

# Oracle Cloud Infrastructure Compute Plugin

## Plugin Information

View Oracle Cloud Infrastructure Compute [on the plugin site](#) for more information.

[Oracle Cloud Infrastructure Compute Plugin](#) allows users to access and manage cloud resources on the Oracle Cloud Infrastructure (OCI) from Jenkins. A Jenkins master instance with Oracle Cloud Infrastructure Compute Plugin can spin up OCI Instances (slaves or agents) on demand within OCI, and remove the Instances and free its resources automatically once the Job completes.

- [Features](#)
- [Prerequisites](#)
- [Compatibility](#)
- [Installation](#)
  - [Using the "Plugin Manager" in the web UI](#)
  - [Using the Jenkins CLI install-plugin command](#)
  - [Copying the .hpi file to the JENKINS\\_HOME/plugins directory](#)
- [Building the Plugin from Repository](#)
  - [Installing OCI Java SDK](#)
  - [Compile the Plugin](#)
  - [Install the Plugin](#)
- [Configuration](#)
  - [Add New Cloud](#)
  - [Add New Template](#)
- [Licensing](#)
- [Changelog](#)
- [Contributing](#)

## Features

[Oracle Cloud Infrastructure Compute Plugin](#) allows users to access and manage cloud resources on the Oracle Cloud Infrastructure (OCI) from Jenkins. A Jenkins master instance with Oracle Cloud Infrastructure Compute Plugin can spin up OCI Instances (slaves or agents) on demand within OCI, and remove the Instances and free its resources automatically once the Job completes.

After installing the Plugin, you can add OCI Clouds and Templates with your required OCI Instance configuration. The Template will have a Label that you can use in your Jenkins Job. Multiple Templates are supported. The Template options include Labels, Domains, Credentials, Shapes, Images, Virtual Cloud Network, Template Instance Cap, etc. After your Jenkins Job completes its work, the OCI Instance is cleanly removed and resources are released back to the OCI pool.

View Oracle Cloud Infrastructure Compute Plugin page on the [plugins.jenkins.io](https://plugins.jenkins.io) site for more information.

## Prerequisites

1. Oracle Cloud Account. To sign up, visit [Oracle Cloud](#).
2. Jenkins installed with JDK 8 or higher.

## Compatibility

1. Plugin is tested on minimum Jenkins version 1.625.3, it may not work with versions early than 1.554.
2. For Jenkins versions between 1.554 and 1.625, pre-install 'bouncycastle API Plugin' plugin.

## Installation

There are a number of ways to install the Oracle Cloud Infrastructure Compute Plugin.

1. Using the "Plugin Manager" in the web UI.
2. Using the Jenkins CLI install-plugin command.
3. Copying the .hpi file to the JENKINS\_HOME/plugins directory.

### Using the "Plugin Manager" in the web UI

The simplest and most common way of installing plugins is through the Manage Jenkins > Manage Plugins view, available to administrators of a Jenkins environment.

To install the Plugin in Jenkins:

1. Click on **Manage Jenkins** in Home
2. Click **Manage Plugins**
3. Click **Available** tab
4. Search for "Oracle Cloud Infrastructure Compute Plugin" or "oracle-cloud-infrastructure-compute"

5. Click **Install**
6. Restart Jenkins

## Using the Jenkins CLI install-plugin command

Administrators may also use the [Jenkins CLI](#) which provides a command to install plugins.

The Oracle Cloud Infrastructure Compute Plugin is available via Jenkins Update Center or can be installed manually.

```
java -jar jenkins-cli.jar -s http://localhost:8080/ install-plugin SOURCE ... [-deploy] [-name VAL] [-restart]
```

Installs a plugin either from a file, an URL, or from update center.

```
SOURCE      : If this points to a local file, that file will be installed. If
              this is an URL, Jenkins downloads the URL and installs that as a
              plugin. Otherwise the name is assumed to be the short name of the
              plugin i.e. "oracle-cloud-infrastructure-compute", and the
              plugin will be installed from the update center.
-deploy     : Deploy plugins right away without postponing them until the reboot.
-name VAL   : If specified, the plugin will be installed as this short name
              (whereas normally the name is inferred from the source name
              automatically).
-restart    : Restart Jenkins upon successful installation.
```

Link to latest .hpi version can be found [here](#).

## Copying the .hpi file to the JENKINS\_HOME/plugins directory

Using the .hpi file that has been explicitly downloaded by a systems administrator, the administrator can manually copy the downloaded .hpi file into the JENKINS\_HOME/plugins directory on the Jenkins master. Link to latest .hpi version can be found [here](#).

The master will need to be restarted before the plugin is loaded and made available in the Jenkins environment.

## Building the Plugin from Repository

Jenkins plugins are packaged as self-contained .hpi files, which have all the necessary code, images, and other resources which the plugin needs to operate successfully.

If desired, you can build the Oracle Cloud Infrastructure Compute Plugin .hpi from the source code, and then install the .hpi file in Jenkins.

To build the .hpi file, OCI Java SDK is required. OCI Java SDK is currently not published to Maven center so first install OCI Java SDK to the local Maven repository. Steps are outlined below.

Refer to OCI Java SDK licensing [here](#).

## Installing OCI Java SDK

```
$ git clone https://github.com/oracle/oci-java-sdk
$ cd oci-java-sdk
$ mvn compile install
```

## Compile the Plugin

1. git clone repo
2. Update pom.xml with OCI Java SDK you have installed

```
...
    <oci-java-sdk.version>1.3.1</oci-java-sdk.version>
...
```

3. Compile and Install package

```
$ mvn package
```

## Install the Plugin

A logged-in Jenkins administrator may upload the file from within the web UI.

1. Navigate to the Manage Jenkins > Manage Plugins page in the web UI.
2. Click on the Advanced tab.
3. Choose the .hpi file under the Upload Plugin section.
4. Upload the plugin file.

or

The System Administrator can copy the .hpi file into the JENKINS\_HOME/plugins directory on the Jenkins master.

The master will need to be restarted before the plugin is loaded and made available in the Jenkins environment.

## Configuration

### Add New Cloud

1. From Jenkins Server console, click Manage Jenkins > Configure System
2. In Cloud section, click **Add a new cloud** and select **Oracle Cloud Infrastructure Compute**. Enter credentials to access your OCI account. You can create multiple Clouds.

Oracle Cloud Infrastructure Compute

This plugin allows you to run dynamic slaves in the Oracle Cloud Infrastructure(OCI) Compute environment.

Name	OCI
Fingerprint	8f:80:b8:7f:64:ab:a6:12:64:e3:aa:bb:ac:cb:ad:3f
API Key	-----BEGIN RSA PRIVATE KEY----- MIIEpQAA... BB CC
PassPhrase	
Tenant Id	ocid1.tenancy.oc1..aaaaaaaabbbbbbbbbbbbbbcccccccccccccccccc
User Id	ocid1.user.oc1..aaaaaaaabbbbbbbbbbbbbbcccccccccccccccccc
Region	us-phoenix-1

- **Name** - Provide a name for this OCI Compute Cloud.
  - **Fingerprint** - Enter the Fingerprint from your OCI API Signing Key. If you do not have one, it can be left blank. For more information see [Security Credentials](#).
  - **API Key** - Enter the OCI API Signing Private Key. For more information see [Security Credentials](#).
  - **User Id** - Enter your User OCID.
  - **Tenant Id** - Enter your Tenant OCID.
  - **Region** - Enter Region, for example, us-phoenix-1.
- Click **Advanced** for more options.

Instance Cap	
Max number of async threads	50

**Instance Cap** - Enter a number to limit the maximum number of instances that can be created for this Cloud configuration. Leave this field empty to remove a cap.

**Max number of async threads** - The max number of async threads to use to load the Templates configuration. Consider reducing this value for Cloud configurations with a large number of Templates and if some values fail to load due to OCI API limit being exceeded. In this case the logs will show "User-rate limit exceeded" errors.

- Click **Test Connection** to verify that Jenkins can successfully connect to the Oracle Cloud Infrastructure Service.

### Add New Template

1. Click **Add** in **Instance Templates** section to add the OCI configuration. You can add multiple Templates to a Cloud configuration.



Remote FS Root	<input type="text"/>
Remote SSH User	<input type="text" value="opc"/>
Instance Creation Timeout	<input type="text" value="1200"/>
Instance SSH Connection Timeout	<input type="text" value="1200"/>
Idle Termination Minutes	<input type="text" value="0"/>
Number Of Executors	<input type="text" value="1"/>
Init Script	<input type="text" value="sudo yum -y install java"/>
Init Script Timeout	<input type="text" value="900"/>
Template Instance Cap	<input type="text" value="1"/>

- **Remote FS root** - Dedicated directory for Jenkins agent in instance.
- **Remote SSH user** - ssh user used for Jenkins master to access instance. The ssh user should have written permission on Remote FS root directory.
- **Instance Creation Timeout** - Number of seconds to wait for instance to reach state "ready", default value is 300.
- **Instance SSH Connection Timeout** - Number of seconds to wait for instance from state "ready" to be able to ssh connect from Jenkins master. Default value is 60.
- **Idle Termination Minutes** - Number of minutes for Jenkins to wait before deleting and completely removing an idle instance. A value of 0 (or an empty string) indicates that instance should never be stopped/deleted.
- **Number of Executors** - Number of concurrent builds that Jenkins can perform. Value should be at least 1.
- **Init Script** - You can define several lines of shell based commands to configure the instance (one-time) before the Jenkins Job runs. For example, if the image selected does not have Java pre-installed, you can add command "sudo yum -y install java"
- **Init Script Timeout** - Number of seconds to wait for the completion of Init Script. Default value is 120 seconds.
- **Template Instance Cap** - Places a limit on the number of OCI Instances that Jenkins may launch from this Template. Leave this field empty to remove the Template Instance Cap.

5. Click **Save** or **Apply**

## Licensing

Copyright (c) 2018, Oracle and/or its affiliates. All rights reserved.

This plugin is licensed under the Universal Permissive License 1.0

This software is dual-licensed to you under the Universal Permissive License (UPL) and Apache License 2.0.

See [LICENSE.txt](#) for more details.

## Changelog

For Changelog, please refer to [CHANGELOG.md](#).

## Contributing

Oracle Cloud Infrastructure Compute Plugin is an open source project. See [CONTRIBUTING.md](#) for more details.

Oracle gratefully acknowledges the contributions to Oracle Cloud Infrastructure Compute Plugin that have been made by the community.