

Linux    debian docker

- [Debian](#)
- [SSH](#)
- [Linux](#)

# Debian

- *eth0* IP

```
cd /etc/sysconfig/  
cat > ifconfig.eth0 << "EOF"  
ONBOOT=yes  
IFACE=eth0  
SERVICE=ipv4-static  
IP=192.168.1.2  
GATEWAY=192.168.1.1  
PREFIX=24  
BROADCAST=192.168.1.255  
EOF
```

## IP

- 

```
ip link
```

- IP

```
nano /etc/network/interfaces
```

```
# The loopback network interface  
auto lo  
iface lo inet loopback
```

^G Help    ^O Write Out    ^W Where Is    ^K Cut    ^T Execute    ^C Location  
^X Exit    ^R Read File    ^\ Replace    ^U Paste    ^J Justify    ^\_ Go To Line

- IP dns-nameservers

```
GNU nano 5.4 /etc/network/interfaces *
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

auto enp0s3
iface enp0s3 inet static
address 192.168.2.2
netmask 255.255.255.0
gateway 192.168.2.2
dns-nameservers 8.8.4.4 8.8.8.8

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

## DNS

- DNS `/etc/resolv.conf`

```
cat > /etc/resolv.conf << "EOF"
# Begin /etc/resolv.conf

domain <Your Domain Name>
nameserver <IP address of your primary nameserver>
nameserver <IP address of your secondary nameserver>

# End /etc/resolv.conf
EOF
```

- `/etc/hostname` `<lfs>` Fully Qualified `/etc/hosts` name FQDN

```
echo "<lfs>" > /etc/hostname
```

## IP

- `/etc/hosts` `<192.168.1.1>` `<FQDN>` `<HOSTNAME>` / IP

```
cat > /etc/hosts << "EOF"
# Begin /etc/hosts

127.0.0.1 localhost
127.0.1.1 <FQDN> <HOSTNAME>
<192.168.1.1> <FQDN> <HOSTNAME> [alias1] [alias2 ...]
```

```
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

# End /etc/hosts
EOF
```

- foo

```
apt-get update #
apt-get -y install foo # "foo"
```

- foo

```
apt-get remove foo # "foo"
apt-get -y autoremove foo # Debian 11 "foo"
```

- foo

```
apt-get -y purge foo # "foo"
apt-get -y autoremove --purge foo # "foo"
```

- 

```
apt-get upgrade #
apt-get dist-upgrade #
```

- 

```
dpkg --get-selections
```

- "foo "

```
apt-cache show foo
```

- 

```
nano /etc/yum.repos.d
```

## ssh

- 22

```
nano /etc/ssh/sshd_config      # sshd_config
#port 22                      # 1
port 22                       # 2  " #"
```

- 1

```
Include /etc/ssh/sshd_config.d/*.conf
#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
```

- 2

```
Include /etc/ssh/sshd_config.d/*.conf
Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
```

- root

```
#PermitRootLogin prohibit-password # 3
PermitRootLogin yes                # 4  " #" "yes" "prohibit-password"

#PasswordAuthentication yes        #
PasswordAuthentication yes         #  " #" 5
```

- 3

```
#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
```

- 4

```
#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
```

- 5

```
PasswordAuthentication yes
```

- nano Ctrl + X "y" 6 " " 7

- 6

```
GNU nano 5.4 /etc/ssh/sshd_config *
# $OpenBSD: sshd_config,v 1.103 2018/04/09 20:41:22 tj Exp $

# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
File Name to Write: /etc/ssh/sshd_config
^G Help      M-D DOS Format  M-A Append     M-B Backup File
^C Cancel    M-M Mac Format  M-P Prepend    ^T Browse
```

○ 7

```
root@H101:~# nano /etc/ssh/sshd_config
root@H101:~#
```

- sshd

```
systemctl restart sshd
```

## Docker

- 

```
curl -fsSL https://get.docker.com | bash -s docker --mirror Aliyun
```

- 

```
apt-get purge docker-ce
```

- 

```
rm -rf /var/lib/docker
```

# SSH

PutTY SSH

Linux

SSH

22

---

SSH

Linux

SSH

- 

```
[root@host ~]$ ssh-keygen <==  
Generating public/private rsa key pair.  
Enter file in which to save the key (/root/.ssh/id_rsa): <== Enter  
Created directory '/root/.ssh'.  
Enter passphrase (empty for no passphrase): <== Enter  
Enter same passphrase again: <==  
Your identification has been saved in /root/.ssh/id_rsa. <==  
Your public key has been saved in /root/.ssh/id_rsa.pub. <==  
The key fingerprint is:  
0f: d3: e7: 1a: 1c: bd: 5c: 03: f1: 19: f1: 22: df: 9b: cc: 08 root@host
```

- root .ssh id\_rsa id\_rsa.pub

- 

```
[root@host ~]$ cd .ssh  
[root@host .ssh]$ cat id_rsa.pub >> authorized_keys
```

- 

```
[root@host .ssh]$ chmod 600 authorized_keys  
[root@host .ssh]$ chmod 700 ~/.ssh
```

## SSH

- /etc/ssh/sshd\_config

```
RSAAuthentication yes  
PubkeyAuthentication yes
```

- root SSH

```
PermitRootLogin yes
```

- 

```
PasswordAuthentication no
```

- SSH

```
[root@host .ssh]$ service sshd restart
```

## PuTTY

WinSCP SFTP

id\_rsa

PuTTYGen

Actions

Load

PuTTYGen

Key comment

Save private key

PuTTY

PuTTY

Connection -> SSH -> Auth

Private key file for authentication:



# Linux

```
arch # (1)
uname -m # (2)
uname -r #
dmidecode -q # - ( SMBIOS / DMI)
hdparm -i /dev/hda #
hdparm -tT /dev/sda #
cat /proc/cpuinfo # CPU info
cat /proc/interrupts #
cat /proc/meminfo #
cat /proc/swaps # swap
cat /proc/version #
cat /proc/net/dev #
cat /proc/mounts #
lspci -tv # PCI
lsusb -tv # USB
date #
cal 2007 # 2007
date 041217002007.00 # - .
clock -w # BIOS
```

( )

```
shutdown -h now # (1)
init 0 # (2)
telinit 0 # (3)
shutdown -h hours:minutes & #
shutdown -c #
shutdown -r now # (1)
reboot # (2)
logout #
```

```

cd /home                #  '/' home'
cd ..                   #
cd ../..               #
cd                      #
cd ~user1              #
cd -                   #
pwd                    #
ls                     #
ls -F                  #
ls -l                  #
ls -a                  #
ls *[0-9]*             #
tree                   #                (1)
lstree                 #                (2)
mkdir dir1             #  'dir1'
mkdir dir1 dir2        #
mkdir -p /tmp/dir1/dir2 #
rm -f file1            #  'file1'
rmdir dir1             #  'dir1'
rm -rf dir1            #  'dir1'
rm -rf dir1 dir2       #
mv dir1 new_dir        #  /
cp file1 file2         #
cp dir/* .             #
cp -a /tmp/dir1 .      #
cp -a dir1 dir2        #
ln -s file1 lnk1       #
ln file1 lnk1          #
touch -t 0712250000 file1 #                - (YYMMDDhhmm)
iconv -l               #

```

```

find / -name file1      #  '/'
find / -user user1      #  'user1'
find /home/user1 -name \*.bin #  '/' home/user1' '.bin'
find /usr/bin -type f -atime +100 #  100
find /usr/bin -type f -mtime -10 #  10
find / -name \*.rpm -exec chmod 755 '{}' \; #  '.rpm'

```

```
find / -xdev -name \*.rpm          #   '.rpm'
locate \*.ps                       #   '.ps'          -   'updatedb'
whereis halt                       #                   man
which halt                         #
```

```
mount /dev/hda2 /mnt/hda2          #   hda2   -   ' / mnt/hda2'
umount /dev/hda2                   #   hda2   -   ' / mnt/hda2'
fuser -km /mnt/hda2                #
umount -n /mnt/hda2                #           /etc/mtab   -
mount /dev/fd0 /mnt/floppy          #
mount /dev/cdrom /mnt/cdrom         #   cdrom dvdrom
mount /dev/hdc /mnt/cdrecorder      #   cdrw dvdrom
mount /dev/hdb /mnt/cdrecorder      #   cdrw dvdrom
mount -o loop file.iso /mnt/cdrom   #   ISO
mount -t vfat /dev/hda5 /mnt/hda5   #   Windows FAT32
mount /dev/sda1 /mnt/usbdisk        #   usb
mount -t smbfs -o username=user,password=pass //WinClient/share /mnt/share #   windows
```

```
df -h                             #
ls -lSr | more                     #
du -sh dir1                        #   'dir1'          '
du -sk * | sort -rn                #
rpm -q -a --qf '%10{SIZE}t%{NAME}n' | sort -k1,1n          #           rpm           ( fedora, red
dpkg-query -W -f='${Installed-Size;10}t${Package}n' | sort -k1,1n          #           deb
debian )
```

```
groupadd group_name                #
groupdel group_name                 #
groupmod -n new_group_name old_group_name          #
useradd -c "Name Surname " -g admin -d /home/user1 -s /bin/bash user1      #           "admin"
useradd user1
userdel -r user1                    #   ( '-r'          )
usermod -c "User FTP" -g system -d /ftp/user1 -s /bin/nologin user1        #
passwd
passwd user1                        #   (   root   )
```

```
chage -E 2005-12-31 user1      #
pwck                          #  '/etc/passwd'
grpck                         #  '/etc/passwd'
newgrp group_name             #
```

- " + " " \_ "

```
ls -lh                        #
ls /tmp | pr -T5 -W$COLUMNS  #      5
chmod ugo+rw directory1      #      (u)  (g)  (o)  r   (w)  (x)
chmod go-rwx directory1      #      (g)  (o)
chown user1 file1            #
chown -R user1 directory1    #
chgrp group1 file1           #
chown user1:group1 file1     #
find / -perm -u+s            #          SUID
chmod u+s /bin/file1          #          SUID  -
chmod u-s /bin/file1          #          SUID
chmod g+s /home/public        #      SGID  -   SUID
chmod g-s /home/public        #      SGID
chmod o+t /home/public        #      STIKY  -
chmod o-t /home/public        #      STIKY
```

- " + " " \_ "

```
chattr +a file1              #
chattr +c file1              #          /
chattr +d file1              #          dump
chattr +i file1              #
chattr +s file1              #
chattr +S file1              #
chattr +u file1              #
lsattr                       #
```



```
cat file1
tac file1
more file1
less file1      'more'
head -2 file1
tail -2 file1
tail -f /var/log/messages
```

```
cat file1 file2 ... | command <> file1_in.txt_or_file1_out.txt general syntax for text
manipulation using PIPE, STDIN and STDOUT
cat file1 | command( sed, grep, awk, grep, etc...) > result.txt
cat file1 | command( sed, grep, awk, grep, etc...) >> result.txt
grep Aug /var/log/messages      '/var/log/messages'      "Aug"
grep ^Aug /var/log/messages      '/var/log/messages'      "Aug"
grep [0-9] /var/log/messages      '/var/log/messages'
grep Aug -R /var/log/*           '/var/log'               "Aug"
sed 's/string1/string2/g' example.txt example.txt        "string1"      "string2"
sed '/^$/d' example.txt example.txt
sed '/ *#/d; /^$/d' example.txt example.txt
echo 'esempio' | tr '[:lower:]' '[:upper:]'
sed -e 'ld' result.txt example.txt
sed -n '/string1/p'              "string1"
sed -e 's/ *$//' example.txt
sed -e 's/string1//g' example.txt      "string1"
sed -n '1,5p;5q' example.txt          5
sed -n '5p;5q' example.txt           5
sed -e 's/00*/0/g' example.txt
cat -n file1
cat example.txt | awk 'NR%2==1' example.txt
echo a b c | awk '{print $1}'
echo a b c | awk '{print $1,$3}'
paste file1 file2
paste -d '+' file1 file2             "+"
sort file1 file2
sort file1 file2 | uniq              (          )
sort file1 file2 | uniq -u
sort file1 file2 | uniq -d           (          )
```

```
comm -1 file1 file2          'file1'
comm -2 file1 file2          'file2'
comm -3 file1 file2
```

```
dos2unix filedos.txt fileunix.txt #      MSDOS  UNIX
unix2dos fileunix.txt filedos.txt #      UNIX   MSDOS
recode ..HTML < page.txt > page.html #      html
recode -l | more                #
```

```
badblocks -v /dev/hda1      hda1
fsck /dev/hda1 / hda1 linux
fsck.ext2 /dev/hda1 / hda1 ext2
e2fsck /dev/hda1 / hda1 ext2
e2fsck -j /dev/hda1 / hda1 ext3
fsck.ext3 /dev/hda1 / hda1 ext3
fsck.vfat /dev/hda1 / hda1 fat
fsck.msdos /dev/hda1 / hda1 dos
dosfsck /dev/hda1 / hda1 dos
```

```
mkfs /dev/hda1 hda1
mke2fs /dev/hda1 hda1 linux ext2
mke2fs -j /dev/hda1 hda1 linux ext3( )
mkfs -t vfat 32 -F /dev/hda1 FAT32
fdformat -n /dev/fd0
mkswap /dev/hda3 swap
```

# SWAP

```
mkswap /dev/hda3 swap
swapon /dev/hda3 swap
swapon /dev/hda2 /dev/hdb3 swap
```

```

dump -0aj -f /tmp/home0.bak /home      ' /home'
dump -1aj -f /tmp/home0.bak /home      ' /home'
restore -if /tmp/home0.bak
rsync -rogpav --delete /home /tmp
rsync -rogpav -e ssh --delete /home ip_address:/tmp  SSH  rsync
rsync -az -e ssh --delete ip_addr:/home/public /home/local  ssh
rsync -az -e ssh --delete /home/local ip_addr:/home/public  ssh
dd bs=1M if=/dev/hda | gzip | ssh user@ip_addr 'dd of=hda.gz'  ssh
dd if=/dev/sda of=/tmp/file1
tar -Puf backup.tar /home/user          ' /home/user'
( cd /tmp/local/ && tar c . ) | ssh -C user@ip_addr 'cd /home/share/ && tar x -p'  ssh
( tar c /home ) | ssh -C user@ip_addr 'cd /home/backup-home && tar x -p'  ssh
tar cf - . | (cd /tmp/backup ; tar xf - )
find /home/user1 -name '*.txt' | xargs cp -av --target-directory=/home/backup/ --parents
'.txt'
find /var/log -name '*.log' | tar cv --files-from=- | bzip2 > log.tar.bz2      '.log'      b
dd if=/dev/hda of=/dev/fd0 bs=512 count=1      MBR (Master Boot Record)
dd if=/dev/fd0 of=/dev/hda bs=512 count=1      MBR

```

```

cdrecord -v gracetime=2 dev=/dev/cdrom -eject blank=fast -force
mkisofs /dev/cdrom > cd.iso              iso
mkisofs /dev/cdrom | gzip > cd_iso.gz      iso
mkisofs -J -allow-leading-dots -R -V "Label CD" -iso-level 4 -o ./cd.iso data_cd      iso
cdrecord -v dev=/dev/cdrom cd.iso        ISO
gzip -dc cd_iso.gz | cdrecord dev=/dev/cdrom -      ISO
mount -o loop cd.iso /mnt/iso            ISO
cd-paranoia -B      CD      wav
cd-paranoia -- "-3"      CD      wav      -3
cdrecord --scanbus      scsi
dd if=/dev/hdc | md5sum      md5sum      CD

```

## - WIFI

```

ifconfig eth0
ifup eth0      'eth0'

```



```
ifdown eth0      'eth0'
ifconfig eth0 192.168.1.1 netmask 255.255.255.0  IP
ifconfig eth0 promisc  'eth0'              (sniffing)
dhclient eth0 dhcp  'eth0'
route -n show routing table
route add -net 0/0 gw IP_Gateway configura default gateway
route add -net 192.168.0.0 netmask 255.255.0.0 gw 192.168.1.1 configure static route to reach
network '192.168.0.0/16'
route del 0/0 gw IP_gateway remove static route
echo "1" > /proc/sys/net/ipv4/ip_forward activate ip routing
hostname show hostname of system
host www.example.com lookup hostname to resolve name to ip address and viceversa(1)
nslookup www.example.com lookup hostname to resolve name to ip address and viceversa(2)
ip link show show link status of all interfaces
mii-tool eth0 show link status of 'eth0'
ethtool eth0 show statistics of network card 'eth0'
netstat -tup show all active network connections and their PID
netstat -tupl show all network services listening on the system and their PID
tcpdump tcp port 80 show all HTTP traffic
iwlist scan show wireless networks
iwconfig eth1 show configuration of a wireless network card
hostname show hostname
host www.example.com lookup hostname to resolve name to ip address and viceversa
nslookup www.example.com lookup hostname to resolve name to ip address and viceversa
whois www.example.com lookup on Whois database
```

# Microsoft Windows networks (SAMBA)

```
nbtscan ip_addr netbios name resolution
nmblookup -A ip_addr netbios name resolution
smbclient -L ip_addr/hostname show remote shares of a windows host
smbget -Rr smb://ip_addr/share like wget can download files from a host windows via smb
mount -t smbfs -o username=user,password=pass //WinClient/share /mnt/share mount a windows
network share
```