

# 望春天

---

None

*aisuhua*

[<a href="https://github.com/aisuhua">](https://github.com/aisuhua) © 2023 *aisuhua*[</a>](https://github.com/aisuhua)

## 首页

---



🕒 2023年10月11日

🕒 2023年5月24日

👤 aisuhua

作者: aisuhua

# 技术

## Docker

### Docker

#### Container

```
1 # 删除所有已停止的容器
2 docker container prune -f
3
4 # 删除所有容器, 除了 php 和 nginx
5 docker rm $(docker ps -a | grep -Ew "php|nginx" | awk 'NR>1 {print $1}')
6
7 # 删除所有容器
8 docker rm -vf $(docker ps -aq)
```

#### Volume

```
1 # 删除所有没在使用的卷
2 docker volume prune
```

#### Image

```
1 # 从小到大排序镜像
2 docker images --format "{{.ID}}\t{{.Size}}\t{{.Repository}}:{{.Tag}}" | sort -k 2 -h
3
4 # 清理 none 镜像
5 docker rmi $(docker images --filter "dangling=true" -q --no-trunc)
6
7 # 删除所有镜像, 除了 php 和 alpine
8 docker rmi $(docker images -a | grep -Ew 'php|alpine' | awk '{print $1}')
9
10 # 清空所有镜像
11 docker rmi -f $(docker images -aq)
12
13 # 导出镜像
14 docker save nginx:latest | gzip > nginx.gz
15
16 # 导入镜像
17 docker load -i nginx.gz
```

#### Build

```
1 # Ubuntu 非交互方式安装软件
2 ARG DEBIAN_FRONTEND=noninteractive
```

#### SSL

```
1 # 单独给仓库配置根证书
2 cp ca.crt /etc/docker/certs.d/yourdomain.com:port/
```

#### Backup

```
1 # 安装
2 sudo curl -sSL https://raw.githubusercontent.com/BretFisher/docker-vackup/main/vackup > /usr/local/bin/vackup
3 sudo chmod +x /usr/local/bin/vackup
4
5 # 备份和恢复 volume
6 vackup export VOLUME_NAME VOLUME_NAME.tar.gz
7 vackup import VOLUME_NAME.tar.gz VOLUME_NAME
```

#### 参考文献

- [Back Up and Share Docker Volumes with This Extension](#)
- <https://github.com/BretFisher/docker-vackup>

🕒 2023年10月12日

🕒 2023年5月26日

👤 aisuhua

作者: aisuhua

## 安装

### 国内源

```
1 cat << EOF | sudo tee /etc/docker/daemon.json
2 {
3     "registry-mirrors":[
4         "https://hub-mirror.c.163.com",
5         "https://mirror.baidubce.com",
6         "https://qby02i3s.mirror.aliyuncs.com",
7         "https://docker.mirrors.ustc.edu.cn"
8     ],
9     "log-driver": "json-file",
10    "log-opts": {
11        "max-size": "10m",
12        "max-file": "3"
13    }
14 }
15 EOF
16
17 sudo systemctl daemon-reload
18 sudo systemctl restart docker
```

### 授权普通用户

```
1 sudo groupadd docker
2 sudo usermod -aG docker $USER
```

### 添加非安全镜像仓库

```
vim /etc/docker/daemon.json
```

```
1 {
2     "insecure-registries" : [
3         "http://registry.example.com:5000"
4     ]
5 }
```

🕒 2023年10月11日

🕒 2023年5月30日

👤 aisuhua

作者: aisuhua

私有仓库

---

登录地址

- <https://cr.console.aliyun.com/cn-shenzhen>

使用方式

```
1 docker login --username=1079087531@qq.com registry.cn-shenzhen.aliyuncs.com
2 docker pull registry.cn-shenzhen.aliyuncs.com/aisuhua/demo:latest
```

 2023年10月11日

 2023年9月9日

 aisuhua

作者: aisuhua

# Podman

---

## Podman

---

### Login

```
1 # 忽略证书验证
2 docker login --tls-verify=false
```

🕒2023年10月11日

🕒2023年6月25日

👤aisuhua

作者: aisuhua

## Git

---

## Git

---

### CheetSheet

```
1 # 设置默认分支
2 git config --global init.defaultBranch master
3
4 # 空目录保持
5 # Ignore everything in this directory
6 *
7 # Except this file
8 !.gitignore
```

### 自动提交

```
1 # 安装
2 sudo wget https://raw.githubusercontent.com/aisuhua/git-auto/master/git-auto -o /usr/local/bin/git-auto
3 sudo chmod a+x /usr/local/bin/git-auto
4
5 # 配置 crontab 自动提交
6 * * * * * /usr/local/bin/git-auto -d /opt/www/aisuhua.github.io -o -p
```

🕒 2023年11月5日

🕒 2023年8月6日

👤 aisuhua

作者: aisuhua



# Ansible

---

## Ansible

---

```
1  # 以 JSON 格式返回结果
2  # 修改配置文件 ansible.cfg
3  [defaults]
4  callback_whitelist=json
5  stdout_callback=json
6
7  # 通过环境变量
8  ANSIBLE_CALLBACK_WHITELIST=json ANSIBLE_STDOUT_CALLBACK=json ansible-playbook ...
9
10 # 修改并发数
11 # ansible.cfg
12 [defaults]
13 forks=50
14
15 # 命令行参数
16 ansible-playbook site.yaml --forks 50
```

### 问题

#### ansible 2.x sudo regression on old systems

```
1  in 1.9x (ansible.cfg)
2  sudo_flags = -H
3
4  in 2.x
5  sudo_flags = -H -S -n
```

### 参考

1. [8 ways to speed up your Ansible playbooks](#)
2. [ansible 2.x sudo regression on old systems](#)

🕒 2024年1月2日

🕒 2023年8月17日

👤 aisuhua

作者: aisuhua

## Book

```
1 # 安装软件
2 ansible all -m dnf -a 'name=php state=present use_backend=dnf4' --become
3 ansible all -m yum -a 'name=haproxy state=present' --become
4
5 # 管理软件
6 ansible all -m service -a "name=haproxy state=started" --become
7
8 # 装软件
9 sudo yum -y install net-tools
10
11 # 查看主机的关系
12 ansible-inventory --graph
13
14 # 查看主机列表
15 ansible -i hosts.yaml icms --list-hosts
16 ansible-playbook -i hosts.yaml playbook.yaml --list-hosts
17
18 # 语法检查
19 ansible-playbook playbook.yaml --syntax-check
20 ansible-playbook playbook.yaml --check
21
22 # 查看 Module 帮助手册
23 ansible-doc service
24
25 # 搜索模块
26 ansible-doc -l | grep yum
27
28 # 检查 playbook 语法是否有误
29 ansible-playbook playbook.yaml --syntax-check
```

## Inventory

```
1 # 最简单的 inventory
2 ansible all -i 'localhost,' -a date
3
4 # 查看 inventory 插件
5 ansible-doc -t inventory -l
6
7 # 简单执行
8 ansible -i 'localhost,' all -a date
9
10 # 查看 facts
11 ansible -i hosts.ini all -m setup
12 ansible -i hosts.ini all -m setup -a 'filter=ansible_all_ipv4_addresses'
13 ansible -i hosts.ini all -m setup -a 'filter=ansible_env'
14
15 # 命令行传递变量
16 ansible-playbook playbook.yaml -e "greeting=hello"
```

## Chapter7

```
1 # 安装依赖
2 sudo apt-get install libjpeg-dev zlib1g-dev
3
4 # 查看 task 清单
5 ansible-playbook --list-tasks playbook.yaml
6
7 # 免费域名解析
8 192.168.1.5.nip.io
9 www.192.168.1.5.nip.io
10
11 # 指定 ansible_python_interpreter 执行
12 ansible-playbook playbook.yaml -e 'ansible_python_interpreter=/usr/bin/python3'
```

## Chapter8

```

1  # 查看详细执行过程
2  ansible -i hosts.yaml -m ping -vvvv
3
4  # 语法检查
5  ansible-playbook --syntax-check playbook.yaml
6
7  # 罗列主机
8  ansible-playbook --list-hosts playbook.yaml
9
10 # 罗列任务清单
11 ansible-playbook --list-tasks playbook.yaml
12
13 # 试运行 dry run
14 ansible-playbook --check playbook.yaml
15
16 # 查看可能更新到的部分
17 ansible-playbook --check --diff playbook.yaml
18
19 # 只运行指定的 tags role
20 ansible-playbook --tags=yuminstall playbook.yaml
21
22 # 只在指定主机上运行
23 ansible-playbook --limit 172.31.96.101 playbook.yaml
24 ansible -i hosts.ini all --limit 172.31.96.102 -m setup

```

## Chapter8

```

1  # 安装 role
2  ansible-galaxy install oefenweb.ntp
3
4  # 查看已安装的 role
5  ansible-galaxy role list
6
7  # 卸载 role
8  ansible-galaxy remove oefenweb.ntp

```

## Chapter10

```

1  # 查看所有 lookup 类型的组件
2  ansible-doc -t lookup --list
3
4  # 操作系统相关 fact 变量
5  "ansible_os_family": "RedHat"
6  "ansible_distribution": "Kylin Linux Advanced Server"
7  "ansible_distribution_major_version": "V10"
8  "ansible_distribution_release": "Tercel"
9  "ansible_distribution_version": "V10"
10
11 # become 无法应用在 include_tasks
12 tasks:
13 - ansible.builtin.include_tasks: redhat.yaml
14   vars:
15     ansible_become: yes
16
17 # become 无法应用在 include_role
18 include_role:
19   name: myrole
20   apply:
21     become: true

```

## Chapter11

```

1  # 分步调试
2  ansible-playbook playbook.yaml --step

```

## Links

- [Using 'creates' arg in Ansible shell module](#)
- [How to install and use Nginx on CentOS 7 / RHEL 7](#)
- [Python 3 Support](#)
- [how can I specify the version of Python to use in an Ansible playbook?](#)

- [How to check if a file exists in Ansible?](#)
- [test-syntax](#)
- [playbooks\\_filters.html](#)
- [ansible\\_error\\_become\\_is\\_not\\_a\\_valid\\_attribute\\_for](#)
- [Use ansible\\_become instead of become in multinode devstack gate](#)
- [How To Use Variables in Ansible Playbooks](#)
- [Ansible - Looking for files and compare their hash](#)
- [Grouping tasks with blocks](#)
- [Controlling playbook execution: strategies and more](#)
- [Send the output from Ansible to a file \[duplicate\]](#)
- [Ansible: how to get output of shell script command on stdout?](#)
- [How to get current role name in an ansible task](#)

🕒 2023年10月9日

🕒 2023年8月24日

👤 aisuhua

作者: aisuhua

# Linux

## Linux

### 初始化

```
1 sudo systemctl disable firewalld
2 sudo systemctl stop firewalld
3 sudo sed -i 's/SELINUX=enforcing/SELINUX=disabled/' /etc/selinux/config
4 sudo setenforce 0
```

### journalctl

```
1 # 查看某个服务的日志
2 sudo journalctl -u sshd --no-pager
```

### User

```
1 # 加入组
2 usermod -a -G sudo suhua
3
4 # sudo 不需要密码, 必须写在 %sudo 组的后面
5 suhua ALL=(ALL:ALL) NOPASSWD:ALL
6
7 # 将用户加入组
8 sudo usermod -aG sudo username
9
10 # 将用户从组中删除
11 sudo gpasswd --delete username sudo
```

### Vim

```
1 # 设置一个 tab 等于 2 个空格
2 cat << EOF | sudo tee -a /etc/vim/vimrc.local
3 " one tab 2 space
4 set tabstop=2
5 set shiftwidth=2
6 set autoindent
7 set expandtab
8 set softtabstop=0
9 EOF
10
11 # 编辑时出现 1H 奇怪字符
12 export LANG=en.UTF-8
```

### grep

```
1 # 包含任意一个关键字
2 grep -E 'google|youtube' file.txt
```

### generate password

```
1
```

🕒 2023年9月5日

🕒 2023年5月28日

👤 aisuhua

作者: aisuhua

## TODO

---

- /var/log/audit/audit.log
- journalctl
- gpart
- grub2
- systemd
- PV 存在 not usable 4.00 MiB [root@localhost ~]# pvdisplay --- Physical volume --- PV Name /dev/sda3 VG Name ROOTVG PV Size 58.63 GiB / not usable 4.00 MiB Allocatable yes PE Size 4.00 MiB Total PE 15009 Free PE 1 Allocated PE 15008 PV UUID HmgWNM-F3Cs-NGAb-Xz1V-DLtN-fN0g-eJJ512

### Links

- [How to Create Your Own Repositories for Packages](#)
- [SUSE Linux Enterprise Server 文档](#)

🕒 2023年11月22日

🕒 2023年9月5日

👤 aisuhua

作者: aisuhua

## 用户

---

```
1 # 添加用户
2 useradd user01 && echo PASSWORD | passwd --stdin user01
3
4 # 分离步骤
5 useradd user01
6 passwd user01
7
8 userdel user01
9
10 # 连同家目录一起删除
11 # userdel -r user01
```

🕒 2023年8月16日

🕒 2023年5月24日

👤 aisuhua

作者: aisuhua



## Shell

### 技巧

```
1 # 在脚本中切换到 root 执行
2 [ `whoami` = root ] || exec su -c $0 root
3 [ `whoami` = root ] || exec sudo su - -c $0 root
```

### tar

```
1 # 备份文件列表里面的所有内容
2 tar -zcvf sos.tar.gz -T mylist.txt
```

### du

```
1 # 不统计某些目录
2 du -sh --exclude='demo01' --exclude='demo02' *
```

### 执行历史记录

```
1 # 将缓冲区内容报错到 ~/.bash_history
2 history -w
3
4 # 清空缓冲区内容
5 history -c
6
7 # 显示时间戳
8 export HISTTIMEFORMAT='%F %T '
9
10 # 持久化配置, 写入 /etc/profile
11 echo "export HISTTIMEFORMAT='%F %T ' >> /etc/profile
12 source /etc/profile
13
14 # 将所有执行过的命名记录到文件
15 # PROMPT_COMMAND 环境变量中设置的内容将在交互式脚本的提示(PS1)出现之前被执行
16 export HISTORY_FILE=/var/log/audit_history.log
17 export PROMPT_COMMAND='{ date "+%y-%m-%d %T #### $(who am i |awk "{print \$1\" \"\$2\" \"\$5}")" #### $(pwd) #### $(history 1 | { read x cmd; echo
18 "$cmd"; }); } >> $HISTORY_FILE'
19
20 # 执行 shell 时将命令输出
21 set -o xtrace
22 # or
23 bash -x myscript.sh
```

### Links

- [谁动了我的 Linux? 原来 history 可以这么强大!](#)
- [Linux 环境变量PROMPT\\_COMMAND](#)
- [How can I print each command before executing?](#)
- [Why doesn't "sudo su" in a shell script run the rest of the script as root?](#)

🕒2023年12月6日

🕒2023年8月17日

👤aisuhua

作者: aisuhua

## SSH

### 使用

```
1 # Disable strict host key checking
2 ssh -o StrictHostKeyChecking=no user@remote-host
3
4 # 查看帮助信息
5 man sshd_config
6
7 # 查看所支持的所有加密算法
8 ssh -Q cipher
9
10 # 测试配置
11 /usr/sbin/sshd -t
12
13 # 打印配置
14 /usr/sbin/sshd -T | grep Ciphers
15
16 # 以指定加密方式连接
17 ssh -c 3des-cbc localhost
18 ssh -c 3des-cbc -o macs=hmac-md5 bakroot02@172.31.96.102
19
20 # 追加加密算法 3des-cbc 和 hmac-md5, 适合 RHEL8/Kylin V10 以下版本服务器
21 $ sudo vi //etc/crypto-policies/back-ends/opensshserver.config
22 /usr/sbin/sshd -D -oCiphers=3des-cbc,aes256-gcm@openssh.com,... -oMACs=hmac-md5,hmac-sha2-256-etm@openssh.com,...
23
24 sudo systemctl daemon-reload
25 sudo systemctl restart sshd
26
27 # 追加加密算法 3des-cbc 和 hmac-md5, 适合 RHEL7 以下版本服务器
28 $ sudo vi /etc/ssh/sshd_config
29 Ciphers +3des-cbc
30 MACs +hmac-md5
31
32 sudo systemctl restart sshd
```

### 问题

```
1 # WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!
2 ssh-keygen -R <host>
```

### Links

- [如何解决 SSH Server 使用了不安全的加密演算法: ARCFOUR、CBC、HMAC-MD5、HMAC-RIPEND160](#)
- [Harden SSH in CentOS 8](#)
- [restart ssh service while users still connected](#)

🕒 2023年12月22日

🕒 2023年6月25日

👤 aisuhua

作者: aisuhua

## 网络

### CheetSheet

#### 配置 DNS

```
1 # 配置 DNS 服务器
2 sudo nmcli con mod ens33 ipv4.dns "223.5.5.5 223.6.6.6"
3
4 # 等同于编辑 /etc/sysconfig/network-scripts/ifcfg-ens33
5 DNS1=223.5.5.5
6 DNS2=223.6.6.6
7
8 # Ignore automatically configured nameservers and search domains
9 nmcli con mod ens33 ipv4.ignore-auto-dns yes
10
11 ## 等同于编辑 /etc/sysconfig/network-scripts/ifcfg-ens33
12 PEERDNS=no
13
14 # 重启 NetworkManager
15 sudo systemctl restart NetworkManager
```

#### 查看默认网关

```
1 $ ip route | grep default
2 default via 192.168.88.1 dev enp60s0 proto static metric 100
```

### Links

- [How to manage DNS in NetworkManager via console \(nmcli\)?](#)
- [Linux: Create virtual network interface](#)
- [How to configure network interface bonding on RHEL 8 / CentOS 8 Linux](#)
- [虚拟机下面实现双网卡绑定](#)

🕒 2023年11月22日

🕒 2023年8月10日

👤 aisuhua

作者: aisuhua

## Chrony

---

```
1 # 手工同步
2 sudo chronyc -a makestep
```

🕒 2023年9月7日

🕒 2023年9月7日

👤 aisuhua

作者: aisuhua

## Ubuntu

---

修改 **HotSpot** 密码

```
1 sudo nm-connection-editor
```

**apt** 使用代理

```
1 sudo apt -o acquire::http::proxy="http://127.0.0.1:1080" install fluent-bit
```

修改网段和**IP**

```
1 sudo vim /etc/NetworkManager/system-connections/Hotspot
2 [ipv4]
3 dns-search=
4 method=shared
5 address1=10.0.0.1/24,10.0.0.1
6
7
8 sudo service network-manager restart
```

## Links

- [How to control IP ranges of Network Manager's hotspots?](#)

🕒 2023年10月26日

🕒 2023年8月18日

👤 aisuhua

作者: aisuhua

## LVM

### 添加磁盘

```
1 # 不重启 VM，将新磁盘加到操作系统
2 echo "- - -" | tee /sys/class/scsi_host/host*/scan
3
4 # 查看块设备
5 lsblk
```

### 划分分区

```
1 # 分区类型选择 Linux LVM 即 8e
2 fdisk /dev/sdb
```

### 创建 PV

```
1 # 将 3 个分区都创建为 PV
2 pvcreate /dev/sdb{1,2,3}
3
4 # 或者单独逐个创建
5 pvcreate /dev/sdb1
6 pvcreate /dev/sdb2
7 pvcreate /dev/sdb3
8
9 # 查看创建出来的 PV
10 pvscan
11 pvdisplay
```

### 创建 VG

```
1 # 创建一个包含 3 个 PV 的 VG
2 vgcreate DATAVG /dev/sdb{1,2,3}
3
4 # 或者先创建 VG 再追加 PV
5 vgcreate DATAVG /dev/sdb1
6 vgextend DATAVG /dev/sdb2
7 vgextend DATAVG /dev/sdb2
8
9 # 查看 VG
10 vgscan
11 vgdisplay
```

### 划分 LV

```
1 # 根据 PE 数量创建，先 vgdisplay 查看剩余可用 PE
2 lvcreate -l 6397 -n lv_data DATAVG
3
4 # 或者根据大小来创建
5 lvcreate -L +25G -n lv_data DATAVG
6 # 或者使用所有剩余容量
7 lvcreate -l +100%free -n lv_data DATAVG
8
9 选项与参数：
10 -L : 后面接容量，容量的单位可以是 M,G,T 等，要注意的是，最小单位为 PE，
11     因此这个数量必须要是 PE 的倍数，若不相符，系统会自行计算最相近的容量。
12 -l : 后面可以接 PE 的『个数』，而不是数量。若要这么做，得要自行计算 PE 数。
13 -n : 后面接的就是 LV 的名称啦！
14
15 # 查看 LV
16 lvscan
17 lvdisplay
```

创建文件系统

```
1 # 格式化文件系统
2 mkfs.xfs /dev/DATAVG/lv_data
3
4 # 挂载使用
5 mkdir /data
6 mount /dev/DATAVG/lv_data /data
7
8 # 配置成开机自动挂载
9 echo '/dev/mapper/DATAVG-lv_data /data xfs defaults 0 0' >> /etc/fstab
```

放大 LV 容量

```
1 fdisk /dev/sdc
2 pvcreate /dev/sdc1
3 vgextend DATAVG /dev/sdc1
4 lvresize -l +100%free /dev/DATAVG/lv_data
5 # lvresize -l +1279 /dev/DATAVG/lv_data
6 # lvresize -L +10G /dev/DATAVG/lv_data
7 xfs_growfs /dev/DATAVG/lv_data
8 # ext4 使用下面的命令
9 # resize2fs /dev/DATAVG/lv_data
```

Links

- 1. [逻辑卷轴管理员 \(Logical Volume Manager\)](#)
- 2. [14.2: 邏輯捲軸管理員 \(Logical Volume Manager\)](#)
- 3. [Rescan the SCSI Bus to Add a SCSI Device Without rebooting the VM](#)
- 4. [Extending a logical volume in a virtual machine running Red Hat or Cent OS](#)

2023年8月11日

2023年8月10日

aisuhua

作者: aisuhua

## iptables

### 安装

```
1  systemctl stop firewalld
2  systemctl disable firewalld
3
4  yum -y install iptables-services
5  systemctl enable iptables
6  systemctl start iptables
7
8  vim /etc/sysconfig/iptables
9  sudo systemctl restart iptables
10
11 $ sudo service iptables save
12 iptables: Saving firewall rules to /etc/sysconfig/iptables: [ OK ]
13
14 # RHEL7、8、v7update6 需单独安装, RHEL5、6、kylin v10 默认自带
15 sudo yum install iptables-services
16
17 # 保存配置, 配置保存位置 /etc/sysconfig/iptables
18 # RHEL 5/6/
19 /etc/init.d/iptables save
20
21 # RHEL7、8/kylin v10/v7update6/
22 service iptables save
23
24 service iptables restart
```

### 使用

```
1  # 查看规则, 相当于 iptables -t filter -L
2  iptables -L
3
4  # 只看 INPUT 链的 filter 表
5  iptables -nvL INPUT --line-numbers
6
7  # 在表头插入规则 -I insert
8  iptables -t filter -I INPUT -s 172.16.230.101 -j ACCEPT
9
10 # 或者忽略 -t filter 默认就是编辑 filter 表
11 iptables -I INPUT -s 172.16.230.101 -j ACCEPT
12
13 # 在表尾追加规则 -A append
14 iptables -A INPUT -s 172.16.230.101 -j ACCEPT
15
16 # 将规则插入特定位置
17 iptables -I INPUT 2 -s 172.16.230.101 -j ACCEPT
18
19 # 清空所有链表的规则
20 iptables -F
21
22 # 仅仅清空 filter 表的规则
23 iptables -t filter -F
24
25 # 仅仅清空 INPUT 链 filter 表的规则
26 iptables -t filter -F INPUT
```

### 注意事项

```
1  # Yes, once you have added a rule to iptables it becomes active immediately - this is why you should be careful with your rules as it is possible to
2  lock yourself out.
3
4  # The iptables-save command writes a series of iptables commands to a file such that the current state of the firewall can be saved. If you want to
5  restore that state you can use the iptables-restore command.
```

### Links

- [iptables 系列文章](#)
- [Is it enough adding iptables rules without restart?](#)
- [Is "iptables-save" will reload iptable configuration?](#)
- [3 ways to make iptables persistent](#)



🕒2023年12月8日

🕒2023年8月11日

👤aisuhua

作者: aisuhua

## VIM

---

```
1  # 设置一个tab等于4个空格
2  # vim /etc/vimrc
3  set tabstop=4
4  set shiftwidth=4
5  set autoindent
6  set expandtab
7  set softtabstop=0
8
9  # 编辑时出现1H奇怪字符
10 export LANG=en.UTF-8
11
12 # 添加多行注释
13 Ctrl + v
14 I
15 #
16 ESC
17
18 # 取消注释
19 Ctrl + v
20 Delete
```

🕒 2023年8月18日

🕒 2023年8月18日

👤 aisuhua

作者: aisuhua

# Security

---

Links

- [RH442之/etc/security/access.conf与tcp\\_wrappers](#)
- [Linux](#) 如何设置密码复杂度?

2023年8月17日

2023年8月17日

 aisuhua

作者: aisuhua

grub

```
1 grub2-setpassword
```

🕒2023年8月17日

🕒2023年8月17日

👤aisuhua

作者: aisuhua

## journalctl

---

```
1 # displays the messages from the current boot
2 journalctl -k
3
4 # view previous boots with the -b option
5 journalctl -b
```

🕒 2023年9月20日

🕒 2023年9月20日

👤 aisuhua

作者: aisuhua

## Kylin

### 查看版本

```
1 nkvers
2 cat /etc/.productinfo
```

### 配置网络

```
1 # 配置 IP
2 [root@test ~]# cat /etc/sysconfig/network-scripts/ifcfg-ens33
3 TYPE=Ethernet
4 PROXY_METHOD=none
5 BROWSER_ONLY=no
6 #BOOTPROTO=dhcp
7 DEFROUTE=yes
8 IPV4_FAILURE_FATAL=no
9 IPV6INIT=yes
10 IPV6_AUTOCONF=yes
11 IPV6_DEFROUTE=yes
12 IPV6_FAILURE_FATAL=no
13 IPV6_ADDR_GEN_MODE=stable-privacy
14 NAME=ens33
15 UUID=c67c9f61-64d2-44ad-999a-80b9f260a775
16 DEVICE=ens33
17 #ONBOOT=no
18
19 ONBOOT=yes
20 BOOTPROTO=static
21 IPADDR=172.16.230.199
22 NETMASK=255.255.255.0
23 GATEWAY=172.16.230.2
24 DNS1=223.5.5.5
25 DNS2=114.114.114.114
26
27 # 重启网卡
28 ifdown ens33
29 ifup ens33
30
31 # nmcli 重新加载网卡配置文件
32 nmcli connection load /etc/sysconfig/network-scripts/ifcfg-ens33
33 nmcli connection up ens33
34
35 # 或者重新加载所有网卡配置文件
36 nmcli connection reload
37 nmcli connection up ens33
```

### 修改字符集

```
1 # 查看当前字符集
2 locale
3
4 # 使用 localectl 命令修改
5 sudo localectl set-locales LANG=en_US.UTF-8
6
7 # 或者直接修改配置文件
8 [root@test ~]# vim /etc/locale.conf
9 LANG=en_US.UTF-8
```

### 修改时区

```
1 # 查看所有可用时区
2 sudo timedatectl list-timezones
3
4 # 设置为东八区
5 sudo timedatectl set-timezone Asia/Shanghai
```

## 时间服务

```

1  # 配置时间服务
2  $ sudo vim /etc/chrony.conf
3  server ntp.aliyun.com iburst
4
5  # 重启服务并设置为自启动
6  sudo systemctl restart chronyd
7  sudo systemctl enable chronyd
8
9  # 查看时间是否同步成功, IP 前面带 * 号表示同步成功
10 [root@test ~]# chronyc sources -v
11 210 Number of sources = 1
12
13  .-- Source mode '^' = server, '=' = peer, '#' = local clock.
14  /  .-- Source state '*' = current synced, '+' = combined , '-' = not combined,
15  | /  '?' = unreachable, 'x' = time may be in error, '~' = time too variable.
16  ||
17  ||      Reachability register (octal) -.      | xxxx [ yyyy ] +/- zzzz
18  ||      Log2(Polling interval) --.      |      | xxxx = adjusted offset,
19  ||      |                          |      | yyyy = measured offset,
20  ||      |                          |      | zzzz = estimated error.
21  ||
22  MS Name/IP address         Stratum Poll Reach LastRx Last sample
23  =====
24  ^* 203.107.6.88             2    6   177    30  -2223us[-5612us] +/-   34ms
25
26  # 手工同步
27  chronyc makestep

```

## 单用户模式

开机后, Grub 界面选择对应 Kernel, 按 **e** 进入编辑模式

光标向下移动至 `linux` 开头的行, 修改 `ro` 为 `rw`, 末尾添加 `rd.break console=tty0`, 按 **Ctrl X** 或 **F10** 保存继续

```

insmod xfs
set root='hd0,msdos1'
if [ x$feature_platform_search_hint = xy ]; then
    search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hin\
t-efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1 --hint='hd0,msdos1' a609d9b6-3\
814-4ab8-a05a-6cfd15d18353
else
    search --no-floppy --fs-uuid --set=root a609d9b6-3814-4ab8-a05a-6cfd\
15d18353
fi
linux /vmlinuz-4.19.90-23.8.v2101.ky10.x86_64 root=/dev/mapper/\
ROOTVG-lv_root rw crashkernel=1024M,high resume=/dev/mapper/ROOTVG-lv_swap rd.\
lvm.lv=ROOTVG/lv_root rd.lvm.lv=ROOTVG/lv_swap video=efifb:on rhgb quiet quiet\
rd.break console=tty0
initrd /initramfs-4.19.90-23.8.v2101.ky10.x86_64.img

```

Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to  
discard edits and return to the menu. Pressing Tab lists  
possible completions.

输入 `chroot /sysroot` 切换至主机系统

```
Generating "/run/initramfs/rdsosreport.txt"

Entering emergency mode. Exit the shell to continue.
Type "journalctl" to view system logs.
You might want to save "/run/initramfs/rdsosreport.txt" to a USB stick or /boot
after mounting them and attach it to a bug report.

switch_root: /# chroot /sysroot
sh-5.0# _
```

## Links

- [KylinV10使用笔记](#)
- [银河麒麟高级服务器操作系统V10\(sp1\)安装手册](#)

🕒 2023年10月26日

🕒 2023年8月10日

👤 aisuhua

作者: aisuhua



# Alpine

其他

```
1 # 运行 Golang 二进制程序
2 apk add gcompat
```

2023年6月2日

2023年5月28日

aisuhua

作者: aisuhua

## FTP

### LFTP

#### 使用

```
1 $ lftp 21.123.39.160 -u fhftp
2 Password:
3 lftp fhftp@21.123.39.160:~> put c.txt
4 6 bytes transferred
5 lftp fhftp@21.123.39.160:~> ls
6 total 1
7 -rw-r----- 1 fhftp fprs 6 Dec 7 12:41 c.txt
```

#### 主动模式

```
1 set ftp:passive-mode off
```

#### 常用命令

```
1 # ls 本地目录内容
2 !dir
3 !dir /opt/www
4
5 # 切换本地目录
6 lcd /tmp
7
8 # 查看现在所在本地位置
9 lpwd
```

🕒 2023年8月6日

🕒 2023年6月12日

👤 aisuhua

作者: aisuhua

## 刻录光盘

### dd

```
1 dd if=rhcos-4.8.14-x86_64-live.x86_64.iso of=<dev/sdx> status=progress
```

### Ventoy

初始化 U 盘

```
1 wget https://github.com/ventoy/Ventoy/releases/download/v1.0.91/ventoy-1.0.91-linux.tar.gz
2 tar -zxvf ventoy-1.0.91-linux.tar.gz
3 cd ventoy-1.0.91
4 sudo sh Ventoy2Disk.sh -i /dev/sdX
```

将 iso 文件拷贝到 U 盘

```
1 cp Win10_22H2_Chinese_Simplified_x64v1.iso /path/to/Ventory
```

插入 U 盘后, 进入 BIOS, 关闭 Secure Boot 并设置成从 U 盘启动。

#### Note

必须关闭 Secure Boot 功能, 否则会出现 Verification failed: (0x1A) Security Violation 报错。

### 参考

- [https://www.ventoy.net/cn/doc\\_start.html#doc\\_linux\\_cli](https://www.ventoy.net/cn/doc_start.html#doc_linux_cli)
- <https://askubuntu.com/a/1456906>
- <https://github.com/pbatard/rufus>

🕒 2023年10月25日

🕒 2023年5月29日

👤 aisuhua

作者: aisuhua

## PDF

### ghostscript

#### 安装

```
1 sudo apt install ghostscript
```

#### 压缩

```
1 # 按电子书比例压缩
2 gs -sDEVICE=pdfwrite -dCompatibilityLevel=1.0 -dPDFSETTINGS=/ebook -dNOPAUSE -dQUIET -dBATCH -sOutputFile=output.pdf input.pdf
3
4 # 自定义分辨率为 120 dpi
5 gs -sDEVICE=pdfwrite -dCompatibilityLevel=1.4 -dDownsampleColorImages=true -dColorImageResolution=120 \
6     -dNOPAUSE -dBATCH -sOutputFile=output.pdf input.pdf
```

#### 参考

- [Reduce PDF File Size in Linux](#)
- [How can I reduce the file size of a scanned PDF file?](#)

🕒 2023年6月3日

🕒 2023年6月2日

👤 aisuhua

作者: aisuhua

概念理解

---

对一些难懂或者容易混淆的概念进行说明。

**chroot**

chroot, 即 **change root directory** (更改 **root** 目录)。在 **Linux** 系统中, 系统默认的目录结构都是以 **/**, 即以根 (**root**) 开始的。而在使用 **chroot** 之后, 系统的目录结构将以指定的位置作为 **/** 位置。

- [Linux chroot 命令详解](#)

 2023年8月5日

 2023年8月5日

 aisuhua

作者: aisuhua

# Kubernetes

---

## minikube

---

安装

```
1 minikube start --driver=docker --container-runtime=containerd --image-mirror-country=cn
```

🕒 2023年10月11日

🕒 2023年6月13日

👤 aisuhua

作者: aisuhua

# CoreDNS

---

## CoreOS

---

同类的 DNS 还有 dnsmasq、bind 等。之所以选择 CoreDNS 是因为它非常轻便和易用，同时也是为了熟悉 Kubernetes 的运用。

2023年5月24日

2023年5月24日

aisuhua

作者: aisuhua

## 安装

可以使用 [Docker](#) 或者二进制方式安装，个人偏向于使用 **Docker**，并在官方镜像基础上做了封装，详见 [aisuhua/docker-coredns](#)。

## 安装

```
1 git clone git@github.com:aisuhua/docker-coredns.git
2 cd docker-coredns
3 docker-compose up
```

## 验证

```
1 dig @127.0.0.1 master1.ocp1.example.com
2 dig @127.0.0.1 -x 192.168.1.20
```

🕒 2023年5月24日

🕒 2023年5月24日

👤 aisuhua

作者: aisuhua



客户端配置

Kylin V10

修改 DNS 配置

```
1 # /etc/systemd/resolved.conf
2 [Resolve]
3 DNS=192.168.1.100
```

重启服务生效

```
1 sudo systemctl daemon-reload
2 sudo systemctl restart systemd-resolved
```

2023年5月25日

2023年5月24日

aisuhua

作者: aisuhua

# OpenShift

---

## OpenShift

---

### Event

```
1 # 按时间顺序查看事件
2 kubectl get events --sort-by='.lastTimestamp'
```

### Namespace

```
1 # 查看当前在哪个命名空间
2 oc config view --minify -o 'jsonpath={..namespace}'
3
4 # 或者
5 kubectl config view --minify -o 'jsonpath={..namespace}'
```

### Pod

```
1 # 强制删除 Pod
2 kubectl delete pod --grace-period=0 --force --namespace <NAMESPACE> <PODNAME>
3
4 # 删除当前命名空间下所有 Terminating 的 Pod
5 for p in $(kubectl get pods | grep Terminating | awk '{print $1}'); do kubectl delete pod $p --grace-period=0 --force;done
6
7 # 删除集群里所有 Terminating 的 Pod
8 kubectl get pods --all-namespaces | awk '{if ($4=="Terminating") print "oc delete pod " $2 " -n " $1 " --force --grace-period=0";}' | sh
```

🕒 2023年10月11日

🕒 2023年5月31日

👤 aisuhua

作者: aisuhua

## 初始化

---

### 新用户不能创建项目

```
1 oc patch clusterrolebinding.rbac self-provisioners -p '{"subjects": null}'
2 oc patch clusterrolebinding.rbac self-provisioners -p '{"metadata": {"annotations": {"rbac.authorization.kubernetes.io/autoupdate": "false" }}}'
```

- [Disabling project self-provisioning](#)

🕒 2023年10月11日

🕒 2023年6月25日

👤 aisuhua

作者: aisuhua

审计日志

---

参考

- [Viewing audit logs](#)

 2023年10月11日

 2023年6月28日

 aisuhua

作者: aisuhua

## 常见问题

### MCP 自动更新并触发节点重启

官方建议 MCP 应该保持开启状态。因为从 4.7.4 版本开始，api-server 证书自动轮转不会再触发节点重启，而关闭自动更新则会导致证书无法正常更新。

执行以下命令，开启 MCP 自动更新

```
1 oc patch --type merge machineconfigpool/master -p '{"spec":{"paused":false}}'
2 oc patch --type merge machineconfigpool/worker -p '{"spec":{"paused":false}}'
```

- [Understand CA cert auto renewal in Red Hat OpenShift 4](#)
- [Disabling the Machine Config Operator from automatically rebooting by using the console](#)

### CoreOS 重启后 hostname 不对

在添加 worker 节点时，虽然使用 `hostnamectl set-hostname` 设置了 hostname，但重启后主机名变成了 localhost。

可以在 ignition 文件里添加主机名解决该问题

```
1 $ cat worker.ign | jq . > worker1.ocpl.example.com.ign
2 $ cat worker1.ocpl.example.com.ign
3 {
4   "ignition": {
5     "config": {
6       "merge": [
7         {
8           "source": "https://api-int.test.example.com:22623/config/worker"
9         }
10      ],
11    },
12    "security": {
13      "tls": {
14        "certificateAuthorities": [
15          {
16            "source": "data:text/plain;charset=utf-8;base64,LS0tLS1CR... "
17          }
18        ]
19      }
20    },
21    "version": "3.1.0"
22  },
23  "storage": {
24    "files": [
25      {
26        "path": "/etc/hostname",
27        "contents": {
28          "source": "data:,worker1.ocpl.example.com"
29        },
30        "mode": 420
31      }
32    ]
33  }
34 }
```

- [Hostname can not persist when installing RHCOS](#)
- [How to configure persistent hostname when creating RHCOS in OpenShift 4.6 or later?](#)

### the cluster operator image-registry is degraded

升级无法正常进行，并长时间停留在以下状态

```
1 $ oc get clusterversion -o wide
2 NAME      VERSION
3 version 4.5.0-0.nightly-s390x-2020-06-29-163732
4 the cluster operator image-registry is degraded
```

AVAILABLE	PROGRESSING	SINCE	STATUS
True	False	15h	Error while reconciling 4.5.0-0.nightly-s390x-2020-06-29-163732:

查看 `co` 状态

```
1 $ oc describe co image-registry
2 ImagePrunerDegraded: Job has reached the specified backoff limit
```

解决方法

```
1 oc patch imagepruner.imageregistry/cluster --patch '{"spec":{"suspend":true}}' --type=merge
2 oc -n openshift-image-registry delete jobs --all
```

- [ImagePrunerDegraded error stalling upgrade](#)
- [OpenShift v4.x - ImagePrunerDegraded: Job has reached the specified backoff limit](#)

## no space left on device

Pod 无法正常启动，一直处于 `ContainerCreating` 状态，查看事件发现报 `no space left on device` 错误

```
1 $ oc describe pods mysql8.0.28-nacos3-67965c8f4-6r4st
2 Events:
3   Type      Reason      Age   From      Message
4   ----      -
5   Normal    Scheduled   79s   default-scheduler   Successfully assigned zfqz/mysql8.0.28-nacos3-67965c8f4-6r4st to worker8.dev3.example.com
6   Warning   FailedMount 79s    kubelet      MountVolume.SetUp failed for volume "kube-api-access-dp55p" : write /var/lib/kubelet/pods/
7   eda464b0-1376-4edb-b0ea-c495f18bbb9e/volumes/kubernetes.io~projected/kube-api-access-dp55p/.2023_05_29_08_16_36.1470493620/ca.crt: no space left on
   device
8   Warning   FailedMount 78s    kubelet      MountVolume.SetUp failed for volume "kube-api-access-dp55p" : write /var/lib/kubelet/pods/
9   eda464b0-1376-4edb-b0ea-c495f18bbb9e/volumes/kubernetes.io~projected/kube-api-access-dp55p/.2023_05_29_08_16_37.1796074709/service-ca.crt: no space
10  left on device
```

发现是 `limit.memory` 和 `request.memory` 的单位写错了

```
1 resources:
2   requests:
3     memory: 256m
4     cpu: 100m
5   limits:
6     memory: 512m
7     cpu: 100m
```

将 `memory` 的单位改成 `Mi` 即可

```
```yaml hl_lines="3,6" resources: requests: memory: 256Mi cpu: 100m limits: memory: 512Mi cpu: 100m
```

```
1 - https://github.com/orgs/strimzi/discussions/6399#discussioncomment-2224453
2
3 ## Cluster operator operator-lifecycle-manager-packageserver is stuck in progressing state
4
5 当 lifecycle operator 证书过期后，会出现该报错
6
7 ```sh
8 # 检查证书是否已过期
9 oc get secret packageserver-service-cert -o json -n openshift-operator-lifecycle-manager | jq -r '.data | .["tls.crt"]' | base64 -d | openssl x509 -
10 noout -dates
11
12 # 更新 secret
13 oc delete secret packageserver-service-cert -n openshift-operator-lifecycle-manager
```

- [Cluster operator operator-lifecycle-manager-packageserver is stuck in progressing state](#)

🕒 2023年10月11日

🕒 2023年5月31日

👤 aisuhua

作者: aisuhua

# Elastic

---

## Elastic

---

**Links**

- [How to Update Security Certificates in Elasticsearch](#)
- [Efficiently Deleting Snapshots in Elasticsearch](#)
- [多es 集群数据迁移方案](#)
- [ES异地双活方案](#)

2023年8月14日

2023年8月14日

aisuhua

作者: aisuhua

## Windows

---

### 工具

---

系统激活

- [Microsoft-Activation-Scripts](#)

录屏截图

- [LICEcap](#)

🕒 2023年5月30日

🕒 2023年5月30日

👤 aisuhua

作者: aisuhua



## WSL

### Cheetsheet

```
1 # View the list of distros and their current state:
2 wsl.exe -l -v
3
4 # Shutdown everything
5 wsl.exe --shutdown
6
7 # Terminate a specific distro:
8 wsl.exe -t <DistroName>
9
10 # Boot up a specific distro:
11 wsl.exe -d <DistroName>
```

### 常见问题

#### WSL UBUNTU ERROR 0X80004002

使用 Administrator 打开 PowerShell, 执行以下命令:

```
1 Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Windows-Subsystem-Linux
```

#### PHPSTORM 访问 WSL2 中的文件

```
1 \\wsl$\\Ubuntu\\opt\\www\\invo
```

#### 新标签默认打开 UBUNTU

设置 -> 启动 -> 默认配置文件 -> 选择 Ubuntu -> 保存

#### 从 WSL 访问 HYPER-V 所创建的虚拟机

```
1 Get-NetIPInterface | where {$_.InterfaceAlias -eq 'vEthernet (WSL)' -or $_.InterfaceAlias -eq 'vEthernet (Default Switch)'} | Set-NetIPInterface -
Forwarding Enabled
```

#### 设置 WSL 静态 IP

1. 打开注册表编辑器, 输入 reg
2. 进入目录 计算机\HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Lxss\
3. 修改 WSL 网段信息 NatNetwork
4. 修改 WSL 网关 IP (即主机的 IP 地址) NatGatewayIpAddress
5. 修改 WSL Ubuntu 的 IP NatIpAddress

### 参考

1. [Windows Subsystem for Linux Documentation](#)
2. [WSL Ubuntu error 0x80004002](#)
3. [Open a project stored in the WSL file system](#)
4. [Connect Hyper-V VM from WSL \(Ubuntu\)](#)
5. [Can I set a static ip address for WSL 2?](#)
6. [Configuring WSL2 distro for static IPv4 address](#)
7. [Rebooting Ubuntu on Windows without rebooting Windows?](#)

🕒 2023年11月15日

🕒 2023年5月30日

👤 aisuhua

作者: aisuhua

# MySQL

## 修改字符集为 utf8mb4

修改数据库的字符集为 `utf8mb4`，让它能存储 4 字节的 emoji 表情。

### 修改数据库

```
1 ALTER DATABASE database_name CHARACTER SET = utf8mb4 COLLATE = utf8mb4_unicode_ci;
```

### 修改表

```
1 ALTER TABLE table_name CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
```

### 自动生成修改语句

创建 `gen_queries.sql` 文件，请用实际的数据库名称替换 `YOUR_DATABASE_NAME`

```
1 USE information_schema;
2 SELECT CONCAT("ALTER DATABASE `",table_schema,"` CHARACTER SET = utf8mb4 COLLATE = utf8mb4_unicode_ci;") AS _sql
3 FROM `TABLES` WHERE table_schema LIKE "YOUR_DATABASE_NAME" AND TABLE_TYPE='BASE TABLE' GROUP BY table_schema UNION
4 SELECT CONCAT("ALTER TABLE `",table_schema,"`.`",table_name,"` CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;") AS _sql
5 FROM `TABLES` WHERE table_schema LIKE "YOUR_DATABASE_NAME" AND TABLE_TYPE='BASE TABLE' GROUP BY table_schema, table_name UNION
6 SELECT CONCAT("ALTER TABLE `",`COLUMNS`.table_schema,"`.`",`COLUMNS`.table_name, "` CHANGE `",column_name,"` `",column_name,"`
7 ",data_type,"(",character_maximum_length,") CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci,IF(is_nullable='YES'," NULL"," NOT NULL"),";") AS _sql
8 FROM `COLUMNS` INNER JOIN `TABLES` ON `TABLES`.table_name = `COLUMNS`.table_name WHERE `COLUMNS`.table_schema like "YOUR_DATABASE_NAME" and data_type
9 in ('varchar','char') AND TABLE_TYPE='BASE TABLE' UNION
10 SELECT CONCAT("ALTER TABLE `",`COLUMNS`.table_schema,"`.`",`COLUMNS`.table_name, "` CHANGE `",column_name,"` `",column_name,"` ",data_type," CHARACTER
11 SET utf8mb4 COLLATE utf8mb4_unicode_ci,IF(is_nullable='YES'," NULL"," NOT NULL"),";") AS _sql
12 FROM `COLUMNS` INNER JOIN `TABLES` ON `TABLES`.table_name = `COLUMNS`.table_name WHERE `COLUMNS`.table_schema like "YOUR_DATABASE_NAME" and data_type
13 in ('text','tinytext','mediumtext','longtext') AND TABLE_TYPE='BASE TABLE';
```

### 生成修改语句

```
1 mysql -u root -p -s < gen_queries.sql > queries.sql
```

### 执行修改语句

```
1 mysql -u root -p < queries.sql
```

### 参考

- [How to easily convert utf8 tables to utf8mb4 in MySQL 5.5](#)
- [utf8mb4-convert](#)

🕒 2023年6月8日

🔗 2023年6月8日

👤 aisuhua

作者: aisuhua

## CSS

---

### CSS

---

```
1  /* 代码过长自动换行，避免出现横向滚动条 */
2  pre code {
3      word-wrap: break-word;
4      white-space: pre-wrap;
5  }
6
7  /* 页面灰色效果 */
8  html{
9      filter: grayscale(100%) !important;
10 }
11
12 * {
13     filter: gray !important;
14 }
```

🕒 2023年6月2日

🕒 2023年5月29日

👤 aisuhua

作者: aisuhua

# Vmware

## 常见问题

### CheetSheet

```
1  # 添加磁盘后，不需要重启虚拟机
2  echo "- - -" | tee /sys/class/scsi_host/host*/scan
3
4  # 调整 NAT 网段
5  Select Edit > Virtual Network Editor.
6  Select the host-only or NAT network.
7  To change the subnet IP address, type a new value in the Subnet IP text box.
8
9  # 从网络启动
10  开机，按 F12
```

### Before you can run VMware several modules must be compiled

#### 解决方法


```
1  sudo vmware-modconfig --console --install-all
```

#### 参考

- [Before you can run VMware several modules must be compiled](#)
- [Change the Subnet Settings for a Host-Only or NAT Network on a Windows Host](#)
- VMware的虚拟网络编辑器，在配置的过程中没有桥接模式！
- [Network Boot of Virtual Machines \(1322\)](#)

2023年11月11日

2023年6月12日

aisuhua, root

作者: aisuhua, root

## 安全

---

### 安全

---

- <https://www.cvedetails.com/>

🕒 2023年7月14日

🕒 2023年7月14日

👤 aisuhua

作者: aisuhua

## 电子书

---

- [Library Genesis](#)
- [SaltTiger](#)

🕒 2023年10月11日

🕒 2023年8月10日

👤 aisuhua

作者: aisuhua