGE.

   d. For IMAGE OPERATING SYSTEM, choose CentOS 7.

   e. For SHAPE TYPE, enable VIRTUAL MACHINE.

   f. For SHAPE, select a VM shape.

      Choose the VM shape that best suits your site's cloud bursting requirements based on number of CPUs, memory, disk space, and network bandwidth.

   g. For IMAGE VERSION, select the latest available one.

   h. For BOOT VOLUME CONFIGURATION, enable CUSTOM BOOT VOLUME SIZE and enter a boot volume size in GBs.

   i. Enable PASTE SSH KEYS and copy the SSH public key (i.e., id_rsa.pub) of a user account that exists on your site's PBS Server and paste it.

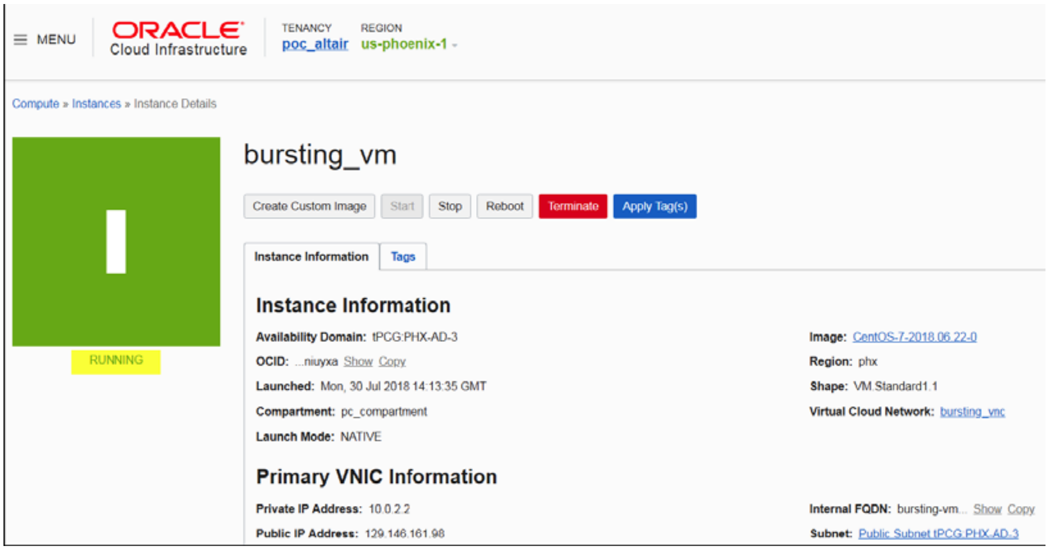
      This user will have sudo rights and will be able to SSH into the VM to install the PBS MoM.

   j. For VIRTUAL CLOUD NETWORK, choose the VCN that was created for cloud bursting.

   k. For SUBNET, choose the subnet associated with the previously chosen availability domain.

8. Click Create Instance.

Creating the virtual machine may take some time. It is done when the state is "Running".



bursting_vm

Instance Information

Availability Domain: tPCG:PHX-AD-3

OCID: ...niuyxa Show Copy

Launched: Mon, 30 Jul 2018 14:13:35 GMT

Compartment: pc_compartment

Launch Mode: NATIVE

Image: CentOS-7-2018.06.22-0

Region: phx

Shape: VM.Standard1.1

Virtual Cloud Network: bursting_vnc

Primary VNIC Information

Private IP Address: 10.0.2.2

Public IP Address: 129.146.161.98

Internal FQDN: bursting-vm... Show Copy

Subnet: Public Subnet tPCG:PHX-AD-3

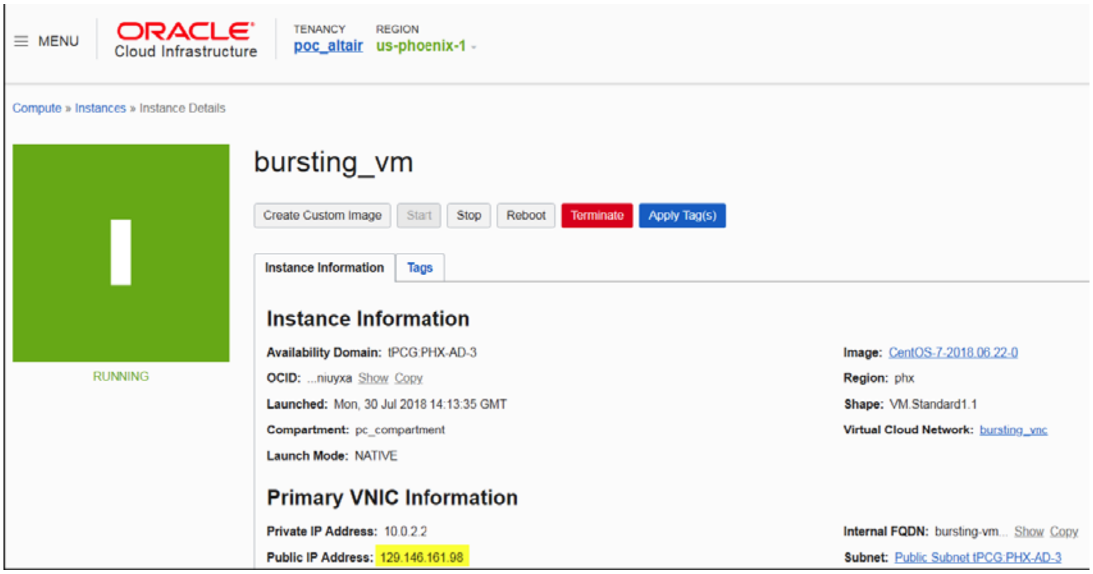
Figure 5-18:Running Virtual Machine

5.4.9 Installing and Configuring a PBS MoM on the VM

1. Log in to your site's PBS Server as the user account provided during the creation of the VM.
2. SSH into the virtual machine using the default user "opc", the private SSH key of the user account provided during the creation of the VM and the External IP assigned to the VM.

```
ssh -i PRIVATE_KEY_PATH opc@PUBLIC_IP_ADDR
```

Where PRIVATE_KEY_PATH is the path to the file that contains the private SSH key of the user account provided during the creation of the VM and PUBLIC_IP_ADDR is the public IP address of the VM.



bursting_vm

Instance Information

Availability Domain: tPCG:PHX-AD-3

OCID: ...niuyxa Show Copy

Launched: Mon, 30 Jul 2018 14:13:35 GMT

Compartment: pc_compartment

Launch Mode: NATIVE

Image: CentOS-7-2018.06.22-0

Region: phx

Shape: VM.Standard1.1

Virtual Cloud Network: bursting_vnc

Primary VNIC Information

Private IP Address: 10.0.2.2

Public IP Address: 129.146.161.98

Internal FQDN: bursting-vm... Show Copy

Subnet: Public Subnet tPCG:PHX-AD-3

Figure 5-19:Bursting Virtual Machine

3. Switch to root:

```
sudo -i
```

4. Using the PBS Professional Installation and Upgrade Guide, install and configure the PBS MOM.
5. Configure the VM to work with your site environment, for example mounting file systems, connecting to the authentication service, installing any applications you need, etc. We recommend that you include the following:
   - Mount /home in the VM
   - Either install applications in the VM or cross-mount them from the head node
   - Either add users to the password file or connect the VM to a service such as NIS
6. If cloud-init is not installed, install it.

5.4.10 Create an OS Image

Oracle documentation can be found at Managing Custom Images.

1. Log in to the Oracle Cloud Infrastructure console.
2. Click ☰ located in the upper left-hand corner of the web page.
3. Click Compute > Instances.

   A list of virtual machines is displayed.
4. Click the name of the virtual machine created for cloud bursting.
5. Click Create Custom Image.
6. Enter the following to create a custom image:
   a. For CREATE IN COMPARTMENT, select the root compartment.
   b. For NAME, enter a name for the image.

      The name can be anything meaningful to your organization, e.g., bursting_image.
7. Click Create Custom Image.

   It may take some time to create the image. Do not proceed until the image is created.
8. You can delete the virtual machine now to avoid storage costs, or keep the virtual machine and update it over time to create updated OS images for bursting. You will incur storage costs, but this is an effective way to keep your OS images up to date when there are changes in packages, patches, or applications.

5.4.11 Collect Information for Oracle Cloud Bursting Scenario

5.4.11.1 Scenario Parameters to Collect at Vendor Interface

Make sure that you capture the following scenario parameters. We will remind you about them:

Table 5-6: Scenario Parameters for Oracle

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Cloud account	Name of your account at cloud vendor	String
Region	Region selected during configuration at cloud vendor	Drop-down list
Domain name	Domain used by PBS Cloud module head node	String

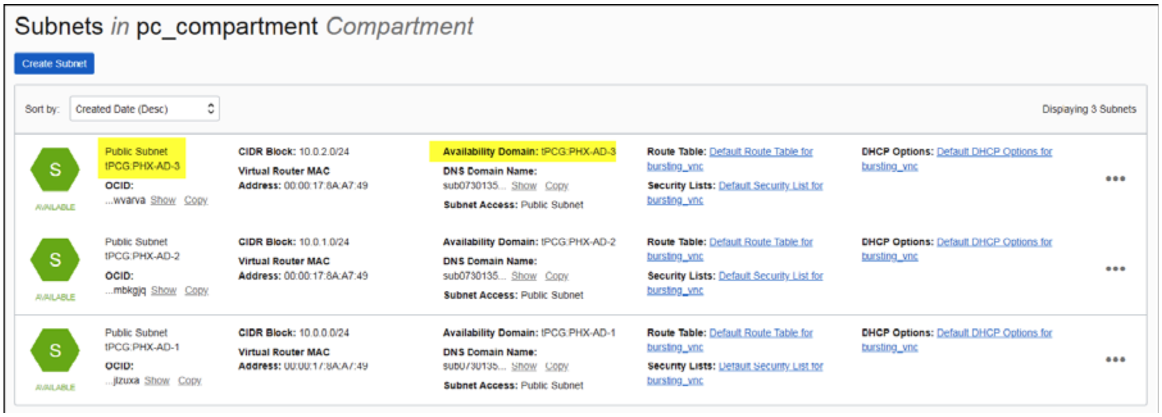
Table 5-6: Scenario Parameters for Oracle

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Hostname prefix	Optional prefix for burst node names; default is "node"; chosen during configuration at vendor	String
Subnet ID	OCID of subnet associated with data center where cloud bursting virtual machine is hosted,	String
OS Image URI	Vendor link to bursting image OCID	String

5.4.11.2 Steps to Collect Information

Open a browser window and log in to the Oracle Cloud Infrastructure console.

9. Click ☰ located in the upper left-hand corner of the web page.
10. Click Networking > Virtual Cloud Networks.
11. Click the name of the VCN created for cloud bursting.
12. Locate the subnet associated with the availability domain where the cloud bursting virtual machine is hosted.



Subnets *in* pc_compartment *Compartment*

Sort by: Created Date (Desc)

Displaying 3 Subnets

S AVAILABLE	Public Subnet tPCG:PHX-AD-3 OCID: ...wvarva Show Copy	CIDR Block: 10.0.2.0/24 Virtual Router MAC Address: 00:00:17:8A:A7:49	Availability Domain: tPCG:PHX-AD-3 DNS Domain Name: sub0730135... Show Copy Subnet Access: Public Subnet	Route Table: Default Route Table for bursting_vnc Security Lists: Default Security List for bursting_vnc	DHCP Options: Default DHCP Options for bursting_vnc
S AVAILABLE	Public Subnet tPCG:PHX-AD-2 OCID: ...mbkgjq Show Copy	CIDR Block: 10.0.1.0/24 Virtual Router MAC Address: 00:00:17:8A:A7:49	Availability Domain: tPCG:PHX-AD-2 DNS Domain Name: sub0730135... Show Copy Subnet Access: Public Subnet	Route Table: Default Route Table for bursting_vnc Security Lists: Default Security List for bursting_vnc	DHCP Options: Default DHCP Options for bursting_vnc
S AVAILABLE	Public Subnet tPCG:PHX-AD-1 OCID: ...jlzuxa Show Copy	CIDR Block: 10.0.0.0/24 Virtual Router MAC Address: 00:00:17:8A:A7:49	Availability Domain: tPCG:PHX-AD-1 DNS Domain Name: sub0730135... Show Copy Subnet Access: Public Subnet	Route Table: Default Route Table for bursting_vnc Security Lists: Default Security List for bursting_vnc	DHCP Options: Default DHCP Options for bursting_vnc

Figure 5-20:Subnet and Associated Availability Domain

13. Click Show located under the name of the subnet.

    The subnet's OCID is displayed.

14. Click Copy to copy the OCID.
15. **Save the subnet OCID to a file**. You will need this later when you create the bursting scenario.
16. Navigate to the Oracle Cloud Infrastructure browser window.
17. Click ☰ located in the upper left-hand corner of the web page.
18. Click Compute > Custom Images. A list of custom images is displayed.
19. Locate the custom image created from the cloud bursting virtual machine.
20. Click the Show link below the name of the image to view the OCID.
21. Click Copy to copy the OCID of the image.
22. **Save the the image OCID to a file**. You will need this later when you create the bursting scenario.

5.5 Configure Orange Cloud Flexible Engine for Cloud Bursting

5.5.1 Purchase an Orange Business Services Account

Purchase an Orange Business Services account. You will use your Orange ID and password to access the Flexible Engine console. For more information visit Orange Cloud.

You should be able to log in to the Orange Cloud Customer space (https://selfcare.cloud.orange-business.com/) with the credentials provided to you with your Orange Cloud account. You will also be provided with a domain name when you sign up for your Orange Cloud account.

5.5.2 Create an Orange Cloud Flexible Engine User Account

PBS Cloud will use an Orange Cloud user account to manage cloud nodes. Make sure that you collect the following information while you are creating the user account (we will also remind you):

Table 5-7: Account Parameters for Orange

Account Parameter	What to Collect During Configuration at Vendor	Format
Auth URL	*https://iam.<orange region>.<console link>*	String
User Domain Name	Orange ID used to log in to Orange account. Same as domain name.	String
Username	Administrator username created at vendor	String
Password	API password generated at vendor	String

1. Go to the Orange Cloud Customer space login page.
2. Enter your Orange Cloud credentials.
3. Click Your services.

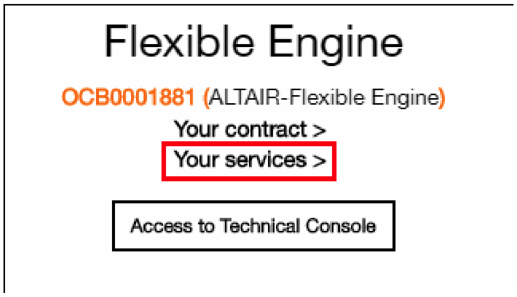

Flexible Engine
OCB0001881 (ALTAIR-Flexible Engine)
Your contract >
Your services >
Access to Technical Console

Figure 5-21:Orange Cloud Customer Space Services

4. In the navigation bar on the top click Users
5. Click Add user.

6. Enter the following user details:
    a. For Civility, choose the form of address
    b. For Last name, enter the user's last name.
    c. For First name, enter the user's first name.
    d. For Login, enter a login name for the user.
    e. For Email, enter the user's email address.
    f. For Phone number, enter the user's phone number.
    g. For Mobile phone, enter the user's mobile phone number.
    h. For Preferred language, choose the language in which the application should be displayed.
    i. Click next.
7. In the Roles section enter these details:
    a. For Billing, choose Visitor.
    b. For Contracts, choose Account Manager.
    c. For Dashboard, choose Visitor.
    d. For Documents, choose Visitor.
    e. For Orders, choose Visitor.
    f. For Services, choose Visitor.
    g. For Subscriptions, choose Visitor.
    h. For Support, choose Visitor.
    i. For Users, choose Manager.
    j. For Flexible Engine Console, choose admin.
    k. Click next.
8. In the Summary section review your choices. Click previous to edit your choices.
9. Click finish.

    The new user account is created and displayed in the list of users. Emails are sent to the email address you specified. The emails will contain:

    - Orange ID (Domain Name). This is the administrator username PBS Cloud will use to manage cloud nodes.
    - Link to set Orange Password. This password is for the administrator to log into Web interface.
    - Link to access Cloud Customer Space.
    - Link to log in to the Flexible Engine Console.
    - Link to define your API password. This password is used by the administrator account for making API requests; PBS Cloud will use this to make API requests.
10. Click the link in the email to set your Orange Password.
11. Click the link in the email to set your API Password.
12. **Store the API Password to a file**. The API Password is used later to create a cloud account in PBS Cloud.

5.5.3 Select a Region

Define a region in the Orange Cloud Flexible Engine console to set up the infrastructure for cloud bursting.

A region is a geographic area where resources used by your ECSes are located. ECSes in the same region can communicate with each other over an intranet, but ECSes in different regions cannot. Before setting up the infrastructure for cloud bursting, it is important to ensure that all the resources are defined in the same region. An Authorization URL is required for adding the Orange Cloud Flexible Engine cloud account in PBS Cloud. This is based on the region selected.

1. Log in to the Orange Cloud Flexible Engine console.
2. In the navigation bar on the top, pull down the region menu and select the region for setting up your infrastructure.
3. For the Authorization URL, (IAM URL), enter the URL in the following format based on the region you chose in the Orange Cloud Flexible Engine console:

   https://iam.<orange region>.<console link>

   e.g. `https://iam.eu-west-0.prod-cloud-ocb.orange-business.com`
4. **Store the region and Auth (IAM) URL in a file**. You will use these to add the provider cloud account to PBS Cloud.

5.5.4 Check Orange Cloud Flexible Engine Account Service Quota

Quotas are used to limit the number of resources available to users. It is important to ensure you are not exceeding your quota while setting up the resources for cloud bursting. If the existing resource quota cannot meet your service requirements, you can submit a work order to increase your quota. Once your application is approved, Orange Cloud Flexible Engine will update your resource quota accordingly and send you a notification.

1. Log in to the Orange Cloud Flexible Engine console.
2. In the navigation bar on the top right hand side, click ◔

   Information about resources usage and availability is displayed.

5.5.5 Create a Virtual Private Cloud

Orange Cloud Flexible Engine documentation for VPC can be found at:Virtual Private Cloud Documentation

1. Log in to the Orange Cloud Flexible Engine console.
2. In the top navigation bar select the region where you wish to deploy your cloud infrastructure.
3. From the Network section click Virtual Private Cloud.
4. Click + Create VPC.
5. In the Basic Information section:

   a. For Region, ensure the VPC is the same region as the other resources.

   a. For Name, enter a name for the VPC.

   a. For CIDR Block, enter CIDR values for the VPC.
6. In the Subnet Settings section:

   a. For the Subnet Settings choose the AZ (Availability Zone) as the same as the region.

   b. For Subnet Name, enter a name to match the VPC Name.

   c. Enter CIDR Block for Subnet.

   d. For Advanced Settings, click Default.
7. Review the Configuration information.

8. Click Create Now.
9. Once the VPC is created, click the Back to VPC List.
10. Click Security Group in the left hand side menu.
11. Click + Create Security Group.
12. For Name, enter a name for the Security Group.
13. For Description, enter a suitable Description.
14. Click OK
15. By default, the Outbound and Inbound traffic over IPv4 is open. You can add firewall rules to this security group if required.

5.5.6 Creating a Virtual Machine

1. Log in to the Orange Cloud Flexible Engine console.
2. In the Computing section and click Elastic Cloud Server.
3. From the menu on the left hand side click Key Pair.
4. Click + Create Key Pair.
5. Enter a Name for the Key Pair.
6. Click OK.
7. Save the Key Pair (.pem) file to your local disk in a secure location. The information in this .pem file is used later to SSH into the VM.
8. Click OK to confirm that you have downloaded the Key Pair file.
9. From the menu on the left hand side, click Elastic Cloud Server.
10. Click + Create ECS.
11. For Region, click the region you selected for setting up the infrastructure.
12. For AZ (Availability Zone), select the AZ related to the region.
13. In the Specifications section:
    a. For ECS type, click one of the flavor names. Orange Cloud Flexible Engine provides a set of predefined ECS types for specific requirements. Click a flavor name to get the list of available configurations.
    b. Review the specifications you have selected.
14. In the Image section:
    a. Click Public image.
    b. From the drop down menu select CentOS
    c. From the version drop down menu select Select OBS_U_CentOS_7.2(40GB)
15. In the Disk section select the defaults.

16. In the VPC section:
    a. For VPC, select the VPC you created from the drop down menu.
    a. For NIC, choose the default primary NIC.
    b. For Security Group, select the Security Group you created for the VPC.
    c. For EIP, click Automatically assign
    d. For Bandwidth, specify it as 5 Mbit/s.
17. For Login Mode, select the Key Pair you generated earlier from the drop down menu.
18. For Auto Recovery, click Enable.
19. For Advanced Settings, click Do not configure.
20. For ECS Name, enter a name.
21. For Quantity, specify 1.
22. Review the Current Configuration.
23. Click Create Now.
24. Review the Specifications.
25. Click Submit.

    The ECS (Virtual Machine) is created and displayed in the list of ECS.

5.5.7 Installing and Configuring a PBS MoM on the VM

1. Log in to your site's PBS Server.
2. Log in to the Flexible Engine console.
3. In the Computing section, click Elastic Cloud Server.
4. In the search box above the upper right corner of the ECS list, enter the ECS name, IP address, or ID, and click the search icon.
5. Click the name of the target ECS.
6. The page providing details about the ECS is displayed.
7. Copy the Public IP address (External IP) of the ECS.
8. SSH into the VM using the default user "cloud", the .pem file you generated when creating the VM, and the External IP assigned to the VM. For more information about logging into the Linux ECS, refer to the Elastic Cloud Server User Guide.

```
ssh -i /<path to .pem file>/<name of .pem file>.pem cloud@<public IP address of VM>
```

9. Switch to root:

```
sudo -i
```

10. Using the PBS Professional Installation and Upgrade Guide, install and configure the PBS MOM.
11. Save the file.

12. Configure the VM to work with your site environment, for example mounting file systems, connecting to the authentication service, installing any applications you need, etc. We recommend that you include the following:
    - Mount `/home` in the VM
    - Either install applications in the VM or cross-mount them from the head node
    - Either add users to the password file or connect the VM to a service such as NIS
13. If cloud-init is not installed, install it.

5.5.8 Create an OS Image

Orange Cloud Flexible Engine documentation can be found at Creating a Linux Private Image Using an ECS.

5.5.8.1 Prerequisites

Before creating the Image from the ECS, you must have.

- A Linux ECS in the Stopped state.
- Configured DHCP for the NICs of the ECS
- Configured Network attributes of the ECS
- Detached Data Disks from the ECS

5.5.8.2 Steps to Create OS Image

1. Log in to the Flexible Engine console.
2. In the Computing section, click Image Management Service.
3. On the Image Management Service page, click + Create Private Image.
4. In the Image Type and Source section, .
    a. For Type, click System disk image
    b. For Source, click ECS
    c. Select the target ECS from the ECS list.
5. Set the required information, such as Name and Description.
6. Click Create Now.
7. Confirm the parameters and click Submit.
8. Switch back to the Image Management Service page to view the image status.

    The time required for creating an image varies depending on the image file size. Generally, it takes about 20 minutes to create an image. The image is successfully created when its image status changes to Normal.

    Do not perform any operation on the selected ECS or its associated resources during image creation.
9. You can delete the virtual machine now to avoid storage costs, or keep the virtual machine and update it over time to create updated OS images for bursting. You will incur storage costs, but this is an effective way to keep your OS images up to date when there are changes in packages, patches, or applications.

5.5.9 Create Orange Cloud Cloud Bursting Scenario

5.5.9.1 Scenario Parameters to Collect at Vendor Interface

Make sure that you capture the following scenario parameters. We will remind you about them:

Table 5-8: Scenario Parameters for Orange

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Cloud account	Name of your administrator account at cloud vendor	String
Region	Region selected during configuration at cloud vendor	Drop-down list
Domain name	Domain used by PBS Cloud head node	String
Hostname prefix	Optional prefix for burst node names; default is "node"; chosen during configuration at vendor	String
Subnet ID	ID of subnet for VPC created at cloud vendor	String
Security Group ID	ID of security group created at cloud vendor	String
OS Image URI	ID of OS image created at cloud vendor	String

5.5.9.2 Steps to Collect Information

Open a browser window and log in to the Orange Cloud Flexible Engine console.

10. Click Service List in the menu bar.

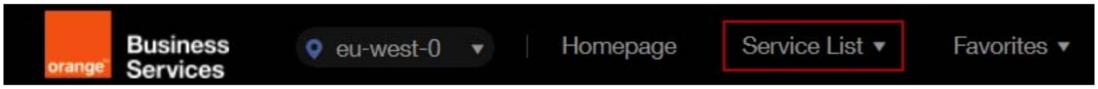


Figure 5-22:Orange Cloud Flexible Engine Console

11. Under Network, click Virtual Private Cloud.
12. Click Virtual Private Cloud from the menu located on the left-hand side of the web page.
13. Click the name of the VPC you created for cloud bursting.
14. Click the name of the Subnet for the VPC.
15. Copy the Subnet ID.
16. **Save the subnet ID to a file**. You will need this later when you create a bursting scenario.
17. Click Security Group from the left hand side menu.
18. Click the name of the Security Group you created for the VPC.
19. Copy the ID of the Security Group.
20. **Save the security group to a file**. You will need this later when you create a bursting scenario.
21. Click Service List in the menu bar.
22. Under Computing, click Image Management Service.
23. Click the Private Images tab.
24. Click the name of the VM image you created for cloud bursting.

25. Copy the ID of the image.
26. **Save the OS Image ID to a file**. You will need this later when you create a bursting scenario.

5.6 Configure HUAWEI Cloud for Cloud Bursting

5.6.1 Create and Activate HUAWEI Account

Create and activate a HUAWEI Cloud account.

5.6.2 Get the HUAWEI Cloud Administrator Credentials

PBS Cloud will use the HUAWEI Cloud administrative user account to manage cloud nodes. While you are getting the credentials for the administrative user account, make sure you capture the following information:

Table 5-9: Account Parameters for Huawei

Account Parameter	What to Collect During Configuration at Vendor	Format
Auth URL	*https://iam.ap-southeast-1.myhwclouds.com*	String
User Domain Name	Domain Name provided when your subscription to HUAWEI Cloud was created If you do not know your Domain name, contact HUAWEI Cloud support.	String
Username	Administrator username created at vendor	String
Password	Administrator password created at vendor	String

5.6.2.1 Choose Administrative User

You can create a new user and give the user administrative privileges, or you can use the administrative user account that is automatically created when you subscribe to HUAWEI Cloud.

The automatically-created user is an administrative user account with permissions for all system operations.

If you create a new user, give the user administrative privileges by setting its User Group to "admin".

5.6.2.2 Get Credentials

1. Log in to the HUAWEI Cloud Console.
2. Click Service List.

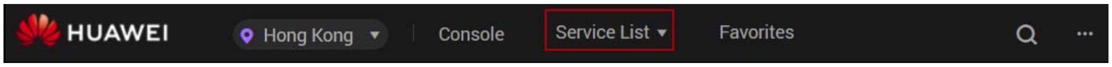
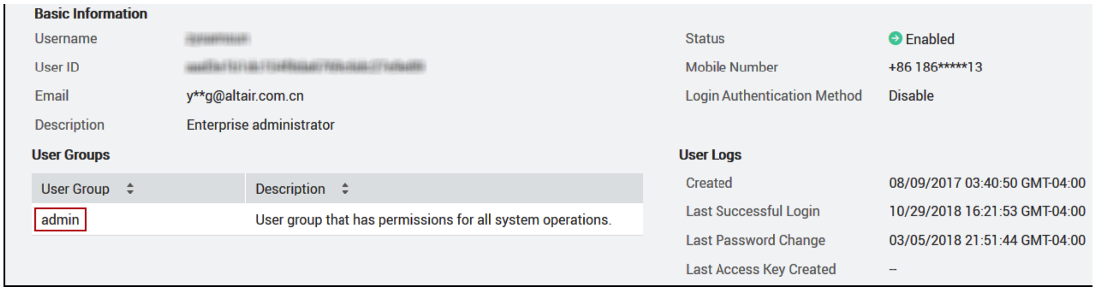


Figure 5-23:HUAWEI Cloud Console

3. Under Management & Deployment, click Identity and Access Management.
4. Click Users from the menu located on the left-hand side of the web page. A list of users is displayed.
5. Click the down-arrow located next to a username to display the user's details.

The user account listed as an "admin" is the account to use to create the cloud account in PBS Cloud.



Basic Information

Username	[redacted]	Status	Enabled
User ID	[redacted]	Mobile Number	+86 186*****13
Email	y**g@altair.com.cn	Login Authentication Method	Disable
Description	Enterprise administrator		

User Groups

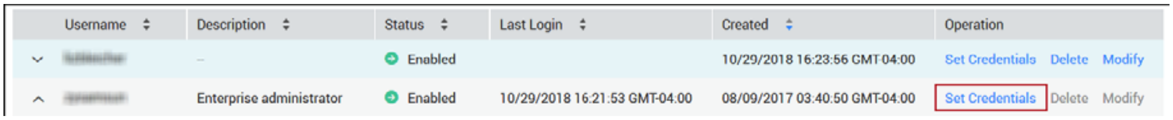
User Group	Description
admin	User group that has permissions for all system operations.

User Logs

Created	08/09/2017 03:40:50 GMT-04:00
Last Successful Login	10/29/2018 16:21:53 GMT-04:00
Last Password Change	03/05/2018 21:51:44 GMT-04:00
Last Access Key Created	--

Figure 5-24:User Details

6. If you do not know the password for the administrative user account, click Set Credentials.



	Username	Description	Status	Last Login	Created	Operation
⌄	[redacted]	--	Enabled		10/29/2018 16:23:56 GMT-04:00	Set Credentials Delete Modify
⌃	[redacted]	Enterprise administrator	Enabled	10/29/2018 16:21:53 GMT-04:00	08/09/2017 03:40:50 GMT-04:00	Set Credentials Delete Modify

Figure 5-25:Set Password

a. Enable Set manually.

b. For Password, enter a password for the user account.

c. For Confirm Password, enter the password a second time.

d. **Save the password**. You will need this later when you add the administrative user account to PBS Cloud.

e. Click OK.

You may have to confirm the password change either by email or by a SMS text.

5.6.3 Check HUAWEI Cloud Account Service Quotas

View your OTC account resource usage and limits.

Quotas are used to limit the number of resources available to users. It is important to ensure you are not exceeding your quota while setting up the resources for cloud bursting. If the existing resource quota cannot meet your service requirements, you can submit a work order to increase your quota. Once your application is approved, HUAWEI Cloud will update your resource quota accordingly and send you a notification.

1. Log in to the HUAWEI Cloud console.
2. Click Resources > My Quota.

   Information about resources usage and availability is displayed.

5.6.4 Create a Virtual Private Cloud

HUAWEI Cloud documentation for creating a VPC can be found at: Creating a VPC and Regions and AZs.

1. Log in to the HUAWEI Cloud Console.
2. Click Service List in the menu bar.
3. Under Network, click Virtual Private Cloud.
4. Click + Create VPC.

5. In the Basic Information section:
    a. For Region, select a region.

       Regions are geographic areas isolated from each other. Resources are region-specific and cannot be used across regions through internal network connections. For low network latency and quick resource access, select the nearest region.
    b. For Name, enter a name for the VPC.
    c. For CIDR Block, enter an address range for the network using CIDR notation.
6. In the Subnet Settings section:
    a. For the Subnet Settings choose the AZ (Availability Zone).

       An Availability Zone is a physical location where resources use independent power supplies and networks. AZs are physically isolated and AZs in the same VPC are interconnected through an internal network.
    b. For Subnet Name, enter a name for the subnet.
    c. For CIDR, enter an address range for the subnet using CIDR notation.
    d. For Advanced Settings, click Default.
7. Click Create Now.
8. Click Back to VPC List.
9. Click Security Group in the left hand side menu.
10. Click + Create Security Group.
11. For Name, enter a name for the security group.
12. For Description, enter a suitable description.
13. Click OK.
14. By default, the Outbound and Inbound traffic over IPv4 is open. You can add firewall rules to this security group if required.

5.6.5 Creating a Virtual Machine

HUAWEI Cloud documentation for creating an ECS (virtual machine) can be found at Purchase an ECS.

1. Log in to the HUAWEI Cloud Console.
2. Click Service List in the menu bar.
3. Under Computing, click Elastic Cloud Server.
4. Click Key Pair from the menu located on the left-hand side of the web page.
5. Click + Create Key Pair.
6. For Name, enter a name for the key pair.
7. Click OK.
8. Save the key pair (.pem) file to your local disk in a secure location. The information in this .pem file is used later to SSH into the VM.
9. Click OK to confirm that you have downloaded the key pair file.
10. Click Elastic Cloud Server from the menu located on the left-hand side of the web page.
11. Click Buy ECS.

12. For Billing Mode, click Pay-per-use.
13. For Region, select the same region that was chosen for the previously created VPC.
14. For AZ (Availability Zone), select the same AZ that was chosen for the previously created VPC.
15. In the Type section:
    a. Choose an ECS type category:
       - General computing
       - General computing-plus
       - Memory-optimized
       - Large-memory
       - High-performance computing
       - Disk-intensive

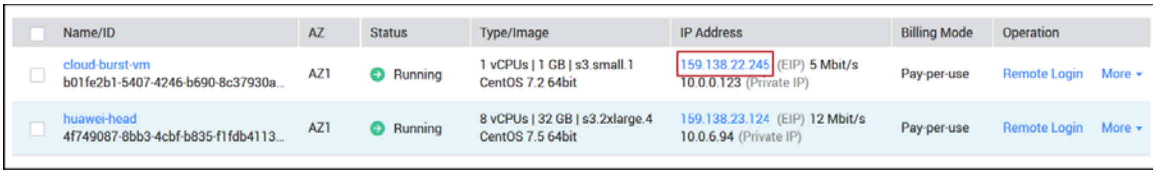
    b. For ECS type, click one of the flavor based on the needs of your site.
16. In the Image section:
    a. Click Public image.
    b. For Select an OS, select CentOS.
    c. For Select an OS version, select CentOS 7.2 64bit(40GB).
17. In the Disk section, select your system disk requirements.
18. In the VPC section:
    a. For VPC, select the VPC you created for cloud bursting. The NIC information is automatically populated.
    b. For Security Group, select the security group you created for cloud bursting.
    c. For EIP, click Automatically assign
    d. For Bandwidth, specify it as 5 Mbit/s.
19. For Login Mode, select Key Pair.
20. For Key Pair, select the key pair file you generated earlier.
21. For Advanced Settings, click Not required.
22. For ECS Name, enter a name for the virtual machine.
23. For Quantity, specify 1.
24. Click Next.
25. Review the specifications.
26. Enable the I have read and agree to the Huawei Image Disclaimer checkbox.
27. Click Submit Application.
28. Click Back to ECS List.

    It may take some time to create the virtual machine. Once the ECS is created it is displayed in the ECS list.

5.6.6 Installing and Configuring a PBS MoM on the VM

1. Log in to the HUAWEI Cloud console.
2. Click Service List in the menu bar.

3. Under Computing, click Elastic Cloud Server.
4. Copy the Public IP address (External IP) of the ECS.



□	Name/ID	AZ	Status	Type/Image	IP Address	Billing Mode	Operation
□	cloud-burst-vm b01fe2b1-5407-4246-b690-8c37930a...	AZ1	Running	1 vCPUs | 1 GB | s3.small.1 CentOS 7.2 64bit	159.138.22.245 (EIP) 5 Mbit/s 10.0.0.123 (Private IP)	Pay-per-use	Remote Login More ▾
□	huawei-head 4f749087-8bb3-4cbf-b835-f1fdb4113...	AZ1	Running	8 vCPUs | 32 GB | s3.2xlarge.4 CentOS 7.5 64bit	159.138.23.124 (EIP) 12 Mbit/s 10.0.6.94 (Private IP)	Pay-per-use	Remote Login More ▾

Figure 5-26:VM IP Address

5. Log in to your site's PBS Server.
6. SSH into the VM using the default user "root", the .pem file you generated when creating the VM and the External IP assigned to the VM.

   ```
   sh -i /path/my-key-pair.pem root@IPV4PublicIP
   ```

   where /path/my-key-pair.pem is the path to the .pem file downloaded while creating the virtual machine and IPV4PublicIP is the public IP address of the virtual machine.
7. Using the PBS Professional Installation and Upgrade Guide, install and configure the PBS MOM.
8. Configure the VM to work with your site environment, for example mounting file systems, connecting to the authentication service, installing any applications you need, etc. We recommend that you include the following:
   - Mount /home in the VM
   - Either install applications in the VM or cross-mount them from the head node
   - Either add users to the password file or connect the VM to a service such as NIS
9. If cloud-init is not installed, install it.

5.6.7 Create an OS Image

HUAWEI Cloud documentation can be found at Creating a Linux Private Image.

1. Log in to the HUAWEI Cloud console.
2. Click Service List in the menu bar.
3. Under Computing, click Image Management Service.
4. Click + Create Image.
5. For Region, select the same region that was chosen for the previously created VPC and ECS.
6. In the Image Type and Source section, .
   a. For Type, click System disk image.
   b. For Source, click ECS.
   c. Select the virtual machine created for cloud bursting.
   d. For Name, enter a name for the virtual machine.
7. Click Next.
8. Review the specifications.
9. Enable the I have read and agree to the Statement of Commitment to Image Creation and Huawei Image Disclaimer checkbox.
10. Click Submit.

11. Click Back to Image List.

    The time required for creating an image varies depending on the image file size. Generally, it takes about 20 minutes to create an image. The image is successfully created when its image status changes to Normal.

    Do not perform any operation on the selected ECS or its associated resources during image creation.

12. You can delete the virtual machine now to avoid storage costs, or keep the virtual machine and update it over time to create updated OS images for bursting. You will incur storage costs, but this is an effective way to keep your OS images up to date when there are changes in packages, patches, or applications.

5.6.8 Collect HUAWEI Cloud Bursting Scenario Information

5.6.8.1 Scenario Parameters to Collect at Vendor Interface

Make sure that you capture the following scenario parameters. We will remind you about them:

Table 5-10: Scenario Parameters for Huawei

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Cloud account	Name of your account at cloud vendor	String
Region	Region selected during configuration at cloud vendor	Drop-down list
Domain name	Domain used by PBS Cloud module head node	String
Hostname prefix	Optional prefix for burst node names; default is "node"; chosen during configuration at vendor	String
Subnet ID	ID of subnet for VPC created at cloud vendor	String
Security Group ID	ID of security group created at cloud vendor	String
OS Image URI	ID of OS image created at cloud vendor	String

5.6.8.2 Steps to Collect Information

Open a browser window and log in to the HUAWEI Cloud console.

13. Click Service List in the menu bar.
14. Under Network, click Virtual Private Cloud.
15. Click Virtual Private Cloud from the menu located on the left-hand side of the web page.
16. Click the name of the VPC you created for cloud bursting.
17. Click the name of the VPC's subnet.
18. Copy the Subnet ID.
19. **Save the subnet ID to a file**. You will need this later when you create a bursting scenario.
20. Click Security Group from the menu located on the left-hand side of the web page.
21. Click the name of the security group you created for the VPC.
22. Copy the ID of the security group.
23. **Save the security group to a file**. You will need this later when you create a bursting scenario.

24. Click Service List in the menu bar.
25. Under Computing, click Image Management Service.
26. Click the Private Images tab.
27. Click the name of the VM image you created for cloud bursting.
28. Copy the ID of the image.
29. **Save the OS Image ID to a file**. You will need this later when you create a bursting scenario.

5.7 Configure Open Telekom Cloud for Cloud Bursting

5.7.1 Create and Activate OTC Cloud Account

Create and activate an OTC Cloud account.

5.7.2 Obtain the OTC Administrator Credentials

PBS Cloud will use the OTC administrative user account to manage cloud nodes. While you are getting the credentials for the administrative user account, make sure you capture the following information:

Table 5-11: Account Parameters for Deutsche Telekom OTC

Account Parameter	What to Collect During Configuration at Vendor	Format
Auth URL	*https://iam.eu-de.otc.t-systems.com/v3*	String
User Domain Name	Deutsche Telekom: OTC domain name used to log in to OTC console at vendor	String
Username	Administrator username created at vendor	String
Password	Administrator password created at vendor	String

5.7.2.1 Choose Administrative User

You can create a new user and give the user administrative privileges, or you can use the administrative user account that is automatically created when you subscribe to OTC.

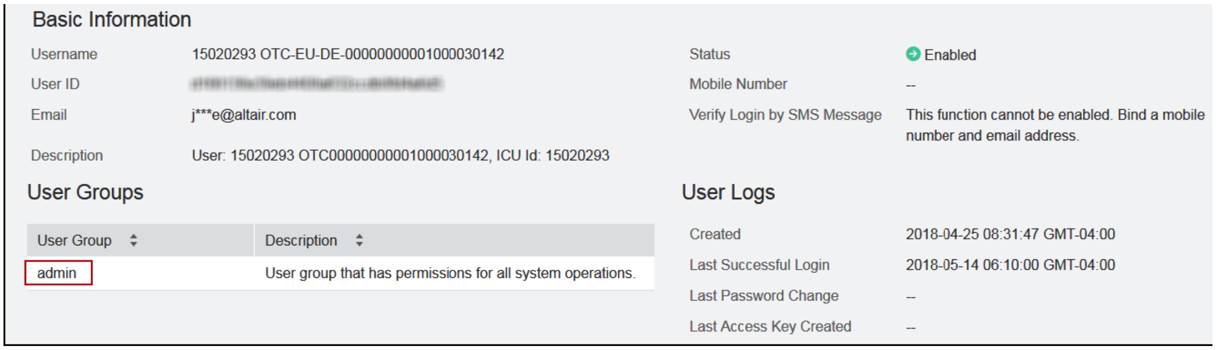
The automatically-created user is an administrative user account with permissions for all system operations.

If you create a new user, give the user administrative privileges by setting its User Group to "admin".

5.7.2.2 Get Credentials

1. Log in to the OTC Console.
2. Click Service List in the menu bar.
3. Under Management & Deployment, click Identity and Access Management.
4. Click Users from the menu located on the left-hand side of the web page. A list of users is displayed.
5. Click the down-arrow located next to a username to display the user's details.

The user account listed as an "admin" is the account to use to create the cloud account in PBS Cloud.



Basic Information

Username	15020293 OTC-EU-DE-00000000001000030142	Status	Enabled
User ID	[redacted]	Mobile Number	--
Email	j***e@altair.com	Verify Login by SMS Message	This function cannot be enabled. Bind a mobile number and email address.
Description	User: 15020293 OTC00000000001000030142, ICU Id: 15020293		

User Groups

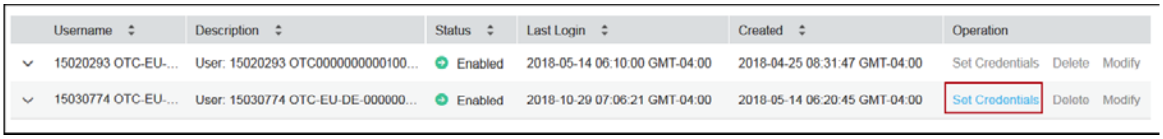
User Group	Description
admin	User group that has permissions for all system operations.

User Logs

Created	2018-04-25 08:31:47 GMT-04:00
Last Successful Login	2018-05-14 06:10:00 GMT-04:00
Last Password Change	--
Last Access Key Created	--

Figure 5-27:User Details

6. If you do not know the password for the admin user account, click Set Credentials



Username	Description	Status	Last Login	Created	Operation
15020293 OTC-EU-...	User: 15020293 OTC0000000000100...	Enabled	2018-05-14 06:10:00 GMT-04:00	2018-04-25 08:31:47 GMT-04:00	Set Credentials Delete Modify
15030774 OTC-EU-...	User: 15030774 OTC-EU-DE-000000...	Enabled	2018-10-29 07:06:21 GMT-04:00	2018-05-14 06:20:45 GMT-04:00	Set Credentials Delete Modify

Figure 5-28:Set Password

7. Enable Set manually.
8. For Password, enter a password for the user account.
9. For Confirm Password, enter the password a second time.
10. Click OK.

    You may have to confirm the password change either by email or by a SMS text.

5.7.3 Check OTC Account Service Quotas

Quotas are used to limit the number of resources available to users. It is important to ensure you are not exceeding your quota while setting up the resources for cloud bursting. If the existing resource quota cannot meet your service requirements, you can submit a work order to increase your quota. Once your application is approved, OTC will update your resource quota accordingly and send you a notification.

1. Log in to the OTC Console.
2. Click the three vertical bars in the menu bar:

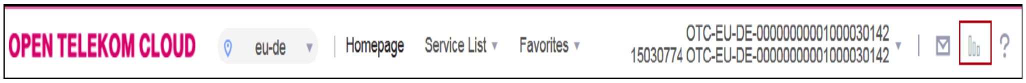
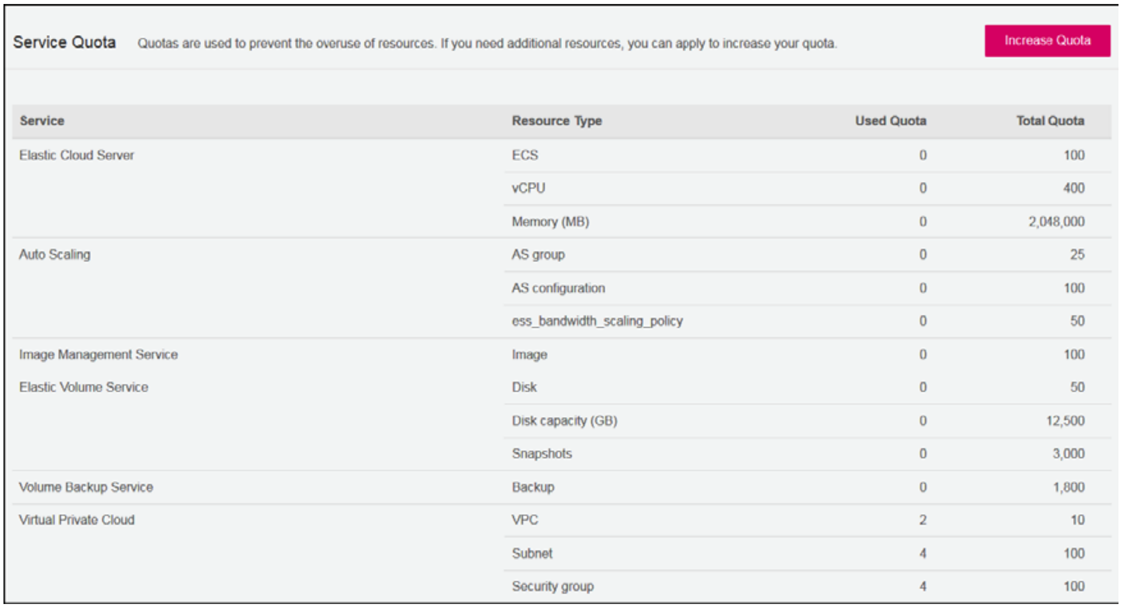

OPEN TELEKOM CLOUD
eu-de
Homepage
Service List
Favorites
OTC-EU-DE-00000000001000030142
15030774 OTC-EU-DE-00000000001000030142
?

Figure 5-29:Viewing Quotas

You can see resource usage and availability:



Service Quota Quotas are used to prevent the overuse of resources. If you need additional resources, you can apply to increase your quota.

Service	Resource Type	Used Quota	Total Quota
Elastic Cloud Server	ECS	0	100
	vCPU	0	400
	Memory (MB)	0	2,048,000
Auto Scaling	AS group	0	25
	AS configuration	0	100
	ess_bandwidth_scaling_policy	0	50
Image Management Service	Image	0	100
Elastic Volume Service	Disk	0	50
	Disk capacity (GB)	0	12,500
	Snapshots	0	3,000
Volume Backup Service	Backup	0	1,800
Virtual Private Cloud	VPC	2	10
	Subnet	4	100
	Security group	4	100

Figure 5-30:Resource quotas and availability

5.7.4 Create a Virtual Private Cloud

OTC documentation for creating a VPC can be found at: Creating a VPC and Regions.

1. Log in to the OTC Console.
2. Click Service List.
3. Under Network, click Virtual Private Cloud.
4. Click + Create VPC.
5. In the Basic Information section:
    a. For Region, select a region.

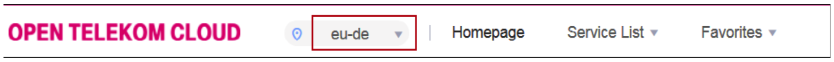


Figure 5-31:Regions

A region is a geographical areas and can comprise one or more availability zones (AZs). A region is completely isolated from other regions. Only AZs in the same region can communicate with one another through an internal network.

    b. For Name, enter a name for the VPC.
    c. For CIDR Block, enter an address range for the network using CIDR notation.
6. In the Subnet Settings section:
    a. For Subnet Name, enter a name for the subnet.
    b. For CIDR, enter an address range for the subnet using CIDR notation.
    c. For Advanced Settings, click Default.
7. Click Create Now.
8. Click Back to VPC List.

9. Click Security Group in the left hand side menu.
10. Click + Create Security Group.
11. For Name, enter a name for the security group.
12. For Description, enter a suitable description.
13. Click OK.

    The security group rules are displayed.
14. Click the Inbound tab.
15. Click Add Rule.
    a. For Protocol/Application, select TCP.
    b. For Port, enter 22.
    c. For Source, select IP Address. and enter the PBS Cloud firewall IP address.
    d. Click OK.

5.7.5 Creating a Virtual Machine

OTC documentation for creating an ECS (virtual machine) can be found at Create an ECS.

1. Log in to the OTC Console.
1. Click Service List in the menu bar.
2. Under Computing, click Elastic Cloud Server.
3. Click Key Pair from the menu located on the left-hand side of the web page.
4. Click + Create Key Pair.
5. For Name, enter a name for the key pair.
6. Click OK.
7. Save the key pair (.pem) file to your local disk in a secure location. The information in this .pem file is used later to SSH into the VM.
8. Click OK to confirm that you have downloaded the key pair file.
9. Click Elastic Cloud Server from the menu located on the left-hand side of the web page.
10. Click Create ECS.
11. For Region, select the same region that was chosen for the previously created VPC.
12. For AZ (Availability Zone), select an availability zone.

13. In the Specifications section:
    a. Choose an ECS type category:
        - General-purpose
        - Dedicated general-purpose
        - Memory-optimized
        - Large-memory
        - High-performance
        - Disk-intensive
        - GPU-accelerated

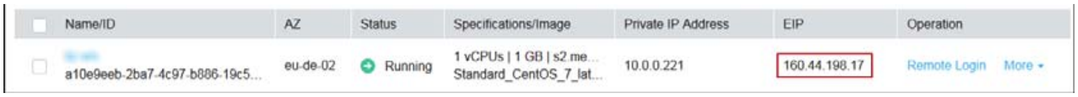
    b. For ECS type, choose one of the flavors based on the needs of your site.
14. In the Image section:
    a. Click Public image.
    b. Select a supported OS as the operating system.
    c. Select a supported version of the OS.
15. In the Disk section, select your system disk requirements.
16. In the VPC section:
    a. For VPC, select the VPC you created for cloud bursting. The NIC information is automatically populated.
    b. For Security Group, select the security group you created for cloud bursting.
    c. For EIP, click Automatically assign
    d. For Bandwidth, specify it as 5 Mbit/s.
17. For Log in Mode, select Key Pair.
18. For Key Pair, select the key pair file you generated earlier.
19. For Advanced Settings, click Do not configure.
20. For ECS Name, enter a name for the virtual machine.
21. For Quantity, specify 1.
22. Click Create Now.
23. Review the specifications.
24. Click Submit.
25. Click Back to ECS List.

    It may take some time to create the virtual machine. Once the ECS is created it is displayed in the ECS list.

5.7.6 Installing and Configuring a PBS MoM on the VM

1. Log in to the OTC Console.
2. Click Service List in the menu bar.
3. Under Computing, click Elastic Cloud Server.

4. Copy the Public IP address (External IP) of the ECS.



□	Name/ID	AZ	Status	Specifications/Image	Private IP Address	EIP	Operation
□	[redacted] a10e9eeb-2ba7-4c97-b886-19c5...	eu-de-02	Running	1 vCPUs | 1 GB | s2.me... Standard_CentOS_7_lat...	10.0.0.221	160.44.198.17	Remote Login More ▾

Figure 5-32:VM IP Address

5. Log in to your site's PBS Server.

6. SSH into the VM using the default user "root", the .pem file you generated when creating the VM and the External IP assigned to the VM.

```
ssh -i /path/my-key-pair.pem root@IPV4PublicIP
```

   where /path/my-key-pair.pem is the path to the .pem file downloaded while creating the virtual machine and IPV4PublicIP is the public IP address of the virtual machine.

7. Using the PBS Professional Installation and Upgrade Guide, install and configure the PBS MOM.

8. Configure the VM to work with your site environment, for example mounting file systems, connecting to the authentication service, installing any applications you need, etc. We recommend that you include the following:

   - Mount /home in the VM
   - Either install applications in the VM or cross-mount them from the head node
   - Either add users to the password file or connect the VM to a service such as NIS

9. If cloud-init is not installed, install it.

5.7.7 Create an OS Image

OTC documentation can be found at Creating a Linux Private Image.

1. Log in to the OTC Console.
2. Click Service List in the menu bar.
3. Under Computing, click Image Management Service.
4. Click + Create System Disk Image.
   a. For Region, select the same region that was chosen for the previously created VPC and ECS.
   b. For Source, click Server.
   c. For Server Type, click ECS.
   d. For ECS, select the virtual machine created for cloud bursting.
   e. If the virtual machine is not stopped, stop it.
   f. Click OK when prompted to verify that certain operations have been performed on the ECS. You do not need to configure or optimize the ECS.
   g. For Name, enter a name for the virtual machine.
5. Click Create Now.
6. Review the specifications.
7. Click Submit.
8. Click Back to Image List.

   The time required for creating an image varies depending on the image file size. Generally, it takes about 20 minutes to create an image. The image is successfully created when its image status changes to Normal.

Do not perform any operation on the selected ECS or its associated resources during image creation.

9. You can delete the virtual machine now to avoid storage costs, or keep the virtual machine and update it over time to create updated OS images for bursting. You will incur storage costs, but this is an effective way to keep your OS images up to date when there are changes in packages, patches, or applications.

5.7.8 Create an OTC Cloud Bursting Scenario

5.7.8.1 Scenario Parameters to Collect at Vendor Interface

Make sure that you capture the following scenario parameters. We will remind you about them:

Table 5-12: Scenario Parameters for OTC

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Cloud account	Name of your account at cloud vendor	String
Region	Region selected during configuration at cloud vendor	Drop-down list
Domain name	Domain used by PBS Cloud module head node	String
Hostname prefix	Optional prefix for burst node names; default is "node"; chosen during configuration at vendor	String
Subnet ID	ID of subnet for VPC created at cloud vendor	String
Security Group ID	ID of security group created at cloud vendor	String
OS Image URI	ID of OS image created at cloud vendor	String

5.7.8.2 Steps to Collect Information

1. Open a browser window and log in to the OTC Console.
2. Click Service List in the menu bar.
3. Under Network, click Virtual Private Cloud.
4. Click Virtual Private Cloud from the menu located on the left-hand side of the web page.
5. Click the name of the VPC you created for cloud bursting.
6. Click the name of the VPC's subnet.
7. Copy the Subnet ID.
8. **Save the subnet ID to a file**. You will need this later when you create a bursting scenario.
9. Click Security Group from the menu located on the left- hand side of the web page.
10. Click the name of the security group you created for the VPC.
11. Copy the ID of the security group.
12. **Save the security group to a file**. You will need this later when you create a bursting scenario.
13. Click Service List in the menu bar.
14. Under Computing, click Image Management Service.
15. Click the Private Images tab.

16. Click the name of the VM image you created for cloud bursting.
17. Copy the Image ID of the image.
18. **Save the OS Image ID to a file**. You will need this later when you create a bursting scenario.

5.8 Configure OpenStack Cloud Bursting

You can find OpenStack documentation for the Stein release at https://docs.openstack.org/stein/index.html. For specific details, contact Altair support.

5.8.1 Get OpenStack Administrator Credentials

Get administrator credentials. While you work through the process of getting the administrator credentials, collect the following, and save them to a file:

Table 5-13: Account Parameters for OpenStack

Account Parameter	What to Collect During Configuration in Cloud Interface	Format
Auth URL	OpenStack: contact Altair support	String
User Domain Name	Domain name used to log in to your private cloud; see your OpenStack administrator	String
Username	Administrator username created using cloud interface	String
Password	Administrator password created using cloud interface	String

1. Find the username of the administrative user.
2. If you do not know the password, reset it.
3. **Save the password**. You will need this later when you add the administrative user account to PBS Cloud.

5.8.2 Create Virtual Private Cloud and OS Image

In the following sections, we touch on the steps to create a VPC and an OS image. While you are in the process, collect the following information, and save it to a file:

Table 5-14: Scenario Parameters for OpenStack

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Cloud account	Name of your account at cloud vendor	String
Region	Region selected during configuration at cloud vendor	Drop-down list
Domain name	Domain used by PBS Cloud module head node	String
Hostname prefix	Optional prefix for burst node names; default is "node"; chosen during configuration at vendor	String

Table 5-14: Scenario Parameters for OpenStack

Scenario Parameter	What to Collect During Configuration at Vendor	Format
Subnet ID	ID of subnet for VPC created at cloud vendor	String
Security Group ID	ID of security group created at cloud vendor	String
OS Image URI	ID of OS image created at cloud vendor	String

5.8.2.1 Create a Virtual Private Cloud

Create a virtual private cloud using the Stein release of OpenStack:

1. Choose a region, a name, and an address range for the VPC.
2. Create a subnet and choose a name and an address range for it.
3. Create a security group and choose a name for it.
4. Add an inbound rule:
    - Use TCP
    - Use port 22
    - Choose IP Address
    - Specify your PBS Cloud firewall IP address

      Warning: 0.0.0.0/0 enables all IPv4 addresses to access your instance. ::/0 enables all IPv6 addresses to access your instance. This is acceptable for a short time in a test environment, but it's unsafe for production environments. In production, authorize only a specific IP address or range of addresses to access your instance.

5.8.2.2 Create a Virtual Machine

Create a virtual machine using the instructions for your cloud. We recommend the following:

- Create a key pair, give it a name, and save the .pem file containing the key pair to your local disk in a secure location. PBS Cloud will use this later to manage cloud nodes.
- Choose the same region you chose for the VPC.
- Choose an availability zone, if you have more than one
- Choose a type that fits your needs.
- Use a public image.
- Choose a supported OS
- Choose the disk characteristics you need.
- Assign the VPC you created.
- Assign the security group you created.
- Choose the Key Pair login mode.
- For the key pair, choose the .pem file you generated.
- Give the VM a name.
- Select a quantity of 1 for the number of nodes.

  It may take some time to create the virtual machine.

5. You can delete the virtual machine now to avoid storage costs, or keep the virtual machine and update it over time to create updated OS images for bursting. You will incur storage costs, but this is an effective way to keep your OS images up to date when there are changes in packages, patches, or applications.

5.8.2.3 Install and Configure a PBS MoM on the Virtual Machine

1. Get the public IP address of the VM.
2. Log in to your site's PBS server host.
3. SSH into the VM using the default user "root", the .pem file you generated when creating the VM and the external IP address assigned to the VM.

   ```
   ssh -i /<path to .pem>/<.pem filename>.pem root@<public IP of VM>
   ```
4. Using the PBS Professional Installation and Upgrade Guide, install and configure the PBS MoM.
5. Configure the VM to work with your site environment, for example mounting file systems, connecting to the authentication service, installing any applications you need, etc. We recommend that you include the following:
   - Mount /home in the VM
   - Either install applications in the VM or cross-mount them from the head node
   - Either add users to the password file or connect the VM to a service such as NIS
6. If cloud-init is not installed, install it.

5.8.2.4 Create OS Image from VM

Follow your cloud instructions to create an OS image from the VM you created. Use the same region you chose for the VPC.

On Azure, you can now delete the virtual machine so that you are no longer charged for it.

On AWS, you can delete the virtual machine now to avoid storage costs, or keep the virtual machine and update it over time to create updated OS images for bursting. You will incur storage costs, but this is an effective way to keep your OS images up to date when there are changes in packages, patches, or applications.

5.9 Windows Bursting on AWS and Azure

Bursting of Windows virtual machines is supported on AWS and Azure. Windows cloud bursting is similar to cloud bursting on Linux platforms. Three special requirements are necessary to burst Windows cloud nodes.

5.9.1 OS Image Name

When creating the OS image, the name of the image must contain the term "windows" (case insensitive). For example, on AWS, the AMI Name should look something like this:

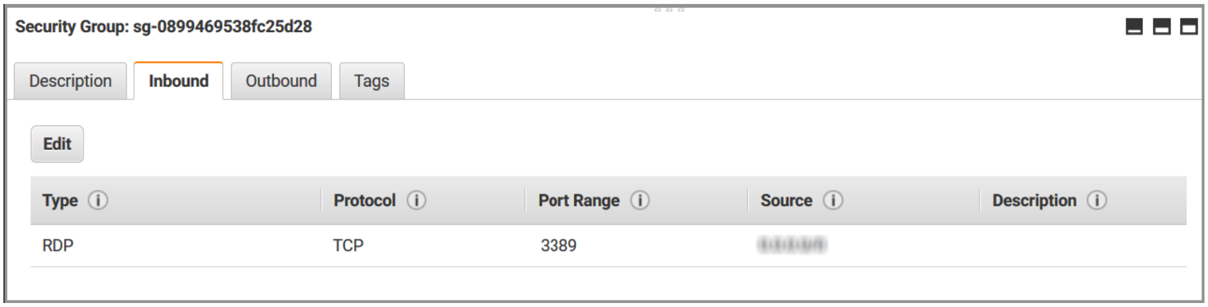
```
Windows_Server-2012-R2_RTM-English-64Bit- Base-2019.11.13
```

On Azure, the Image Name should look something like Windows Server 2012 R2 Datacenter.

5.9.2 Inbound Security Rule for RDP

Secondly, an inbound rule to open the port 3389 must be added to the AWS security group or the Azure network security group that is associated with the cloud provider virtual network. This allows a connection to be made to the Windows VM via RDP so that the PBS MoM can be installed.

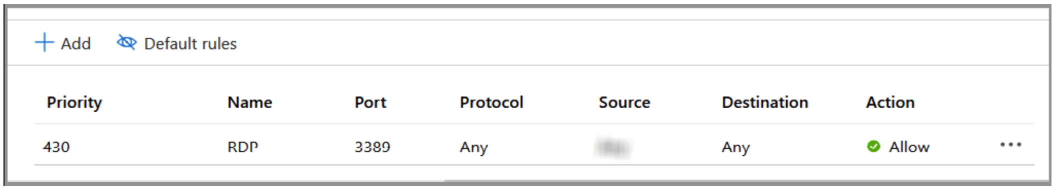
For more information see AWS: Authorizing Inbound Traffic for Your Windows Instances and Azure: Cannot connect remotely to a VM because RDP port is not enabled in NSG.



Security Group: sg-0899469538fc25d28

Type	Protocol	Port Range	Source	Description
RDP	TCP	3389	[redacted]	

Figure 5-33:AWS Inbound Security Rule for RDP



Priority	Name	Port	Protocol	Source	Destination	Action
430	RDP	3389	Any	[redacted]	Any	✓ Allow

Figure 5-34:Azure Inbound Security Rule for RDP

5.9.3 Startup Script

The cloud node startup script must use a PowerShell script. For more information see PowerShell Scripting. The below PowerShell script example generates a file in C:\Windows\Temp:

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
```

5.9.4 See Also

- "Configuring Amazon Web Service Cloud Bursting” on page 49
- "Configuring Microsoft Azure Cloud Bursting” on page 60

6 The Cloud Node Startup Script

6.1 Introduction

For each scenario, you will want to do some configuration of each cloud node after it boots. The simplest way to configure an instance at boot time is to use a shell script. You can add a startup script to each bursting scenario that runs when a scenario instance boots, in order to perform automated tasks to customize your cloud nodes. For example, you may want to install packages, perform updates, start services, add users, configure filesystems (/etc/fstab), configure NIS (/etc/yp.conf), or mount necessary filesystems.

The cloud-init tool is a utility for initializing cloud instances. PBS Cloud uses the cloud-init tool to launch its own startup script first to configure each freshly burst cloud node so that the PBS server can connect to the cloud node. If you use a startup script, it runs after the built-in script.

For more information on cloud-init, see the cloud-init documentation at cloud-init.

Note that you may not need a startup script. If you configure the VM that you use to create your OS image to have everything you need to run jobs, you do not need a startup script.

6.1.1 Making cloud-init Tool Available in OS Image

For each scenario, you create an OS image that includes the cloud-init tool. Make sure that cloud-init is installed in the VM that you use to create the image (we include that step in the instructions). Sometimes it is pre-installed; if it's not there, install it. For example, on CentOS:

```
yum install -y cloud-init
```

6.1.2 Adding a cloud-init Script to a Scenario

You add a cloud-init script to each scenario via the PBS Cloud web interface; see section 3.3.4.9, "Specifying the Cloud Node Startup Script", on page 32. You can add a script while creating a scenario, and you can edit the script or choose a new script later. The startup script to be included in a scenario can be located anywhere that your PBS Cloud web interface can browse to. Once a script has been added to a scenario, the script is stored in the PBS Cloud database.

Make sure you develop the startup script for each scenario. See section 6.2.5, "Developing the Startup Script", on page 117.

6.1.3 Startup Script Prerequisites

- The startup script must run using a shell or language available in the freshly burst node. For example, if you have bash and Python available, your script can use bash, or it could use a bash script to launch a Python script.
- On Windows cloud nodes, use a PowerShell startup script. Enclose the content of the PowerShell script in <powershell> and </powershell>. Refer to Microsoft documentation for more information about PowerShell.
- The startup script can have any name.

6.1.4 Startup Script Recommendations

- We strongly recommend mounting /home to simplify making each cloud node usable for jobs.
- We recommend installing your applications somewhere and then mounting that directory in each cloud node.

6.2 Customizing Your Startup Script

Make sure that your startup script uses the correct locations for PBS_HOME, PBS_EXEC, etc.

6.2.1 Mounting /home Directory

We strongly recommend that you mount /home so that users can run jobs in the cloud node. This makes user SSH keys available, and /home can be used for PBS data transfer. If you do not mount /home, you have to create a /home directory in the image, pre-fill it for each user who will submit jobs, and set up the user space.

```
echo "<PBS server host IP address> <head node> <head node>.<domain name>" >> /etc/hosts
...
...
yum install -y nfs-utils
mount -t nfs <hostname of machine with /home>:/home /home
```

6.2.2 Configuring MoM for Local Copy

We recommend that you set the $usecp MoM configuration parameter to tell the MoM which local directories are mapped to mounted directories, so that MoM can use the local copy mechanism for them:

```
echo "<PBS server host IP address> <head node> <head node>.<domain name>" >> /etc/hosts
...
...
echo "\$usecp <hostname of machine with /home>:/home/ /home/" >> /var/spool/pbs/mom_priv/config
```

See "Configuring MoM for Local Copy" on page 563 in the PBS Professional Administrator’s Guide.

6.2.3 Creating Local Scratch Space

You can create local scratch on a fast local disk and use it as the default location where PBS runs jobs, if PBS will be creating the directories where jobs run. To do this, each job must have its sandbox attribute set to *PRIVATE*, and the $jobdir_root MoM configuration parameter has to be set to /scratch.

```
mkdir /scratch
chmod 1777 /scratch
```

For example, if /scratch is not shared:

```
echo "\$jobdir_root /scratch" >> /var/spool/pbs/mom_priv/config
```

6.2.3.1 Creating Job-specific Staging and Execution Directories

Whether or not PBS creates job-specific staging and execution directories for a job is controlled by the job's sandbox attribute:

- If the job's sandbox attribute is set to *PRIVATE*, PBS creates a staging and execution directory for each job, in the location specified by the $jobdir_root MoM parameter. If the $jobdir_root parameter is unset, PBS creates job-specific staging and execution directories in the job submitter's home directory.
- If the job's sandbox attribute is set to *HOME* or is unset, PBS does not create job-specific staging and execution directories. Instead PBS uses the job submitter's home directory.

6.2.3.2 Using Shared Directories for Staging and Execution

Using a shared directory for job staging and execution is a little more complicated when nodes are released early from a job. Normally each MoM on a sister node that is being released cleans up its own files upon release. However, if the directory is shared, you need to prevent those sister MoM(s) from prematurely cleaning up job files before the job has finished. This is an issue whether or not the directory is the user home directory. You take care of this by specifying whether the directory is shared via the $jobdir_root MoM parameter:

- When staging and execution directories are to be created in a shared (e.g. NFS) directory specified in `$jobdir_root`, set the *shared* directive after the directory name:

  $jobdir_root <directory name> shared

  For example:

  ```
  echo "\$jobdir_root /scratch shared " >> /var/spool/pbs/mom_priv/config
  ```

- If job submitter home directories are shared, tell MoM:

  $jobdir_root PBS_USER_HOME shared

  For example:

  ```
  echo "\$jobdir_root PBS_USER_HOME shared " >> /var/spool/pbs/mom_priv/config
  ```

See "Staging and Execution Directories for Job" on page 525 in the PBS Professional Administrator’s Guide.

6.2.4 Example cloud-init Startup Script for Linux

Example 6-1: cloud-init startup script for Linux. Do not use this script as is; make sure you adapt it for your site.

```
#!/bin/sh
# Map PBS server host IP address to hostnames via /etc/hosts
echo "/etc/hosts setup"
rm -f /etc/hosts
echo "127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4" > /etc/hosts
echo "<PBS server host IP address> <PBS server host> <PBS server host>.<cloud node domain>" >>
    /etc/hosts

# Disable NetworkManager and use network interface
# so that it does not overwrite the /etc/resolv.conf file
systemctl disable NetworkManager
systemctl stop NetworkManager
systemctl enable network
systemctl start network

# Configure PBS via /etc/pbs.conf
echo "PBS setup"
systemctl stop pbs
rm -f /etc/pbs.conf
echo "PBS_EXEC=/opt/pbs/default" > /etc/pbs.conf
echo "PBS_HOME=/var/spool/PBS" >> /etc/pbs.conf
echo "PBS_START_SERVER=0" >> /etc/pbs.conf
echo "PBS_START_MOM=1" >> /etc/pbs.conf
echo "PBS_START_SCHED=0" >> /etc/pbs.conf
echo "PBS_START_COMM=0" >> /etc/pbs.conf
echo "PBS_SERVER=PBS_SERVER_HOSTNAME" >> /etc/pbs.conf
echo "PBS_CORE_LIMIT=unlimited" >> /etc/pbs.conf
echo "PBS_SCP=/bin/scp" >> /etc/pbs.conf

# Update pbs.conf with the cloud node's IP address
IP=$(ip addr show eth0 | grep "inet\b" | awk '{print $2}' | cut -d/ -f1)
echo "PBS_MOM_NODE_NAME=$IP" >> /etc/pbs.conf

# Configure the MoM by updating PBS_HOME/mom_priv/config
echo "MoM configuration setup"
. /etc/pbs.conf
echo "\$clienthost $PBS_SERVER" >> /var/spool/pbs/mom_priv/config
echo "\$clienthost ${PBS_SERVER//.*}" >> /var/spool/pbs/mom_priv/config
echo "\$restrict_user_maxsysid 999" >> /var/spool/pbs/mom_priv/config

# Restart PBS
systemctl start pbs
```

6.2.5 Developing the Startup Script

Troubleshooting issues with cloud bursting can be difficult and time consuming. Issues that can arise include the following:

- Network issues
- SSH key issues
- Missing users or groups
- Missing packages
- Mounted file system errors

6.2.5.1 Prerequisites for Developing a Startup Script

- You have installed PBS Professional and PBS Cloud, configured PBS Professional, created a cloud administrator account at your cloud provider and added that account to PBS Cloud, created and configured your cloud provider components including an OS image, and configured the PBS Cloud module. These steps are outlined in section 3.1, "Overview of Configuring PBS Cloud", on page 15.
- Make sure the cloud bursting hook is disabled.
- You should have at least one PBS Cloud scenario to test.
- This scenario must be enabled.
- The Add Public IP to VMs scenario option is enabled; see section 3.3.4.4, "Temporarily Adding Public IP for Debugging", on page 29
- The SSH keys parameter has an administrator SSH key; see section 3.3.4.8, "Adding SSH Key for Access to Burst Nodes", on page 32
- You have the corresponding private key
- Port 22 in the vendor firewall has to be open (already covered when you were configuring vendor components)

6.2.5.2 What to Test For

- You can log into the node
- The `/home` directory is mounted
- The directory containing your applications is mounted
- Make sure applications work
- Make sure the node was added to PBS (use `pbsnodes -a`)
- The cloud node can be reached from the PBS server host
- The cloud infrastructure (VPN connectivity, networks, firewalls etc.) is working

6.2.5.3 Steps to Develop the Startup Script

We recommend that you burst a single cloud node, test it, and resolve one issue at a time. Each time you resolve an issue, keep a log of your changes, and incorporate your solution into the cloud-init script. You can log your command-line input as you work to resolve the issue, and convert this record into script inputs.

Once the cloud node is configured correctly, you can enable the cloud bursting hook, use it to burst cloud nodes, and troubleshoot the cloud bursting hook.

6.2.5.4 Example of Developing a cloud-init Script

This example is intended to be used after you have installed PBS Professional and PBS Cloud, configured PBS Professional, created a cloud administrator account at your cloud provider and added that account to PBS Cloud, created and configured your cloud provider components including an OS image, and configured the PBS Cloud module. These steps are outlined in section 3.1, “Overview of Configuring PBS Cloud”, on page 15.

Make sure the cloud bursting hook is disabled.

You should have at least one PBS Cloud scenario to test. This scenario must be enabled.

1. If you have not already prepared the scenario for developing a startup script:
   a. Create a cloud-init script by copying the example cloud-init script
   b. In the script, map the PBS server host to its IP address:
   ```
   echo "<PBS server host IP address> <PBS server hostname> <PBS server hostname>.<cloud node domain
       name>" >> /etc/hosts
   ```
   c. Make any other necessary changes to the script
   d. Upload the cloud-init script to the scenario. Enable the Add public IP to VMs option for the scenario, and add a suitable public SSH key to the scenario. Make sure you have the corresponding private key.
1. Log in to PBS Cloud.
2. Click the Cloud tab.
3. Under Infrastructure, click Bursting.
4. Select a bursting scenario by clicking on its name.
5. Under Machines (manually burst), click Manual bursting.
6. Choose the instance type and OS. Set the number of cloud nodes to be 1.
7. Once the cloud node is burst, attempt to SSH into the cloud node using the value of the private_ip parameter that is returned from the PBS Cloud CLI bursting command.
8. If you encounter a "Permission denied" error, there is a problem with the cloud node configuration and/or the cloud-init script. Each user's SSH keys are stored in the user's home directory under the .ssh directory and the public key has been added to .ssh/authorized_keys in the user's home directory, so SSH should be working. .
9. Using PuTTY or a similar SSH client, SSH into the cloud node using the cloud node's public IP address (the value of the parameter public_ip address returned from the PBS Cloud bursting command) and your previously generated private SSH key (matching the public key used in the bursting scenario).
10. Check the contents of the /etc/hosts file. If the PBS server hostname is mapped to its IP address, the cloud-init script is being run.
11. Check which filesystems have been mounted using the df command. If /home is not mounted, this is the cause of the SSH failure to the private IP address of the cloud node.
12. Use the ls command to see whether /home exists.
13. To mount the directory, install nfs-utils and then mount /home:
    ```
    yum install nfs-utils
    mount -t nfs headnode:/home /home
    ```
1. Log in to PBS Cloud.
2. Click the Cloud tab.
3. Under Infrastructure, click Bursting.

4. Select a bursting scenario by clicking on its name.
5. Under Machines (manually burst), click Manual unbursting page.
6. Enabling the check box to the right of the cloud node.
7. Click Unburst.
8. Click Unburst machines to confirm the action.
9. Edit the bursting scenario.
10. Edit the cloud-init script.
11. Copy the two commands to install `nfs-utils` and mount `/home` and paste them into the cloud-init script.
12. Save the cloud-init script.
13. Burst another cloud node and repeat the process.

6.2.5.5 Caveats for Testing Startup Script

Occasionally you will burst a node that appears to be correct in all respects, but it simply won't run a job. Sometimes you just get a bad node. In this case, unburst the node and burst a new one.

7
Managing Cloud Bursting

7.1 Managing Cloud Bursting

7.1.1 Viewing Cloud Account Details

1. Log in to PBS Cloud.
2. Click the Cloud tab.
3. Under Infrastructure, click Cloud.
4. Select a cloud account by clicking on its name.

   The information that was entered to create the cloud account is displayed.
5. Click Close.

7.1.2 Viewing Burst Cloud Nodes

Information that is displayed about cloud nodes:

Machine Name

Hostname of the node.

IP Address

IP address assigned to the node.

Instance Type

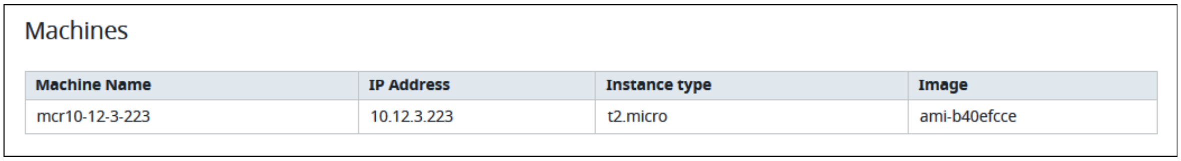
Cloud provider instance type (machine type, shape or flavor) of cloud node.

Image

OS image used to burst the node.

1. Log in to PBS Cloud.
2. Click the Cloud tab.
3. Under Infrastructure, click Bursting.
4. Select a bursting scenario by clicking on its name.

   Any cloud nodes that have been burst are displayed under the Machines category.



Machines

Machine Name	IP Address	Instance type	Image
mcr10-12-3-223	10.12.3.223	t2.micro	ami-b40efcce

Figure 7-1:Burst Cloud Nodes

7.1.3 Enabling or Disabling a Bursting Scenario in PBS Cloud

Each bursting scenario is enabled by default when you create it.

1. Log in to PBS Cloud.
2. Click Cloud.
3. Click the Bursting tab on the left-hand side of the web page.
4. Select the name of the bursting scenario.
   - To enable the bursting scenario, click Enable.
   - To disable the bursting scenario, click Disable.

7.1.4 Disabling Bursting for a Scenario and Queue

How you stop the bursting process depends on whether the remaining jobs in the scenario queue can run using either the existing or edited scenario.

- If the jobs in the queue can run using the changed scenario:
  a. Stop the cloud queue associated with the scenario (prevent jobs in the queue from starting):
     ```
     qmgr -c "set queue <queue name> started=false"
     ```
  b. Allow the burst nodes for that scenario to drain and unburst, or requeue them
- If the jobs in the cloud queue must use the existing scenario:
  a. Disable the cloud queue (prevent jobs being enqueued):
     ```
     qmgr -c "set queue <queue name> enabled=false"
     ```
  b. Log in to PBS Cloud.
  c. Click Cloud.
  d. Click the Bursting tab on the left-hand side of the web page.
  e. Under Infrastructure, click Bursting.
  f. Select the name of the bursting scenario.
  g. Click Disable.
  h. Allow the cloud queue associated with the bursting scenario to drain; allow time for the jobs that are waiting in the queue to run and finish; allow automated unbursting to finish. You can shorten the time to unburst using the PBS Cloud interface or the PBS Cloud CLI.

Verify that all cloud nodes are unburst:

a. Click on the name of the cloud bursting scenario.
b. Look under the Machines heading. The following message indicates that all cloud nodes are unburst:
   ```
   No machines are available
   ```

7.1.5 Re-enabling Bursting for a Scenario and Queue

How you start the bursting process depends on how you stopped it.

- If you stopped the queue to prevent jobs from starting, start the cloud queue associated with the scenario (allow jobs to start):

  ```
  qmgr -c "set queue <queue name> started=true"
  ```

- If you disabled the queue to prevent jobs from being enqueued:
  a. Enable the cloud queue (allow jobs to be enqueued):

     ```
     qmgr -c "set queue <queue name> enabled=true"
     ```

  b. Log in to PBS Cloud.
  c. Click Cloud.
  d. Click the Bursting tab on the left-hand side of the web page.
  e. Under Infrastructure, click Bursting.
  f. Select the name of the bursting scenario.
  g. Click Enable.

7.2 Starting and Stopping PBS Cloud

7.2.1 Start PBS Cloud

Starting PBS Cloud must be done as root or as a user with sudo permissions using the sudo command.

When your server hosting the PBS Cloud component reboots, containers are restarted automatically. If you need to manually start PBS Cloud containers, please follow the below instructions.

1. Log in to the machine where PBS Cloud is installed.
2. Enter the following command to start PBS Cloud:

   ```
   pkr start
   ```

7.2.2 Stop PBS Cloud

Stop PBS Cloud after a manual installation.

Stopping PBS Cloud must be done as root or as a user with sudo permissions using the sudo command.

1. Log in to the machine where PBS Cloud is installed.
2. Enter the following command to stop PBS Cloud:

   ```
   pkr stop
   ```

7.2.3 Restart PBS Cloud

Restarting PBS Cloud must be done as root or as a user with sudo permissions using the sudo command.

When your server hosting the PBS Cloud component reboots, containers are restarted automatically. If you need to manually restart PBS Cloud containers, please follow below instructions.

1. Log in to the machine where PBS Cloud is installed.
2. Enter the following command to restart PBS Cloud:

```
pkr restart
```

7.2.4 Determine the Status of PBS Cloud

Determine whether PBS Cloud is up or down.

1. Log in to the machine where PBS Cloud is installed.
2. Enter the following command to display the status of PBS Cloud:

```
pkr status
```

7.2.5 Monitoring Logs and Workflows

PBS Cloud includes a Loki interface for monitoring logs and workflows. To use this, log into PBS Cloud, then choose *Monitoring*, *Logs*, or *Workflows*.

The Monitoring section shows the status of the cloud services, including how many nodes are in the cloud, etc.

The Logs section shows all the PBS Cloud logs in chronological order.

The Workflows section shows you specific information about each workflow stage.

7.3 Logging into PBS Cloud

To log into PBS Cloud, go to the PBS Cloud interface in your web browser:

```
http://<PBS Cloud hostname or IP address>:<port>/pbspro-cloud/#/login
```

The default port is 9980.

7.4 Troubleshooting Cloud Bursting

Log messages can help you troubleshoot cloud bursting:

- Check log messages written to `PBS_HOME/server_logs` on the PBS server host.
- For debugging issues with node creation or issues with starting MoM on the cloud node, SSH to the cloud node and check `PBS_HOME/mom_logs`. This requires:
  - The Add Public IP to VMs scenario option is enabled; see section 3.3.4.4, “Temporarily Adding Public IP for Debugging”, on page 29
  - The SSH keys parameter has an administrator SSH key; see section 3.3.4.8, “Adding SSH Key for Access to Burst Nodes”, on page 32
- You can use PBS Cloud to view the logs through a Loki interface; see section 7.2.5, “Monitoring Logs and Workflows”, on page 124

7.4.1 PBS MoM is Stopped or Down

When you are using cloud bursting and all PBS MoMs are stopped or down, you may find error messages similar to the following in the PBS Server logs:

```
Server@server;Hook;Server@server;CLBR: Error: /opt/pbs/bin/pbsnodes: Server has no node list
Server@server;Hook;Server@server;CLBR: Error: Failed to get nodes info
```

Resolve the issue by starting at least one MoM.

8 Managing Cloud Jobs

8.1 Manage Cloud and On-Premise Jobs

We present solutions to handle various job distribution needs.

Your site may want to run certain types of jobs on-premises or in the cloud. PBS Professional provides various methods to collect and distribute jobs. For more information see Routing Jobs in the PBS Professional Administrator's Guide. Here are some solutions to handle various job distribution needs.

8.1.1 Associate Nodes with Queues

In general, start by associating nodes with queues, so that a job in a specific queue is sent to the appropriate nodes. Associate each on-premise vnode with the appropriate local queue, and associate each bursted vnode with the appropriate cloud queue; see "Associating Vnodes with Queues" on page 105 in the PBS Professional Administrator’s Guide.

8.1.2 Use Hooks or Routing Queues to Send Jobs to the Appropriate Queue

To send jobs to the appropriate queue on submission, use hooks or routing queues. For more information see the *PBS Professional Hooks Guide* and "Routing Queues" on page 27 in the PBS Professional Administrator’s Guide.

8.1.3 Job Distribution Examples and Solutions

8.1.3.1 Send Small Jobs to the Cloud

You have big machines for on-premise nodes and want to reserve those big machines for big jobs. You want to send smaller jobs to the cloud.

Solution 1:

- Create three queues: a routing queue, a local queue, and a cloud queue.
- Use the routing queue to collect jobs on submission.
- Set resource gating on the local queue to filter out smaller jobs.
- Allow smaller jobs into the cloud queue.

Solution 2:

- Create two queues: a local queue, and a cloud queue.
- Use a queuejob hook to route jobs into the appropriate queue.

8.1.3.2 Send Specific Jobs Only to the Cloud

You want to send specific jobs to the cloud because:

- an application needed by some jobs runs well in the cloud.
- a resource that is available in the cloud is not available locally.
- a department has exhausted its share of local resources, and wants to send its jobs to the cloud.

Solution :

- Create two queues: a local queue, and a cloud queue.
- Use a queuejob hook to route jobs into the appropriate queue.

8.1.3.3 Charge Departments for Resources Used

You have multiple departments and each department should be charged for the resources it uses.

Solution 1:

- Consider using PBS Budget to monitor consumption of on premise and cloud resources

Solution 2:

- Create a cloud queue for each department
- Set separate limits on each cloud queue

8.2 Run Cloud Jobs On-Premise Before Bursting

Use placement sets to run jobs on-premise first and burst cloud nodes when local nodes are not available.

Assign a bursting scenario to an on-premise vnode so that cloud jobs are run on-premise when there is capacity. Cloud nodes are only burst when on-premise nodes are not available.

For more information on placement sets, see "Placement Sets" on page 170 in the PBS Professional Administrator’s Guide.

1. Create a custom resource to indicate that a node is on-premise.

   ```
   qmgr -c "create resource node_location type=string,flag=h"
   ```

2. For each on-premise vnode, set the resources_available.node_location resource to "local" to indicate that it is on-premise.

   ```
   qmgr -c "active node <vnode name>"
   qmgr -c "set node <vnode name> resources_available.node_location=local"
   ```

3. For any on-premise vnodes where you want to run cloud jobs, associate a bursting scenario or a list of bursting scenarios to the vnode:

   - To associate a list of bursting scenarios with a vnode:

     ```
     qmgr -c "set node <vnode name> resources_available.cloud_scenario= SCENARIO_1,
         SCENARIO_2,...,SCENARIO_n"
     ```

   - To associate a single bursting scenario with a vnode:

     ```
     qmgr -c "set node <vnode name> resources_available.cloud_scenario= SCENARIO_1"
     ```

   Where SCENARIO_1,..., SCENARIO_n are scenario names that have been added to the cloud bursting hook configuration file.

   ```
   qmgr -c "set node <vnode name> resources_available.cloud_scenario=
       azure_scenario_1,aws_scenario_1,aws_scenario_2"
   ```

4. Create a dummy vnode with a high resources_available.ncpus value and unset resources_available.node_location.
5. Mark the dummy node as offline so that jobs cannot run on it.

   ```
   pbsnodes -C "<comment>" -o HOSTNAME
   ```

   Where <comment> is a string explaining that this is a dummy node for cloud bursting and HOSTNAME is the hostname of the vnode.
6. Enable placement sets:

   ```
   qmgr -c "set server node_group_enable=true"
   qmgr -c "set server node_group_key=node_location"
   ```
7. Allow creation of placement sets from unset resources:

   ```
   qmgr -c "set sched only_explicit_psets=false"
   ```
8. Optionally, prevent any single job from running on both local and cloud vnodes:

   ```
   qmgr -c "set sched do_not_span_psets=true"
   ```
9. Navigate to PBS_HOME/sched_priv.
10. Edit the sched_config file.
11. Add cloud_scenario and node_location to the resources line:

    ```
    "... , cloud_scenario, node_location"
    ```
12. For each local (non-cloud) queue set its node_location to "local":

    ```
    qmgr -c "set queue <queue name> resources_default.node_location=local"
    ```

    Where <queue name> is the name of the local queue.
13. For each cloud queue, verify that the value of resources_default.node_location is unset.
14. Restart or HUP the scheduler:

    ```
    kill -HUP <scheduler PID>
    ```

8.2.1 Override Instance Type or Image at Job Submission

The OS image that is to be used when a cloud node is burst can be specified at job submission via the qsub command. If this information is not provided at job submission, then the OS image defaults to the value of the cloud_default_image parameter in the cloud bursting hook configuration file.

The job submitter can use only the instance types that have been allowed by the administrator. If the job submitter tries to use a non-allowed instance type, the job does not run.

Additionally, the instance type that is burst can be specified at job submission via the qsub command (e.g. Standard_DS3_v2). Instance types are defined by the cloud provider. The name of the instance type must exactly match the cloud provider's. If this information is not provided at job submission, then the instance type defaults to the instance type defined for the cloud queue.

1. Override the cloud image at job submission via the qsub command:

   ```
   qsub -l select=1:ncpus=4 -q <queue name> -v CLOUD_IMAGE=<OS image> <job script>
   ```

   Where:

   - IMAGE is the name of the OS image to be used when the node is burst.
2. Override the cloud instance at job submission via the qsub command:

   ```
   qsub -l select=1:ncpus=4 -q <queue name> -v CLOUD_INSTANCE=INSTANCE_TYPE <job script>
   ```

Where:

- INSTANCE_TYPE is the name of cloud provider instance type. This is the type of machine that will be burst.

3. You can also override both the OS image and the instance type as part of the qsub command:

```
qsub -l select=1:ncpus=4 -q <queue name> -v CLOUD_IMAGE=<OS image>,CLOUD_INSTANCE=INSTANCE_TYPE
    <job script>
```

8.2.2 Request InfiniBand Nodes

You will need to create an InfiniBand enabled OS image before you can submit a job to an InfiniBand cloud node.

When the user wants to run HPC workloads like MPI jobs, the user needs InfiniBand supported nodes deployed on the same InfiniBand network. Among the cloud providers, only Azure currently supports InfiniBand nodes. Azure limits the number of nodes on a single Infiniband network (current default limit is 100), however your site may set up multiple Infiniband networks. A restriction for bursting on Infiniband networks is that PBS jobs cannot run across multiple Infiniband networks. To prevent this from happening, placement sets can be used.

At job submission, the user can request InfiniBand nodes by requesting an InfiniBand supported instance type, an InfiniBand enabled OS image, and an InfiniBand network.

Upon bursting an InfiniBand node, the name of the network is obtained from the cloud provider. The custom resource cloud_network on the Infiniband node is updated with this value. All nodes on the same Infiniband network have this custom resource set to the same value. This value is then used to restrict jobs to run within the same network, using placement sets.

1. Enable placement sets:

```
qmgr -c "set server node_group_key=cloud_network"
qmgr -c "set server node_group_enable=true"
```

2. Prevent any single job from running across multiple Infiniband networks:

```
qmgr -c "set sched do_not_span_psets=true"
```

3. Request Infiniband nodes by overriding the cloud image and the cloud instance and requesting an Infiniband network via the qsub command:

```
qsub -l select=1:ncpus=4 -q <queue name> -v CLOUD_IMAGE=<OS
    image>,CLOUD_INSTANCE=INSTANCE_TYPE,CLOUD_NETWORK=IB <job script>
```

Where:

- INSTANCE_TYPE is the name of cloud provider instance type (machine types, shapes or flavors) and should be an instance type that supports Infiniband.
- IMAGE is the name of the Infiniband enabled OS image to be used when the node is burst.
- CLOUD_NETWORK=IB requests an Infiniband network
- JOB_SCRIPT is the name of the script to be executed.

8.3 Adding Application Licenses

If cloud jobs need to use externally-managed licenses and you have not already configured PBS Cloud to handle them, make sure you do so:

1. Read section 1.3.2, “Tracking Application Licenses”, on page 3
2. Follow the steps in section 3.2.3, “Create Resources and Scripts to Manage Application Licenses”, on page 19

3. For each scenario that uses licenses, follow the instructions in section 4.2.4.2, “Steps to Define a Scenario in Hook Configuration File”, on page 44 to set the value for check_resources

9
Running Cloud Jobs

9.1 Introduction

Cloud jobs are submitted to one or more designated cloud queues, and a server periodic hook monitors the cloud queues, estimates the demand for cloud nodes, and dynamically adjusts the number of nodes by bursting or unbursting as needed. Each cloud queue gives its job access to a specific scenario; that scenario offers specific instance types and application licenses.

9.1.1 Running Your Job in the Cloud

Each cloud scenario is associated with a specific cloud queue. Each scenario offers specific instance types. In order to run your job in the cloud, you need to submit the job to the appropriate cloud queue.

9.1.2 Requesting Instance Type

You can choose any of the instance types available in the scenario you’ve chosen. To request an instance type:

```
CLOUD_INSTANCE=<instance type>
```

You can choose an OS image:

```
CLOUD_IMAGE=<name of image file>
```

9.1.2.1 Requesting Preemptable and Spot Instances

If the scenario includes them, you can request preemptable instances, including spot instances. When requesting cloud nodes, the request should be for either non-preemptable instances or preemptable instances, but not both. Do not request some cloud nodes that are on-demand and some that are preemptable or spot.

9.1.3 Running Jobs Requiring Application Licenses

To run a job that needs an application license, choose a scenario that offers that application license.

Each application license is represented by two PBS resources; one is static, and one is dynamic. If your job requires an application license, your job script must include requests for both resources. For example, if your job requires an App1 license, represented by the resources app1_static and app1_dynamic, your job script should contain the following:

```
#PBS -l app1_static=1
#PBS -l app1_dynamic=1
```

9.2 Sample Job Scripts for Cloud Jobs

9.2.1 Example of Simple Sleep Job Script

Simple cloud job script: simple job requesting 10 minutes of walltime that will sleep for 1 minute (or tune $sleeptime as appropriate) and then exit. It requests *cloudq*; adjust the name depending on your site configuration. You can save the following job script as sleep.sh. Then you can submit it to PBS:

```
qsub sleep.sh
```

Example 9-1:

```
#!/bin/bash
#PBS -N testjob
#PBS -j oe
#PBS -m n
#PBS -q cloudq
#PBS -l select=1:ncpus=2:mem=16mb
#PBS -l walltime=0:10:00
sleeptime=60
cmd="sleep $sleeptime"
echo $cmd
$cmd
exit
```

9.2.2 Example of Radioss Cloud Job Script

Example 9-2: Job script for cloud job that uses 25 Radioss licenses. This script uses Intel MPI. The static resource is named "Rad_stat" and the dynamic resource is named "Rad_dyn":

```
#!/bin/bash
#PBS -N RunRad
#PBS -j oe
#PBS -m n
#PBS -q CloudRadq
#PBS -P project1
#PBS -l select=1:ncpus=16:mem=16gb
#PBS -l walltime=2:00:00
#PBS -l Rad_stat=25
#PBS -l Rad_dyn=25
/usr/local/altair/scripts/radioss -mpi i -nt $NCPUS -np 1 -hostfile $PBS_NODEFILE -both
    SEAT_DYREL_0000.rad
```

9.2.3 Viewing Job Output

When the job completes you should see the job's output. This will appear where the job was submitted.

9.3 Logging into PBS Cloud

To log into PBS Cloud, go to the PBS Cloud interface in your web browser:

```
http://<PBS Cloud host name or IP address>:<port>/pbspro-cloud/#/login
```

The default port is *9980*.

10 Example Azure Head Node

10.1 Example Configuration of Cloud Head Node in Azure

Here we show an example of how to configure a cloud head node using Azure:

1. Create a new VM based on Centos 7, using Rogue Wave Software's CentOS-based 7.6 image, not an HPC tagged version
   - Use Standard_D14 instance type for head node (Good price/performance option)
   - Add additional storage: 1000GB SSD to allow an ext4 volume to be added; the Centos image uses XFS filesystem which has a bug with docker
2. Provide a public ssh key for the username "centos" via Azure GUI

   This is the only default user in this image
3. Once the node is up, use the provided external IP and your private key to connect to it. PuTTY is more robust for connection to cloud than Moba xTerm
4. Unless noted otherwise all the following commands must be run as root. Use sudo or the following to switch to the root user:

   ```
   sudo su -
   ```
5. Use `cfdisk` to partition `/dev/sdc` using all space for a single volume. Make absolutely sure that `/dev/sdc` is the right target. The target depends on instance type chosen and how many disks are added; if you followed the procedure exactly this should be correct in this case, but formatting is destructive to data on the disk.

   ```
   cfdisk /dev/sdc
   ```
   a. Select New

   b. Select Primary

   c. Accept Default Size (Should be whole disk)

   d. Select Write

   e. Answer yes (You will need to type yes)

   f. Select Quit
6. Create a filesystem on your drive (ext4 preferred):

   ```
   mkfs -t ext4 /dev/sdc1
   ```
7. Make a folder to mount your new volume to:

   ```
   mkdir /data
   ```
8. Set suitable permissions on the folder:

   ```
   chmod 777 /data
   ```

9. Find UUID from /dev/disk/by-uuid, e.g.

```
lrwxrwxrwx 1 root root 10 Apr 1 12:48 640a03fd-aa69-4f8d-98d5-2f0d3d12bb26 -> ../../sdb1
lrwxrwxrwx 1 root root 10 Apr 1 12:48 a505f591-5a7d-424f-a5a1-06dcb72f944c -> ../../sda1
lrwxrwxrwx 1 root root 10 Apr 1 12:48 c6cd262b-3930-48a8-9b21-8981bb479cee -> ../../sdc1
lrwxrwxrwx 1 root root 10 Apr 1 12:48 e0d6ff47-4c69-4a4c-b44a-13ea19d80f96 -> ../../sda2
```

10. Add a line to end of `/etc/fstab` for this mount (Note: ID of `/dev/sdc1` above):

```
UUID=c6cd262b-3930-48a8-9b21-8981bb479cee /data ext4 defaults 0 0
```

11. Mount the new filesystem:

```
mount /data
```

12. Upgrade all system packages to the latest versions:

```
yum upgrade
```

13. Disable SELinux by editing `/etc/selinux/config` and changing, then save:

```
SELINUX=enforcing -> SELINUX=disabled
```

14. Disable and stop `firewalld`

```
systemctl disable firewalld
systemctl stop firewalld
```

15. SELinux is still enforcing, so prevent that. Reboot.

16. Add password to user "centos":

```
passwd centos
```

17. Log in as centos

18. Create ssh keys for centos and enable passwordless ssh

    Accept all the defaults for ssh-keygen step:

```
cd $HOME
ssh-keygen
cd .ssh
cat id_rsa.pub >> authorized_keys
```

19. For a cloud head node, make sure you validate that external ssh still works externally, before you disconnect your first session. If there is an issue with ssh keys, which can be sometimes caused by errors in step 18 above, and you disconnect your first session, you could be permanently locked out of your cloud VM. Use ssh from your local machine to your cloud head node to create a second session using your public ssh key. Troubleshoot as required while your first session is still up.

20. Add the key PBS Professional service users:

```
useradd -rm pbsdata; useradd pbsadmin
```

21. Follow steps 17, 18, and 19 for user pbsadmin

22. Install/Start/Enable `docker-ce`

```
yum install -y yum-utils
yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo
yum install docker-ce docker-ce-cli containerd.io
systemctl enable docker
systemctl start docker
```

23. Stop docker:

```
systemctl stop docker
```

24. Move docker data storage to new filesystem mounted on /data

25. Add -g /data/docker \ in file /lib/systemd/system/docker.service. ExecStart should look like this:

```
ExecStart=/usr/bin/dockerd-current ...
--seccomp-profile=/etc/docker/seccomp.json -g /data/docker $OPTIONS ...
$REGISTRIES
```

26. Reread daemon config files:

```
systemctl daemon-reload
```

27. Look in /data/docker; there should be subdirectories for Docker data

28. Start docker:

```
systemctl start docker
```

29. Create a repository for the PBS package:

```
mkdir /home/centos/software
chown -R centos:centos /home/centos/software
chmod 777 /home/centos/software
```

30. Download all the PBS package modules to your software directory (PBS Professional, PBS Cloud)

31. Install and configure NFS Server to share /home from head node, so you can avoid setting up ssh keys for client nodes and have a convenient mount:

```
yum install nfs-utils
systemctl enable nfs-blkmap
systemctl enable nfs-rquotad
systemctl enable nfs-server
systemctl enable nfs
systemctl start nfs-blkmap
systemctl start nfs-rquotad
systemctl start nfs-server
systemctl start nfs
```

32. Edit /etc/exports to add

```
/home *(rw,sync,no_root_squash,no_all_squash)
```

33. Restart NFS server:

```
systemctl restart nfs-server
```

34. Add local machine name and IP to /etc/hosts, remove IPv6 loopback, e.g.

```
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
<PBS server IP address> <PBS server hostname>
```

This would look like:

```
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
172.17.0.6 myhost
```

35. Follow the *PBS Professional Installation & Upgrade Guide*, the *PBS Professional Administrator's Guide,* and the *PBS Professional Licensing Guide* to install, configure and license PBS Professional
    - Set PBS to have MoM running on the head node before first start
36. Run test job and ensure job output is returned without issues (e.g. `scp` problems)

    Sample test script using cloudq:

    ```
    #!/bin/bash
    #PBS -N testjob
    #PBS -j oe
    #PBS -m n
    #PBS -q cloudq
    #PBS -l select=1:ncpus=2:mem=16mb
    #PBS -l walltime=0:10:00
    sleeptime=60
    cmd="sleep $sleeptime"
    echo $cmd
    $cmd
    Exit
    ```

11
Command Reference

11.1 PBS Cloud CLI pclm Commands

11.1.1 CLI Bursting Scenario Commands

Command line interface commands that can be used to enable and disable a bursting scenario and get bursting scenario information.

A cloud bursting scenario is created using PBS Cloud and provides information needed for cloud bursting including the cloud provider, region where the node is burst, VPC details, cloud nodes booting script, SSH keys, valid instance types, etc.

Use the PBS Cloud CLI to obtain information about previously created bursting scenarios including its status, enable a bursting scenario so that nodes can be burst, or disable a bursting scenario to restrict nodes from being burst.

11.1.1.1 Bursting Scenario States

READY

When scenario is enabled, this indicates the bursting scenario can be used to burst nodes

When scenario is disabled, the scenario could be used to burst nodes if you enabled it

PENDING

The scenario is added but not validated.

BUSY

The bursting or unbursting workflow is running; we are in the process of bursting or unbursting.

DELETING

The bursting scenario is being deleted.

ERROR

The bursting scenario contains an error.

11.1.1.2 PBS Cloud Account States

READY

The account is ready and usable.

PENDING

Your login credentials are being validated with the cloud vendor.

BUSY

The account is busy.

DELETING

The account is being deleted.

ERROR

The account contains an error.

11.1.1.3 Some Outputs of PCLM Commands

Scenario ID

Unique label that identifies a bursting scenario. Required to enable, disable or get the status of a bursting scenario. Use the Display a List of Bursting Scenarios command to obtain a list of bursting scenarios including the scenario ID.

Cloud node hostname

Hostname assigned to the cloud node. This information can be obtained by executing the command to display cloud node details.

Machine ID

A unique label that identifies the cloud node. This information can be obtained by executing the command to display cloud node details.

Private IP

Private IP address assigned to the cloud node. This information can be obtained by executing the command to display cloud node details.

11.1.1.4 Display a List of Bursting Scenarios

The below command displays a list of bursting scenarios that have been previously created using PBS Cloud.

```
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper scenario list
```

The output of the command is a list of bursting scenarios including the scenario ID, the state of the scenario, and whether the scenario is enabled or disabled.

11.1.1.5 Enable a Bursting Scenario

The below command enables a bursting scenario so that cloud nodes can be burst using the scenario.

```
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper scenario enable <scenario ID>
```

To verify that the bursting scenario is enabled, use the command Display Bursting Scenario Details.

11.1.1.6 Disable a Bursting Scenario

The below command disables a bursting scenario. Cloud nodes cannot be burst using a scenario that is disabled.

```
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper scenario disable <scenario ID>
```

To verify that the bursting scenario is disabled, use the command Display Bursting Scenario Details.

11.1.1.7 Display Bursting Scenario Details

The below command returns information about the bursting scenario:

```
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper scenario show --id <scenario ID>
```

The output of the command displays information about the bursting scenario including the scenario ID, the state of the scenario, the associated cloud account. and whether the scenario is enabled or disabled.

If you want to display the amount of time before an idle node is unburst, then use the below command (the --raw option must be provided)

```
pclm --raw --api-endpoint=http://HOST:9980/pbspro-cloud/ --api-key KEY bootstrapper scenario show
```

The output of the command is in a JSON format. The idle time is displayed as a key-value pair in the output: {"idle_before_unburst": 100}.

11.1.1.8 Update the Minimum Time Before an Idle Node is Unburst

We recommend setting the Idle Before Unburst scenario parameter to more than double the PBS scheduler cycle time. The below command updates the bursting scenario setting the minimum time that a cloud node can be idle before it is unburst.

```
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper scenario patch --idle-before-unburst <max idle time before
    unbursting> <scenario ID>
```

To verify that the idle before unburst time is updated, use the command Display Bursting Scenario Details with the --raw option.

11.1.2 CLI Node Bursting Commands

Command line interface commands that can be used to burst and unburst cloud nodes and get status information about a bursting activity.

These commands can be used to test cloud bursting without using the cloud bursting hook. This will ensure that the connectivity from the PBS Server to the cloud infrastructure and the bursting scenarios are working properly. These commands can also be used when a site wants to burst a cloud node so that it remains burst indefinitely.

Cloud nodes burst manually remain up and running until explicitly unburst using the PBS Cloud CLI.

11.1.2.1 Command Options

Options that can be used with the `pclm` command:

--wait

Causes the application to wait for the node to be burst/unburst. This option must follow the keyword "bootstrapper".

```
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper --wait scenario burst \
```

--raw

Displays the output of the command in JSON format. This option must follow the keyword "pclm".

```
pclm --raw --api-endpoint==http://<PBS Cloud host name or IP address>:<PBS Cloud
    port>/pbspro-cloud/ --api-key <API key>
```

11.1.2.2 Command for Bursting Cloud Nodes

The below command bursts a number of cloud nodes based on <burst count> of type <instance type> for the bursting scenario identified by <API key>.

```
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper --wait scenario burst '{"mom":[{"deployable_id":"<instance
    type>","count":"<burst count>","tags":{"burst-by":"user"}}]}'
```

The PBS cloud bursting hook always adds a tag called "burst-by" and its value is set to "pbs-cloudhook".

By using the same tag with a different value or a different tag altogether, manually burst nodes can be distinguished from those burst via the bursting hook.

The JSON that describes the type of node to burst and how many to burst can be included directly on the command-line, as shown above, or loaded from a separate file. For example, the following JSON can be copied to a file called node_to_burst.json:

```
{"mom":[{"deployable_id": "<instance type>", "count":"10", "tags": {"burst-by": "user"}}]}
```

The file can then be used as part of the bursting command to load the information.

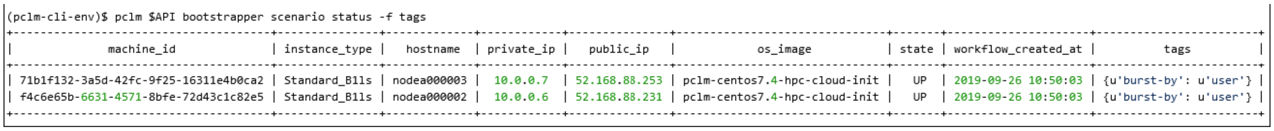
```
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper --wait scenario burst node_to_burst.json
```

11.1.2.3 Command to Display Cloud Node Details

The below command displays details for cloud nodes:

```
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper scenario status -f tags
```

The status is displayed as a table. Use --raw to display the output of the command in JSON format. The status command displays the machine ID, instance type, hostname, private and public IP addresses, the OS image used to create the node, the node's state, and creation time. When the option -f tags is used, any associated tags are displayed as well.



```
(pclm-cli-env)$ pclm $API bootstrapper scenario status -f tags
+--------------------------------------+---------------+-------------+------------+---------------+-------------------------------+-------+---------------------+------------------------+
|              machine_id              | instance_type |   hostname  | private_ip |   public_ip   |           os_image            | state | workflow_created_at |          tags          |
+--------------------------------------+---------------+-------------+------------+---------------+-------------------------------+-------+---------------------+------------------------+
| 71b1f132-3a5d-42fc-9f25-16311e4b0ca2 | Standard_B1ls | nodea000003 |  10.0.0.7  | 52.168.88.253 | pclm-centos7.4-hpc-cloud-init |  UP   | 2019-09-26 10:50:03 | {u'burst-by': u'user'} |
| f4c6e65b-6631-4571-8bfe-72d43c1c82e5 | Standard_B1ls | nodea000002 |  10.0.0.6  | 52.168.88.231 | pclm-centos7.4-hpc-cloud-init |  UP   | 2019-09-26 10:50:03 | {u'burst-by': u'user'} |
+--------------------------------------+---------------+-------------+------------+---------------+-------------------------------+-------+---------------------+------------------------+
```

Figure 11-1:Command Output

11.1.2.4 Cloud Node States

The following is a list of possible virtual machine states. Only a virtual machine that is UP is guaranteed to have an IP addresses and a hostname. A virtual machine that has been fully unburst will not be present in the output at all.

DOWN

Indicates the machine is created in the local database, but not yet in the cloud provider. This status is short-lived and rarely visible.

DEPLOYING

Indicates deployment is ongoing.

FAILED_DEPLOYING

Indicates that something has gone wrong and the machine deployment has failed. The virtual machine is automatically unburst in this case.

FAILED_STARTING

Indicates that something has gone wrong and the machine has not started.

FAILED_STOPPING

Indicates that something has gone wrong and the machine has not been stopped.

FAILED_TERMINATING

Indicates that something has gone wrong and the machine has not been terminated.

STARTING

Indicates machine is starting.

STOPPED

Indicates the machine is stopped.

STOPPING

Indicates machine is stopping.

TERMINATING

Indicates that machine is being removed.

UP

Indicates that the machine deployment has completed and the machine is ready.

11.1.2.5 Command for Unbursting Cloud Nodes

The below command unbursts cloud nodes. Use the command to display cloud bursting activity to obtain information about cloud nodes. A cloud node can be unburst using its machine ID, private IP address, or hostname.

pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/ --api-key <API key> bootstrapper --wait scenario unburst <machine ID1>|<private IP address1>|<cloud node hostname1> <machine ID2>|<private IP address2>|<cloud node hostname2> [...]

Multiple nodes can be unburst in a single call and the parameter used to unburst the nodes can be mixed:

unburst <machine ID1> <private IP address2> <cloud node hostname3>

This information can also be supplied in JSON format on the command-line or in a JSON file:

```
unburst '["<machine ID1>", "<private IP address2>", "<cloud node hostname3>"]'
unburst /tmp/machines_to_unburst.json
```

11.1.2.6 Sizing the Network Disk for the Cloud Node Root System

Use the disk_size_gb parameter to define the size of the network disk for the root file system in GBs when bursting a cloud node. By default the minimum size is compatible with the image associated with the bursting scenario defined by <API key>.

```
"mom" : [
    {
        "deployable_id": "c3.xlarge",
        "disk_size_gb": 12
    },
    {
        "deployable_id": "d2.xlarge",
        "count": 2,
        "disk_size_gb": 15
    }
]
```

11.1.2.7 Specifying the Image to Use when Bursting

You can manually burst a virtual machine from a specific image using the image parameter. By default, the virtual machine is burst based on the image specified when creating a bursting scenario (defined by the <API key>). The value of image depends on the cloud provider. For example, for AWS the name of the AMI is specified:

```
"mom":[{"deployable_id": "t2.medium", "count":"10", "image":"ami-123456"}]
```

For Azure, the resource group into which to place the VM and the name of the image must be specified:

```
"mom":[{"deployable_id": "Standard_DS1_v2", "count":"10", "image":"res-group/imagename"}]
```

11.1.2.8 Allowing Users to Request Infiniband Nodes

When the user wants to run HPC workloads like MPI jobs, the user needs InfiniBand supported nodes deployed on the same InfiniBand network.   A restriction for bursting on Infiniband networks is that jobs cannot run across multiple Infiniband networks. To prevent this from happening, scale sets can be used. Only virtual machines within the same scale set have Infiniband connectivity. Use the infiniband_network parameter to ensure that all nodes for a bursting request get deployed into the same scaleset.

When bursting Infiniband nodes, the OS image associated with the bursting scenario defined by <API key> must contain everything that is needed to use Infiniband.

To create a new scaleset , set the value of the infiniband_network parameter to "*new*":

```
"deployable_id":"Standard_H16mr", "infiniband_network": "new"
```

To use an existing scale set, set the value of the infiniband_network parameter to "*auto*", which will re-use the existing scale sets as long as all requested nodes fit inside the scale set. If they do not, a new scale set is created for the requested nodes.

```
"deployable_id":"Standard_H16mr", "infiniband_network": "auto"
```

Use the command to display the status of the cloud bursting activity to get information about a node's scaleset. For example:

```
{ ...,
    "scaleset": {"nr": "2", "name": "pclmDEVvhoiAACrxTOHioWSkwg"}, ... }
```

Nodes with the same name value are in the same scaleset.

11.1.2.9 Bursting Asynchronously

Sometimes it can take several minutes to burst a cloud node. If you do not want to wait for the bursting command to complete, then eliminate the --wait option and provide a unique request identifier. The bursting call will return instantaneously. For example:

```
REQUEST_ID=$(uuidgen)
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper scenario burst --request-id $REQUEST_ID
    '{"mom":[{"deployable_id": "<instance type>", "count":"<burst count>"}]}'
```

Using the request identifier, poll the list of notifications for the bursting operation to determine its status.

```
pclm --raw --api-endpoint==http://<PBS Cloud host name or IP address>:<PBS Cloud
    port>/pbspro-cloud/ --api-key <API key>
notif thread list --request-id $REQUEST_ID --expand
```

Output from the command will look something like this:

```
[
    {
        "title": "Workflow bootstrapper.deploy_deployables",
        "created_at": "2019-10-01T09:02:57.383000+00:00",
        "open": false,
        "related": [ ... ],
        "notifications": [
            ... ,
            {
                "notification_id": "5d93170000b64a0001c4fe00",
                "sender": "executor",
                "message": "Workflow \"deploy_deployables\" execution succeeded",
                "timestamp": "2019-10-01T09:06:08.752000+00:00",
                "content": {
                    "workflow": "deploy_deployables",
                    "workflow_name": "bootstrapper.deploy_deployables",
                    "machine_ids": [
                        "a8d1681f-173d-4476-9d52-05e7e1b405d5",
                        "535ec0fc-f6f4-4b51-8f9d-abb13984f42d"
                    ],
                    "state": "SUCCEEDED",
                    "machines": [ ... ],
                    ... ,
                },
                ...
            }
        },
    ],
    ...
}
```

When a final message of success or failure is received, the open parameter is set to "false". There can be many notifications so be sure to read the last one (the latest timestamp is usually the last one in the list). The state of the notification content can be SUCCEEDED, RUNNING or FAILED.

11.1.2.10 Bursting Preemptable Instances

You can use the preemptable parameter to indicate that the cloud node is preemptable (currently implemented for AWS. AWS refers to this as spot instances). This parameter takes a Boolean value.

The bursting request must either contain only preemptable instances or only non-preemptable instances.

The bursting scenario associated with the <API key> must have spot instances enabled.

```
pclm --api-endpoint=http://<PBS Cloud host name or IP address>:<PBS Cloud port>/pbspro-cloud/
    --api-key <API key> bootstrapper --wait scenario burst "mom":[{"deployable_id":
    "c3.xlarge","count":"10","preemptable": true}]
```

Possible errors that may occur (returned as JSON, when the --raw option is used):

```
The current spot instance price for Instance Type xx in region yy is higher than the limit
    configured in the Bursting Scenario zz
The Instance Type xx has not been configured in the Bursting Scenario to be used for spot
    instances.
```

11.2 Command Reference and Sample Output for pkr

A kard defines all the container images and container services you need for PBS Cloud. Each kard is specific to its version of PBS Cloud. This version of PBS Cloud comes packaged with the PBS Cloud kard already fully defined. The PBS administrator is not expected to make any changes to the kard.

We use pkr to manage all the containers in the current kard. In this context that means all the services around the Cloud Bursting feature in PBS Professional. Each service runs in its own container.

To start all pkr services:

```
pkr start
```

To stop all pkr services:

```
pkr stop
```

To list all pkr services:

```
pkr ps
```

11.2.1 Sample pkr Output on Startup

```
[root@myhost ~]# pkr start
Starting postgres      ... done
Starting mongodb       ... done
Starting cadvisor      ... done
Starting loki          ... done
Starting fluentd       ... done
Starting prometheus           ... done
Starting fluentd              ... done
Starting bootstrapper-worker  ... done
Starting bootstrapper-worker1 ... done
Starting grafana              ... done
Starting guardian             ... done
Starting notification-center  ... done
Starting websocket-bridge     ... done
Starting pacioli              ... done
Starting keeper               ... done
Starting hype                 ... done
Starting hubble               ... done
Starting mistral-api          ... done
Starting ui                   ... done
Starting executor-api         ... done
Starting mistral-executor     ... done
Starting cloudflow            ... done
Starting bootstrapper-api     ... done
```

11.2.2 Sample pkr Output on Stop

```
[root@myhost ~]# pkr stop
Stopping bootstrapper-api     ... done
Stopping cloudflow            ... done
Stopping ui                   ... done
Stopping mistral-executor     ... done
Stopping executor-api         ... done
Stopping keeper               ... done
Stopping notification-center  ... done
Stopping pacioli              ... done
Stopping websocket-bridge     ... done
Stopping hype                 ... done
Stopping mistral-api          ... done
Stopping hubble               ... done
Stopping guardian             ... done
Stopping grafana              ... done
Stopping bootstrapper-worker1 ... done
Stopping bootstrapper-worker  ... done
Stopping prometheus           ... done
Stopping fluentd              ... done
Stopping node-exporter        ... done
Stopping postgres             ... done
Stopping rabbitmq             ... done
Stopping loki                 ... done
Stopping mongodb              ... done
Stopping cadvisor             ... done
```

11.2.3 Sample pkr Output while Running

A missing IP address indicates that a service is unhealthy, although an IP address does not guarantee health. System is running:

```
[root@myhost ~]# pkr ps
[root@pm-lwood-testbed current]# pkr ps
- cadvisor: 172.18.0.4
- loki: 172.18.0.6
- fluentd: 172.18.0.9
- mongodb: 172.18.0.3
- node-exporter: 172.18.0.5
- postgres: 172.18.0.2
- prometheus: 172.18.0.8
- grafana: 172.18.0.12
- rabbitmq: 172.18.0.7
- guardian: 172.18.0.13
- pacioli: 172.18.0.14
- notification-center: 172.18.0.16
- mistral-api: 172.18.0.20
- mistral-executor: 172.18.0.23
- keeper: 172.18.0.17
- hype: 172.18.0.18
- hubble: 172.18.0.19
- executor-api: 172.18.0.22
- cloudflow: 172.18.0.24
- bootstrapper-worker1: 172.18.0.10
- bootstrapper-worker: 172.18.0.11
- bootstrapper-api: 172.18.0.25
- websocket-bridge: 172.18.0.15
- ui: 172.18.0.21
```

11.2.4 Sample pkr Output while Stopped

System is stopped:

```
[root@myhost ~]# pkr ps
- cadvisor: stopped
- loki: stopped
- fluentd: stopped
- mongodb: stopped
- node-exporter: stopped
- postgres: stopped
- prometheus: stopped
- grafana: stopped
- rabbitmq: stopped
- guardian: stopped
- pacioli: stopped
- notification-center: stopped
- mistral-api: stopped
- mistral-executor: stopped
- keeper: stopped
- hype: stopped
- hubble: stopped
- executor-api: stopped
- cloudflow: stopped
- bootstrapper-worker1: stopped
- bootstrapper-worker: stopped
- bootstrapper-api: stopped
- websocket-bridge: stopped
```

Index