



.com



The Most Customizable eCommerce Platform
For Building Your Online Business.

Get Started >

DOCKER / VIRTUALIZATION



7

I ♥ TecMint :



Install Docker and Learn Basic



BEGINNER'S GUIDE FOR LINUX



Start learning Linux in minutes →

51 Useful Lesser Known Commands for Linux Users

Container Manipulation in CentOS and RHEL 7/6 – Part 1

by Matei Cezar | Published: January 27, 2016 |

Last Updated: February 3, 2016

AdChoices

Docker

Cent

Download Your Free eBooks NOW - [10 Free Linux eBooks for Administrators](#) | [4 Free Shell Scripting eBooks](#)

In this 3-article series, we will discuss about **Docker**, is an open-source lightweight virtualization tool which runs at top of Operating System level, allowing users to create, run and deploy applications, encapsulated into small containers.

Get Started with Docker - Part 1



Install Docker and Learn Container Manipulation in CentOS and RHEL 7/6

Install Docker and Learn Basic

Container Manipulation – Part 1

This type of Linux containers are proven to be fast, portable and secure

51 Useful Lesser Known Commands for Linux Users

WHEELS NEAR YOU TO CONQUER EVERY TO-DO.



zipcar

JOIN & DRIVE



Vi/Vim Editor BEGINNER'S GUIDE



Learn vi/vim as a Full Text Editor



Advertise Here

Earn up to 11.00%*

with a 3 year TD Canadian Banking and Utilities GIC.

Learn more

*Return linked to market index performance. Max. return over 3-year term.



Linux Foundation Certification



Exam Study Guide to **LFCS** and **LFCE**

The processes that run in a Docker container are always isolated from the main host, preventing outside tampering.

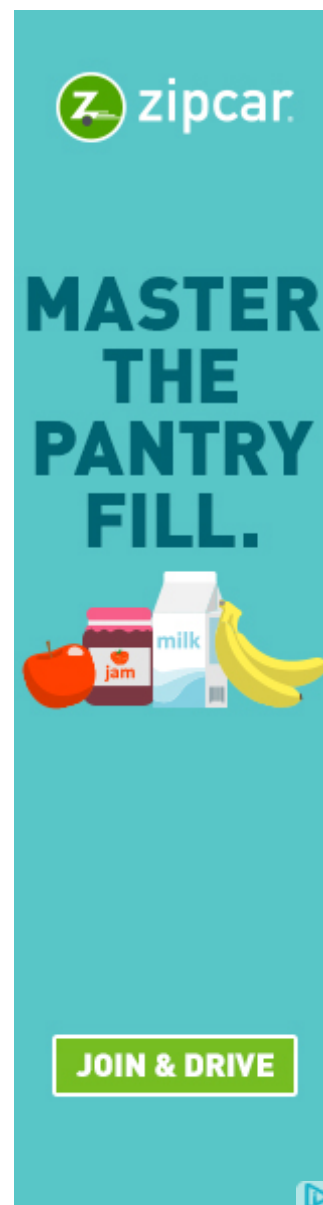
Part 1: Install Docker and Learn Basic Container Manipulation in CentOS and RHEL 7/6

Part 2: **How to Deploy and Run Applications into Docker Containers on CentOS/RHEL 7/6**

Part 3: **Automatically Build and Configure Docker Images with Dockerfile on CentOS/RHEL 7/6**

This tutorial provides a starting point on how to install Docker, create and run Docker containers on CentOS/RHEL 7/6, but barely scratches the surface of Docker.

Step 1: Install and Configure Docker



How to Add Linux Host to Nagios Monitoring Server Using NRPE Plugin

Nagios 4.2.0 Released – Install on RHEL/CentOS 7.x/6.x/5.x and Fedora 24-19

Install Cacti (Network Monitoring) on RHEL/CentOS 7.x/6.x/5.x and Fedora 24-12

Google Chrome 56 Released – Install on RHEL/CentOS 7/6 and Fedora 25-20

51 Useful Lesser Known Commands for Linux Users

1. Docker binaries are incorporated into RHEL/CentOS 7 extras repositories, the installation process being pretty simple. Install Docker package by issuing the following command with root privileges:

Install Docker on RHEL and CentOS 7

```
# yum install docker
```

```
loading mirror speeds from cached hostfile
* base: centos.euclidmedia.net
* epel: mirrors.hustanig.com
* extras: centos.euclidmedia.net
* updates: centos.euclidmedia.net
Resolving Dependencies
--> Running transaction check
--> Package docker.x86_64 1:1.8.2-10.el7.centos will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
Installing:
docker x86_64 1:1.8.2-10.el7.centos extras 10 M
Transaction Summary
-----
Install 1 Package
Total download size: 10 M
Installed size: 40 M
Is this ok [y/d/N]:
```

Install Docker on CentOS and RHEL 7

Install Docker on RHEL and CentOS 6

Stacki Open Source

Automate the Deployment, Scaling, and I Metal Go to stackiq.com/kubernetes

To install Docker, the [Epel repositories](#) must be enabled on your system by issuing the following command:

Ebook: Introducing the RHCSA and RHCE Exam Preparation Guide

How to Install Ubuntu 16.10/16.04 Alongside With Windows 10 or 8 in Dual-Boot



Red Hat RHCSA/RHCE Certification Preparation Study Guide

RedHat RHCSA / RHCE 7

RHCSA (EX200) and RHCE (EX300) exams

* EX200 - Red Hat Certified System Administrator (RHCSA)
* EX300 - Red Hat Certified Engineer (RHCE)

Buy Now \$35.00

Linux System Administrator Bundle
with 7-Courses (96% off)

Add to Cart - \$69

🕒 Ending In: 3 days

Computer Hacker Professional Certification Course (96% Off)

Add to Cart - \$59

🕒 Ending In: 4 days

51 Useful Lesser Known Commands for Linux Users

```
# yum install epel-release
# yum install docker-io
```

```
--> Package libfilesystem.x86_64 0.1.4-2.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
Installing:
docker-io x86_64 1.7.1-2.el6 epel 4.6 M
Installing for dependencies:
libnghttp x86_64 0.40.rc1-16.el6 base 129 k
lua-all-getopt noarch 0.2.0-1.el6 epel 6.9 k
lua-libsystem x86_64 1.4.2-1.el6 epel 24 k
lua-lua x86_64 1.0.0-1.el6 epel 16 k
lua x86_64 1.0.0-1.el6 epel 322 k
lua-lua x86_64 1.0.0-1.el6 epel 255 k
Transaction Summary
-----
Install : 7 Packages
Total download size: 5.1 M
Installed size: 28 M
Is this ok [y/N]: y
```

Install Docker on RHEL and CentOS 6

2. After, Docker package has been installed, start the daemon, check its status and enable it system wide using the below commands:

On RHEL/CentOS 7

```
# systemctl start docker
# systemctl status docker
# systemctl enable docker
```

```
root@tecmint:~# systemctl start docker.service
root@tecmint:~# systemctl status docker.service
docker.service - Docker Application Container Engine
Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; vendor preset: disabled)
Active: active (running) since Wed 2016-01-27 02:19:12 EST; 9s ago
Docs: http://docs.docker.com
Main PID: 2988 (docker)
CGroup: /system.slice/docker.service
└─2988 /usr/bin/docker daemon --linux-enabled

Jan 27 02:19:12 tecmint docker[2988]: time="2016-01-27T02:19:12.50727845-0500" msg="Starting Docker Application Container Engine"
Jan 27 02:19:12 tecmint docker[2988]: time="2016-01-27T02:19:12.507329885-0500" msg="Docker daemon"
Jan 27 02:19:12 tecmint docker[2988]: time="2016-01-27T02:19:12.509096374-0500" msg="Docker daemon"
Jan 27 02:19:12 tecmint docker[2988]: time="2016-01-27T02:19:12.533211386-0500" msg="Docker daemon"
Jan 27 02:19:12 tecmint docker[2988]: time="2016-01-27T02:19:12.002175913-0500" msg="Docker daemon"
Jan 27 02:19:12 tecmint docker[2988]: time="2016-01-27T02:19:12.025420986-0500" msg="Docker daemon"
Jan 27 02:19:12 tecmint docker[2988]: time="2016-01-27T02:19:12.025477280-0500" msg="Docker daemon"
Jan 27 02:19:12 tecmint docker[2988]: time="2016-01-27T02:19:12.025515140-0500" msg="Docker daemon"
Jan 27 02:19:12 tecmint systemctl[1]: Started Docker Application Container Engine.
Hint: some lines were ellipsized, use -l to show in full.
root@tecmint:~# systemctl enable docker.service
```

Enable Docker on RHEL and CentOS 7

On RHEL/CentOS 6

```
# service docker start
# service docker status
# chkconfig docker on
```

DOWNLOAD FREE LINUX EBOOKS

- Complete Linux Command Line Cheat Sheet
- The GNU/Linux Advanced Administration Guide
- Securing & Optimizing Linux Servers
- Linux Patch Management: Keeping Linux Up To Date
- Introduction to Linux – A Hands on Guide
- Understanding the Linux® Virtual Memory Manager
- Getting Started with Ubuntu 16.04
- A Newbie's Getting Started Guide to Linux
- Linux from Scratch – Create Your Own Linux OS
- Linux Shell Scripting Cookbook, Second Edition
- Securing & Optimizing Linux: The Hacking Solution
- User Mode Linux – Understanding and Administration



Never Miss Any Linux Tutorials, Guides, Tips and Free eBooks

Join Our Community Of **150,000+ Linux Lovers** and get a weekly newsletter in your inbox

YES! SIGN ME UP

```
root@tecmint ~# service docker start
Starting docker:
root@tecmint ~# service docker status
docker (pid 3790) is running...
root@tecmint ~# chkconfig docker on
root@tecmint ~#
```

Enable Docker on RHEL and CentOS 6

3. Finally, run a container test image to verify if Docker works properly, by issuing the following command:

```
# docker run hello-world
```

If you can see the below message, then everything is in the right place.

```
"Hello from Docker. This message shows that your installation appears to be working correctly."
```

```
025b0d410011: Pull complete
Digest: sha256:1bae98ef2aeb1bdc83271d4fc2010f4603d13f6dfb8c72074cc1ce30766a7
Status: Downloaded newer image for hello-world:latest
Hello from Docker.
This message shows that your installation appears to be working correctly.
To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.
To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash
Share images, automate workflows, and more with a free Docker Hub account:
https://hub.docker.com
For more examples and ideas, visit:
https://docs.docker.com/userguide/
root@tecmint ~#
```

Docker Hello World

4. Now, you can run a few basic Docker commands to get some info about Docker:

For system-wide information on Docker

51 Useful Lesser Known Commands for Linux Users

```
# docker info
```

```
root@node: docker-0.1.3-527d1-pool
root@node:~# docker info
Docker Version: 0.1.3-527d1-pool
Docker Root Dir: /var/lib/docker
Data File: /dev/loop0
Metadata File: /dev/loop0
Data Space Used: 387.7 MB
Data Space Total: 107.4 GB
Data Space Available: 36.47 GB
Metadata Space Used: 227.4 MB
Metadata Space Total: 2.147 GB
Metadata Space Available: 2.147 GB
User Space Supported: true
Deferred Removal Enabled: false
Data loop file: /var/lib/docker/devicemapper/devicemapper/data
Metadata loop file: /var/lib/docker/devicemapper/devicemapper/metadata
Library Version: 1.02.05-30216 (2015-03-08)
Execution Driver: native-0.2
Logging Driver: json-file
Kernel Version: 2.6.32-573.12.1.el6.x86_64
Operating System: (unknown)
CPU: 1
Total Memory: 996.3 MB
Name: node1
ID: 724H115A:RHX:Q48:4L04-JF8G:FEYP:YR6:Y2DC:350L:87Q1:U3KT
root@node:~#
```

Check Docker Info

For Docker version

```
# docker version
```

```
root@node:~# docker version
Client version: 1.7.1
Client API version: 1.19
Go version (client): go1.4.2
git commit (client): 7083292:1.7.1
OS/Arch (client): linux/amd64
Server version: 1.7.1
Server API version: 1.19
Go version (server): go1.4.2
git commit (server): 7083292:1.7.1
OS/Arch (server): linux/amd64
root@node:~#
```

Check Docker Version

5. To get a list of all available Docker commands type docker on your console.

```
# docker
```

```
usage: docker [OPTIONS] COMMAND [arg...]
A self-sufficient runtime for Linux containers.

Options:
  -api-cors-headers          Set CORS headers in the remote API
  -b, --bridge=bridge       Attach containers to a network bridge
  --bip=ip                   Specify network bridge IP
  -D, --debug=false         Enable debug mode
  --daemon=false            Enable daemon mode
  --default-gateway=ip       Container default gateway IP/v4 address
  --default-gateway-v6=ip   Container default gateway IP/v6 address
  --default-ulimits=ulimits Set default ulimits for containers
  --dns=server[,server...]  DNS server to use
  --dns-search=server[,server] DNS search domains to use
  -e, --exec-driver=native   Exec driver to use
  --exec-opt=opt[=value]    Set exec driver options
  --exec-root=/var/run/docker Root of the Docker execdriver
  --fixed-cidr=ip4           IP/v4 subnet for fixed IPs
  --fixed-cidr-v6=ip6       IP/v6 subnet for fixed IPs
  -g, --group=docker         Group for the unix socket
  -H, --graph=/var/lib/docker Root of the Docker runtime
  -h, --help                Daemon socket(s) to connect to
```

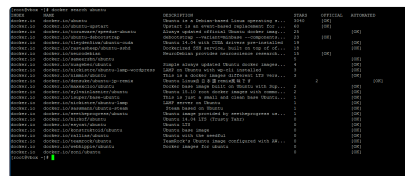
List Docker Commands

Step 2: Download a Docker Image

6. In order to start and run a Docker container, first an image must be downloaded from [Docker Hub](#) on your host. Docker Hub offers a great deal of free images from its repositories.

To search for a Docker image, Ubuntu for instance, issue the following command:

```
# docker search ubuntu
```



Search Docker Images

7. After you decided on what image you want to run based on your needs, download it locally by running the below command (in this case an **Ubuntu** image is downloaded and used):

```
# docker pull ubuntu
```

SHARE

+



0



187



93



Download Docker Images

8. To list all the available Docker images on your host issue the following command:

```
# docker images
```

List Docker Images

9. If you don't need a Docker image anymore and you want to remove it from the host issue the following command:

```
# docker rmi ubuntu
```

51 Useful Lesser Known Commands for Linux Users

Remove Docker Image

Step 3: Run a Docker Container

When you execute a command against an image you basically obtain a container. After the command that is executing into container ends, the container stops (you get a non-running or exited container). If you run another command into the same image again a new container is created and so on.

All the containers created will remain on the host filesystem until you choose to delete them by using the `docker rm` command.

10. In order to create and run a container, you need to run a command into a downloaded image, in this case **Ubuntu**, so a basic command would be to display the distribution version file inside the container using **cat** command, as in the following example:

```
# docker run ubuntu cat
```

Run Docker Containers

The above command is divided as follows:

```
# docker run [local ima
```

11. To run one of the containers again with the command that was

executed to create it, first

you must get the container ID (or the name automatically generated by Docker) by issuing the below command, which displays a list of the running and stopped (non-running) containers:

```
# docker ps -l
```

List Running Docker Containers

12. Once the container ID has been obtained, you can start the container again with the command that was used to create it, by issuing the following command:

```
# docker start c629b7d7
```

Here, the string

c629b7d70666 represents the container ID.

Start Docker Containers

13. In case the container is running state, you can get its **ID** by issuing

`docker ps` command. To stop the running container issue `docker stop`

command by specifying the container **ID** or auto-generated name.

```
# docker stop dreamy_mo
# docker ps
```

```
centos@centos ~$ docker stop dreamy_mo
dreamy_mo
centos@centos ~$ docker ps
CONTAINER ID        IMAGE               COMMAND             Status
centos@centos ~$
```

Start Stop Docker Containers

14. A more elegant alternative so you don't have to remember the container **ID** would be to allocate a unique name for every container you create by using the `--name` option on

command line, as in the following example:

```
# docker run --name myname
```

```
[root@vbox ~]# docker run --name myname ubuntu cat /etc/debian_version
0.7.1
[root@vbox ~]#
```

Add Name to Docker Container

15. Then, using the name that you allocated for the container, you can manipulate container (**start**, **stop**, **remove**, **top**, **stats**) further just by addressing its name, as in the below examples:

```
# docker start myname
# docker stats myname
# docker top myname
```

Be aware that some of the above commands might display no output if the process of command that was used to create the container finishes. When the process that runs inside the container finishes, the container stops.

Step 4: Run an Interactive Session into a Container

16. In order to interactively connect into a container shell session, and run commands as you do on any other Linux session, issue the following command:

```
# docker run -it ubuntu
```

Start Docker Container Interactive Shell

The above command is divided as follows:

- `-i` is used to start an interactive session.
- `-t` allocates a tty and attaches stdin and stdout.

51 Useful Lesser Known Commands for Linux Users

- `ubuntu` is the image that we used to create the container.
- `bash` (or `/bin/bash`) is the command that we are running inside the Ubuntu container.

17. To quit and return to host from the running container session you must type `exit` command. The `exit` command terminates all the container processes and stops it.

```
# exit
```

18. If you're interactively logged on container terminal prompt and you need to keep the container in running state but `exit` from the interactive session, you can quit the console and return to host terminal by pressing `Ctrl+p` and `Ctrl+q` keys.

Keep Docker Shell Session Active

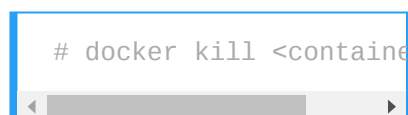
19. To reconnect to the running container you need the container **ID** or **name**.

Issue `docker ps` command to get the **ID** or **name** and, then, run `docker attach` command by specifying container **ID** or **name**, as illustrated in the image above:



```
# docker attach <container ID or name>
```

20. To stop a running container from the host session issue the following command:



```
# docker kill <container ID or name>
```

That's all for basic container manipulation. In the next

tutorial we will discuss **51 Useful Lesser Known Commands for Linux Users**

to save, delete and run a web server into a Docker container.

If You Appreciate What We Do Here On TecMint, You Should Consider:

1. Stay Connected to: [Twitter](#) | [Facebook](#) | [Google Plus](#)
2. Subscribe to our email updates: [Sign Up Now](#)
3. Use our [Linode referral link](#) if you plan to buy VPS (it starts at only \$10/month).
4. Support us via PayPal donate - [Make a Donation](#)
5. Support us by [purchasing our premium books](#) in PDF format.
6. Support us by taking our [online Linux courses](#)

We are thankful for your never ending support.

Tags: virtualization**Matei Cezar** [View all Posts](#)

I'm a computer addicted guy, a fan of open source and linux based system software, have about 4 years experience with Linux distributions desktop, servers and bash scripting.

Your name can also be listed here. Got a tip? [Submit it here](#) to become an TecMint author.



PREVIOUS

STORY

Deal: Learn Android and iOS Programming with Complete 2016 Coder Bundle (10-Courses)

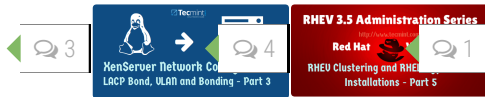


NEXT STORY

Deal: Get proXPN VPN Lifetime Subscription for \$39 – Save 89% off

[51 Useful Lesser Known Commands for Linux Users](#)

👍 YOU MAY ALSO LIKE...



How to Automatically Build and Configure Custom Docker Images with Dockerfile – Part 3

XenServer Network (LACP Bond, VLAN and Bonding) Configuration – Part 3

RHEV Clustering and RHEL Hypervisors Installation – Part 5

28 SEP, 2015

13 OCT, 2015

3 FEB, 2016

7 RESPONSES

💬 **Comments** 7 ➡ **Pingbacks** 0

david 🕒 March 3, 2016 at 10:55 pm

Hi – A genuine question please. I use Proxmox for LXC containers which works great..what does Docker offer over proxmox ?

Reply

Matei Cezar

🕒 March 4, 2016 at 2:07 pm

Basically, I would say that the main difference is that LXC containers behave as a virtual machine mainly, opposed to Docker which runs an ephemeral single process. But

that's only just a simple

51 Useful Lesser Known Commands for Linux Users

difference. You should google it for more detailed info.

Reply

alex ☉ February 13, 2016 at 11:04 pm

Thanks for the article..

I completed a 6 month contract for a large publisher involving the evaluation of various triple stores. I created a dockerfile for each triple store allowing us to quickly install the systems when required on various servers. I also created a system to monitor and graph the memory use of any number docker containers. They are a number of stumbling blocks that takes some experience to understand.

Regards

Alex

<http://Www.tiloge.com>

Reply

Raghu

☉ October 24, 2016 at 11:21 pm

Seems tiloge.com link is down & unavailable.

Reply

Shambhu Rajput

☉ February 7, 2016 at 12:39 pm

Thank you

Reply

Matei Cezar ☉ February 3, 2016 at 8:46 pm

You could use Docker on any machine, it doesn't matter if you are running LAMP stack on it. As a recommendation try to run Docker on a fresh server or a test server, never play on production machines for the sake of security and resources.

Reply

Shambhu ☉ February 2, 2016 at 3:24 pm

hi

For docker install 51 Useful Lesser Known Commands for Linux Users

or can install at running LAMP server?

Regard

Reply

GOT SOMETHING TO SAY? JOIN THE DISCUSSION.

Comment

Name *

Email *

Website

☐

Notify me of followup comments via e-mail. You can also [subscribe](#) without commenting.

Post Comment

LINUX MONITORING TOOLS

Collectl: An Advanced All-in-One Performance Monitoring Tool for Linux
20 APR, 2014

How to Setup and Manage Log Rotation Using

LINUX INTERVIEW QUESTIONS

10 Useful 'ls' Command Interview Questions – Part 2
7 MAR, 2015

10 VsFTP (Very Secure File Transfer Protocol) Interview Questions and Answers
51 Useful Lesser Known Commands for Linux Users

OPEN SOURCE TOOLS

Linux Directory Structure and Important Files Paths Explained
16 SEP, 2013

How to Use Ansible Playbooks to Automate

Logrotate in Linux

25 AUG, 2016

3 FEB, 2014

10 Advance VsFTP Interview
Questions and Answers –
Part II

10 FEB, 2014

Observium: A Complete
Network Management and
Monitoring System for
RHEL/CentOS

1 JUL, 2014

10 Basic Interview
Questions and Answers on
Linux Networking – Part 1

2 AUG, 2014

Suricata 1.4.4 Released – A
Network Intrusion Detection,
Prevention and Security
Monitoring System

23 JUL, 2013

Complex Tasks on Multiple
Remote Servers – Part 2

10 OCT, 2015

Alternatives of 13 Most
Commonly Used Windows
Applications for Linux

29 MAY, 2013

Create Your Own Video
Sharing Website using
'CumulusClips Script' in
Linux

28 MAY, 2014



Tecmint: Linux Howtos, Tutorials & Guides © 2017.

All Rights Reserved.

This work is licensed under a (cc) BY-NC

The material in this site cannot be republished
either online or offline, without our permission.

