Oracle VirtualBox

*Estimated reading time: 4 minutes*

Create machines locally using [VirtualBox](https://www.virtualbox.org/). This driver requires VirtualBox 5+ to be installed on your host. Using VirtualBox 4.3+ should work but will give you a warning. Older versions will refuse to work.

Usage

$ docker-machine create --driver=virtualbox vbox-test

You can create an entirely new machine or you can convert a Boot2Docker VM into a machine by importing the VM. To convert a Boot2Docker VM, you’d use the following command:

$ docker-machine create -d virtualbox --virtualbox-import-boot2docker-vm boot2docker-vm b2d

The size of the VM’s disk can be configured this way:

$ docker-machine create -d virtualbox --virtualbox-disk-size "100000" large

Options

* --virtualbox-boot2docker-url: The URL of the boot2docker image. Defaults to the latest available version.
* --virtualbox-cpu-count: Number of CPUs to use to create the VM. Defaults to single CPU.
* --virtualbox-disk-size: Size of disk for the host in MB.
* --virtualbox-host-dns-resolver: Use the host DNS resolver. (Boolean value, defaults to false)
* --virtualbox-hostonly-cidr: The CIDR of the host only adapter.
* --virtualbox-hostonly-nicpromisc: Host Only Network Adapter Promiscuous Mode. Possible options are deny , allow-vms, allow-all
* --virtualbox-hostonly-nictype: Host Only Network Adapter Type. Possible values are ‘82540EM’ (Intel PRO/1000), ‘Am79C973’ (PCnet-FAST III), and ‘virtio’ Paravirtualized network adapter.
* --virtualbox-hostonly-no-dhcp: Disable the Host Only DHCP Server
* --virtualbox-import-boot2docker-vm: The name of a Boot2Docker VM to import.
* --virtualbox-memory: Size of memory for the host in MB.
* --virtualbox-nat-nictype: Specify the NAT Network Adapter Type. Possible values are are ‘82540EM’ (Intel PRO/1000), ‘Am79C973’ (PCnet-FAST III) and ‘virtio’ Paravirtualized network adapter.
* --virtualbox-no-dns-proxy: Disable proxying all DNS requests to the host (Boolean value, default to false)
* --virtualbox-no-share: Disable the mount of your home directory
* --virtualbox-no-vtx-check: Disable checking for the availability of hardware virtualization before the vm is started
* --virtualbox-share-folder: Mount the specified directory instead of the default home location. Format: dir:name

|  |  |  |  |
| --- | --- | --- | --- |
| --virtualbox-ui-type: Specify the UI Type: (gui | sdl | headless | separate) |

The --virtualbox-boot2docker-url flag takes a few different forms. By default, if no value is specified for this flag, Machine will check locally for a boot2docker ISO. If one is found, that will be used as the ISO for the created machine. If one is not found, the latest ISO release available on [boot2docker/boot2docker](https://github.com/boot2docker/boot2docker) will be downloaded and stored locally for future use. Note that this means you must run docker-machine upgrade deliberately on a machine if you wish to update the “cached” boot2docker ISO.

This is the default behavior (when --virtualbox-boot2docker-url=""), but the option also supports specifying ISOs by the http:// and file:// protocols. file:// will look at the path specified locally to locate the ISO: for instance, you could specify --virtualbox-boot2docker-url file://$HOME/Downloads/rc.iso to test out a release candidate ISO that you have downloaded already. You could also just get an ISO straight from the Internet using the http:// form.

To customize the host only adapter, you can use the --virtualbox-hostonly-cidr flag. This will specify the host IP and Machine will calculate the VirtualBox DHCP server address (a random IP on the subnet between .1 and .25) so it does not clash with the specified host IP. Machine will also specify the DHCP lower bound to .100 and the upper bound to .254. For example, a specified CIDR of 192.168.24.1/24 would have a DHCP server between 192.168.24.2-25, a lower bound of 192.168.24.100 and upper bound of 192.168.24.254.

ENVIRONMENT VARIABLES AND DEFAULT VALUES

| **CLI option** | **Environment variable** | **Default** |
| --- | --- | --- |
| --virtualbox-boot2docker-url | VIRTUALBOX\_BOOT2DOCKER\_URL | *Latest boot2docker url* |
| --virtualbox-cpu-count | VIRTUALBOX\_CPU\_COUNT | 1 |
| --virtualbox-disk-size | VIRTUALBOX\_DISK\_SIZE | 20000 |
| --virtualbox-host-dns-resolver | VIRTUALBOX\_HOST\_DNS\_RESOLVER | false |
| --virtualbox-hostonly-cidr | VIRTUALBOX\_HOSTONLY\_CIDR | 192.168.99.1/24 |
| --virtualbox-hostonly-nicpromisc | VIRTUALBOX\_HOSTONLY\_NIC\_PROMISC | deny |
| --virtualbox-hostonly-nictype | VIRTUALBOX\_HOSTONLY\_NIC\_TYPE | 82540EM |
| --virtualbox-hostonly-no-dhcp | VIRTUALBOX\_HOSTONLY\_NO\_DHCP | false |
| --virtualbox-import-boot2docker-vm | VIRTUALBOX\_BOOT2DOCKER\_IMPORT\_VM | boot2docker-vm |
| --virtualbox-memory | VIRTUALBOX\_MEMORY\_SIZE | 1024 |
| --virtualbox-nat-nictype | VIRTUALBOX\_NAT\_NICTYPE | 82540EM |
| --virtualbox-no-dns-proxy | VIRTUALBOX\_NO\_DNS\_PROXY | false |
| --virtualbox-no-share | VIRTUALBOX\_NO\_SHARE | false |
| --virtualbox-no-vtx-check | VIRTUALBOX\_NO\_VTX\_CHECK | false |
| --virtualbox-share-folder | VIRTUALBOX\_SHARE\_FOLDER | - |
| --virtualbox-ui-type | VIRTUALBOX\_UI\_TYPE | headless |

Known Issues

Vboxfs suffers from a [longstanding bug](https://www.virtualbox.org/ticket/9069) causing [sendfile(2)](http://linux.die.net/man/2/sendfile) to serve cached file contents.

This will often cause problems when using a web server such as nginx to serve static files from a shared volume. For development environments, a good workaround is to disable sendfile in your server configuration.