

KOOL
SHARE

01. `reboot`
[复制代码](#)

```
Please press Enter to activate this console.
```

```
BusyBox v1.30.0 () built-in shell (ash)
```

```

 _   _          _ 
| | | | _   _  / \  
| |_| | | | | / _ \ 
|  __/| |_| |/_\_\_ 
|_|   |_|_|_|___\___
      W I R E L E S S   F R E E D O M

```

```
-----
HomeLede v2020.05.01 based on OpenWrt R20.4.8
-----
```

```
root@OpenWrt:/# reboot
root@OpenWrt:/# [ 1216.959580] br-lan: port 1(eth0) entered disabled state
[ 1216.961683] device eth0 left promiscuous mode
[ 1216.963010] br-lan: port 1(eth0) entered disabled state
[ 1217.179303] IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready
```

KOOL SHARE

koolshare.cn

高级 TCP/IP 设置

IP 设置 DNS WINS

IP 地址(R)

IP 地址	子网掩码
192.168.1.240	255.255.255.0
192.168.66.240	255.255.255.0

默认网关(F):

```

Last login: Sun Aug 2 07:55:27 from 192.168.5.1
[root@k8s-node02 ~]# vi /etc/sysconfig/network-scripts/ifcfg-ens33

```

```
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=static
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=f81d9407-0825-4681-9358-eae19f5a9f23
DEVICE=ens33
ONBOOT=yes
IPADDR=192.168.66.20
NETMASK=255.255.255.0
GATEWAY=192.168.66.1
DNS1=192.168.66.1
DNS2=114.114.114.114
```

```
[root@k8s-node02 ~]# reboot_
```

Ping www.baidu.com

5. 配置 docker

先配置好视频的再按一下来

<http://www.jufanshare.com/content/188.html>

yum添加软件源

```
1 | sudo yum-config-manager \
2 | --add-repo \
3 | https://mirrors.ustc.edu.cn/docker-ce/linux/centos/docker-ce.repo
```

然后刷新缓存

```
1 | sudo yum makecache fast
```

然后安装docker-ce

```
1 | sudo yum install docker-ce
```

OK!

6. 拖动文件到 xshell 进行上传

```
1 ~]# yum -y install lrzsz
```

yum -y install lrzsz

7. pull 不了镜像

```
latest: Pulling from library/busybox
1c5ed1cbdf8: Pulling fs layer
error pulling image configuration: Get https://production.cloudflare.docker.com/registry-v2/docker/registry/v2/
957533667279843acf9a46c973067c8d1dff31ea8b4/data?verify=1596450720-FQXF69V%2BSB0gB6YCDL820Ll%2BpMg%3D: dial t
root@k8s-node02 sysconfig# docker login https://hug.atguigu.com
```

```
root@k8s-node02 ~]# docker pull busybox
Using default tag: latest
error response from daemon: Get https://registry-1.docker.io/v2/: net/http: request canceled while
g for connection (Client.Timeout exceeded while awaiting headers)
root@k8s-node02 ~]# vim /etc/docker/daemon.json
root@k8s-node02 ~]# diff 0114 114 114 114 registry-1.docker.io
```

亲，先看看你的网有没开，ping www.baidu.com

加个镜像

```
"exec-opts": ["native.cgroupdriver=systemd"],
"log-driver": "json-file",
"log-opts": { "max-size": "100m" },
"registry-mirrors": ["https://registry.docker-cn.com", "http://hub-mirror.c.163.com"],
"insecure-registries": ["https://hub.atguigu.com"]
}
~
~
```

"registry-mirrors": ["https://registry.docker-cn.com", "http://hub-mirror.c.163.com"],

8. harbor 登陆不上

```
root@k8s-harbor ~]# docker login https://hub.atguigu.com
authenticating with existing credentials...
login did not succeed, error: Error response from daemon: login attempt to http://hub.atguigu.com/v2/ fa
led with status: 502 Bad Gateway
Username (admin):
Password:
```

docker ps -a | grep harbor

```

root@k8s-harbor ~]# docker ps --help .
root@k8s-harbor ~]# docker ps -a | grep harbor
49f16c936dd        vmware/harbor-jobservice:v1.2.0    "/harbor/harbor_jobs..."   18 hours ago    Up 48 seconds
b3879f09f69        vmware/harbor-ui:v1.2.0            "/harbor/harbor_ui"         18 hours ago    Up 48 seconds
924b50241e6        vmware/harbor-adminserver:v1.2.0   "/harbor/harbor_admini..."   18 hours ago    Exited (2) 9 min
ed6f0bf0dc4        vmware/harbor-db:v1.2.0            "docker-entrypoint.s..."   18 hours ago    Up 8 minutes
3b5ea9a7b26        vmware/harbor-log:v1.2.0           "/bin/sh -c 'cron..."   18 hours ago    Up 2 minutes
root@k8s-harbor ~]# docker restart a924b50241e6

```

docker restart id 绿色那个

```

root@k8s-harbor ~]# docker login https://hub.atguigu.com
Authenticating with existing credentials...
Login did not succeed, error: Error response from daemon: Get http://hub.atguigu.com/v2/: dial tcp 192.168.66.100:443: connect: connection refused
Username (admin):
Password:

```

```

root@k8s-harbor 2020-08-05]# cd /usr/local/harbor
root@k8s-harbor harbor]# ls
common                docker-compose.yml    harbor-offline-installer-v1.2.0
docker-compose        harbor                harbor.v1.2.0.tar.gz
docker-compose.clair.yml  harbor_1_1_0_template  install.sh
docker-compose.notary.yml harbor.cfg             LICENSE
root@k8s-harbor harbor]# docker-compose up -d
harbor-log is up-to-date
Starting harbor-db ...
Starting harbor-db ... done
Starting harbor-adminserver ... done
Starting harbor-ui ... done
Starting harbor-jobservice ... done
Starting nginx ... done
root@k8s-harbor harbor]# docker login https://hub.atguigu.com
Authenticating with existing credentials...
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store
Login Succeeded

```

Cd /usr/local/harbor
docker-compose up -d

9.改命令行

```

[root@centos6 ~]# PS1="\[\e[1;5;41;33m\][\u@\h \W\t]\[\e[0m\]"
[root@centos6 ~12:48:57]#
[root@centos6 ~12:48:58]#
[root@centos6 ~12:48:58]#

```

9. raw.githubusercontent.com 无法连接问题

https://blog.csdn.net/cxxxxxxxxxxxxx/article/details/106152542?utm_medium=distribute.pc_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-1.channel_param&depth_1-utm_source=distribute.pc_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-1.channel_param

1. 在查询网址上查 ip: <https://site.ip138.com/raw.githubusercontent.com/>
2. Vim /etc/hosts

在打开的hosts文件中, 添加下面的内容 (ip地址更换为第1步你旋转的ip)

```
1 | 199.232.68.133 raw.githubusercontent.com
```

- 3.
4. 尝试 ping

10. 下载 docker

简易下载方法见 docker

11. root@youyou-virtual-machine:~# `curl -sSL https://get.daocloud.io/docker | sh`
- 12.
13. Command 'curl' not found, but can be installed with:
- 14.
15. `apt install curl`
- 16.
17. root@youyou-virtual-machine:~# `^C`
18. root@youyou-virtual-machine:~# `dpkg --get-selections | grep docker`
19. root@youyou-virtual-machine:~# `apt-get install curl`
20. Reading package lists... Done
21. Building dependency tree
22. Do you want to continue? [Y/n] Y
23. Get:1 `https://mirrors.tuna.tsinghua.edu.cn/ubuntu focal-updates/main amd64 libcurl4 amd64 7.68.0-1ubuntu2.2 [233 kB]`
24. root@youyou-virtual-machine:~# `curl -sSL https://get.daocloud.io/docker | sh`
25. # `Executing docker install script, commit: 26ff363bcf3b3f5a00498ac43694bflc7d9ce16c`
26. + `sh -c apt-get update -qq >/dev/null`
- 27.
- 28.


```

29. + sh -c DEBIAN_FRONTEND=noninteractive apt-get install -y -qq
apt-transport-https ca-certificates curl >/dev/null
30. + sh -c curl -fsSL "https://download.docker.com/linux/ubuntu/gpg" |
apt-key add -qq - >/dev/null
31. Warning: apt-key output should not be parsed (stdout is not a terminal)
32. + sh -c echo "deb [arch=amd64]
https://download.docker.com/linux/ubuntu focal stable" >
/etc/apt/sources.list.d/docker.list
33. + sh -c apt-get update -qq >/dev/null
34. + [ -n ]
35. + sh -c apt-get install -y -qq --no-install-recommends
docker-ce >/dev/null
36. + sh -c docker version
37. Client: Docker Engine - Community
38. Version: 19.03.13
39. API version: 1.40
40. Go version: go1.13.15
41. Git commit: 4484c46d9d
42. Built: Wed Sep 16 17:02:52 2020
43. OS/Arch: linux/amd64
44. Experimental: false
45.
46. Server: Docker Engine - Community
47. Engine:
48. Version: 19.03.13
49. API version: 1.40 (minimum version 1.12)
50. Go version: go1.13.15
51. Git commit: 4484c46d9d
52. Built: Wed Sep 16 17:01:20 2020
53. OS/Arch: linux/amd64
54. Experimental: false
55. containerd:
56. Version: 1.3.7
57. GitCommit: 8fba4e9a7d01810a393d5d25a3621dc101981175
58. runc:
59. Version: 1.0.0-rc10
60. GitCommit: dc9208a3303feef5b3839f4323d9beb36df0a9dd
61. docker-init:
62. Version: 0.18.0
63. GitCommit: fec3683
64. If you would like to use Docker as a non-root user, you should now
consider
65. adding your user to the "docker" group with something like:
66.

```

67. `sudo usermod -aG docker your-user`
68.
69. Remember that you will have to log out and back in for this to take effect!
70.
71. WARNING: Adding a user to the "docker" group will grant the ability to run
72. containers which can be used to obtain root privileges on
73. the docker host.
74. Refer <https://docs.docker.com/engine/security/security/#docker-daemon-attack-surface> to
75. for more information.
76. root@youyou-virtual-machine:~#
77. root@youyou-virtual-machine:~#
78. root@youyou-virtual-machine:~# **sudo docker run hello-world**
79. Unable to find image 'hello-world:latest' locally
80. latest: Pulling from library/hello-world
81. 0e03bdcc26d7: Pull complete
82. Digest:
sha256:4cf9c47f86df71d48364001ede3a4fcd85ae80ce02ebad74156906caff5378bc
83. Status: Downloaded newer image for hello-world:latest
84.
85. Hello from Docker!
86. This message shows that your installation appears to be working correctly.
87.
88. To generate this message, Docker took the following steps:
89. 1. The Docker client contacted the Docker daemon.
90. 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
91. (amd64)
92. 3. The Docker daemon created a new container from that image which runs the
93. executable that produces the output you are currently reading.
94. 4. The Docker daemon streamed that output to the Docker client, which sent it
95. to your terminal.
96.
97. To try something more ambitious, you can run an Ubuntu container wi