

Certification Training Install Lab Environment

Table of Contents

Introduction.....	1
Local VM: Vagrant + Virtualbox	1
Common Requirements	1
Hardware Requirements	2
Software Requirements	2
Getting Lab Resources.....	3
Starting the Lab Environment	3
Accessing the Lab Environment	5
Setting up CloudBees Jenkins licenses.....	5
Troubleshooting	5
General workflow	5
Technical Troubleshooting.....	6

Introduction

This document explains how to install and start your CloudBees Lab Environment.

Please follow all the steps carefully, before running any Lab exercise.

Local VM: Vagrant + Virtualbox

A Virtual Machine (VM) will be used for hosting your **Lab Environment**:

- It does not break any local environment
- It does not depend on the host OS you are using
- It is portable with the same behavior for everyone

This VM runs using the VirtualBox hypervisor, and is managed and automated by Vagrant (see requirements below).

Both of those tools are Open Source, free and multi-platforms, but they require:

- Having admin rights on your computer for installation only
- Your computer must not be already a Virtual Machine. Nested virtualization is not supported.

Common Requirements

- An HTML5 compliant Web Browser is required: **Mozilla Firefox**, **Google Chrome**, **Microsoft Edge**, **Apple Safari**, **Opera**

IMPORTANT

Internet Explorer is not supported

- The following ports must be allowed access to your instance's domain (which is `localhost`):
 - 5000
 - 5001
 - 5002
 - 5003
 - 20000
 - 30001
 - 30002
- The following protocols must be allowed by any antivirus/firewall software:
 - HTTP
 - HTTPS
 - Websockets
 - Some antivirus software like **Kasperky** and **McAfee** might block websocket silently
 - You can test websockets from this page: [WebSockets Test](#)

- For more about security and websockets: [Testing WebSockets on OWASP](#)
- Even if the training lab is running in offline mode, an open Internet access is **recommended**
 - HTTP Proxy can **only** be configured for Jenkins operations

Hardware Requirements

Your machine must meet the following **hardware** requirements:

- Intel 64 Bits Dual-Core CPU compliant (Intel Pentium/i3 at least)
- 6GB of RAM (the VM will allocate 4GB for itself)
- 20GB of free space on your local hard drive
- One OS from this list:
 - Windows >= 8.1
 - Mac OS >= 10.10
 - Linux "classic" distribution (Ubuntu >= 12.04, Debian >= Jessie, RHEL >= 6)
- The "Virtualization instructions" of your CPU must be enabled (Intel VT-x or AMD SVM)
 - More information here: <https://forums.virtualbox.org/viewtopic.php?f=6&t=58820>
 - Intel official VT page: <http://www.intel.com/content/www/us/en/virtualization/virtualization-technology/intel-virtualization-technology.html>

Software Requirements

Your machine must meet the following **software** requirements:

- For **All** OSes, download and install the latest (64 Bits) versions of:
 - **VirtualBox 5.2.6** (An Open Source Hypervisor from Oracle):
 - For Windows: <https://download.virtualbox.org/virtualbox/5.2.6/VirtualBox-5.2.6-120293-Win.exe>

IMPORTANT

If you have HyperV installed, VirtualBox may throw some errors with the code **VERR_VMX_NO_VMX**.

In this case ([Stack Overflow - Vagrant up - VBoxManage.exe error: VT-x is not available \(VERR_VMX_NO_VMX\)](#)), please disable HyperV temporarily : ([Disable HyperV](#))

```
bcdedit /set hypervisorlaunchtype off
```

and reboot

- For MacOS: <https://download.virtualbox.org/virtualbox/5.2.6/VirtualBox-5.2.6-120293-OSX.dmg>
- For Linux: https://www.virtualbox.org/wiki/Linux_Downloads
- **Vagrant 2.0.1** (An Open Source VM manager):

- For Windows **64 bits**: https://releases.hashicorp.com/vagrant/2.0.1/vagrant_2.0.1_x86_64.msi
- For MacOS: https://releases.hashicorp.com/vagrant/2.0.1/vagrant_2.0.1_x86_64.dmg
- For Linux Debian/Ubuntu (64 Bits): https://releases.hashicorp.com/vagrant/2.0.1/vagrant_2.0.1_x86_64.deb
- For Linux Centos (64 Bits): https://releases.hashicorp.com/vagrant/2.0.1/vagrant_2.0.1_x86_64.rpm
- For **Windows** only, download latest version of [Git for Windows](#)

TIP

Git for Windows provides a bash-compliant shell and OpenSSH client

Getting Lab Resources

After installing the software requirements:

- Start by downloading the **ZIP** archive for your training:
 - Use the left-section of the Lesson named **Summary**, the **ZIP** file is attached as downloadable content:
 - This archive contains your VM image and automated settings

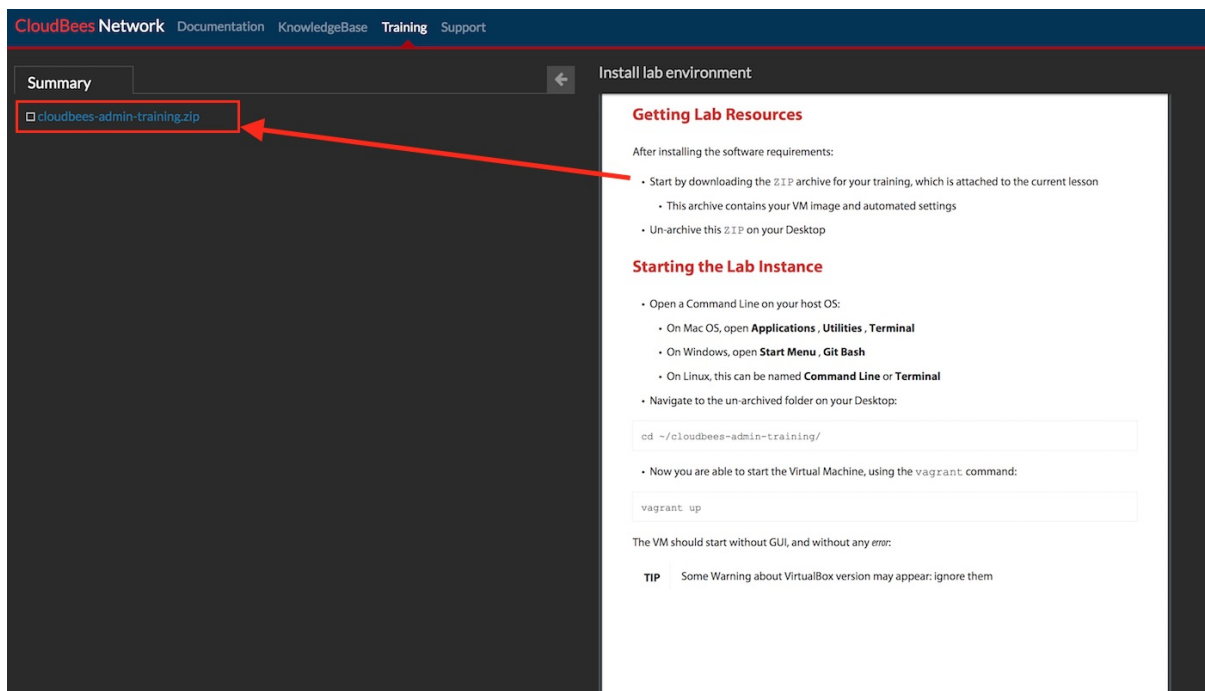


Figure 1. Download the ZIP archive from SkillJar

- Un-archive this **ZIP** archive on your Desktop. A folder named **cloudbees-training-certification** will show up.

Starting the Lab Environment

- Open a Command Line on your host OS:
 - On Mac OS, open **Applications** , **Utilities** , **Terminal**
 - On Windows, open **Start Menu** , **Git Bash**

- On Linux, this can be named **Command Line** or **Terminal**

TIP The command line is required to start the Virtual Machine without having to care to any specific configuration.

- Using the command line `cd`, navigate to the un-archived folder that should be located on your Desktop:

```
cd ~/Desktop/cloudbees-training-certification/
```

TIP

- The `~` special character means "Full path to the user home folder"
- Desktop may vary depending on your operating system: can be lower case, or localized in your operating system's language.

- Use the command line `ls` to check the content of this directory. We need to have a file named **Vagrantfile** here:

```
ls -l
Vagrantfile
```

- Now you are able to start the Virtual Machine, using the **vagrant** command:

```
vagrant up
```

The VM should start without a GUI, and without any **error**:

TIP If some warnings about VirtualBox version appears, ignore them as long as everything is working well.

```
MacBook-Pro-de-Dadou:cloudbees-admin-training dadou$ vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Box 'cb-admin-training-1.2.0-SNAPSHOT' could not be found. Attempting to find and install...
default: Box Provider: virtualbox
default: Box Version: >= 0
==> default: Box file was not detected as metadata. Adding it directly...
==> default: Adding box 'cb-admin-training-1.2.0-SNAPSHOT' (v0) for provider: virtualbox
default: Unpacking necessary files from: file:///Users/dadou/workspace/trainings-cloudbees/training-admin/labs-vm/cloudbees-admin-training/cb-admin-training-1.2.0-SNAPSHOT.box
==> default: Successfully added box 'cb-admin-training-1.2.0-SNAPSHOT' (v0) for 'virtualbox'!
==> default: Importing base box 'cb-admin-training-1.2.0-SNAPSHOT'...
==> default: Matching MAC address for NAT networking...
==> default: Setting the name of the VM: cloudbees-admin-training_default_1491332749574_5874
==> default: Fixed port collision for 22 => 2222. Now on port 2200.
==> default: Clearing any previously set network interfaces...
==> default: Preparing network interfaces based on configuration...
default: Adapter 1: nat
==> default: Forwarding ports...
default: 5000 (guest) => 5000 (host) (adapter 1)
default: 5001 (guest) => 5001 (host) (adapter 1)
default: 5002 (guest) => 5002 (host) (adapter 1)
default: 5003 (guest) => 5003 (host) (adapter 1)
default: 22 (guest) => 2200 (host) (adapter 1)
==> default: Running 'pre-boot' VM customizations...
==> default: Booting VM...
==> default: Waiting for machine to boot. This may take a few minutes...
default: SSH address: 127.0.0.1:2200
default: SSH username: butler
default: SSH auth method: private key
default: Warning: Remote connection disconnect. Retrying...
==> default: Machine booted and ready!
==> default: Checking for guest additions in VM...
default: No guest additions were detected on the base box for this VM! Guest
default: additions are required for forwarded ports, shared folders, host only
default: networking, and more. If SSH fails on this machine, please install
default: the guest additions and repackage the box to continue.
default:
default: This is not an error message; everything may continue to work properly,
default: in which case you may ignore this message.
MacBook-Pro-de-Dadou:cloudbees-admin-training dadou$
```

Figure 2. Vagrant starting the VM for you

- You need to be able to stop and start the Virtual Machine whenever you want. Let's do it now:
 - From the cloudbees-training-certification folder that contains a **Vagrantfile**:
 - Stop the VM "gracefully" with the vagrant "halt" command:

```
vagrant halt
```

TIP

Once the VM is in the **stopped** state, you can safely do anything else, like stopping your computer

- Start again the Virtual Machine:

```
vagrant up
```

TIP

Any **Vagrant** command can be used here. For more informations, please check [Vagrant Documentation](https://www.vagrantup.com/docs/cli/) - <https://www.vagrantup.com/docs/cli/>

Accessing the Lab Environment

Your **Lab Environment** provides a Home Page to use as the entrypoint.

- This page is available on your web browser at this URL: [Lab Home Page](#)

IMPORTANT

Unless specified differently, any authentication asked by any service in the Lab Environment uses the following:

- Username: butler
- Password: butler

- You will see an HTML page that lists the services hosted on your Lab Environment.
- Each service will be detailed in the next steps.

Setting up CloudBees Jenkins licenses

Please follow the lesson **Get a Student License** to know how to ask for a CloudBees Jenkins training license.

Troubleshooting

General workflow

If you face any issue with the lab during this course, please read this troubleshooting guide first.

If you still cannot use the lab environment, depending on your training type:

- "Trainer led": please ask your trainer for help.
- "Self Paced": please open a ticket by following the instructions found in the "Open a training ticket" at the start of the course.

TIP

Always double-check your settings: peer review is the best !

Technical Troubleshooting

- If an error was raised during the initial VM startup:
 - If the error is related to **GuestAdditions** like the one below:

```
==> default: Machine booted and ready!
[default] GuestAdditions versions on your host (5.1.8) and guest
(5.0.18_Ubuntu r106667) do not match.
...
The following SSH command responded with a non-zero exit status.
Vagrant assumes that this means the command failed!

apt-get update
...
```

Then remove the plugin **vagrant-vbguest**, by using the command

```
vagrant plugin uninstall vagrant-vbguest
```

- If the error is related to **VT-x is not available**

```
...
Stderr: VBoxManage.exe: error: VT-x is not available (VERR_VMX_NO_VMX)
```

Make sure you disable the HyperV service as stated in the 'Software Requirements' of this document

- Is your VM started ?
 - Open VirtualBox GUI and check the state.
 - With you command line, use **vagrant status** within your labs directory.
 - On your process manager, verify that you have a **VBoxHeadless** process.
- Is your VM reachable with SSH ?
 - Is Vagrant aware of port forwarding (using **vagrant port**) ?
 - In the VirtualBox GUI, do you see the port forwarding ?
 - Do you have any firewall rule that would block any traffic on your **localhost** (lo, loopback, etc.) interface, on the forwarded port (2222 generally)?
- When stuck, always try rebooting the VM one time
- If you need to submit an issue (Self Paced training only), try to run your latest **vagrant** command in debug mode (see example below), and copy-paste the result in a text file or in <https://gist.github.com/>


```
VAGRANT_LOG=debug vagrant up
```