

Docker - 2 - CentOS 安装 Docker

一、Docker 安装

1.0 Docker 的版本

1.1 下载 Wget

1.2 替换 YUM 镜像

1.3 查看CentOS版本

1.4 安装GCC

1.5 安装 GCC-C++

1.6 安装软件包

1.7 设置Stable仓库

1.8 更新 YUM 软件包索引

1.9 查看 YUM 中 Docker 版本

1.10 安装 Docker Engine

1.11 启动 Docker

1.12 查看 Docker信息

1.13 运行 HelloWorld

1.14 开启阿里云镜像加速

Docker - 2 - CentOS 安装 Docker

一、Docker 安装

1.0 Docker 的版本

随着 Docker 的飞速发展，企业级功能的上线，更好的服务意味着需要支付一定的费用，目前 Docker 被分为两个版本：

- community-edition 社区版
- enterprise-edition 企业版

Docker 企业版（EE）专为企业开发和 IT 团队设计，可在大规模生产中构建，运送和运行关键业务应用程序。Docker EE 集成，认证和支持，为企业提供业界最安全的容器平台，实现所有应用程序的现代化。作为一个以应用为中心的平台，Docker EE 旨在加速和保护整个软件供应链，从开发到在任何基础设施上运行的生产。

我们学习 Docker 使用 CE 社区版即可。

Pricing & Subscriptions

Choose a plan that's right for you.

		TEAM POWER
Free FOR INDIVIDUALS The basics of Docker for every developer, including unlimited public repos and one private repo. \$0 /month Signup for Free	Pro FOR INDIVIDUALS Pro tools for developers with advanced requirements for parallel builds and unlimited private repos. \$5 /month With annual plan Buy Now	Team FOR ORGANIZATIONS Advanced collaboration and management tools for teams, including organization and team management with role-based access controls, parallel builds, and unlimited private repositories. \$7 user/month Starts at \$25 for 5 users Buy Now

1.1 下载 Wget

```
1 [root@localhost ~]# yum -y install wget
2 已加载插件: fastestmirror
3 Loading mirror speeds from cached hostfile
4 * base: ftp.sjtu.edu.cn
5 * extras: ftp.sjtu.edu.cn
6 * updates: ftp.sjtu.edu.cn
7 正在解决依赖关系
8 --> 正在检查事务
9 ---> 软件包 wget.x86_64.0.1.14-18.el7_6.1
    将被 安装
10 --> 解决依赖关系完成
11
12 依赖关系解决
13
```

14 =====
=====
=====
=====

15 Package 架构
版本
源
大小

16 =====
=====
=====
=====

17 正在安装：
18 wget x86_64
1.14-18.el7_6.1
base
547 k

19
20 事务概要
21 =====
=====
=====
=====

22 安装 1 软件包
23
24 总下载量：547 k
25 安装大小：2.0 M
26 Downloading packages:

27 wget-1.14-18.el7_6.1.x86_64.rpm

547 kB 00:00:00

28 Running transaction check

29 Running transaction test

30 Transaction test succeeded

31 Running transaction

32 正在安装 : wget-1.14-
18.el7_6.1.x86_64

1/1

33 验证中 : wget-1.14-
18.el7_6.1.x86_64

1/1

34

35 已安装:

36 wget.x86_64 0:1.14-18.el7_6.1

37

38 完毕!

39 [root@localhost ~]#

40

1.2 替换 YUM 镜像

```
1 [root@localhost yum.repos.d]# pwd
2 /etc/yum.repos.d
3 [root@localhost yum.repos.d]# ls -al
4 总用量 52
5 drwxr-xr-x.  2 root root  220 10月  2
6 2020 .
7 drwxr-xr-x. 75 root root 8192 12月 28
8 19:33 ..
9 -rw-r--r--.  1 root root 1664 11月 23
10 2020 CentOS-Base.repo
11 -rw-r--r--.  1 root root 1309 11月 23
12 2020 CentOS-CR.repo
13 -rw-r--r--.  1 root root  649 11月 23
14 2020 CentOS-Debuginfo.repo
15 -rw-r--r--.  1 root root  314 11月 23
16 2020 CentOS-fasttrack.repo
17 -rw-r--r--.  1 root root  630 11月 23
18 2020 CentOS-Media.repo
19 -rw-r--r--.  1 root root 1331 11月 23
20 2020 CentOS-Sources.repo
21 -rw-r--r--.  1 root root 8515 11月 23
22 2020 CentOS-Vault.repo
23 -rw-r--r--.  1 root root  616 11月 23
24 2020 CentOS-x86_64-kernel.repo
25 [root@localhost yum.repos.d]#
```

```
-rw-r--r--. 1 root root 1664 11月 23 2020
```

CentOS-Base.repo 这是 CentOS 自带的 YUM 镜像，使用的是国外的源，容易超时，更换成国内的：网易的或者阿里的。

更换流程3步：

- 备份默认的
- 下载阿里镜像 | 网易镜像
- 重新生成缓存

```
1 [root@localhost yum.repos.d]# mv CentOS-Base.repo CentOS-Base.repo.back
2 [root@localhost yum.repos.d]# ls -al
3 总用量 52
4 drwxr-xr-x. 2 root root 225 12月 28 19:39 .
5 drwxr-xr-x. 75 root root 8192 12月 28 19:33 ..
6 -rw-r--r--. 1 root root 1664 11月 23 2020 CentOS-Base.repo.back
7 -rw-r--r--. 1 root root 1309 11月 23 2020 CentOS-CR.repo
8 -rw-r--r--. 1 root root 649 11月 23 2020 CentOS-Debuginfo.repo
9 -rw-r--r--. 1 root root 314 11月 23 2020 CentOS-fasttrack.repo
10 -rw-r--r--. 1 root root 630 11月 23 2020 CentOS-Media.repo
```

```
11 -rw-r--r--. 1 root root 1331 11月 23
    2020 CentOS-Sources.repo
12 -rw-r--r--. 1 root root 8515 11月 23
    2020 CentOS-Vault.repo
13 -rw-r--r--. 1 root root 616 11月 23
    2020 CentOS-x86_64-kernel.repo
14 [root@localhost yum.repos.d]#
15
```

```
1 [root@localhost yum.repos.d]# wget -O
  /etc/yum.repos.d/CentOS-Base.repo
  https://mirrors.aliyun.com/repo/Centos-
  7.repo
2 --2022-12-28 19:40:47--
  https://mirrors.aliyun.com/repo/Centos-
  7.repo
3 正在解析主机 mirrors.aliyun.com
  (mirrors.aliyun.com)... 58.216.14.238,
  140.249.32.239, 140.249.32.248, ...
4 正在连接 mirrors.aliyun.com
  (mirrors.aliyun.com)|58.216.14.238|:443.
  .. 已连接。
5 已发出 HTTP 请求，正在等待回应... 200 OK
6 长度: 2523 (2.5K) [application/octet-
  stream]
7 正在保存至: “/etc/yum.repos.d/CentOS-
  Base.repo”
8
```



```

9 100%
  [=====
  =====
  =====>] 2,523      --.-
K/s 用时 0s
10
11 2022-12-28 19:40:48 (106 MB/s) - 已保存
   "/etc/yum.repos.d/CentOS-Base.repo"
   [2523/2523])
12
13 [root@localhost yum.repos.d]#
14

```

`-o`的作用是指定下载文件保存在本地的文件名。重新查询可以看到文件已经下载完成！

```

1 [root@localhost yum.repos.d]# ls -al
2 总用量 56
3 drwxr-xr-x. 2 root root 249 12月 28
   19:40 .
4 drwxr-xr-x. 75 root root 8192 12月 28
   19:33 ..
5 -rw-r--r--. 1 root root 2523 8月 4
   15:04 CentOS-Base.repo
6 -rw-r--r--. 1 root root 1664 11月 23
   2020 CentOS-Base.repo.back
7 -rw-r--r--. 1 root root 1309 11月 23
   2020 CentOS-CR.repo
8 -rw-r--r--. 1 root root 649 11月 23
   2020 CentOS-Debuginfo.repo

```

```
9 -rw-r--r--. 1 root root 314 11月 23
  2020 CentOS-fasttrack.repo
10 -rw-r--r--. 1 root root 630 11月 23
   2020 CentOS-Media.repo
11 -rw-r--r--. 1 root root 1331 11月 23
   2020 CentOS-Sources.repo
12 -rw-r--r--. 1 root root 8515 11月 23
   2020 CentOS-Vault.repo
13 -rw-r--r--. 1 root root 616 11月 23
   2020 CentOS-x86_64-kernel.repo
14 [root@localhost yum.repos.d]#
15
```

最后一步：重新生成缓存信息

```
1 [root@localhost yum.repos.d]# yum clean
  all
2 已加载插件：fastestmirror
3 正在清理软件源： base extras updates
4 Cleaning up list of fastest mirrors
5 [root@localhost yum.repos.d]# yum
  makecache
6 已加载插件：fastestmirror
7 Determining fastest mirrors
8 * base: mirrors.aliyun.com
9 * extras: mirrors.aliyun.com
10 * updates: mirrors.aliyun.com
```

11	base		
	3.6 kB	00:00:00	
12	extras		
	2.9 kB	00:00:00	
13	updates		
	2.9 kB	00:00:00	
14	base/7/x86_64/primary_db	FAILED	
15	http://mirrors.cloud.aliyuncs.com/centos/7/os/x86_64/repo/6d0c3a488c282fe537794b5946b01e28c7f44db79097bb06826e1c0c88bad5ef-primary.sqlite.bz2: [Errno 14] curl#6 - "Could not resolve host: mirrors.cloud.aliyuncs.com; Unknown error"		
16	正在尝试其它镜像。		
17	(1/10): base/7/x86_64/group_gz		
	153 kB	00:00:00	
18	(2/10): extras/7/x86_64/other_db		
	149 kB	00:00:00	

```
19 (3/10): extras/7/x86_64/filelists_db
|
276 kB 00:00:00
20 (4/10): base/7/x86_64/other_db
|
2.6 MB 00:00:01
21 (5/10): updates/7/x86_64/other_db
|
1.2 MB 00:00:00
22 (6/10): updates/7/x86_64/filelists_db
|
10 MB 00:00:04
23 (7/10): base/7/x86_64/primary_db
|
6.1 MB 00:00:04
24 base/7/x86_64/filelists_db FAILED
25 http://mirrors.aliyuncs.com/centos/7/os/
x86_64/repodata/d6d94c7d406fe7ad4902a971
04b39a0d8299451832a97f31d71653ba982c955b
-filelists.sqlite.bz2: [Errno 14] curl#7
- "Failed connect to
mirrors.aliyuncs.com:80; Connection
refused"
26 正在尝试其它镜像。
```

```
27 | extras/7/x86_64/primary_db      FAILED
```

```
28 http://mirrors.aliyuncs.com/centos/7/extras/x86_64/repo/511a574b4c3f907f9acf
e5603566958fb6f09137292c5cd5d5d57c298864790f-primary.sqlite.bz2: [Errno 14]
```

```
curl#7 - "Failed connect to
mirrors.aliyuncs.com:80; Connection
refused"
```

29 正在尝试其它镜像。

```
30 updates/7/x86_64/primary_db FAILED
```

```
31 http://mirrors.aliyuncs.com/centos/7/updates/x86_64/repodata/4679d0dbfe073cbbfdb2475610c0718f361697c45a9009ea30f6eb611bffa622-primary.sqlite.bz2: [Errno 14]
```

```
curl#7 - "Failed connect to
mirrors.aliyuncs.com:80; Connection
refused"
```

32 正在尝试其它镜像。

```
33 (8/10): extras/7/x86_64/primary_db
```

249 kB 00:00:00

```
34 (9/10): base/7/x86_64/filelists_db
```

7.2 MB 00:00:04

35 (10/10): updates/7/x86_64/primary_db

|

19 MB 00:00:08

36 元数据缓存已建立

37 [root@localhost yum.repos.d]#

38

3. 运行 yum makecache 生成缓存

4. 其他

非阿里云ECS用户会出现 Couldn't resolve host 'mirrors.cloud.aliyuncs.com' 信息, 不影响使用。用户也可自行修改相关配置: eg:

```
sed -i -e '/mirrors.cloud.aliyuncs.com/d' -e '/mirrors.aliyuncs.com/d' /etc/yum.repos.d/Cent
```

```
1 [root@localhost yum.repos.d]# sed -i -e  
  '/mirrors.cloud.aliyuncs.com/d' -e  
  '/mirrors.aliyuncs.com/d'  
  /etc/yum.repos.d/CentOS-Base.repo  
2 [root@localhost yum.repos.d]#  
3
```



```
10 centos-scl0-rh
    | 3.0 kB 00:00:00
11 centos-scl0-scl0
    | 3.0 kB 00:00:00
12 extras
    | 2.9 kB 00:00:00
13 updates
    | 2.9 kB 00:00:00
14 (1/3): extras/7/x86_64/primary_db
    | 249 kB 00:00:00
15 (2/3): centos-scl0-rh/x86_64/primary_db
    | 3.4 MB 00:00:00
16 (3/3): updates/7/x86_64/primary_db
    | 18 MB 00:00:02
17 正在解决依赖关系
18 --> 正在检查事务
19 ---> 软件包 gcc.x86_64.0.4.8.5-44.e17 将被
    安装
20 --> 正在处理依赖关系 cpp = 4.8.5-44.e17, 它
    被软件包 gcc-4.8.5-44.e17.x86_64 需要
21 --> 正在检查事务
22 ---> 软件包 cpp.x86_64.0.4.8.5-44.e17 将被
    安装
```


23 --> 解决依赖关系完成

24

25 依赖关系解决

26

27

=====

=====

=====

28

Package

架构

版本

源

大小

29

=====

=====

=====

30

正在安装：

31

gcc

x86_64

4.8.5-44.e17

base

16 M

32

为依赖而安装：

33

cpp

x86_64

4.8.5-44.e17

base

5.9 M

34

35

事务概要

36

=====

=====

=====

37

安装 1 软件包 (+1 依赖软件包)

38

39

总下载量：22 M

40

安装大小：52 M

```
41 Downloading packages:
42 (1/2): cpp-4.8.5-44.el7.x86_64.rpm
                                     | 5.9 MB 00:00:01
43 (2/2): gcc-4.8.5-44.el7.x86_64.rpm
                                     | 16 MB 00:00:02
44 -----
                                     -----
                                     -----
45 总计
                                     10 MB/s | 22 MB 00:00:02
46 Running transaction check
47 Running transaction test
48 Transaction test succeeded
49 Running transaction
50 正在安装      : cpp-4.8.5-44.el7.x86_64
                                     1/2
51 正在安装      : gcc-4.8.5-44.el7.x86_64
                                     2/2
52 验证中        : gcc-4.8.5-44.el7.x86_64
                                     1/2
53 验证中        : cpp-4.8.5-44.el7.x86_64
                                     2/2
54
```

```
55 已安装：
56     gcc.x86_64 0:4.8.5-44.e17

57
58 作为依赖被安装：
59     cpp.x86_64 0:4.8.5-44.e17

60
61 完毕！
62 [root@localhost ~]#
63
```

1.5 安装 GCC-C++

```
1 [root@localhost yum.repos.d]# yum -y
  install gcc-c++
2 已加载插件：fastestmirror
3 Loading mirror speeds from cached
  hostfile
4 正在解决依赖关系
5 --> 正在检查事务
6 ---> 软件包 gcc-c++.x86_64.0.4.8.5-44.e17
  将被 安装
7 --> 正在处理依赖关系 libstdc++-devel =
  4.8.5-44.e17，它被软件包 gcc-c++-4.8.5-
  44.e17.x86_64 需要
8 --> 正在检查事务
```

```
9  ---> 软件包 libstdc++-
    devel.x86_64.0.4.8.5-44.el7 将被 安装
10  --> 解决依赖关系完成
11
12  依赖关系解决
13
14  =====
    =====
    =====
    =====
15  Package
    架构                                版本
                                源
                                大小
16  =====
    =====
    =====
    =====
17  正在安装:
18  gcc-c++
    x86_64                                4.8.5-
    44.el7                                base
                                7.2 M
19  为依赖而安装:
20  libstdc++-devel
    x86_64                                4.8.5-
    44.el7                                base
                                1.5 M
21
22  事务概要
```

23 =====
=====
=====
=====

24 安装 1 软件包 (+1 依赖软件包)

25

26 总下载量: 8.7 M

27 安装大小: 25 M

28 Downloading packages:

29 (1/2): libstdc++-devel-4.8.5-
44.el7.x86_64.rpm

| 1.5 MB 00:00:02

30 (2/2): gcc-c++-4.8.5-44.el7.x86_64.rpm

|

7.2 MB 00:00:05

31 -----

32 总计

1.5 MB/s |

8.7 MB 00:00:05

33 Running transaction check

34 Running transaction test

35 Transaction test succeeded

36 Running transaction

37 正在安装 : libstdc++-devel-4.8.5-
44.e17.x86_64

1/2

38 正在安装 : gcc-c++-4.8.5-
44.e17.x86_64

2/2

39 验证中 : gcc-c++-4.8.5-
44.e17.x86_64

1/2

40 验证中 : libstdc++-devel-4.8.5-
44.e17.x86_64

2/2

41

42 已安装:

43 gcc-c++.x86_64 0:4.8.5-44.e17

44

45 作为依赖被安装:

46 libstdc++-devel.x86_64 0:4.8.5-44.e17

47

48 完毕!

1.6 安装软件包

```
1 [root@localhost yum.repos.d]# yum -y
install yum-utils
2 已加载插件：fastestmirror
3 Loading mirror speeds from cached
hostfile
4 正在解决依赖关系
5 --> 正在检查事务
6 ---> 软件包 yum-utils.noarch.0.1.1.31-
54.el7_8 将被 安装
7 --> 正在处理依赖关系 python-kitchen，它被软件
包 yum-utils-1.1.31-54.el7_8.noarch 需要
8 --> 正在处理依赖关系 libxml2-python，它被软件
包 yum-utils-1.1.31-54.el7_8.noarch 需要
9 --> 正在检查事务
10 ---> 软件包 libxml2-
python.x86_64.0.2.9.1-6.el7_9.6 将被 安装
11 ---> 软件包 python-
kitchen.noarch.0.1.1.1-5.el7 将被 安装
12 --> 正在处理依赖关系 python-chardet，它被软件
包 python-kitchen-1.1.1-5.el7.noarch 需要
13 --> 正在检查事务
14 ---> 软件包 python-
chardet.noarch.0.2.2.1-3.el7 将被 安装
15 --> 解决依赖关系完成
16
17 依赖关系解决
```

18				
19				
20	Package			架构
		版本		
		源		
	大小			
21				
22	正在安装:			
23	yum-utils			
	noarch		1.1.31-	
	54.e17_8		base	
		122 k		
24	为依赖而安装:			
25	libxml2-python			
	x86_64		2.9.1-	
	6.e17_9.6			
	updates		247 k	
26	python-chardet			
	noarch		2.2.1-3.e17	
			base	
		227 k		


```
27  python-kitchen
    noarch                                1.1.1-5.e17
                                         base
                                         267 k
28
29  事务概要
30  =====
    =====
    =====
    =====
31  安装   1  软件包 (+3  依赖软件包)
32
33  总下载量: 863 k
34  安装大小: 4.3 M
35  Downloading packages:
36  (1/4): python-chardet-2.2.1-
    3.e17.noarch.rpm
           | 227 kB  00:00:00
37  (2/4): libxml2-python-2.9.1-
    6.e17_9.6.x86_64.rpm
           | 247 kB  00:00:00
38  (3/4): python-kitchen-1.1.1-
    5.e17.noarch.rpm
           | 267 kB  00:00:00
```

39 (4/4): yum-utils-1.1.31-
54.el7_8.noarch.rpm

| 122 kB 00:00:00

40

41 总计

1.9 MB/s |

863 kB 00:00:00

42 Running transaction check

43 Running transaction test

44 Transaction test succeeded

45 Running transaction

46 正在安装 : python-chardet-2.2.1-
3.el7.noarch

1/4

47 正在安装 : python-kitchen-1.1.1-
5.el7.noarch

2/4

48 正在安装 : libxml2-python-2.9.1-
6.el7_9.6.x86_64

3/4

49 正在安装 : yum-utils-1.1.31-
54.e17_8.noarch

4/4

50 验证中 : python-kitchen-1.1.1-
5.e17.noarch

1/4

51 验证中 : yum-utils-1.1.31-
54.e17_8.noarch

2/4

52 验证中 : libxml2-python-2.9.1-
6.e17_9.6.x86_64

3/4

53 验证中 : python-chardet-2.2.1-
3.e17.noarch

4/4

54

55 已安装:

56 yum-utils.noarch 0:1.1.31-54.e17_8

57

58 作为依赖被安装:

```
59    libxml2-python.x86_64 0:2.9.1-  
    6.e17_9.6                python-chardet.noarch  
    0:2.2.1-3.e17            python-  
    kitchen.noarch 0:1.1.1-5.e17  
60  
61 完毕!  
62 [root@localhost yum.repos.d]#  
63
```

1.7 设置Stable仓库

```
1 [root@localhost yum.repos.d]# yum-config-  
  manager --add-repo  
  http://mirrors.aliyun.com/docker-  
  ce/linux/centos/docker-ce.repo  
2 已加载插件: fastestmirror  
3 adding repo from:  
  http://mirrors.aliyun.com/docker-  
  ce/linux/centos/docker-ce.repo  
4 grabbing file  
  http://mirrors.aliyun.com/docker-  
  ce/linux/centos/docker-ce.repo to  
  /etc/yum.repos.d/docker-ce.repo  
5 repo saved to /etc/yum.repos.d/docker-  
  ce.repo  
6 [root@localhost yum.repos.d]#
```

1.8 更新 YUM 软件包索引

```
1 [root@localhost yum.repos.d]# yum
  makecache fast
2 已加载插件: fastestmirror
3 Loading mirror speeds from cached
  hostfile
4 base
                                     |
    3.6 kB  00:00:00
5 docker-ce-stable
                                     |
    3.5 kB  00:00:00
6 extras
                                     |
    2.9 kB  00:00:00
7 updates
                                     |
    2.9 kB  00:00:00
8 (1/2): docker-ce-
  stable/7/x86_64/updateinfo
                                     |
                                     55 B  00:00:00
```

```
9 (2/2): docker-ce-  
stable/7/x86_64/primary_db
```

```
| 91 kB 00:00:00
```

```
10 元数据缓存已建立
```

```
11 [root@localhost yum.repos.d]#
```

```
12
```

1.9 查看 YUM 中 Docker 版本

```
1 [root@localhost yum.repos.d]# yum list  
docker-ce --showduplicates | sort -r
```

```
2 已加载插件: fastestmirror
```

```
3 可安装的软件包
```

```
4 Loading mirror speeds from cached  
hostfile
```

```
5 docker-ce.x86_64          3:20.10.9-  
3.e17                      docker-ce-  
stable
```

```
6 docker-ce.x86_64          3:20.10.8-  
3.e17                      docker-ce-  
stable
```

```
7 docker-ce.x86_64          3:20.10.7-  
3.e17                      docker-ce-  
stable
```

```
8 docker-ce.x86_64          3:20.10.6-  
3.e17                      docker-ce-  
stable
```

9	docker-ce.x86_64 3.e17 stable	3:20.10.5- docker-ce-
10	docker-ce.x86_64 3.e17 stable	3:20.10.4- docker-ce-
11	docker-ce.x86_64 3.e17 stable	3:20.10.3- docker-ce-
12	docker-ce.x86_64 3.e17 stable	3:20.10.2- docker-ce-
13	docker-ce.x86_64 3.e17 stable	3:20.10.22- docker-ce-
14	docker-ce.x86_64 3.e17 stable	3:20.10.21- docker-ce-
15	docker-ce.x86_64 3.e17 stable	3:20.10.20- docker-ce-
16	docker-ce.x86_64 3.e17 stable	3:20.10.19- docker-ce-
17	docker-ce.x86_64 3.e17 stable	3:20.10.18- docker-ce-
18	docker-ce.x86_64 3.e17 stable	3:20.10.17- docker-ce-

19	docker-ce.x86_64 3.e17 stable	3:20.10.16- docker-ce-
20	docker-ce.x86_64 3.e17 stable	3:20.10.15- docker-ce-
21	docker-ce.x86_64 3.e17 stable	3:20.10.14- docker-ce-
22	docker-ce.x86_64 3.e17 stable	3:20.10.1- docker-ce-
23	docker-ce.x86_64 3.e17 stable	3:20.10.13- docker-ce-
24	docker-ce.x86_64 3.e17 stable	3:20.10.12- docker-ce-
25	docker-ce.x86_64 3.e17 stable	3:20.10.11- docker-ce-
26	docker-ce.x86_64 3.e17 stable	3:20.10.10- docker-ce-
27	docker-ce.x86_64 3.e17 stable	3:20.10.0- docker-ce-
28	docker-ce.x86_64 3.e17 stable	3:19.03.9- docker-ce-

29	docker-ce.x86_64 3.e17 stable	3:19.03.8- docker-ce-
30	docker-ce.x86_64 3.e17 stable	3:19.03.7- docker-ce-
31	docker-ce.x86_64 3.e17 stable	3:19.03.6- docker-ce-
32	docker-ce.x86_64 3.e17 stable	3:19.03.5- docker-ce-
33	docker-ce.x86_64 3.e17 stable	3:19.03.4- docker-ce-
34	docker-ce.x86_64 3.e17 stable	3:19.03.3- docker-ce-
35	docker-ce.x86_64 3.e17 stable	3:19.03.2- docker-ce-
36	docker-ce.x86_64 3.e17 stable	3:19.03.15- docker-ce-
37	docker-ce.x86_64 3.e17 stable	3:19.03.14- docker-ce-
38	docker-ce.x86_64 3.e17 stable	3:19.03.1- docker-ce-

39	docker-ce.x86_64 3.e17 stable	3:19.03.13- docker-ce-
40	docker-ce.x86_64 3.e17 stable	3:19.03.12- docker-ce-
41	docker-ce.x86_64 3.e17 stable	3:19.03.11- docker-ce-
42	docker-ce.x86_64 3.e17 stable	3:19.03.10- docker-ce-
43	docker-ce.x86_64 3.e17 stable	3:19.03.0- docker-ce-
44	docker-ce.x86_64 3.e17 stable	3:18.09.9- docker-ce-
45	docker-ce.x86_64 3.e17 stable	3:18.09.8- docker-ce-
46	docker-ce.x86_64 3.e17 stable	3:18.09.7- docker-ce-
47	docker-ce.x86_64 3.e17 stable	3:18.09.6- docker-ce-
48	docker-ce.x86_64 3.e17 stable	3:18.09.5- docker-ce-

49	docker-ce.x86_64 3.e17 stable	3:18.09.4- docker-ce-
50	docker-ce.x86_64 3.e17 stable	3:18.09.3- docker-ce-
51	docker-ce.x86_64 3.e17 stable	3:18.09.2- docker-ce-
52	docker-ce.x86_64 3.e17 stable	3:18.09.1- docker-ce-
53	docker-ce.x86_64 3.e17 stable	3:18.09.0- docker-ce-
54	docker-ce.x86_64 3.e17 stable	18.06.3.ce- docker-ce-
55	docker-ce.x86_64 3.e17 stable	18.06.2.ce- docker-ce-
56	docker-ce.x86_64 3.e17 stable	18.06.1.ce- docker-ce-
57	docker-ce.x86_64 3.e17 stable	18.06.0.ce- docker-ce-
58	docker-ce.x86_64 1.e17.centos stable	18.03.1.ce- docker-ce-

59	docker-ce.x86_64 1.e17.centos stable	18.03.0.ce- docker-ce-
60	docker-ce.x86_64 1.e17.centos stable	17.12.1.ce- docker-ce-
61	docker-ce.x86_64 1.e17.centos stable	17.12.0.ce- docker-ce-
62	docker-ce.x86_64 1.e17.centos stable	17.09.1.ce- docker-ce-
63	docker-ce.x86_64 1.e17.centos stable	17.09.0.ce- docker-ce-
64	docker-ce.x86_64 1.e17.centos stable	17.06.2.ce- docker-ce-
65	docker-ce.x86_64 1.e17.centos stable	17.06.1.ce- docker-ce-
66	docker-ce.x86_64 1.e17.centos stable	17.06.0.ce- docker-ce-
67	docker-ce.x86_64 1.e17 stable	17.03.3.ce- docker-ce-
68	docker-ce.x86_64 1.e17.centos stable	17.03.2.ce- docker-ce-

```
69 docker-ce.x86_64          17.03.1.ce-  
   1.e17.centos              docker-ce-  
   stable  
70 docker-ce.x86_64          17.03.0.ce-  
   1.e17.centos              docker-ce-  
   stable  
71 [root@localhost yum.repos.d]#  
72
```

1.10 安装 Docker Engine

```
1 [root@localhost yum.repos.d]# sudo yum  
  install docker-ce docker-ce-cli  
  containerd.io docker-compose-plugin  
2 已加载插件：fastestmirror  
3 Loading mirror speeds from cached  
  hostfile  
4 正在解决依赖关系  
5 --> 正在检查事务  
6 ---> 软件包  
   containerd.io.x86_64.0.1.6.14-3.1.e17 将  
   被 安装  
7 --> 正在处理依赖关系 container-selinux >=  
   2:2.74, 它被软件包 containerd.io-1.6.14-  
   3.1.e17.x86_64 需要  
8 ---> 软件包 docker-ce.x86_64.3.20.10.22-  
   3.e17 将被 安装
```

```
9 --> 正在处理依赖关系 docker-ce-rootless-  
extras, 它被软件包 3:docker-ce-20.10.22-  
3.el7.x86_64 需要  
10 --> 正在处理依赖关系 libcgroup, 它被软件包  
3:docker-ce-20.10.22-3.el7.x86_64 需要  
11 ---> 软件包 docker-ce-  
cli.x86_64.1.20.10.22-3.el7 将被 安装  
12 --> 正在处理依赖关系 docker-scan-  
plugin(x86-64), 它被软件包 1:docker-ce-  
cli-20.10.22-3.el7.x86_64 需要  
13 ---> 软件包 docker-compose-  
plugin.x86_64.0.2.14.1-3.el7 将被 安装  
14 --> 正在检查事务  
15 ---> 软件包 container-  
selinux.noarch.2.2.119.2-  
1.911c772.el7_8 将被 安装  
16 --> 正在处理依赖关系 policycoreutils-  
python, 它被软件包 2:container-selinux-  
2.119.2-1.911c772.el7_8.noarch 需要  
17 ---> 软件包 docker-ce-rootless-  
extras.x86_64.0.20.10.22-3.el7 将被 安装  
18 --> 正在处理依赖关系 fuse-overlayfs >=  
0.7, 它被软件包 docker-ce-rootless-  
extras-20.10.22-3.el7.x86_64 需要  
19 --> 正在处理依赖关系 slirp4netns >= 0.4, 它  
被软件包 docker-ce-rootless-extras-  
20.10.22-3.el7.x86_64 需要  
20 ---> 软件包 docker-scan-  
plugin.x86_64.0.0.23.0-3.el7 将被 安装
```

```
21 ---> 软件包 libcgrouper.x86_64.0.0.41-  
21.e17 将被 安装  
22 --> 正在检查事务  
23 ---> 软件包 fuse-  
overlayfs.x86_64.0.0.7.2-6.e17_8 将被 安  
装  
24 --> 正在处理依赖关系  
libfuse3.so.3(FUSE_3.2)(64bit), 它被软件  
包 fuse-overlayfs-0.7.2-6.e17_8.x86_64  
需要  
25 --> 正在处理依赖关系  
libfuse3.so.3(FUSE_3.0)(64bit), 它被软件  
包 fuse-overlayfs-0.7.2-6.e17_8.x86_64  
需要  
26 --> 正在处理依赖关系 libfuse3.so.3(  
(64bit), 它被软件包 fuse-overlayfs-0.7.2-  
6.e17_8.x86_64 需要  
27 ---> 软件包 polycoreutils-  
python.x86_64.0.2.5-34.e17 将被 安装  
28 --> 正在处理依赖关系 setools-libs >=  
3.3.8-4, 它被软件包 polycoreutils-  
python-2.5-34.e17.x86_64 需要  
29 --> 正在处理依赖关系 libsemanage-python >=  
2.5-14, 它被软件包 polycoreutils-  
python-2.5-34.e17.x86_64 需要  
30 --> 正在处理依赖关系 audit-libs-python >=  
2.1.3-4, 它被软件包 polycoreutils-  
python-2.5-34.e17.x86_64 需要
```

```
31 --> 正在处理依赖关系 python-IPy, 它被软件包
    polycoreutils-python-2.5-
    34.el7.x86_64 需要
32 --> 正在处理依赖关系
    libqpol.so.1(VERS_1.4)(64bit), 它被软件包
    polycoreutils-python-2.5-
    34.el7.x86_64 需要
33 --> 正在处理依赖关系
    libqpol.so.1(VERS_1.2)(64bit), 它被软件包
    polycoreutils-python-2.5-
    34.el7.x86_64 需要
34 --> 正在处理依赖关系
    libapol.so.4(VERS_4.0)(64bit), 它被软件包
    polycoreutils-python-2.5-
    34.el7.x86_64 需要
35 --> 正在处理依赖关系 checkpolicy, 它被软件包
    polycoreutils-python-2.5-
    34.el7.x86_64 需要
36 --> 正在处理依赖关系 libqpol.so.1()
    (64bit), 它被软件包 polycoreutils-
    python-2.5-34.el7.x86_64 需要
37 --> 正在处理依赖关系 libapol.so.4()
    (64bit), 它被软件包 polycoreutils-
    python-2.5-34.el7.x86_64 需要
38 ---> 软件包 slirp4netns.x86_64.0.0.4.3-
    4.el7_8 将被 安装
39 --> 正在检查事务
40 ---> 软件包 audit-libs-
    python.x86_64.0.2.8.5-4.el7 将被 安装
```



```
41 ---> 软件包 checkpolicy.x86_64.0.2.5-
    8.el7 将被 安装
42 ---> 软件包 fuse3-libs.x86_64.0.3.6.1-
    4.el7 将被 安装
43 ---> 软件包 libsemanage-
    python.x86_64.0.2.5-14.el7 将被 安装
44 ---> 软件包 python-IPy.noarch.0.0.75-
    6.el7 将被 安装
45 ---> 软件包 setools-libs.x86_64.0.3.3.8-
    4.el7 将被 安装
46 --> 解决依赖关系完成
47
48 依赖关系解决
49
50 =====
    =====
    =====
    =====
51 Package
    架构                                版本
                                         源
                                大小
52 =====
    =====
    =====
    =====
53 正在安装:
```

54	containerd.io		
	x86_64	1.6.14-3.1.e17	
		docker-ce-	
	stable	33	M
55	docker-ce		
	x86_64	3:20.10.22-3.e17	
		docker-ce-	
	stable	22	M
56	docker-ce-cli		
	x86_64	1:20.10.22-3.e17	
		docker-ce-	
	stable	30	M
57	docker-compose-plugin		
	x86_64	2.14.1-3.e17	
		docker-ce-	
	stable	10	M
58	为依赖而安装:		
59	audit-libs-python		
	x86_64	2.8.5-4.e17	
		base	
		76	k
60	checkpolicy		
	x86_64	2.5-8.e17	
		base	
		295	k
61	container-selinux		
	noarch	2:2.119.2-	
	1.911c772.e17_8		extras
		40	k

62	docker-ce-rootless-extras	
	x86_64	20.10.22-3.e17
		docker-ce-
	stable	8.5 M
63	docker-scan-plugin	
	x86_64	0.23.0-3.e17
		docker-ce-stable
		3.8 M
64	fuse-overlayfs	
	x86_64	0.7.2-6.e17_8
		extras
		54 k
65	fuse3-libs	
	x86_64	3.6.1-4.e17
		extras
		82 k
66	libcgroup	
	x86_64	0.41-21.e17
		base
		66 k
67	libsemanage-python	
	x86_64	2.5-14.e17
		base
		113 k
68	policycoreutils-python	
	x86_64	2.5-34.e17
		base
		457 k

```
69  python-IPy
    noarch                                0.75-6.e17
                                         base
                                         32 k
70  setools-libs
    x86_64                                3.3.8-4.e17
                                         base
                                         620 k
71  slirp4netns
    x86_64                                0.4.3-4.e17_8
                                         extras
                                         81 k
72
73  事务概要
74  =====
    =====
    =====
    =====
75  安装   4  软件包 (+13 依赖软件包)
76
77  总下载量: 109 M
78  安装大小: 404 M
79  Is this ok [y/d/N]: y
80  Downloading packages:
81  (1/17): container-selinux-2.119.2-
    1.911c772.e17_8.noarch.rpm

    |   40 kB   00:00:00
```

```
82 (2/17): audit-libs-python-2.8.5-  
4.el7.x86_64.rpm  
  
      | 76 kB 00:00:00  
83 (3/17): checkpolicy-2.5-  
8.el7.x86_64.rpm  
  
      | 295 kB 00:00:00  
84 warning:  
/var/cache/yum/x86_64/7/docker-ce-  
stable/packages/docker-ce-20.10.22-  
3.el7.x86_64.rpm: Header V4 RSA/SHA512  
Signature, key ID 621e9f35: NOKEY  
85 docker-ce-20.10.22-3.el7.x86_64.rpm 的公  
钥尚未安装  
86 (4/17): docker-ce-20.10.22-  
3.el7.x86_64.rpm  
  
      | 22 MB 00:00:13  
87 (5/17): containerd.io-1.6.14-  
3.1.el7.x86_64.rpm  
  
      | 33 MB 00:00:20  
88 (6/17): docker-ce-rootless-extras-  
20.10.22-3.el7.x86_64.rpm  
  
      | 8.5 MB 00:00:04
```

```
89 (7/17): docker-compose-plugin-2.14.1-
    3.el7.x86_64.rpm
    | 10 MB 00:00:04
90 (8/17): fuse-overlayfs-0.7.2-
    6.el7_8.x86_64.rpm
    | 54 kB 00:00:00
91 (9/17): libcgroup-0.41-
    21.el7.x86_64.rpm
    | 66 kB 00:00:00
92 (10/17): fuse3-libs-3.6.1-
    4.el7.x86_64.rpm
    | 82 kB 00:00:00
93 (11/17): libsemanage-python-2.5-
    14.el7.x86_64.rpm
    | 113 kB 00:00:00
94 (12/17): python-IPy-0.75-
    6.el7.noarch.rpm
    | 32 kB 00:00:00
95 (13/17): slirp4netns-0.4.3-
    4.el7_8.x86_64.rpm
    | 81 kB 00:00:00
```

```
96 (14/17): setools-libs-3.3.8-
    4.el7.x86_64.rpm
        | 620 kB 00:00:00
97 (15/17): polycoreutils-python-2.5-
    34.el7.x86_64.rpm
        | 457 kB 00:00:00
98 (16/17): docker-ce-cli-20.10.22-
    3.el7.x86_64.rpm
        | 30 MB 00:00:17
99 (17/17): docker-scan-plugin-0.23.0-
    3.el7.x86_64.rpm
        | 3.8 MB 00:00:01
100 -----
    -----
    -----
    -----
101 总计
        3.4 MB/s |
    109 MB 00:00:32
102 从 https://mirrors.aliyun.com/docker-
    ce/linux/centos/gpg 检索密钥
103 导入 GPG key 0x621E9F35:
104 用户ID      : "Docker Release (CE rpm)
    <docker@docker.com>"
```

```
105 指纹          : 060a 61c5 1b55 8a7f 742b
    77aa c52f eb6b 621e 9f35
106 来自          :
    https://mirrors.aliyun.com/docker-
    ce/linux/centos/gpg
107 是否继续? [y/N]: y
108 Running transaction check
109 Running transaction test
110 Transaction test succeeded
111 Running transaction
112 正在安装      : libcgrou-0.41-
    21.el7.x86_64

                                1/17
113 正在安装      : setools-libs-3.3.8-
    4.el7.x86_64

                                2/17
114 正在安装      : audit-libs-python-2.8.5-
    4.el7.x86_64

                                3/17
115 正在安装      : fuse3-libs-3.6.1-
    4.el7.x86_64

                                4/17
116 正在安装      : fuse-overlayfs-0.7.2-
    6.el7_8.x86_64
```


117 正在安装 : checkpolicy-2.5-
8.el7.x86_64

6/17

118 正在安装 : python-IPy-0.75-
6.el7.noarch

7/17

119 正在安装 : slirp4netns-0.4.3-
4.el7_8.x86_64

8/17

120 正在安装 : libsemanage-python-2.5-
14.el7.x86_64

9/17

121 正在安装 : polycoreutils-python-
2.5-34.el7.x86_64

10/17

122 正在安装 : 2:container-selinux-
2.119.2-1.911c772.el7_8.noarch

11/17

123 正在安装 : containerd.io-1.6.14-
3.1.el7.x86_64

12/17

124 正在安装 : docker-scan-plugin-
0.23.0-3.el7.x86_64

13/17

125 正在安装 : 1:docker-ce-cli-20.10.22-
3.el7.x86_64

14/17

126 正在安装 : docker-ce-rootless-
extras-20.10.22-3.el7.x86_64

15/17

127 正在安装 : 3:docker-ce-20.10.22-
3.el7.x86_64

16/17

128 正在安装 : docker-compose-plugin-
2.14.1-3.el7.x86_64

17/17

129 验证中 : 3:docker-ce-20.10.22-
3.el7.x86_64

1/17

130 验证中 : docker-scan-plugin-
0.23.0-3.el7.x86_64

2/17

131 验证中 : docker-ce-rootless-
extras-20.10.22-3.el7.x86_64

3/17

132 验证中 : fuse-overlayfs-0.7.2-
6.el7_8.x86_64

4/17

133 验证中 : libsemanage-python-2.5-
14.el7.x86_64

5/17

134 验证中 : slirp4netns-0.4.3-
4.el7_8.x86_64

6/17

135 验证中 : 2:container-selinux-
2.119.2-1.911c772.el7_8.noarch

7/17

136 验证中 : 1:docker-ce-cli-
20.10.22-3.el7.x86_64

8/17

137 验证中 : python-IPy-0.75-
6.el7.noarch

9/17

138 验证中 : checkpolicy-2.5-
8.e17.x86_64

10/17

139 验证中 : containerd.io-1.6.14-
3.1.e17.x86_64

11/17

140 验证中 : polycoreutils-python-
2.5-34.e17.x86_64

12/17

141 验证中 : fuse3-libs-3.6.1-
4.e17.x86_64

13/17

142 验证中 : audit-libs-python-2.8.5-
4.e17.x86_64

14/17

143 验证中 : setools-libs-3.3.8-
4.e17.x86_64

15/17

144 验证中 : docker-compose-plugin-
2.14.1-3.e17.x86_64

16/17

```
145 验证中      : libcgroup-0.41-  
      21.e17.x86_64
```

17/17

146

147 已安装：

```
148 containerd.io.x86_64 0:1.6.14-3.1.e17
      docker-ce.x86_64
```

```
3:20.10.22-3.e17          docker-ce-
cli.x86_64 1:20.10.22-3.e17
```

```
149     docker-compose-plugin.x86_64
0:2.14.1-3.el7
```

150

151 作为依赖被安装:

```
152 audit-libs-python.x86_64 0:2.8.5-4.e17
```

checkpolicy.x86_64 0:2.5-8.e17

```
153 container-selinux.noarch 2:2.119.2-  
1.911c772.el7_8
```

```
docker-ce-rootless-extras.x86_64
0:20.10.22-3.el7
```

```
154     docker-scan-plugin.x86_64 0:0.23.0-3.e17
```

fuse-overlayfs.x86_64 0:0.7.2-6.el7_8

```
155 fuse3-libs.x86_64 0:3.6.1-4.e17
```

libcgrouper.x86_64 0:0.41-21.e17

```
156    libsemanage-python.x86_64 0:2.5-  
    14.el7  
    policycoreutils-python.x86_64 0:2.5-  
    34.el7  
157    python-IPy.noarch 0:0.75-6.el7  
  
    setools-libs.x86_64 0:3.3.8-4.el7  
  
158    slirp4netns.x86_64 0:0.4.3-4.el7_8  
  
159  
160 完毕!  
161 [root@localhost yum.repos.d]#  
162
```

This command installs Docker, but it doesn't start Docker. It also creates a group, however, it doesn't add any users to the group by default. `docker`

意思是默认情况会创建一个Group：组，组的名字是docker，但是一般不会向该组中创建任何用户。

1.11 启动 Docker

```
1  # 首先查看是否有docker进程在执行。  
2  [root@localhost yum.repos.d]# ps -  
    ef|grep docker  
3  root          1701    1401    0 19:53 pts/0  
    00:00:00 grep --color=auto docker
```

```
4 [root@localhost yum.repos.d]#
5
6
7 # 上面没有任何docker进程在执行中，启动docker
8 [root@localhost yum.repos.d]# systemctl
start docker
9 [root@localhost yum.repos.d]#
10
11 # 查看docker的服务状态
12 [root@localhost yum.repos.d]# systemctl
status docker
13 • docker.service - Docker Application
Container Engine
14     Loaded: loaded
(/usr/lib/systemd/system/docker.service;
disabled; vendor preset: disabled)
15     Active: active (running) since 三
2022-12-28 19:54:21 CST; 28s ago
16     Docs: https://docs.docker.com
17     Main PID: 1723 (dockerd)
18     Tasks: 7
19     Memory: 26.7M
20     CGroup: /system.slice/docker.service
21             └─1723 /usr/bin/dockerd -H
fd:// --
containerd=/run/containerd/containerd.so
ck
22
```

23 12月 28 19:54:21 localhost.localdomain
dockerd[1723]: time="2022-12-
28T19:54:21.293262130+08:00" level=info
msg="ccResolverWrapper...e=grpc

24 12月 28 19:54:21 localhost.localdomain
dockerd[1723]: time="2022-12-
28T19:54:21.293268403+08:00" level=info
msg="ClientConn switch...e=grpc

25 12月 28 19:54:21 localhost.localdomain
dockerd[1723]: time="2022-12-
28T19:54:21.304860903+08:00" level=info
msg="Loading container...tart."

26 12月 28 19:54:21 localhost.localdomain
dockerd[1723]: time="2022-12-
28T19:54:21.781428044+08:00" level=info
msg="Default bridge (d...dress"

27 12月 28 19:54:21 localhost.localdomain
dockerd[1723]: time="2022-12-
28T19:54:21.876745895+08:00" level=info
msg="Firewalld: interf...rning"

28 12月 28 19:54:21 localhost.localdomain
dockerd[1723]: time="2022-12-
28T19:54:21.954617444+08:00" level=info
msg="Loading containers: done."

29 12月 28 19:54:21 localhost.localdomain
dockerd[1723]: time="2022-12-
28T19:54:21.965726287+08:00" level=info
msg="Docker daemon" co....10.22


```
30 12月 28 19:54:21 localhost.localdomain
    dockerd[1723]: time="2022-12-
    28T19:54:21.966346290+08:00" level=info
    msg="Daemon has comple...ation"
31 12月 28 19:54:21 localhost.localdomain
    systemd[1]: Started Docker Application
    Container Engine.
32 12月 28 19:54:21 localhost.localdomain
    dockerd[1723]: time="2022-12-
    28T19:54:21.979682454+08:00" level=info
    msg="API listen on /va....sock"
33 Hint: Some lines were ellipsized, use -l
    to show in full.
34 [root@localhost yum.repos.d]#
35
36 # 再次查看进程信息
37 [root@localhost yum.repos.d]# ps -
    ef|grep docker
38 root          1723          1   0 19:54 ?
    00:00:00 /usr/bin/dockerd -H fd:// --
    containerd=/run/containerd/containerd.so
    ck
39 root          1858    1401   0 19:55 pts/0
    00:00:00 grep --color=auto docker
40 [root@localhost yum.repos.d]#
41
42
```

1.12 查看 Docker信息

```
1 [root@localhost yum.repos.d]# docker
version
2 Client: Docker Engine - Community
3   Version:           20.10.22
4   API version:       1.41
5   Go version:        go1.18.9
6   Git commit:        3a2c30b
7   Built:             Thu Dec 15 22:30:24
                        2022
8   OS/Arch:           linux/amd64
9   Context:           default
10  Experimental:      true
11
12 Server: Docker Engine - Community
13   Engine:
14     Version:           20.10.22
15     API version:       1.41 (minimum
version 1.12)
16     Go version:        go1.18.9
17     Git commit:        42c8b31
18     Built:             Thu Dec 15 22:28:33
                        2022
19     OS/Arch:           linux/amd64
20     Experimental:     false
21   containerd:
22     Version:           1.6.14
```

```

23   GitCommit:
    9ba4b250366a5ddde94bb7c9d1def331423aa323
24   runc:
25     Version:           1.1.4
26     GitCommit:         v1.1.4-0-g5fd4c4d
27   docker-init:
28     Version:           0.19.0
29     GitCommit:         de40ad0
30 [root@localhost yum.repos.d]#
31

```

1.13 运行 HelloWorld

首先查看本地有哪些镜像

```

1 [root@localhost yum.repos.d]# docker
  images -a
2 REPOSITORY      TAG                IMAGE ID           CREATED
  SIZE
3 [root@localhost yum.repos.d]#
4

```

查看当前是否有容器运行中

```

1 [root@localhost yum.repos.d]# docker ps -
  a
2 CONTAINER ID    IMAGE             COMMAND
  CREATED        STATUS           PORTS          NAMES
3 [root@localhost yum.repos.d]#

```

没有镜像，容器。运行HelloWorld，因为本地没有镜像，所以会先下载然后再运行！

```
1 [root@localhost yum.repos.d]# docker run
  hello-world
2 Unable to find image 'hello-
  world:latest' locally
3 latest: Pulling from library/hello-world
4 2db29710123e: Pull complete
5 Digest:
  sha256:c77be1d3a47d0caf71a82dd893ee61ce0
  1f32fc758031a6ec4cf1389248bb833
6 Status: Downloaded newer image for
  hello-world:latest
7
8 Hello from Docker!
9 This message shows that your
  installation appears to be working
  correctly.
10
11 To generate this message, Docker took
  the following steps:
12 1. The Docker client contacted the
  Docker daemon.
13 2. The Docker daemon pulled the "hello-
  world" image from the Docker Hub.
14    (amd64)
15 3. The Docker daemon created a new
  container from that image which runs the
```

```
16     executable that produces the output
    you are currently reading.
17  4. The Docker daemon streamed that
    output to the Docker client, which sent
    it
18     to your terminal.
19
20 To try something more ambitious, you can
    run an Ubuntu container with:
21  $ docker run -it ubuntu bash
22
23 Share images, automate workflows, and
    more with a free Docker ID:
24  https://hub.docker.com/
25
26 For more examples and ideas, visit:
27  https://docs.docker.com/get-started/
28
29 [root@localhost yum.repos.d]#
```

再次查看信息

```

1 [root@localhost yum.repos.d]# docker ps -
  a
2 CONTAINER ID    IMAGE                COMMAND
   CREATED        STATUS
     PORTS        NAMES
3 87d73c80b8cf    hello-world         "/hello"
   30 seconds ago  Exited (0) 30 seconds
   ago            recursing_pasca1
4 [root@localhost yum.repos.d]#
5

```

```

1 [root@localhost yum.repos.d]# docker
  images -a
2 REPOSITORY      TAG                IMAGE ID
   CREATED        SIZE
3 hello-world     latest            feb5d9fea6a5    15
   months ago    13.3kB
4 [root@localhost yum.repos.d]#
5

```

1.14 开启阿里云镜像加速

```
1 sudo tee /etc/docker/daemon.json <<- 'EOF'
2 {
3     "registry-mirrors":
4     ["https://0nws4w1f.mirror.aliyuncs.com"]
5 }
6 EOF
```

```
1 [root@localhost yum.repos.d]# mkdir -p
   /etc/docker
2 [root@localhost yum.repos.d]# sudo tee
   /etc/docker/daemon.json <<- 'EOF'
3 > {
4 >     "registry-mirrors":
5 >     ["https://0nws4w1f.mirror.aliyuncs.com"]
6 > }
7 > EOF
8 {
9     "registry-mirrors":
10    ["https://0nws4w1f.mirror.aliyuncs.com"]
11 }
```

```
1 [root@localhost ~]# systemctl daemon-
  reload
2 [root@localhost ~]# systemctl restart
  docker
3 [root@localhost ~]# docker run hello-
  world
4
5 Hello from Docker!
6 This message shows that your
  installation appears to be working
  correctly.
7
8 To generate this message, Docker took
  the following steps:
9   1. The Docker client contacted the
  Docker daemon.
10  2. The Docker daemon pulled the "hello-
  world" image from the Docker Hub.
11     (amd64)
12  3. The Docker daemon created a new
  container from that image which runs the
13     executable that produces the output
  you are currently reading.
14  4. The Docker daemon streamed that
  output to the Docker client, which sent
  it
15     to your terminal.
16
17 To try something more ambitious, you can
  run an Ubuntu container with:
```



```
18 $ docker run -it ubuntu bash
```

```
19
```

```
20 Share images, automate workflows, and  
   more with a free Docker ID:
```

```
21 https://hub.docker.com/
```

```
22
```

```
23 For more examples and ideas, visit:
```

```
24 https://docs.docker.com/get-started/
```

```
25
```

```
26 [root@localhost ~]#
```

```
27
```