

Red Hat Container Development Kit

Introducing Red Hat Container Development Kit

Red Hat Container Development Kit is a platform for developing containerized applications on a single, personal system. It enables developers to quickly and easily set up an environment for developing and testing containerized applications on the Red Hat Enterprise Linux platform.

CDK:

- Provides a personal Container Development Environment you can install on your own laptop, desktop, or server system. The Container Development Environment is provided in the form of a Red Hat Enterprise Linux virtual machine. The Container Development Environment itself can also be installed in a virtual machine.
- Includes the same container-development and runtime tools used to create and deploy containers for large data centers.
- CDK version 3 offers an easy installation method based on the [minishift tool](#).
- Runs on Microsoft Windows, macOS, and Linux operating systems as a Linux virtual machine, thus allowing developers to use their favorite platform while producing applications ready to be deployed in the Red Hat Enterprise Linux ecosystem.

CDK is a part of the [Red Hat Developers](#) program, which provides tools, resources, and support for developers who wish to utilize Red Hat solutions and products to create applications, both locally and in the cloud. For additional information and to register to become a part of the program, visit [developers.redhat.com](#).

Preparing to Install Container Development Kit

Overview

Container Development Kit 3.2 is based on [Minishift](#) 1.7.0. Using a single executable file, the `minishift` command deploys CDK as a Red Hat Enterprise Linux virtual machine, running OpenShift (which includes Kubernetes, the Docker service, and other container development and deployment software).

The CDK setup procedure can, and should, be run as a regular user that has special permission to launch virtual machines. In the procedure, you will see how to assign that permission, along with ways to configure your hypervisor and command shell to start and effectively interact with CDK.

The following section describes how to install CDK and the required dependencies.

Prerequisites

CDK requires a hypervisor to start the virtual machine on which the OpenShift cluster is provisioned. Verify that the hypervisor of your choice is installed and enabled on your system before you set up CDK. Once the hypervisor is up and running, additional setup is required for CDK to work with that hypervisor (as described in the coming setup procedure).

Depending on your host operating system, you have the choice of the following hypervisors:

macOS

- [xhyve](#) (default)
- [VirtualBox](#)

Red Hat Enterprise Linux, Fedora, CentOS or other Linux system

- [KVM](#) (default)
- [VirtualBox](#)

Windows

- [Hyper-V](#) (default)
- [VirtualBox](#)

Refer to the documentation for each hypervisor to determine the hardware and operating system versions needed to run that hypervisor.

Understanding CDK Installation

These are the basic steps for setting up CDK on your personal laptop or desktop system:

1. Set up your virtualization environment
2. Download CDK software for your operating system from the [Red Hat Container Development Kit Download Page](#)
3. Set up CDK
4. Start CDK
5. Configure CDK so you can use it efficiently

CDK Installation

The following steps describe how to prepare your virtualization environment (hypervisor) for CDK, download CDK software, set up CDK, and start using it.

Step 1: Set up your virtualization environment

Follow the appropriate procedure to set up virtualization for your particular operating system and hypervisor. CDK uses Docker Machine and its driver plug-in architecture to provide a consistent way to manage the OpenShift VM.

Some hypervisors require manual installation of the driver plug-in. CDK embeds VirtualBox drivers so no additional steps are required to configure the driver. However, a `minishift` command option (`--vm-driver virtualbox`) will be required later to identify VirtualBox to CDK.

Set up hypervisor on Red Hat Enterprise Linux

Choose between KVM (default) and VirtualBox for your hypervisor. Manual driver setup is required for KVM. The driver is automatically configured if you install VirtualBox. However, a `minishift` command will be required later to identify VirtualBox to CDK.

On Red Hat Enterprise Linux with KVM virtualization: CDK is currently tested against `docker-machine-driver-kvm` version 0.7.0. Follow these steps to install the KVM driver and configure your user account to use the **libvirtd** service.

1. Install the KVM binary as follows:

```
sudo curl -L https://github.com/dhiltgen/docker-machine-kvm/releases/download/v0.7.0/docker-machine-driver-kvm
```

2. Make the binary executable as follows:

```
sudo chmod +x /usr/local/bin/docker-machine-driver-kvm
```

For more information, see the GitHub documentation of the [Docker Machine KVM driver](#).

3. Install **libvirt** and **qemu-kvm** on your system:

```
sudo yum install libvirt qemu-kvm
```

4. Add yourself to the *libvirt* group so that you do not need to use `sudo`:

```
sudo usermod -a -G libvirt <username>
```

5. Update your current session to apply the group change:

```
newgrp libvirt
```

6. Start the **libvirtd** service:

```
sudo systemctl start libvirtd
sudo systemctl enable libvirtd
```

Set up hypervisor on macOS

Choose between xhyve (default) and VirtualBox for your hypervisor. Manual driver setup is required for xhyve. The driver is automatically configured if you install VirtualBox. However, a `minishift` command option (`--vm-driver virtualbox`) will be required later to identify VirtualBox to CDK.

On macOS with xhyve virtualization:

CDK on macOS with xhyve virtualization is currently tested against `docker-machine-driver-xhyve` version 0.3.3. To manually install the xhyve driver, you need to download and install the `docker-machine-driver-xhyve` binary and place it in a directory which is on your `PATH` environment variable. The directory */usr/local/bin* is most likely a good choice, since it is the default installation directory for Docker Machine binaries.

The following steps explain the installation of the `docker-machine-driver-xhyve` binary to the */usr/local/bin/* directory:

1. Download the `docker-machine-driver-xhyve` binary using:

```
sudo curl -L https://github.com/zchee/docker-machine-driver-xhyve/releases/download/v0.3.3/docker-machine-driver-xhyve
```

2. Enable root access for the `docker-machine-driver-xhyve` binary and add it to the default `wheel` group:

```
sudo chown root:wheel /usr/local/bin/docker-machine-driver-xhyve
```

3. Set owner User ID (SUID) for the binary as follows:

```
sudo chmod u+s,+x /usr/local/bin/docker-machine-driver-xhyve
```

NOTE

The downloaded `docker-machine-driver-xhyve` binary is compiled against a specific version of macOS. It is possible that the driver fails to work after a macOS version upgrade. In this case you can try to compile the driver from source:

```
go get -u -d github.com/zchee/docker-machine-driver-xhyve
cd $GOPATH/src/github.com/zchee/docker-machine-driver-xhyve

# Install docker-machine-driver-xhyve binary into /usr/local/bin
make install

# docker-machine-driver-xhyve needs root owner and uid
sudo chown root:wheel /usr/local/bin/docker-machine-driver-xhyve
sudo chmod u+s /usr/local/bin/docker-machine-driver-xhyve
```

For more information, see the [xhyve driver](#) documentation on GitHub.

Set up hypervisor on Windows

Choose between Hyper-V (default on Windows 10) and VirtualBox (Windows 7 or Windows 10) for your hypervisor. Manual driver setup is required for Hyper-V. The driver is automatically configured if you install VirtualBox. However, a `minishift` command option (`--vm-driver virtualbox`) will be required later to identify VirtualBox to CDK.

On Windows with Hyper-V virtualization:

1. Install [Hyper-V](#).
2. Add an [External Virtual Switch](#). Verify that you pair the virtual switch with a network card (wired or wireless) that is connected to the network.
3. Set the environment variable `HYPERV_VIRTUAL_SWITCH` to the name of the external virtual switch you want to use for CDK. For more information, see [driver-specific environment variables](#).

For example, on Command Prompt use:

```
C:\> set HYPERV_VIRTUAL_SWITCH=External (Wireless)
```

Note that using quotes in Command Prompt results in the following error:

```
C:\> set HYPERV_VIRTUAL_SWITCH="External (Wireless)"
Error creating the VM. Error with pre-create check: "vswitch \"\\\"External (Wireless)\\\"\" not found"
```

However, on PowerShell you need to use the quotes:

```
PS C:\> $env:HYPERV_VIRTUAL_SWITCH="External (Wireless)"
```

Step 2: Download CDK Software

Before you can download CDK software, you need to either register with the [Red Hat Developer Program](#) site or login to the Red Hat customer portal with Red Hat subscription credentials. Then go to one of the following two sites and download the software associated with your operating system:

- [Red Hat Developer Program CDK Download Page](#)
- [Red Hat Customer Portal CDK Download Page](#)

Copy the downloaded minishift file to a directory in your PATH and make it executable. The downloaded executable is named cdk-3.2.0-1-minishift-darwin-amd64 (for macOS), cdk-3.2.0-1-minishift-linux-amd64 (for Linux) or cdk-3.2.0-1-minishift-windows-amd64.exe (for Windows). Assuming the executable is in the **~/Downloads** directory, follow the procedure for your operating system:

For Red Hat Enterprise Linux:

```
mkdir -p ~/bin
cp ~/Downloads/cdk-3.2.0-1-minishift* ~/bin/minishift
chmod +x ~/bin/minishift
export PATH=$PATH:$HOME/bin
echo 'export PATH=$PATH:$HOME/bin' >> ~/.bashrc
```

For macOS:

```
mkdir -p ~/bin
cp ~/Downloads/cdk-3.2.0-1-minishift* ~/bin/minishift
chmod +x ~/bin/minishift
export PATH=$PATH:$HOME/bin
echo export PATH=$PATH:$HOME/bin >> ~/.bash_profile
```

For Windows:

Create the desired directory and copy the downloaded CDK binary to the directory, renaming the binary to minishift.exe. Add the directory path to the Windows PATH variable.

IMPORTANT

This directory **must** be on the **C:** drive!

If it's difficult to get minishift.exe in your PATH, you can simply run it from the current directory as ./minishift.exe (or .\minishift.exe in some Windows shells).

Step 3: Set up CDK

The minishift setup-cdk command gets and configures the components needed to run CDK on your system. By default, minishift setup-cdk places CDK content in your **~/.minishift** directory (**%USERPROFILE%\minishift** on Windows).

IMPORTANT

To use a directory other than **~/.minishift**, you must set the --minishift-home flag and MINISHIFT_HOME environment variable, as described in [Environment Variables](#).

Run the following command to set up CDK for Red Hat Enterprise Linux:

```
minishift setup-cdk
Setting up CDK 3 on host using '/home/joe/.minishift' as Minishift's home directory
Copying minishift-rhel7.iso to '/home/joe/.minishift/cache/iso/minishift-rhel7.iso'
Copying oc to '/home/joe/.minishift/cache/oc/v3.6.173.0.21/oc'
Creating configuration file '/home/joe/.minishift/config/config.json'
Creating marker file '/home/joe/.minishift/cdk'
Default add-ons anyuid, admin-user, xpaas installed
Default add-ons anyuid, admin-user, xpaas enabled
CDK 3 setup complete.
```

As you can see from the output, a **.minishift** directory is created in the user's home directory to hold various CDK components, and appropriate files and settings are stored there.

For Windows or macOS: Running the same minishift setup-cdk command on Windows and macOS results in slightly different output, based on some different components and pathnames.

Optional: Use OpenShift Container Platform 3.7

Currently, CDK cannot automatically download the oc binary for OpenShift Container Platform 3.7. However, you may manually download the oc 3.7 binary to use it. To do so, follow these steps:

- Ensure that you have run minishift setup-cdk.
- Manually download the oc binary for OpenShift Container Platform 3.7 from <https://mirror.openshift.com/pub/openshift-v3/clients/>.

- Place the downloaded oc binary in the **existing** directory for oc in *MINISHIFT_HOME/cache/oc/v3.6.173.0.21/*.
- Run CDK with the `--ocp-tag` flag with the exact version of the oc3.7 binary. Example:

```
minishift setup-cdk
curl -O https://mirror.openshift.com/pub/openshift-v3/clients/3.7.9/linux/oc.tar.gz
tar -zxvf oc.tar.gz
cp oc ~/.minishift/cache/oc/v3.6.173.0.21
minishift start --ocp-tag v3.7.9
```

NOTE

When creating new profiles, CDK extracts the v3.6.173.0.21 oc binary from minishift rather than copying the oc binary placed in the cache. You must manually add the oc 3.7 binary to the cache for each profile to use OpenShift Container Platform 3.7. Use the following for each profile making use of the oc3.7 binary:

- Set the active profile via `minishift profile set <profile-name>`.
- Follow the previous procedure using *MINISHIFT_HOME/profiles/<profile-name>/cache/oc/v3.6.173.0.21/* as the cached oc location. Example:

```
minishift profile set example
minishift setup-cdk
cp oc ~/.minishift/profiles/example/cache/oc/v3.6.173.0.21
minishift start --ocp-tag v3.7.9
```

IMPORTANT

The `--ocp-tag` flag for `minishift start` expects version information in a specific format. Ensure that you prepend the version with "v" and replace any dashes with periods for use with the `--ocp-tag` flag. For example, if your oc version is 3.7.0-0.198.0, use the following:

```
minishift start --ocp-tag v3.7.0.0.198.0
```

Step 4: Start CDK

The `minishift start` command launches CDK, which consists of a Red Hat Enterprise Linux virtual machine running OpenShift Container Platform. Follow these steps to launch CDK.

1. If you are using the default hypervisor for your operating system (KVM for Linux, xhyve for macOS, or Hyper-V for Windows), you can skip this step. If you have set up VirtualBox as your hypervisor, you need to configure the VirtualBox Driver. To switch the hypervisor used by CDK to VirtualBox, you have two choices:
 - Temporary: Add the `--vm-driver virtualbox` option to the `minishift start` command line to use VirtualBox immediately.
 - Persistent: To persistently change the hypervisor, run the `minishift config set vm-driver virtualbox` command. See the [Persistent Configuration](#) section for examples.
2. By default, `minishift start` prompts you for your Red Hat account username and password. You can enter that information or choose instead to:
 - Skip registration: Add the `--skip-registration` option to `minishift start` to not register the CDK VM.
 - Register permanently: You can export registration information to environment variables so that `minishift` picks it up automatically each time it starts as shown here:

For Red Hat Enterprise Linux:

```
export MINISHIFT_USERNAME=<RED_HAT_USERNAME>
export MINISHIFT_PASSWORD='<RED_HAT_PASSWORD>'
echo export MINISHIFT_USERNAME=$MINISHIFT_USERNAME >> ~/.bashrc
echo export MINISHIFT_PASSWORD=$MINISHIFT_PASSWORD >> ~/.bashrc
```

For macOS:

```
export MINISHIFT_USERNAME='<RED_HAT_USERNAME>'
export MINISHIFT_PASSWORD='<RED_HAT_PASSWORD>'
echo export MINISHIFT_USERNAME=$MINISHIFT_USERNAME >> ~/.bash_profile
echo export MINISHIFT_PASSWORD=$MINISHIFT_PASSWORD >> ~/.bash_profile
```

For Windows:

Using Command Prompt:

```
C:\> setx MINISHIFT_USERNAME <RED_HAT_USERNAME>
C:\> setx MINISHIFT_PASSWORD <RED_HAT_PASSWORD>
```

Using PowerShell:

```
PS> $env:MINISHIFT_USERNAME = '<RED_HAT_USERNAME>'
PS> $env:MINISHIFT_PASSWORD = '<RED_HAT_PASSWORD>'
```

3. Run `minishift start` to set up and start the virtual machine with the default configuration. See the [minishift start command reference](#) to see other options for modifying `minishift start`. Here is an example of how that would look run from a Red Hat Enterprise Linux system:

```
-- Minishift VM will be configured with ...
Memory: 4 GB
vCPUs: 2
```

```

    Disk size: 20 GB
-- Starting Minishift VM ..... OK
-- Registering machine using subscription-manager
-- Registration in progress ..... OK [42s]
-- Checking for IP address ... OK
-- Checking if external host is reachable from the Minishift VM ...
...
Extracting
Image pull complete
OpenShift server started.
The server is accessible via web console at:
  https://192.168.42.60:8443
You are logged in as:
  User: developer
  Password: developer
To login as administrator:
  oc login -u system:admin
...

```

4. Run this command to see if the virtual machine is running:

```

$ minishift status
Running

```

You will see similar output from `minishift start` when run in a macOS terminal or Windows command prompt or PowerShell.

Step 5: Configure CDK

With the CDK virtual machine running, you can configure and start using CDK through two primary interfaces: `oc` (OpenShift client binary command) or the **OpenShift web console**.

- `oc`: Use `minishift oc-env` to display the command you need to type into your shell in order to add the `oc` binary to your `PATH` environment variable. The output of `minishift oc-env` will differ depending on your user directory, operating system and shell type.

For Red Hat Enterprise Linux:

```

minishift oc-env
export PATH="/home/joe/.minishift/cache/oc/v3.6.173.0.21:$PATH"
# Run this command to configure your shell:
# eval $(minishift oc-env)

```

Use the following to add the `oc` command to your `PATH` via the `~/.bashrc` file:

```

$ echo export PATH="\home/joe/.minishift/cache/oc/v3.6.173.0.21:$PATH\" >> ~/.bashrc
$ source ~/.bashrc

```

For macOS:

```

$ minishift oc-env
export PATH="/home/joe/.minishift/cache/oc/v3.6.173.0.21:$PATH"
# Run this command to configure your shell:
# eval $(minishift oc-env)
$ echo export PATH="/Users/joe/.minishift/cache/oc/v3.6.173.0.21:$PATH" >> ~/.bash_profile
$ source ~/.bash_profile

```

For Windows:

Add the path to the `oc` binary to your `PATH` environment variable.

You are now ready to start using OpenShift using either of those two interfaces.

Step 6: Disable anyuid addon

Anyuid allows to run Docker images with a preset user id and that is not allowed in OpenShift production environment. This addon should be disabled.

```

$minishift addons disable anyuid

```