

一. 课前导读

教学任务

1. vmware14/15的安装(提前安装好)
2. 虚拟机的安装(提前安装好)
3. linux系统的目录结构介绍
4. linux的基本命令操作
5. Linux环境配置
6. 用户管理命令
7. 进程管理命令
8. 权限管理
9. 软件安装管理
10. 免密登录
11. 定时器

教学目标

1. 能够独立安装虚拟机
2. 了解linux系统及目录结构
3. 熟悉linux的基本命令操作
4. 熟练进行环境配置
5. 了解基本的用户管理,进程管理命令
6. 了解简单的权限设置
7. 熟练通过rpm,yum源进行软件安装
8. 熟练配置免密登录
9. 熟练使用定时器

二. 正课内容

第一章 Linux简介与安装

1.1 Linux的简介

1.1.1 Linux介绍

Linux是一种自由和开放源码的操作系统，存在着许多不同的Linux发行版本，但它们都使用了Linux内核。

Linux内核网站：[www.kernel.org](<http://www.kernel.org>)

1.1.2 Unix与Linux发展史

Unix在1969年，美国贝尔实验室的肯汤普森在DEC PDP-7机器上开发出了UNIX系统。

Linux出现于1991年，是由芬兰赫尔辛基大学学生李纳斯·托瓦兹（Linus Torvalds）和后来加入的众多爱好者共同开发完成。

1.1.3 Linux应用领域

Linux企业服务器

嵌入式系统：Linux可安装在各种计算机硬件设备中，比如手机、平板电脑、路由器、智能家电等嵌入式系统。

电影娱乐行业：电影后期特效处理。

1.1.4 Linux特点

开源，多用户，多任务，丰富的网络功能，可靠的系统安全，良好的可移植性，具有标准兼容性，良好的用户界面，出色的速度性能

1.1.5 服务器不使用图形化界面

原因：

- 1 图形化占用资源
- 2 机房部署方便，无需配置操作界面

1.1.6 常用的Linux版本

1.1.6.1 Linux两大阵营

RedHat系列：RedHat, Suse, CentOS, Fedora等

Red Hat：

RHEL(Redhat Enterprise Linux, 也就是所谓的Redhat Advance Server收费版本. 红帽企业级Linux

Fedora Core:

由原来的Redhat桌面版本发展而来，免费版本

CentOS:

RHEL的社区克隆版本，免费

Fedora:

想尝试最先进的技术，等不及程序的稳定版出来。其实，Fedora就是红帽公司的一个测试平台；产品在成为企业级发行版之前，在该平台上进行开发和测试。

SuSe:

最华丽的Linux发行版，企业服务器端 X windows和程序应用方面做的确实不错。尤其与Microsoft的合作关系，应该是在所有的Linux发行版本中最亲密的。

Debian系列：Debian, Ubuntu等

Debian:

运行起来极其稳定，这使得它非常适合用于服务器

Ubuntu:

是Debian的一款衍生版，也是当今最受欢迎的免费操作系统，最流行的linux桌面系

区别：命令细微的区别；安装方式

1.1.6.2 CentOS社区版介绍

主流：目前的Linux操作系统主要应用于生产环境，主流企业级Linux系统仍旧是RedHat或者CentOS

免费：RedHat 和CentOS差别不大，CentOS是一个基于Red Hat Linux 提供的可自由使用源代码的企业级Linux发行版本

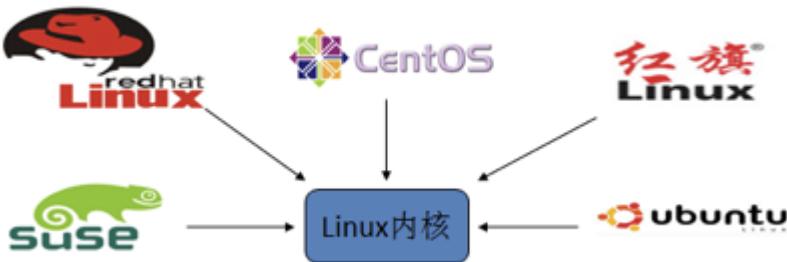
更新方便：CentOS独有的yum命令支持在线升级，可以即时更新系统，不像RedHat 那样需要花钱购买支持服务！

CentOS官网：<http://www.centos.org/>

CentOS搜狐镜像：<http://mirrors.sohu.com/centos/>

CentOS网易镜像：<http://mirrors.163.com/centos/>

基于内核，再增加一些桌面，应用程序，就可以发行，称为开发版。



1.2 VMWare的安装

1.2.1 环境说明

安装环境： Windows7/win10

VMware版本： VMware Workstation14/15或者Parallels Desktop(PD)，

Linux版本： CentOS 7.7

软件在网盘上的链接地址：

链接: <https://pan.baidu.com/s/1TU8VAHyJFfGvNsOBdIp5dw> 密码: uanc

软件介绍如下：

全部文件 > 大数据精品视频录制 > Linux > 资料			已全部加载, 共7个
<input type="checkbox"/> 文件名	大小	修改时间	
<input type="checkbox"/> VMware.Workstation.v14.0.0.Win	虚拟机VM14-主要用在windows系统	2020.08.05 23:11	
<input type="checkbox"/> Notepad++	文本编辑-增强版的记事本,可以进行各种格式文件编写,还可以远程连接服务器	2020.08.05 23:11	
<input type="checkbox"/> VMware-workstation-full-15.5.1-x64.exe	虚拟机VM15-主要用在windows系统 541.16MB	2020.08.05 23:05	
<input type="checkbox"/> Parallels Desktop 13 2 0 43213 TNT已破解直接可用.zip	虚拟机PD-主要用于mac系统 240.93MB	2020.08.05 23:02	
<input type="checkbox"/> finalshell_install.pkg	远程连接工具-用于mac系统 95.24MB	2020.08.05 23:07	
<input type="checkbox"/> finalshell_install.exe	用于windows系统 66.42MB	2020.08.05 23:07	
<input type="checkbox"/> CentOS-7-x86_64-Minimal-1908.iso	CentOS7.7版 942.00MB	2020.08.05 23:05	

1.2.2 VMWare介绍

VMWare是一个虚拟PC的软件，可以在现有的操作系统上虚拟出一个新的硬件环境，相当于模拟出一台新的PC，以此来实现在一台机器上真正同时运行两个独立的操作系统。

VMware主要特点：

- 1 不需要分区或重新开机就能在同一台PC上使用两种以上的操作系统
- 2 本机系统可以与虚拟机系统网络通信
- 3 可以设定并且随时修改虚拟机操作系统的硬件环境

1.2.3 安装注意事项

- 1 安装路径下不能有中文，空格，特殊符号（尽量不要安装在C盘），路径不能太深
- 2 如果想要卸载重新安装，双击其安装文件，选择卸载软件，尽量不要手动去删除其安装目录
- 3 如果使用手动删除VMware的安装目录进行卸载，要自行删除注册表中VMware的相关信息

1.3 Parallels Desktop 安装

1.3.1 软件说明

Parallels Desktop，是Mac上最主流的虚拟机程序。该文档中使用的版本为 13.1.2。

1.3.2 新建虚拟机

选择“安装 Windows或其他操作系统（从DVD或镜像文件）”，然后“继续”。



1.3.3 选择安装镜像



Parallels Desktop会自动的扫描磁盘上的镜像文件，但是如果扫描不到，或者扫描到的不是你想要安装的镜像。可以使用“手动选择”，手动的选择你想要安装的镜像



1.3.4 取消快速安装

只需要取消快速安装集合，其他的都可以不用填写



1.3.5 设置保存位置

设置虚拟机文件（pvm文件）的保存路径，以及勾选安装前设定



1.3.6 设置Linux的硬件



1.3.7 启动即可



1.4 CentOS的安装

1.4.1 准备事项

注意：在bios里面设置虚拟化 将 inter technology vatuializtion设置成 enable

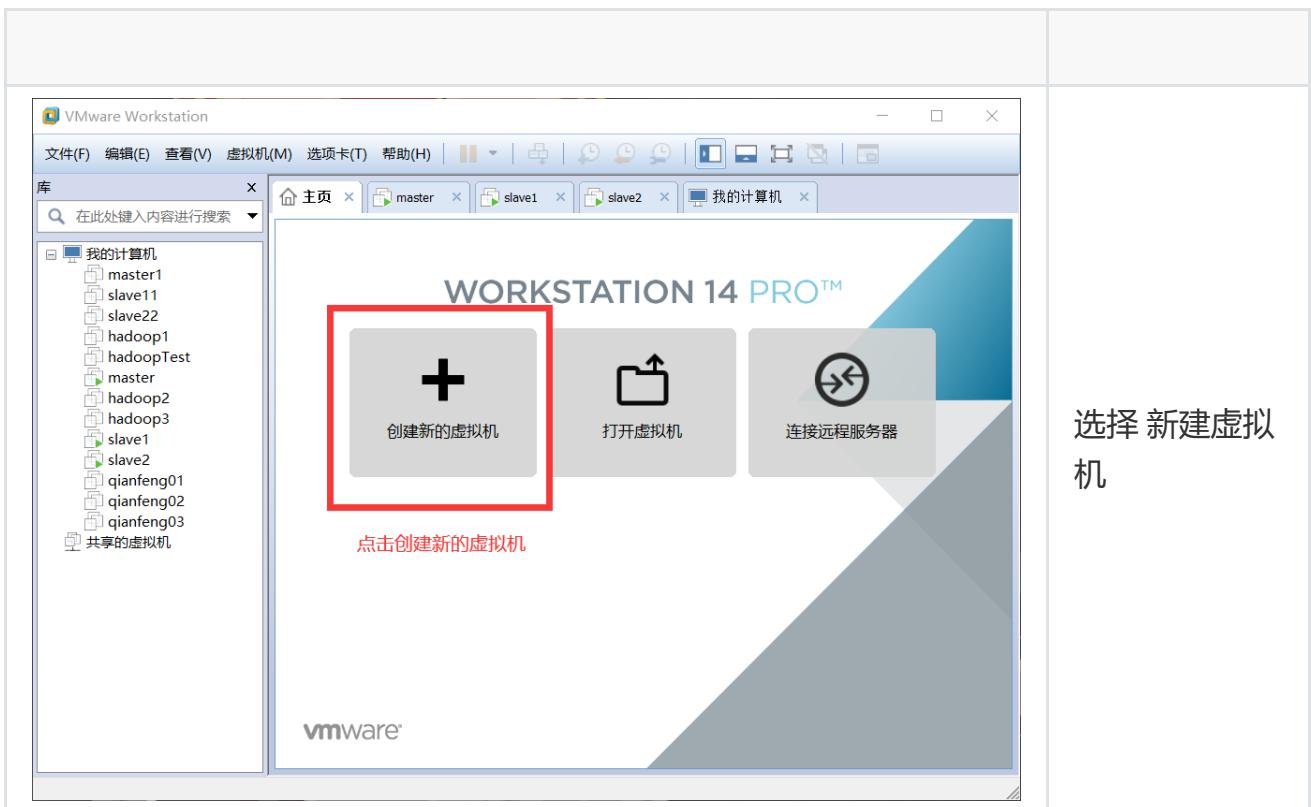
虚拟机：VMWare workstation

虚拟机版本：CentOS-7-x86_64-Minimal-1908.iso

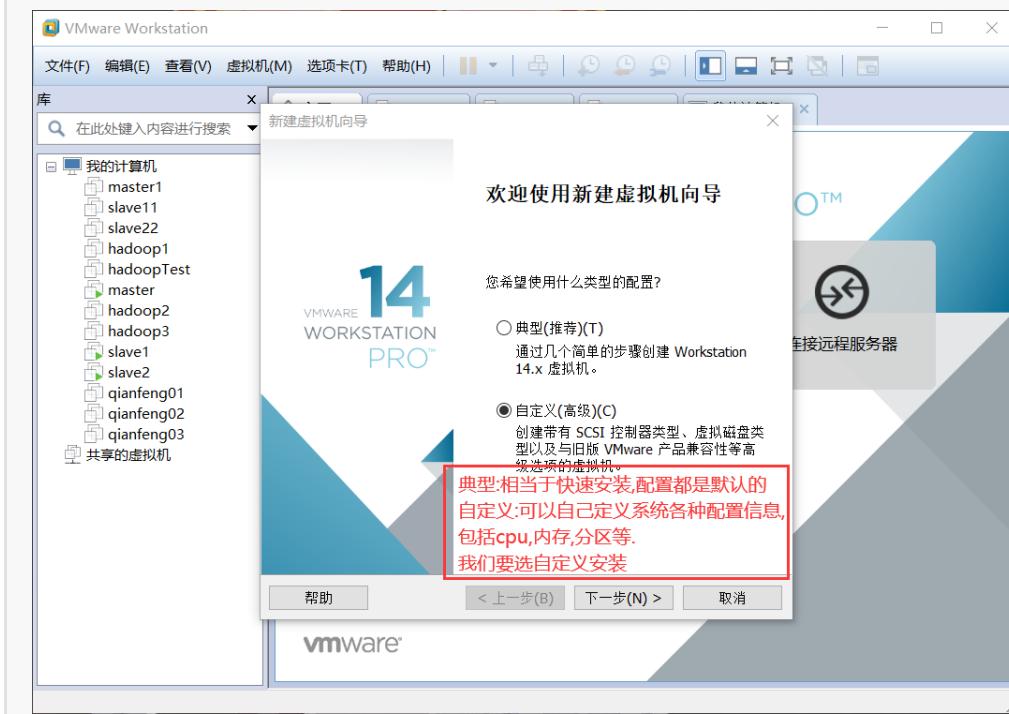
电脑：笔记本或者台式机

系统版本：win7 win8.1 win10 mac 等操作系统

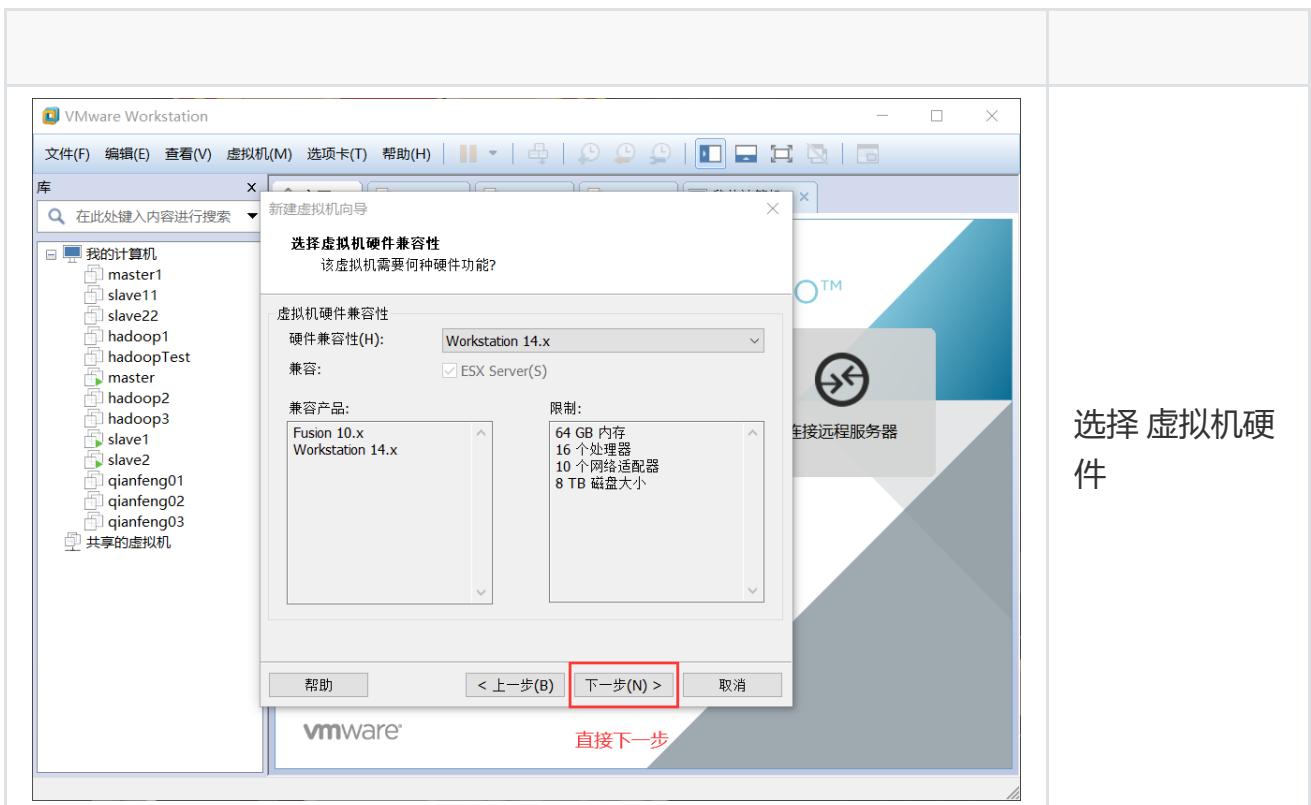
1.4.2 安装



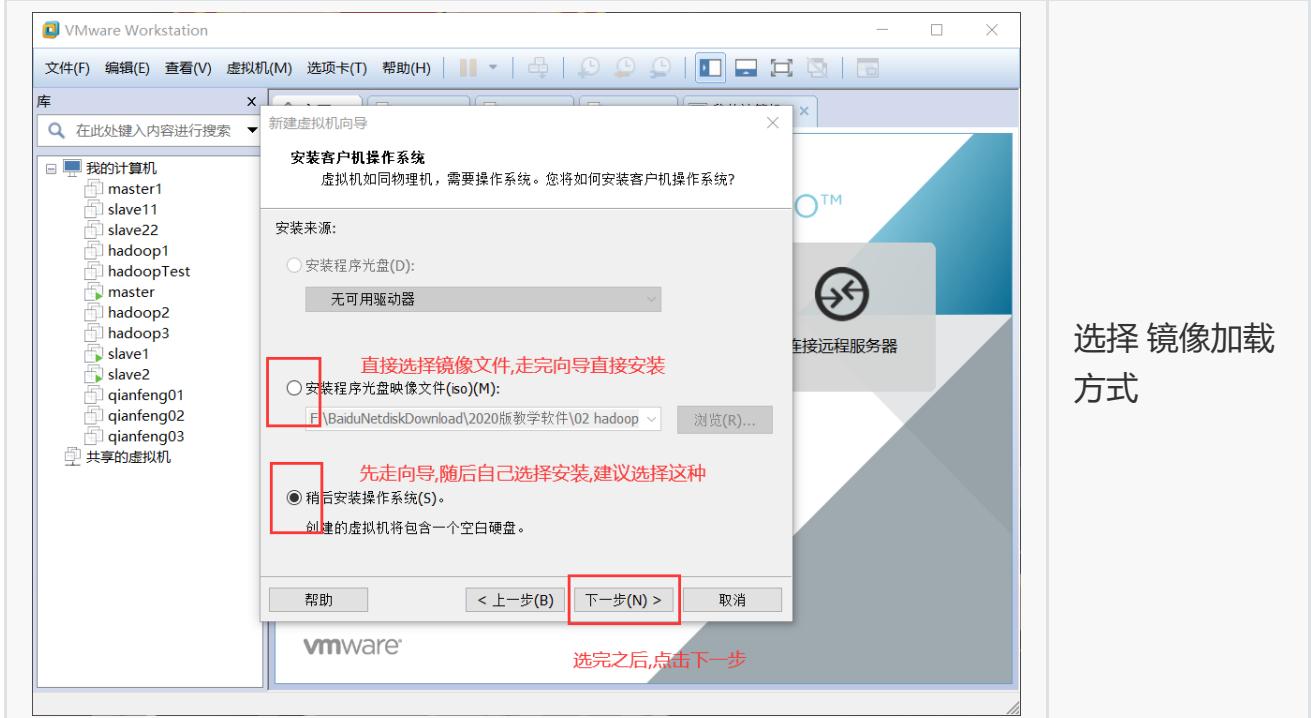
选择 新建虚拟机



选择 自定义

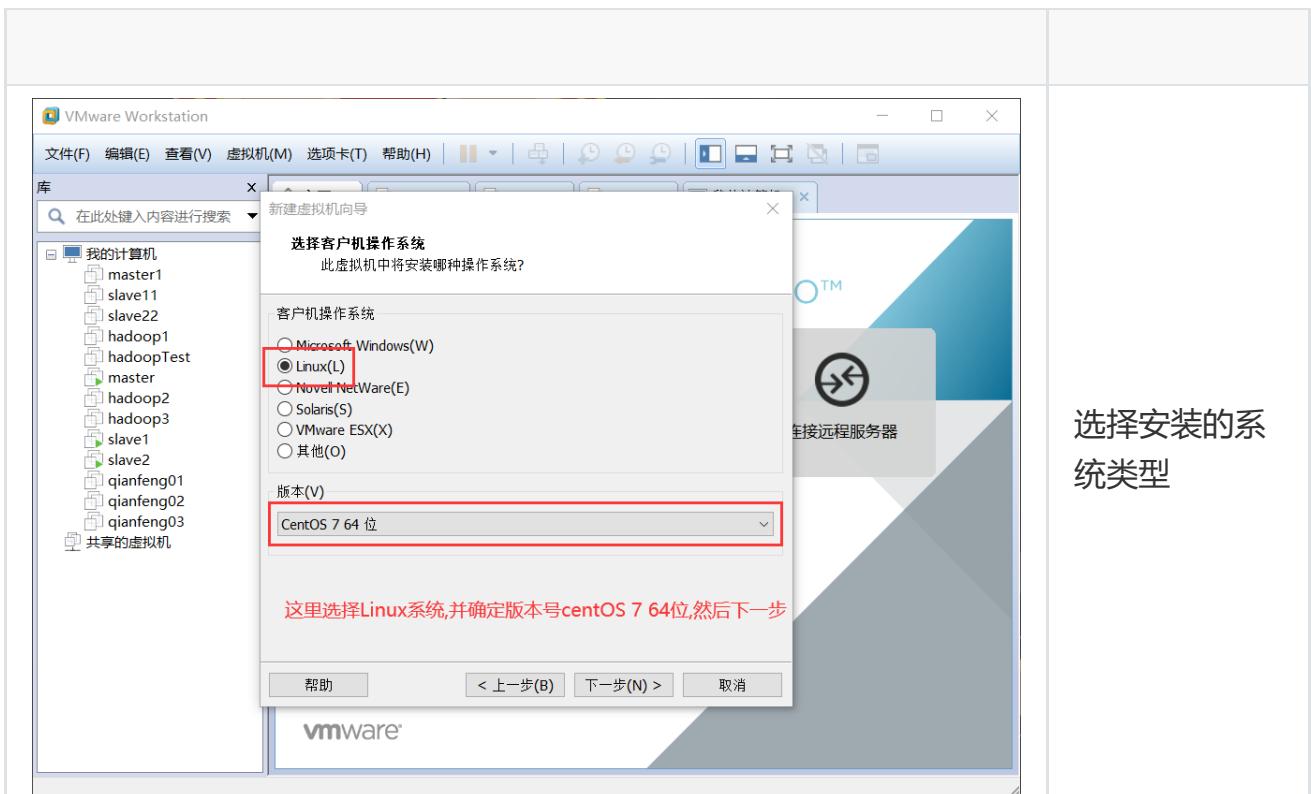


直接下一步

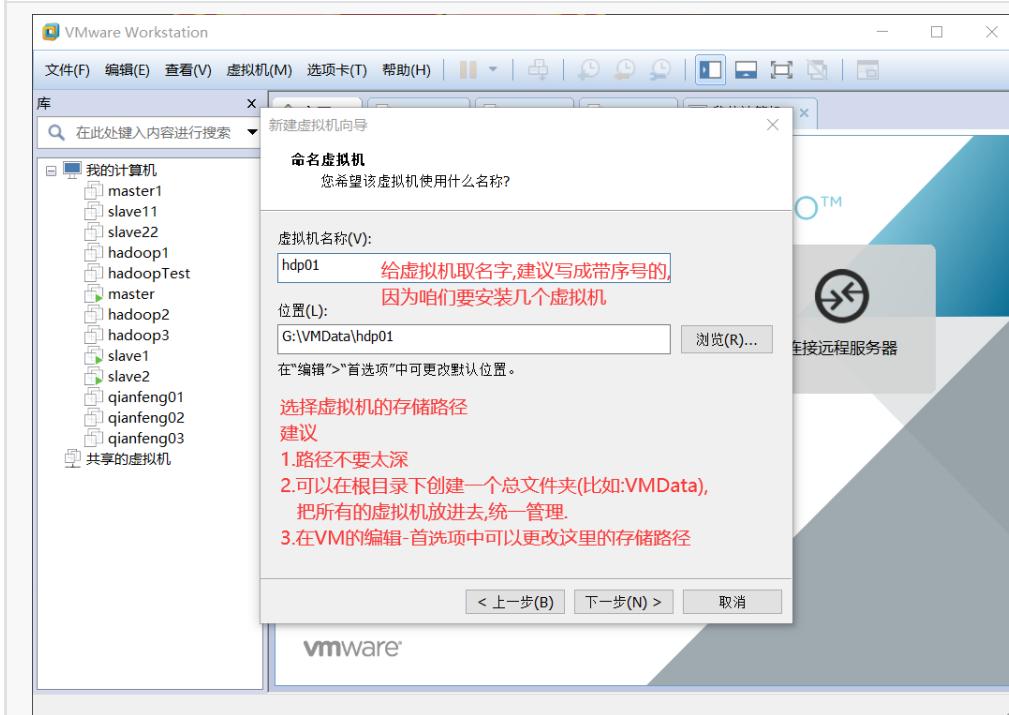


直接选择镜像文件,走完向导直接安装

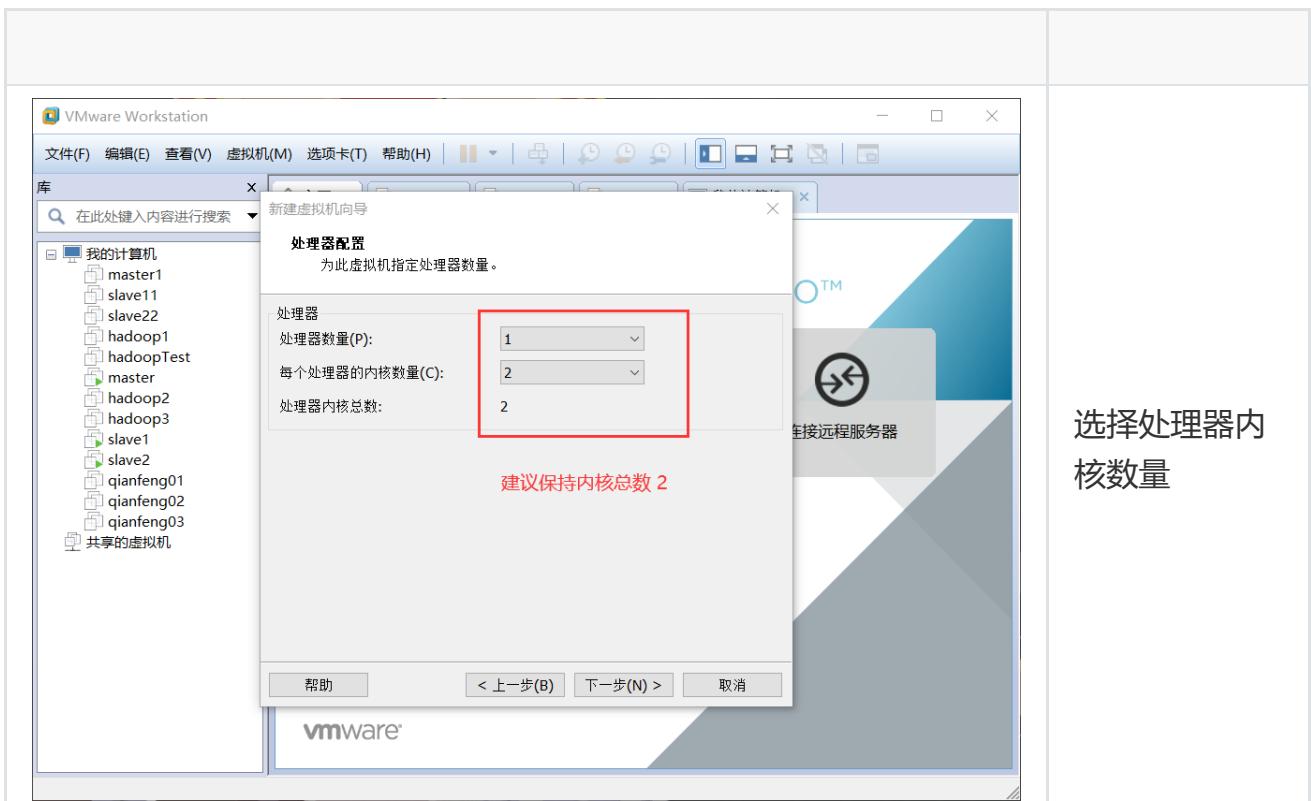
选完之后,点击下一步



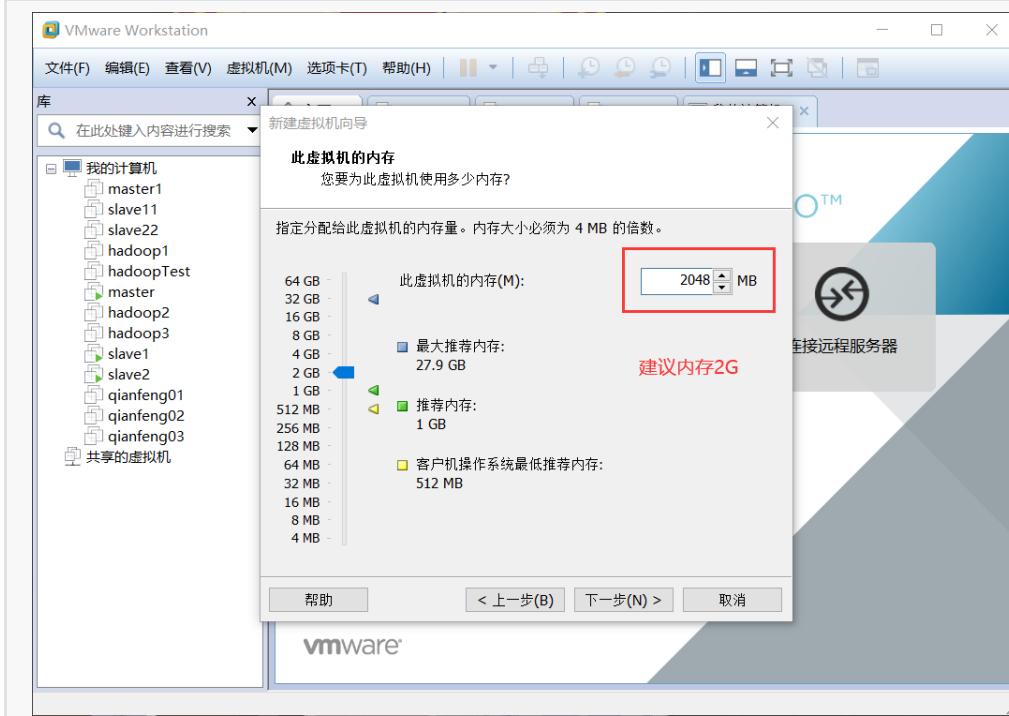
选择安装的系统类型



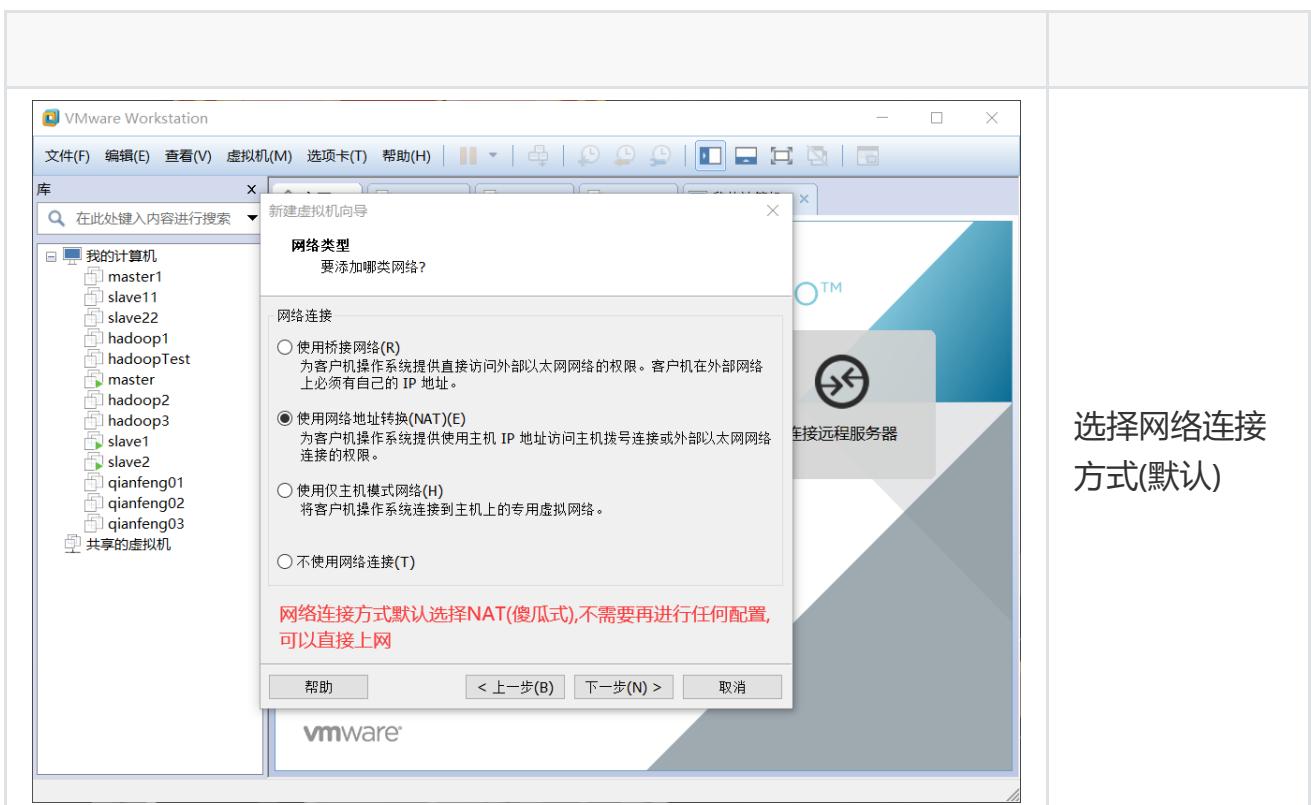
命名虚拟机,选择存放位置



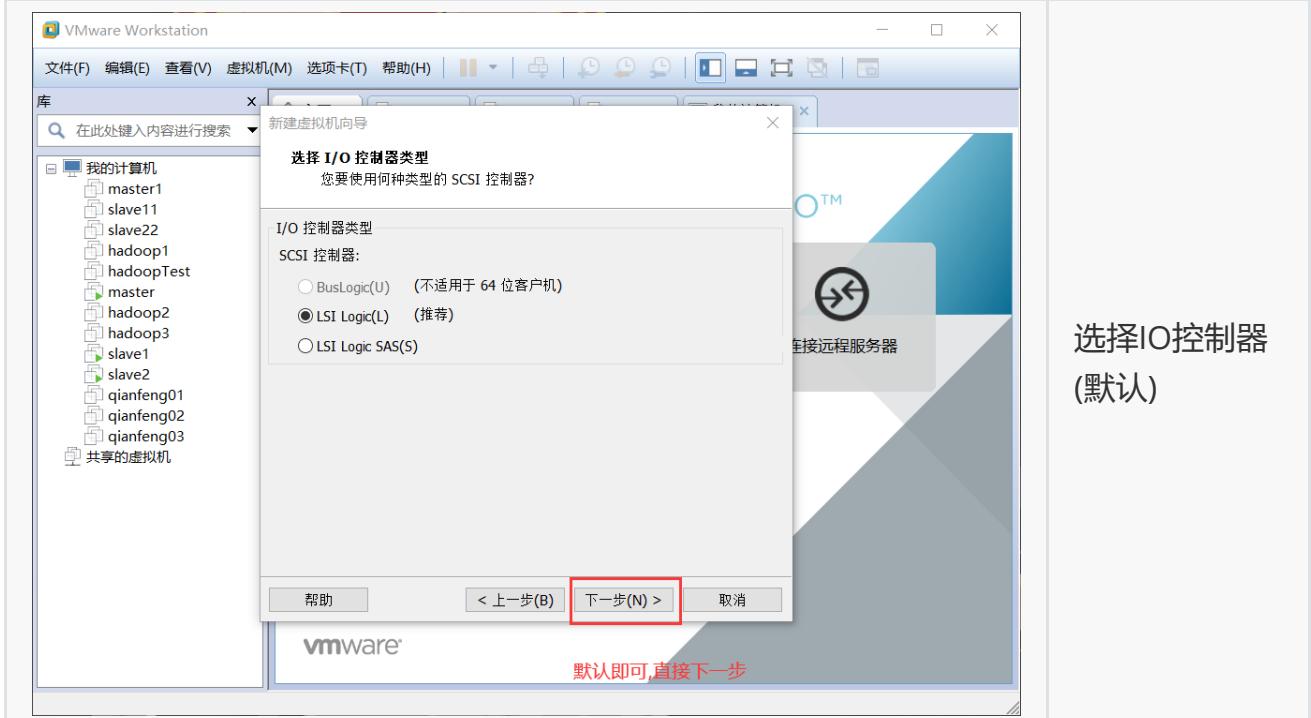
选择处理器内
核数量



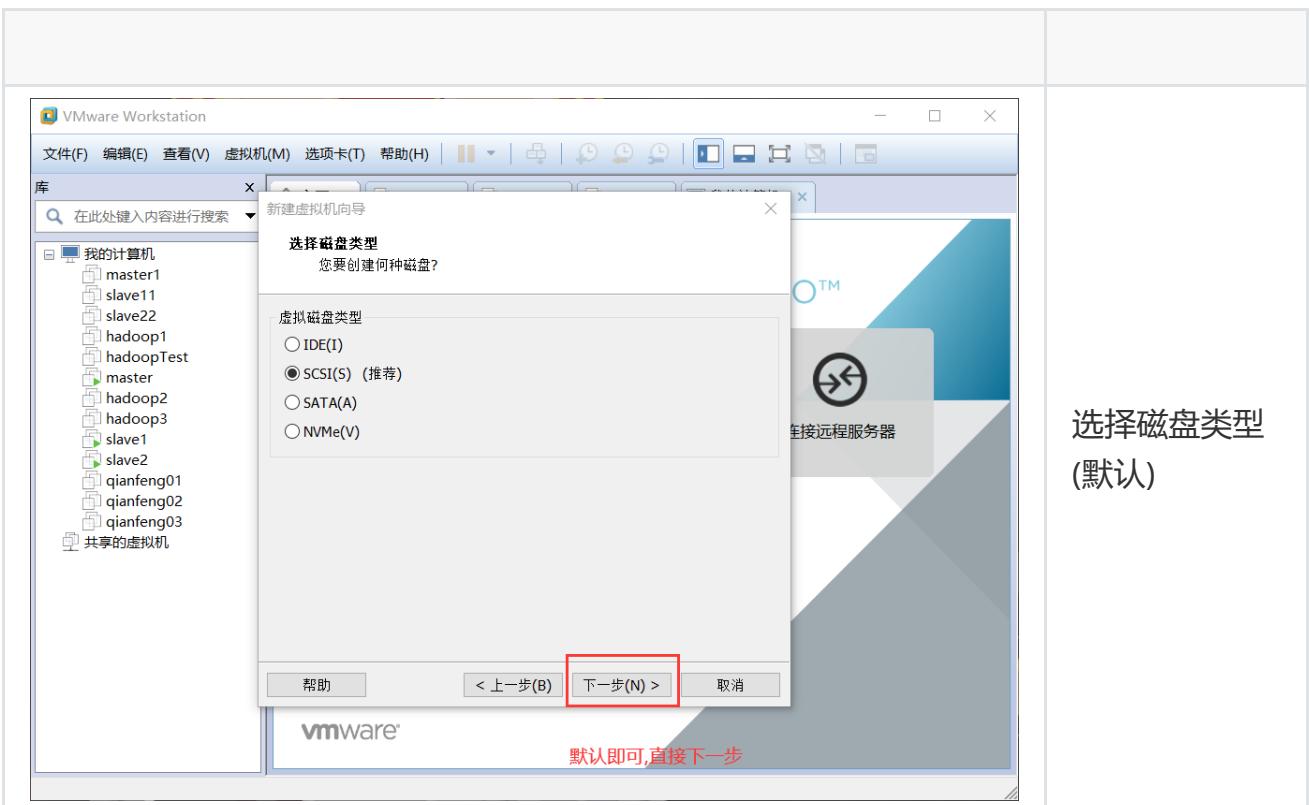
选择内存容量



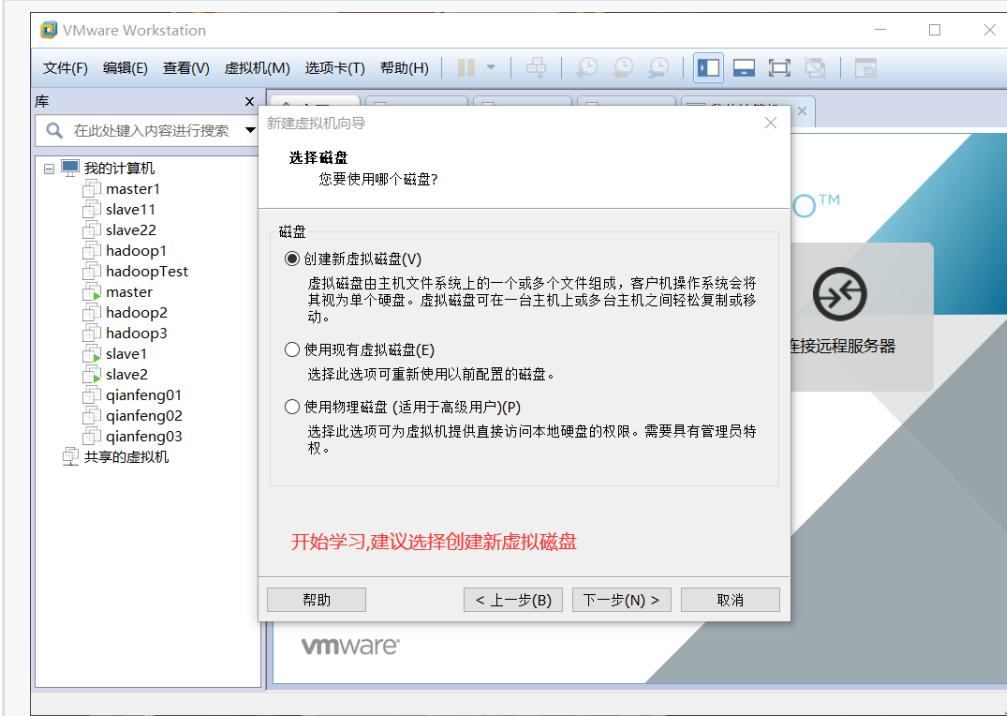
选择网络连接
方式(默认)



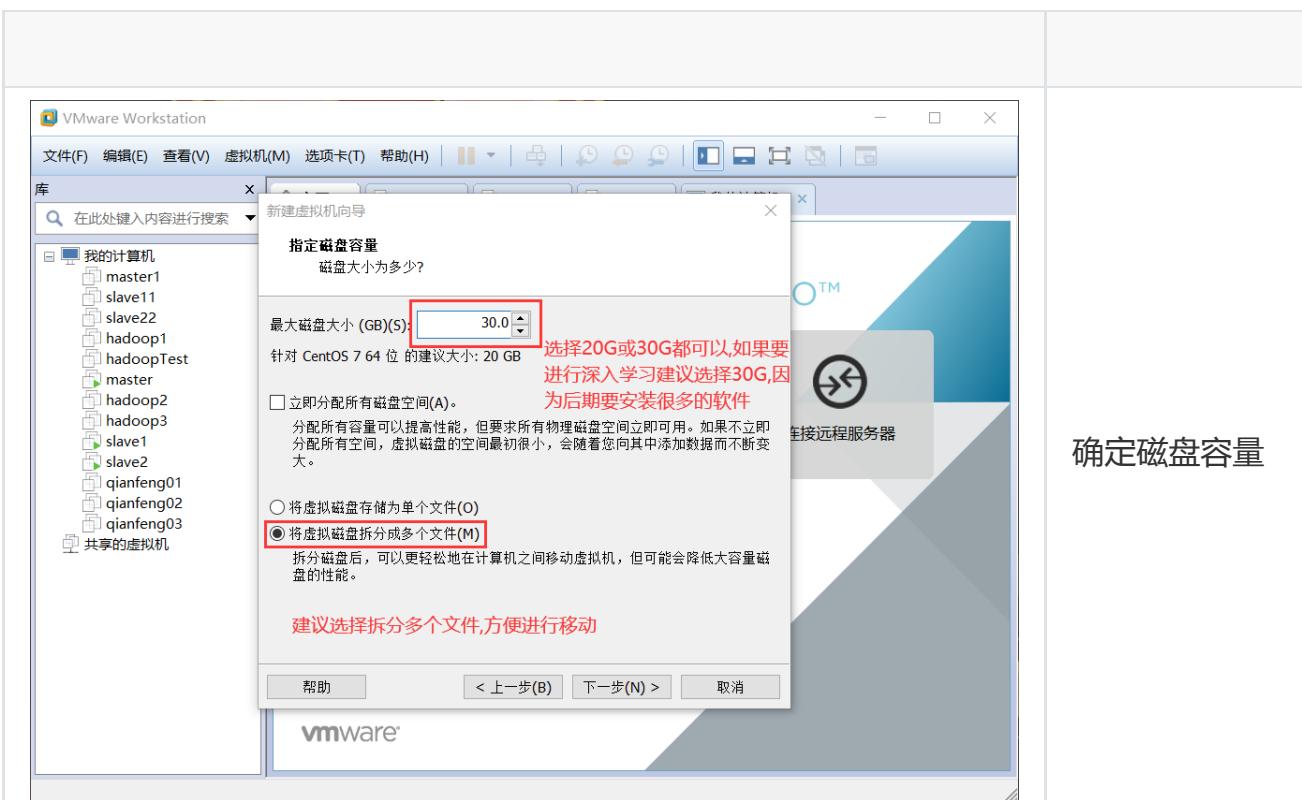
选择I/O控制器
(默认)



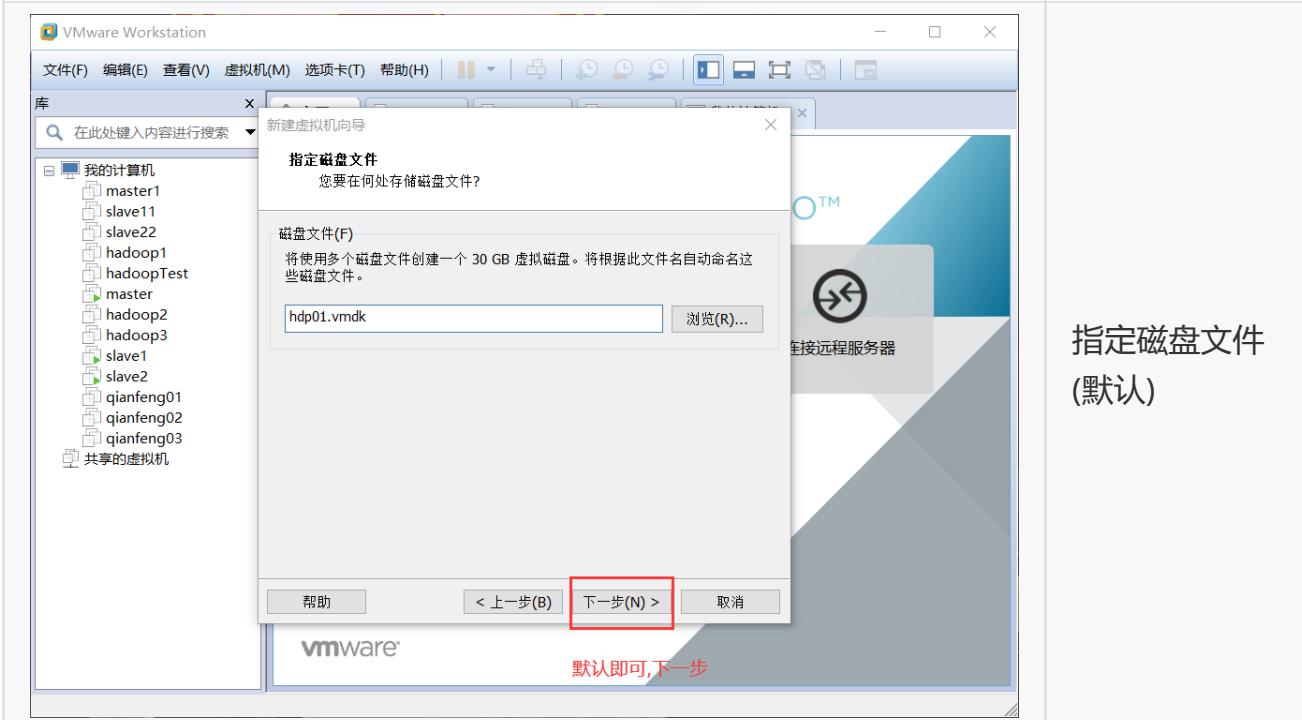
选择磁盘类型
(默认)



选择磁盘(默
认)

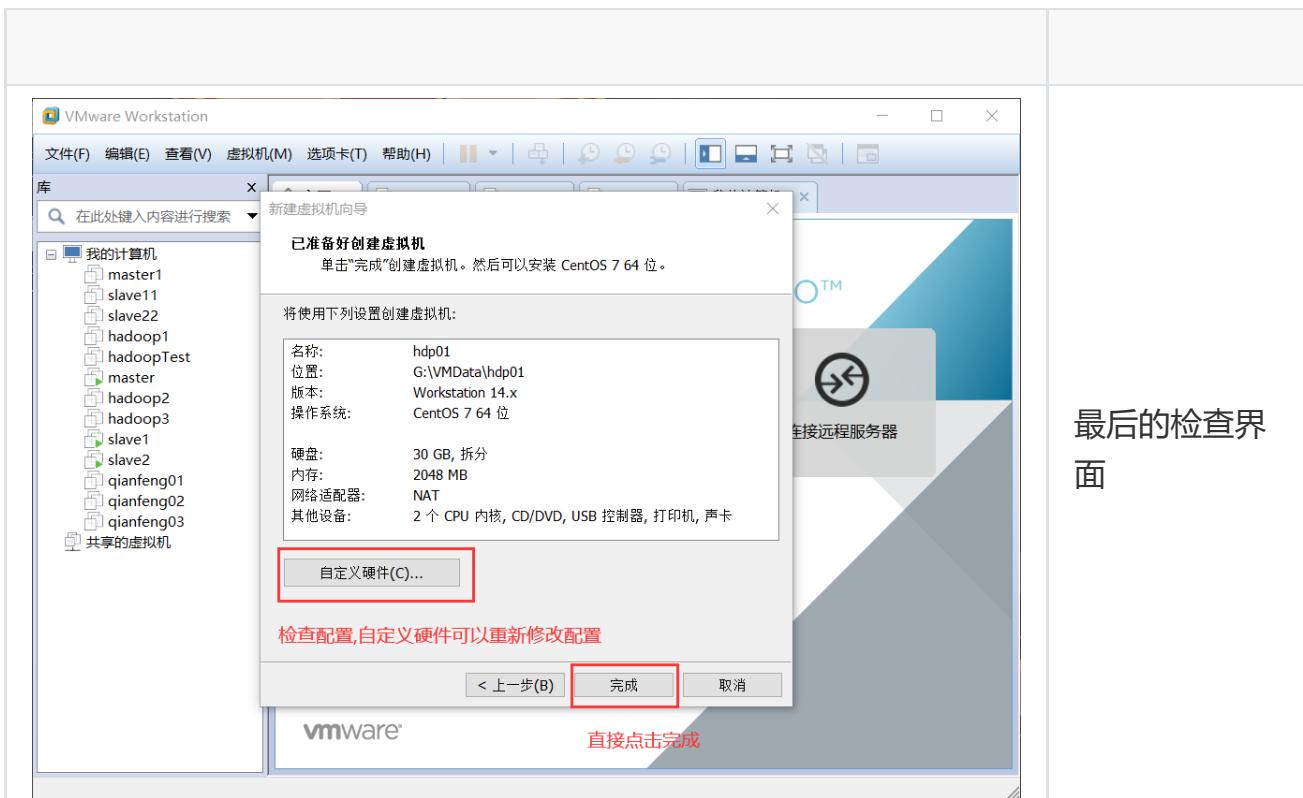


确定磁盘容量

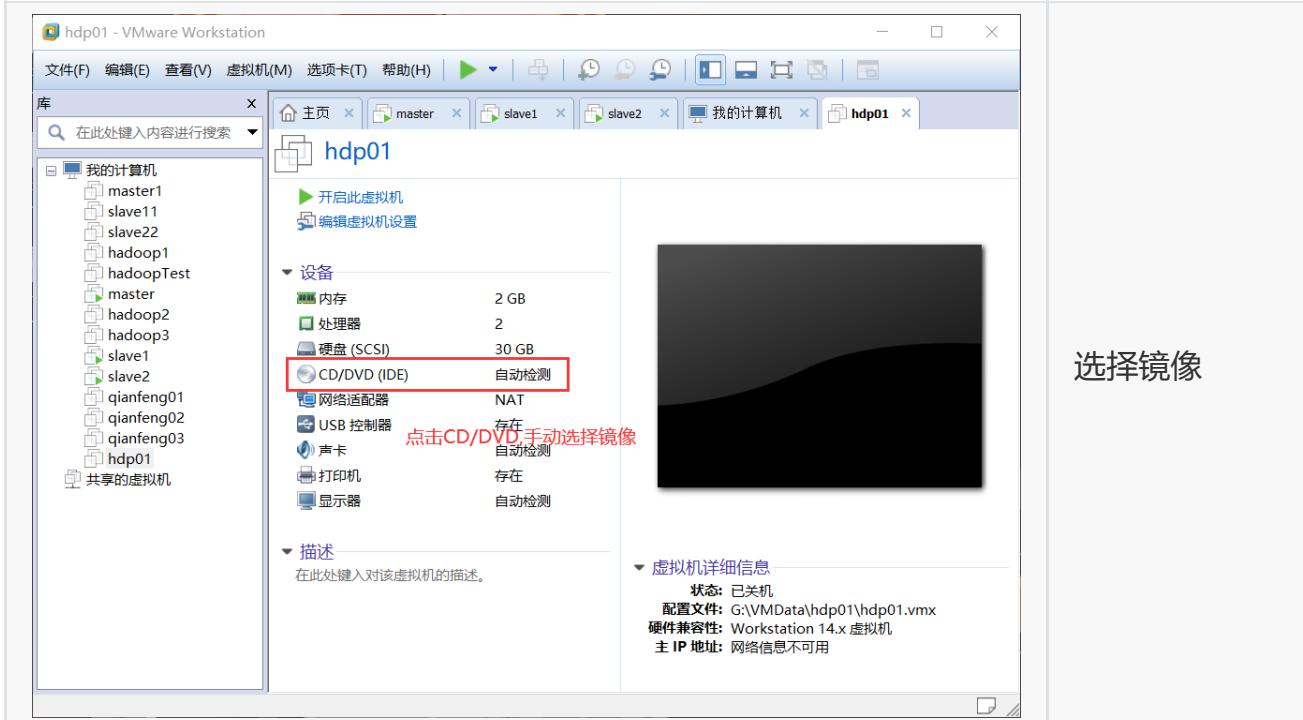


指定磁盘文件
(默认)

默认即可,下一步



最后的检查界
面



选择镜像

虚拟机设置

硬件 选项

设备 简要

内存	2 GB
处理器	2
硬盘 (SCSI)	30 GB
CD/DVD (IDE)	自动检测
网络适配器	NAT
USB 控制器	存在
声卡	自动检测
打印机	存在
显示器	自动检测

设备状态

已连接(C)
 启动时连接(O)

连接

使用物理驱动器(P):
自动检测
 使用 ISO 映像文件(M):
F:\BaiduNetdiskDownload\2020
浏览(B)...

高级(V)...

选择镜像

浏览 ISO 映像

选择要加载的镜像文件

名称	修改日期	类型	大小
hadoop-2.7.6	2020/3/21 2:54	文件夹	
CentOS-7-x86_64-Minimal-1908.iso	2020/3/14 0:56	光盘映像文件	964,608 KB

文件名(N):
CD-ROM 映像 (*.iso)
打开(O) 取消
添加(A)... 移除(R)
确定 取消 帮助

选择镜像 全国

hdp01 - VMware Workstation

文件(F) 编辑(E) 查看(V) 虚拟机(M) 选项卡(T) 帮助(H)

库 在此处键入内容进行搜索

我的计算机 hdp01

▶ 开启此虚拟机 开始安装系统 编辑虚拟机设置

设备

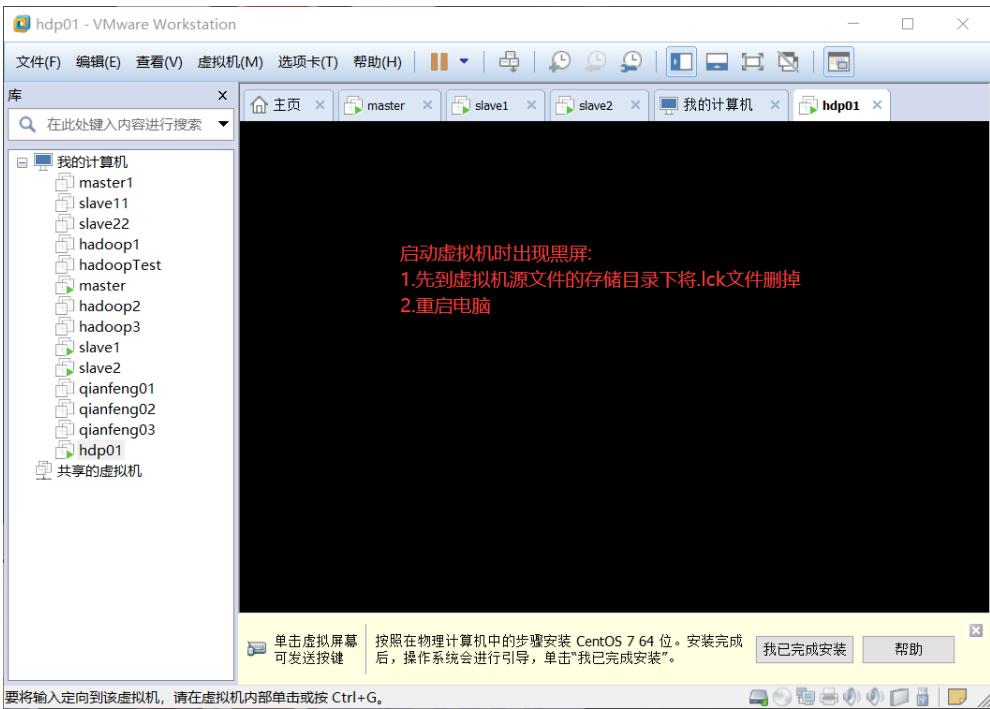
内存	2 GB
处理器	2
硬盘 (SCSI)	30 GB
CD/DVD (IDE)	正在使用 F:
网络适配器	NAT
USB 控制器	存在
声卡	自动检测
打印机	存在
显示器	自动检测

描述 在此处键入对该虚拟机的描述。

虚拟机详细信息

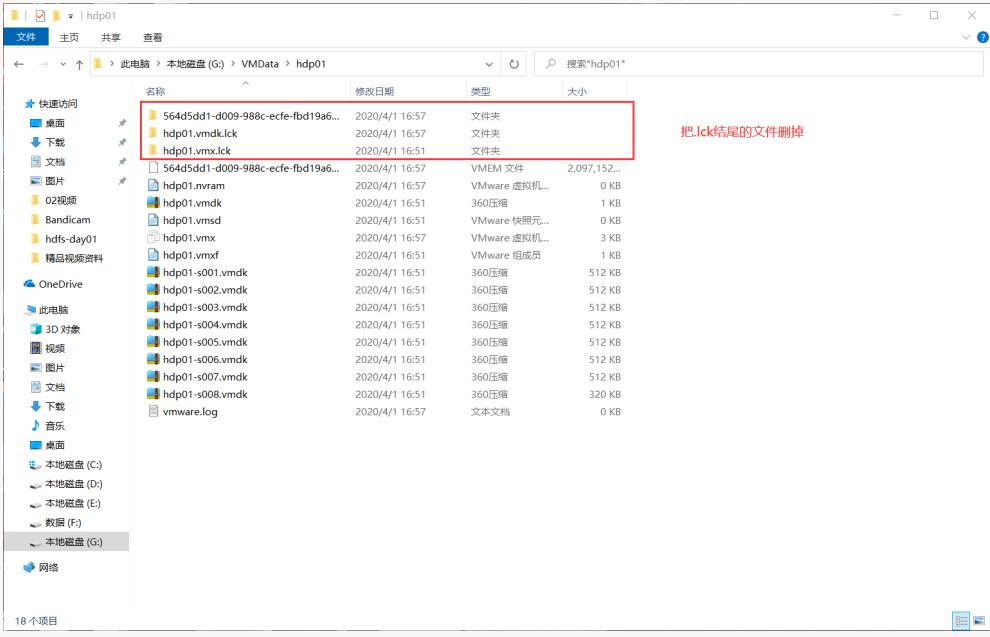
状态: 已关机
配置文件: G:\VMData\hdp01\hdp01.vmx
硬件兼容性: Workstation 14.x 虚拟机
主 IP 地址: 网络信息不可用

开始安装系统



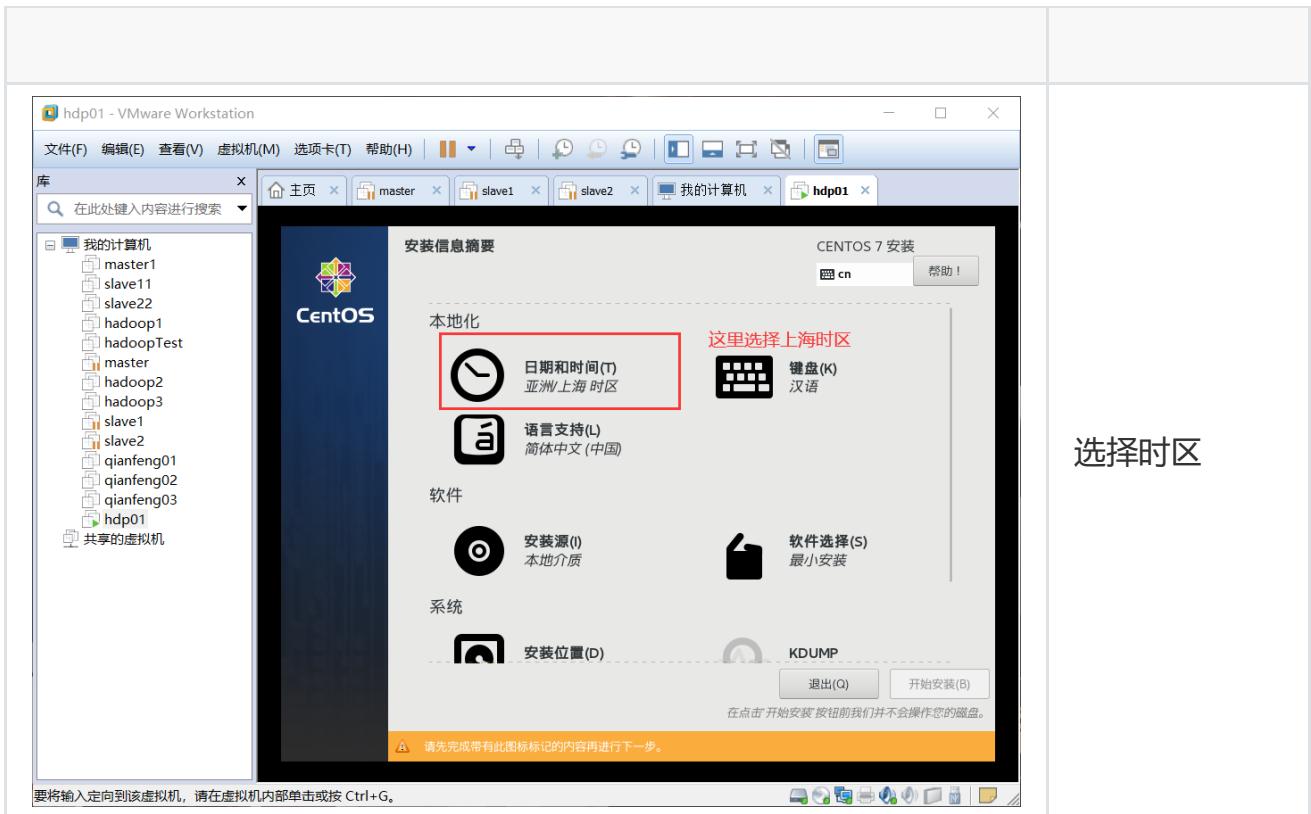
启动虚拟机时出现黑屏:
1.先到虚拟机源文件的存储目录下将.lck文件删掉
2.重启电脑

如果出现黑屏的处理1



把.lck结尾的文件删掉

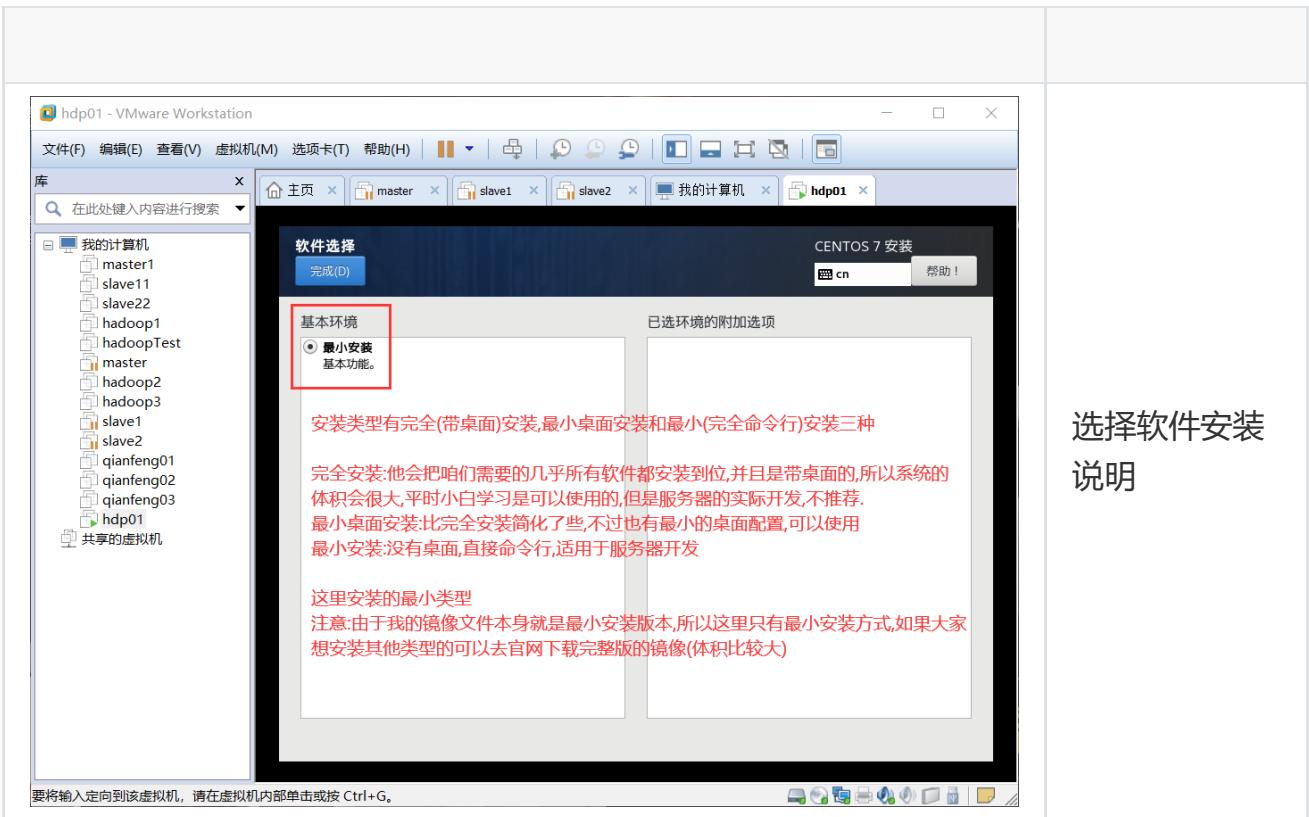
如果出现黑屏的处理2:删除.lck文件



选择时区



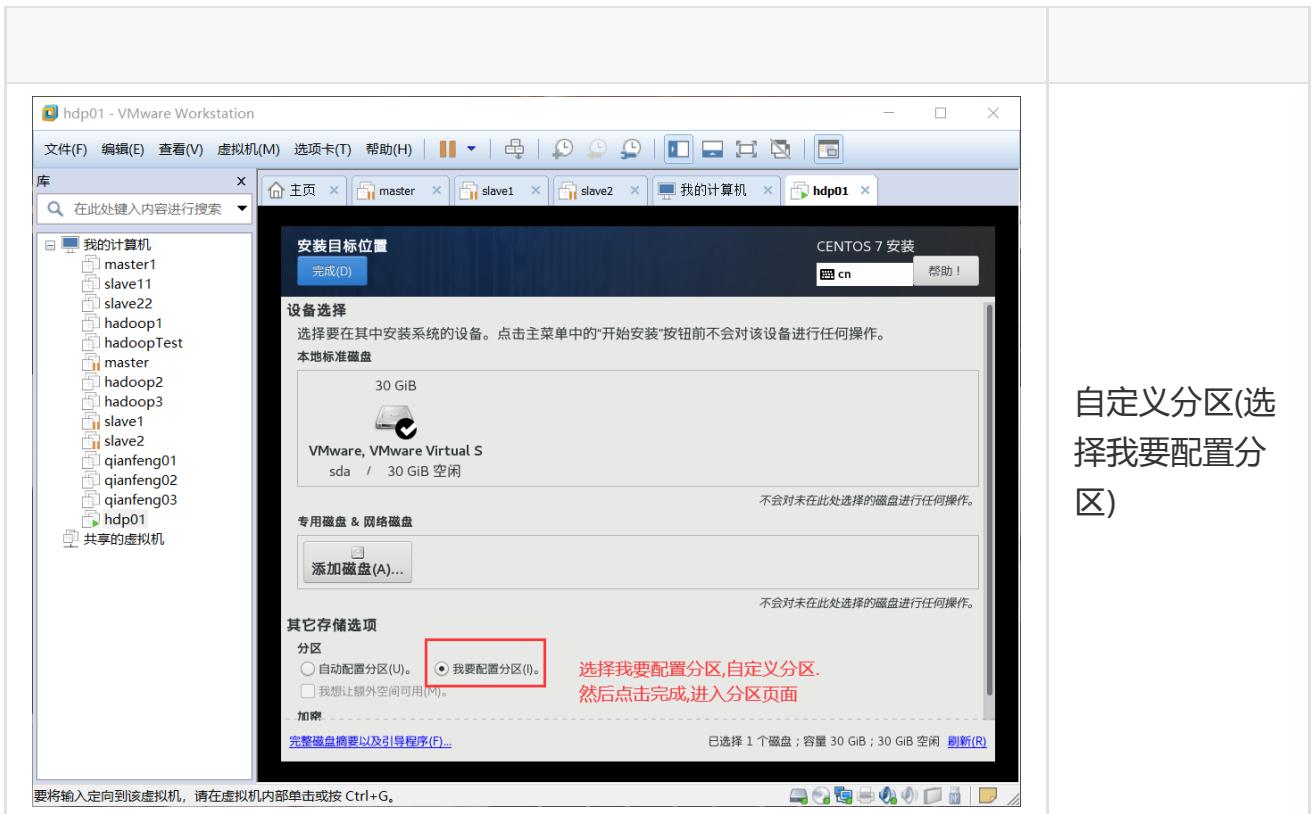
选择软件安装



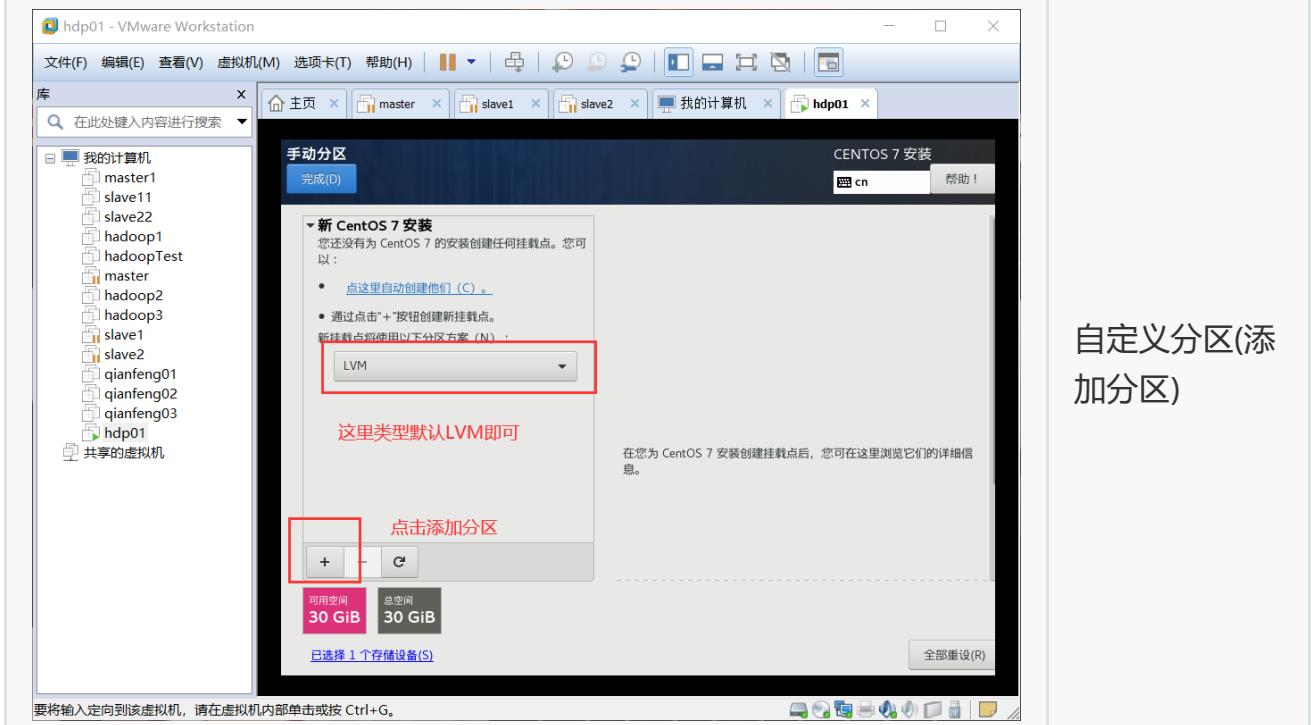
选择软件安装
说明



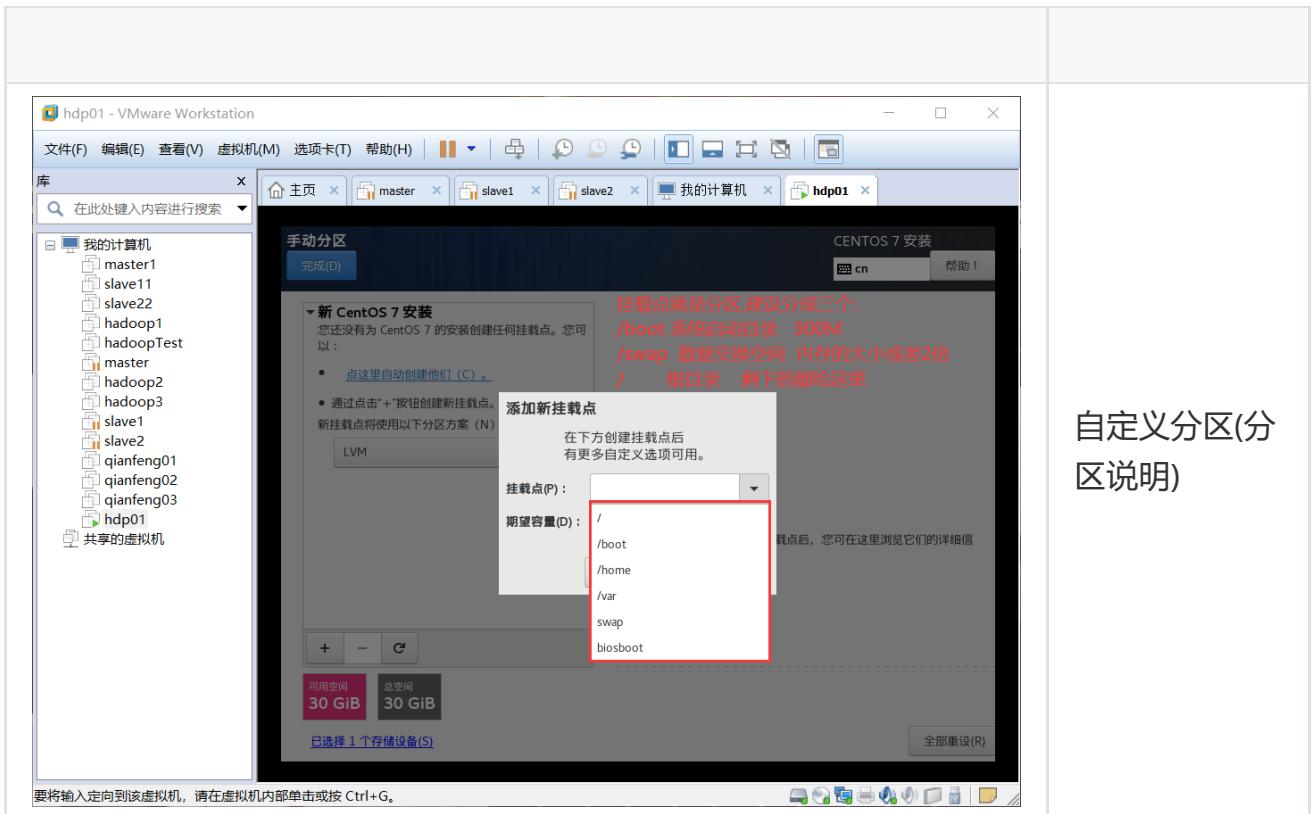
自定义分区(选择自定义分区)



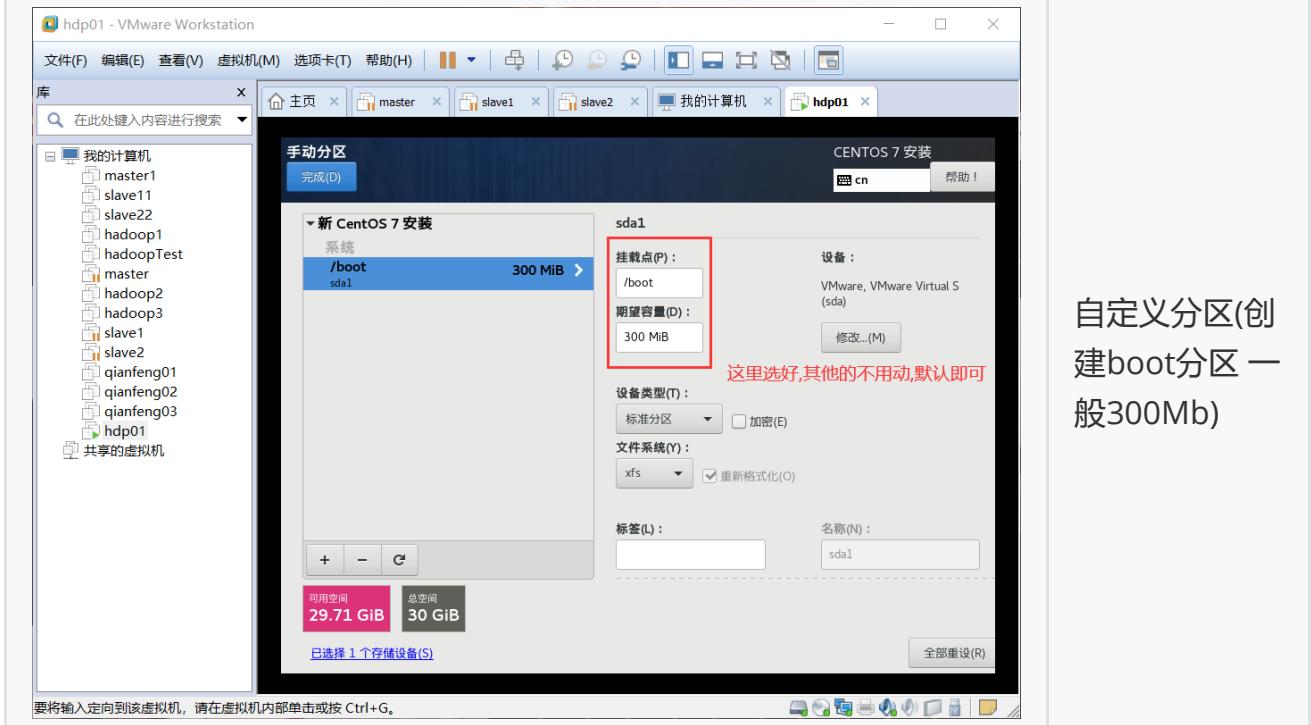
自定义分区(选择我要配置分区)



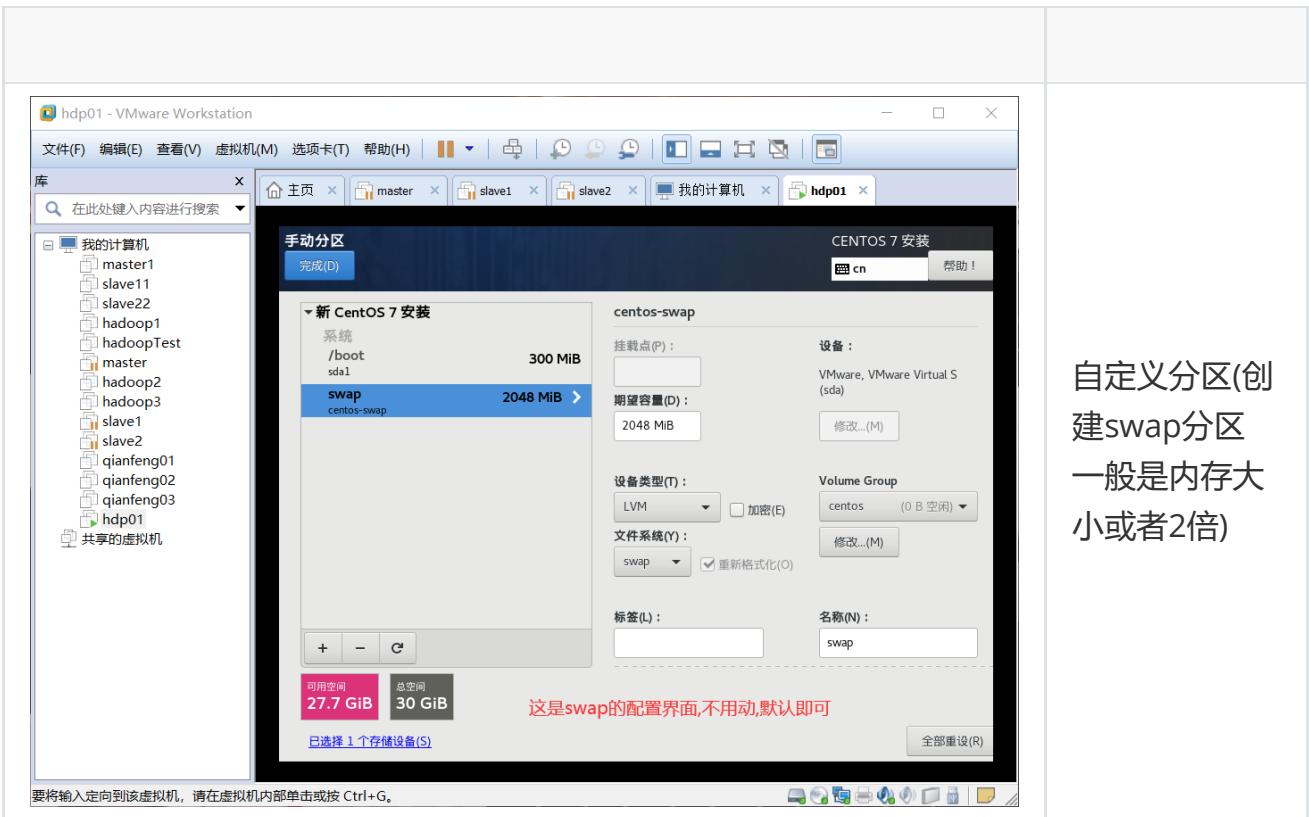
自定义分区(添加分区)



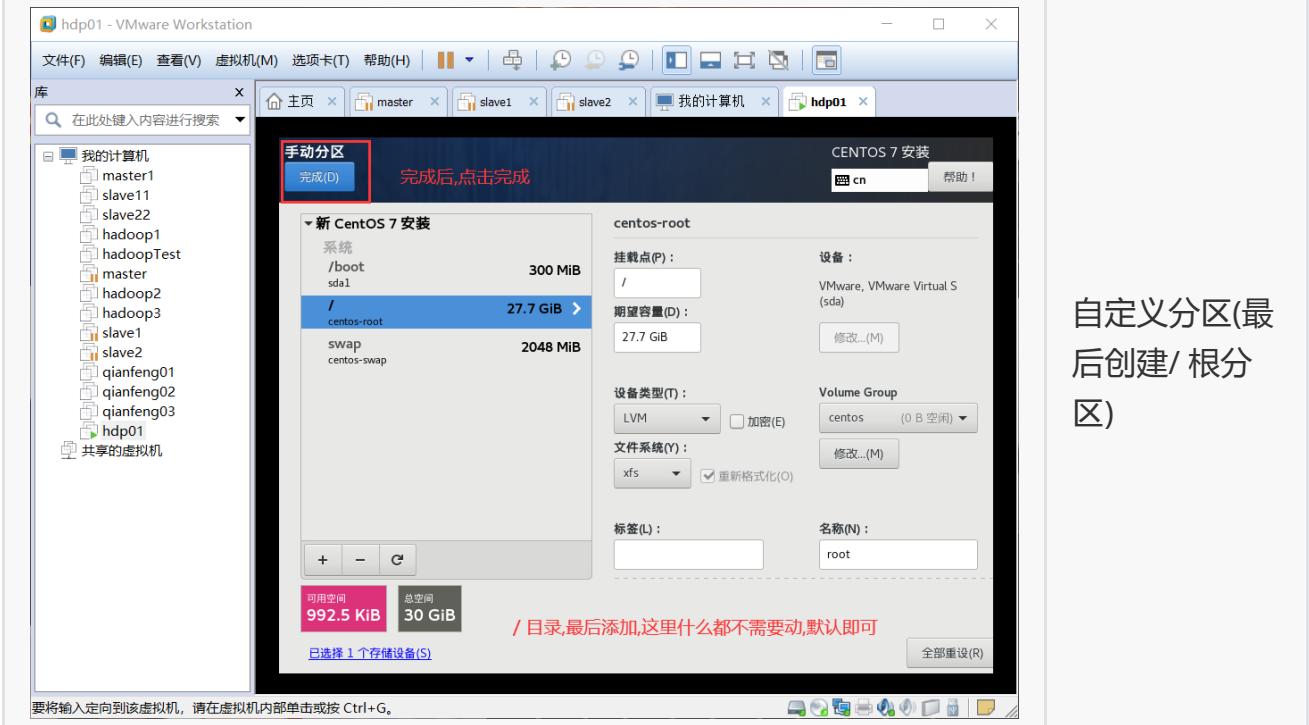
自定义分区(分区说明)



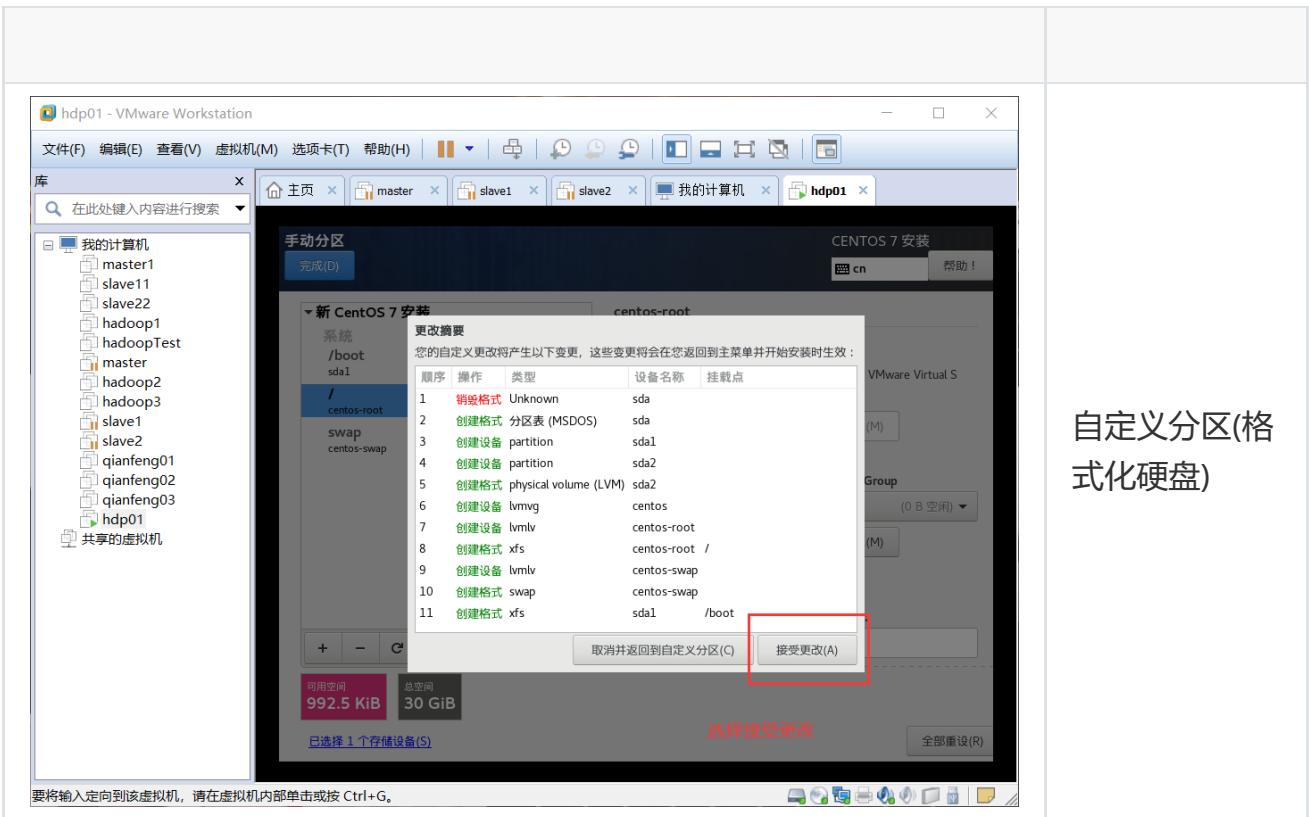
自定义分区(创建boot分区一般300Mb)



自定义分区(创建swap分区
一般是内存大小或者2倍)



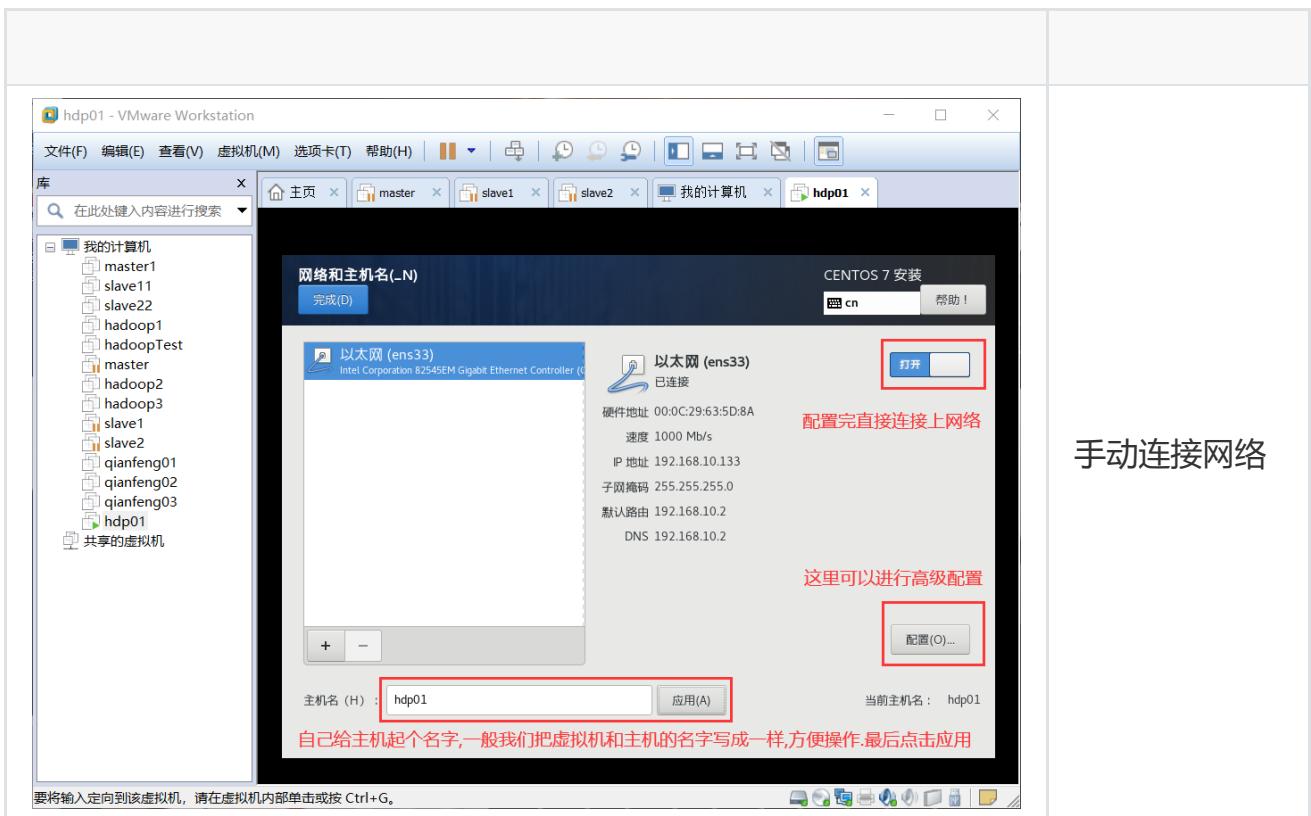
自定义分区(最后创建/根分
区)



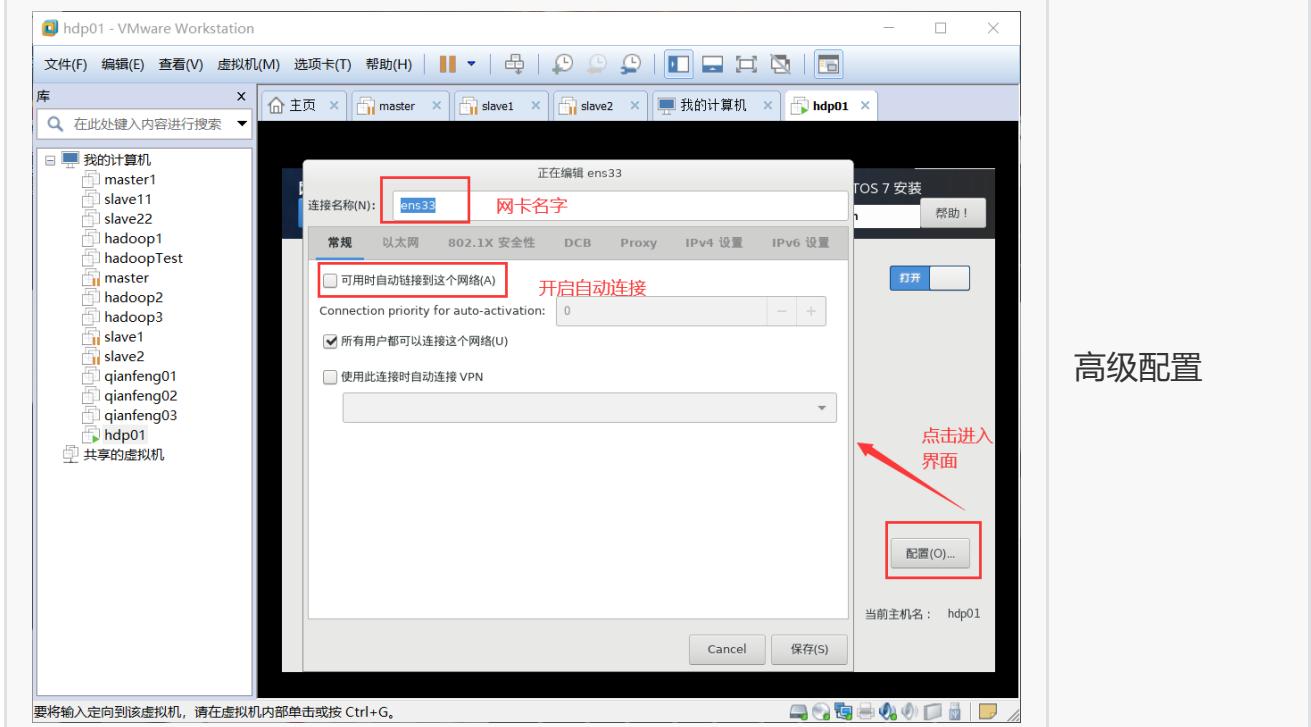
自定义分区(格式化硬盘)



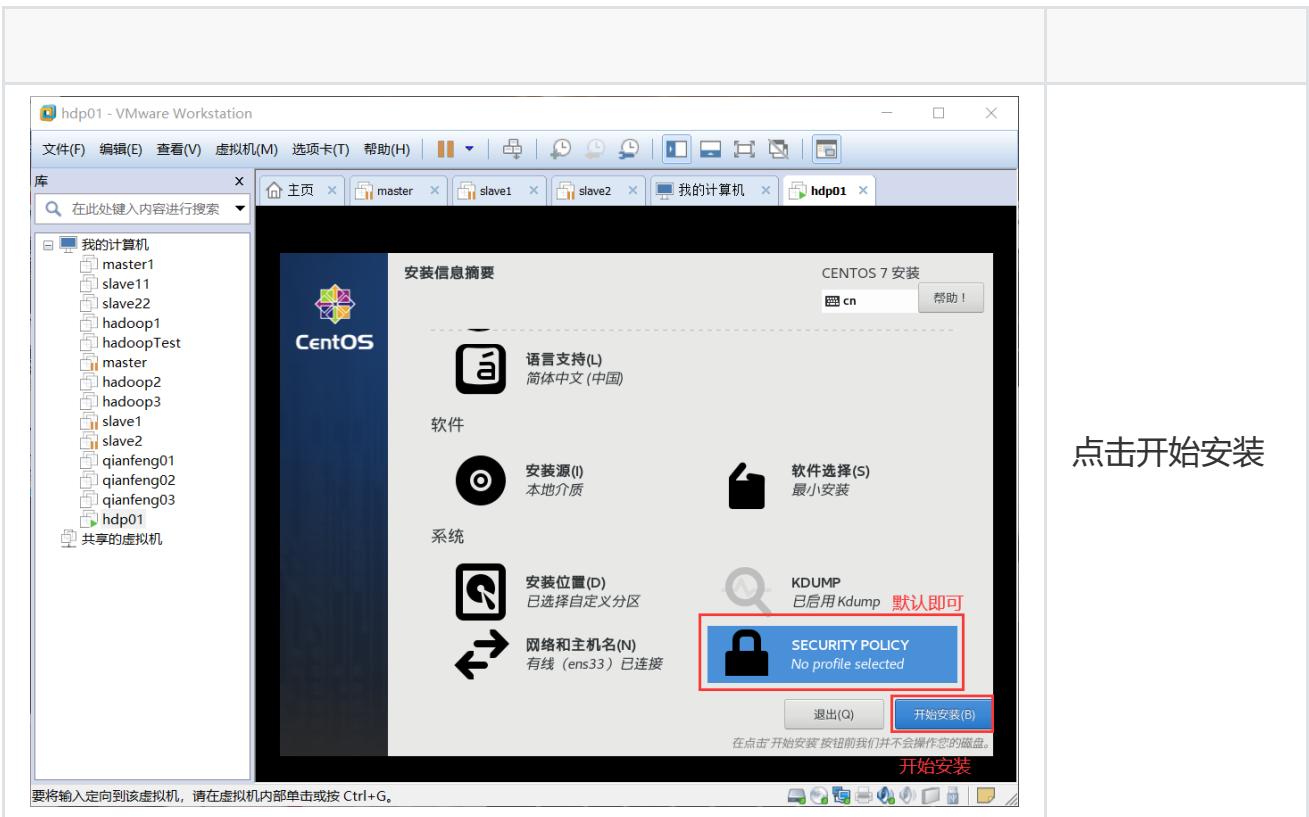
手动连接网络



手动连接网络



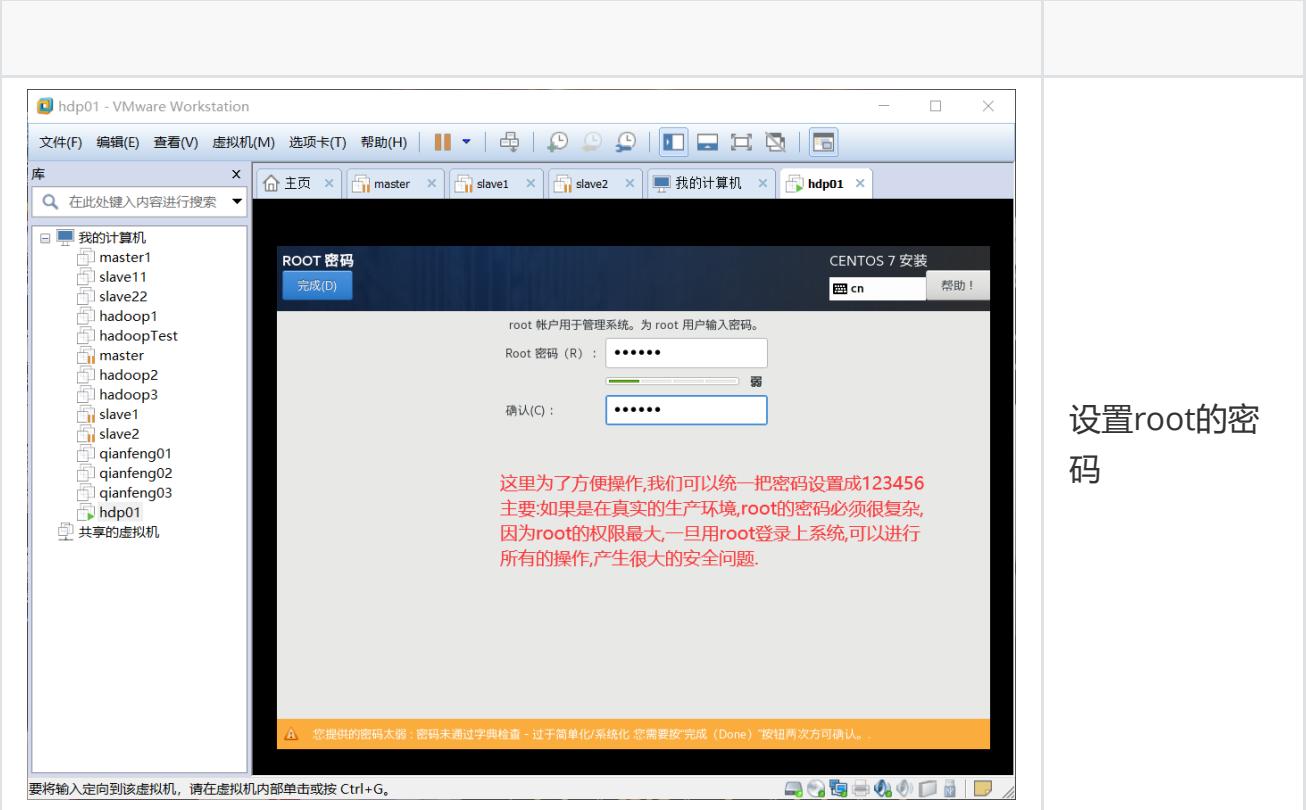
高级配置



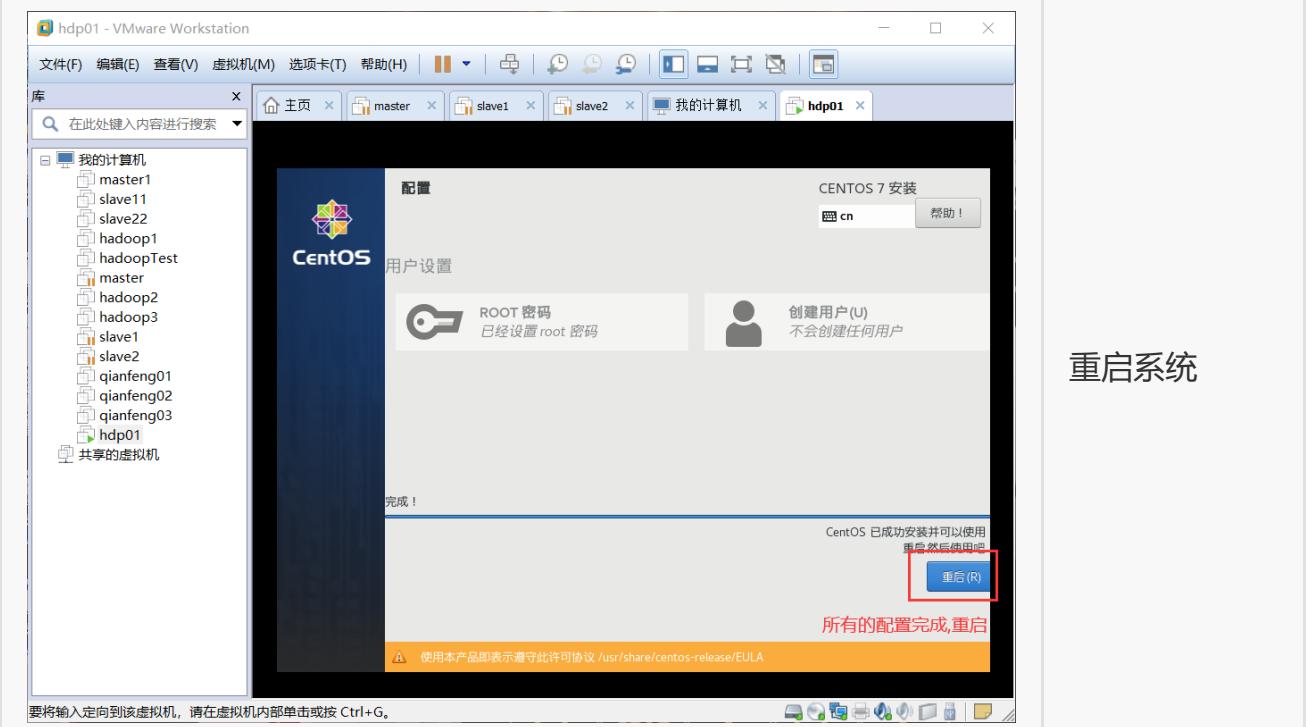
点击开始安装



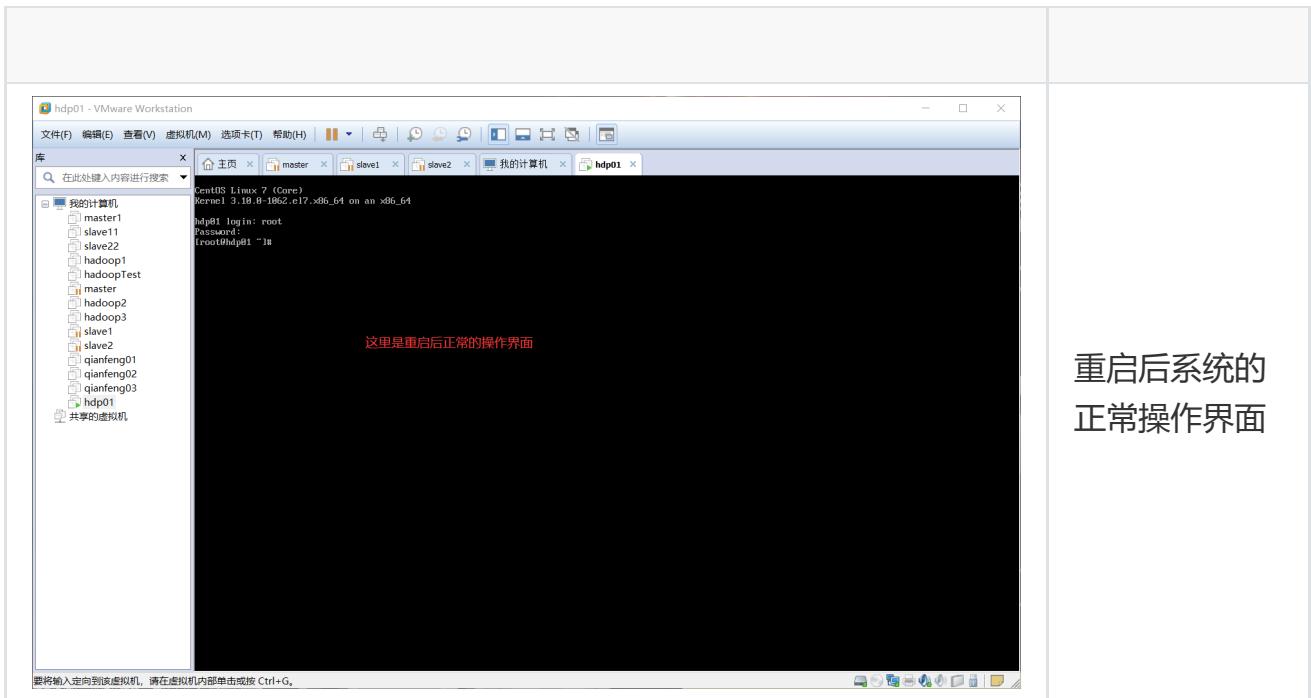
配置用户



设置root的密码



重启系统



重启后系统的
正常操作界面

一些学习方法推荐

转变思维习惯

图形化界面/鼠标 ==> 命令/键盘

Linux 与 windows 主要区别:

Linux严格区分大小写

Linux中一切皆文件

Linux 不区分扩展名 扩展名是为了让管理员使用

不惧英文 勇往直前

记住常用英文

command not found Permission denied

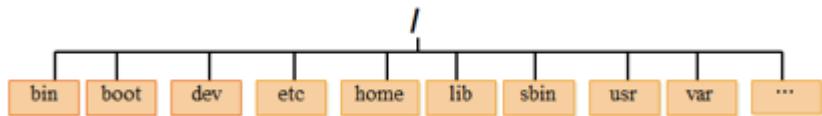
动起来 练习

多练多练多练

命令要敲的999，虚拟机环境可以随意折腾

1.4.3 Linux目录简介

1.4.3.1 目录结构



1.4.3.2 各目录的作用

/bin: (binaries) 存放系统命令的目录，所有用户都可以执行。

/sbin: (super user binaries) 保存和系统环境设置相关的命令，只有超级用户可以使用这些命令，有些命令可以允许普通用户查看。

/usr/bin: 存放系统命令的目录，所有用户可以执行。这些命令和系统启动无关，单用户模式下不能执行

/usr/sbin: 存放根文件系统不必要的系统管理命令，超级用户可执行

/root: 存放root用户的相关文件,root用户的家目录。宿主目录 超级用户

/home/当前普通用户的文件夹,名字默认就是当前的用户 举例:/home/bihai

/cd: 用户缺省宿主目录 eg:/home/spark

/tmp: (temporary)存放临时文件

/etc: (etcetera)系统配置文件

/usr: (unix software resource) 系统软件共享资源目录，存放所有命令、库、手册页等

/proc: 虚拟文件系统，数据保存在内存中，存放当前进程信息

/boot: 系统启动目录

/dev: (devices)存放设备文件

/sys: 虚拟文件系统，数据保存在内存中，主要保存于内存相关信息

/lib: 存放系统程序运行所需的共享库

/lost+found: 存放一些系统出错的检查结果。

/var: (variable) 动态数据保存位置，包含经常发生变动的文件，如邮件、日志文件、计划任务等

/mnt: (mount)挂载目录。临时文件系统的安装点，默认挂载光驱和软驱的目录

/media: 挂载目录。挂载媒体设备，如软盘和光盘

/misc: 挂载目录。挂载NFS服务

/opt: 第三方安装的软件保存位置。习惯放在/usr/local/目录下

/srv: 服务数据目录

1.5 VMWare的三种网络连接

1.5.1 网络环境的相关概念

1 外网: IP是唯一性, 不能重复。范围: 0.0.0.0~255.255.255.255

2 内网: 通过路由器或者交换器设备, 来重新设置IP地址。

不同的内网的ip可以相同。内网A的某一台主机通过自己的网卡与自己的路由器A通信, 然后路由器A通过网络中心

与路由器B通信, 路由器B再与内网中的某一台机器的网卡通信。

3 IP: ip地址是确定一台机器的唯一标识符

4 NETMASK (子网掩码) :与ip连用, 用于确定网络段位

192.168.1.x 范围: 1-254

255.255.255.0

子网掩码有1的位置对应ip的部分就是网络段位。0对应的位置就是内网中主机的位置。

子网掩码作用:用来指定当前机器属于哪个ip号段的

判断方式:将当前的ip的二进制按位与子网掩码的二进制等于网段地址

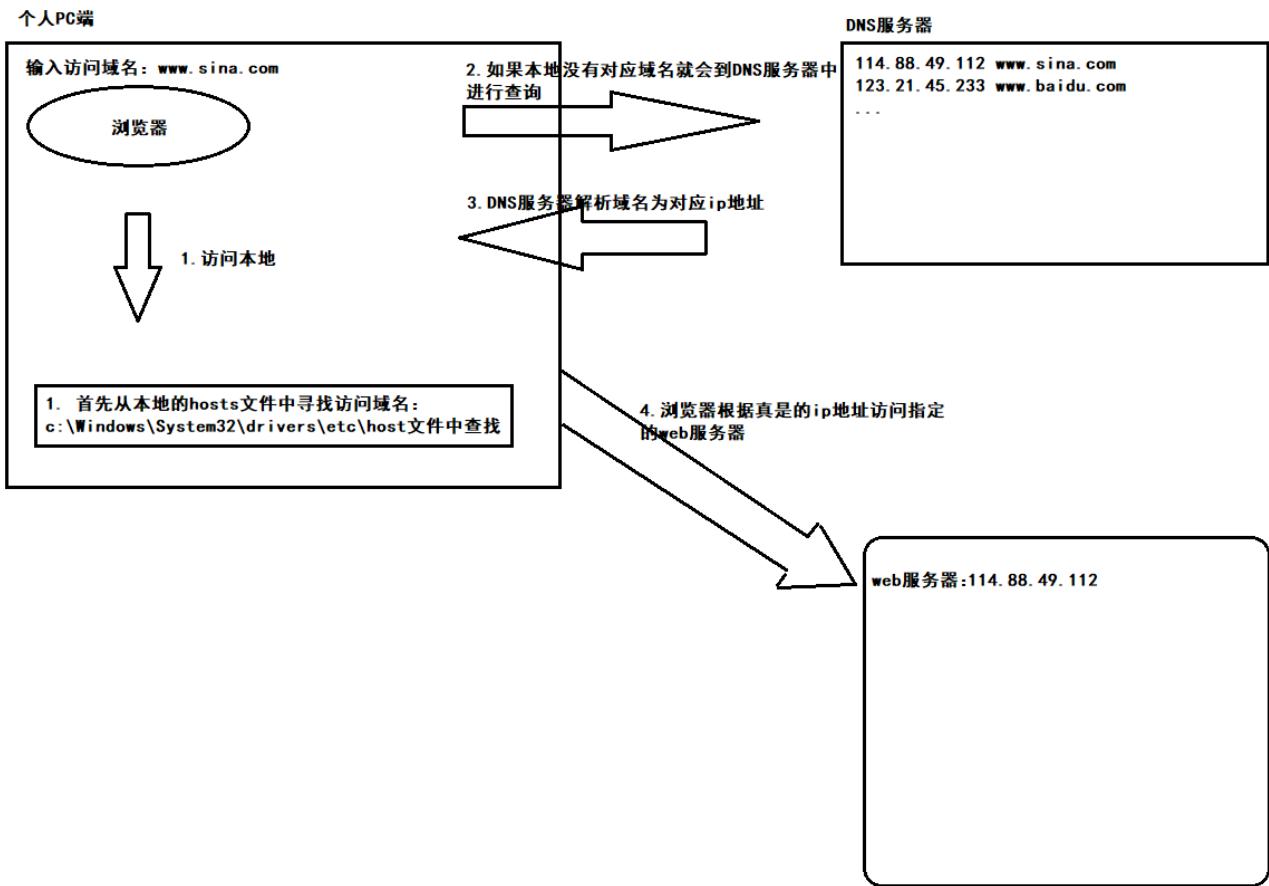
如: 192.168.1.2 & 255.255.255.0 = 192.168.1.0

5 GATEWAY: 网关, 用于与连接外网的机器设备通信 (路由器) 换句话说, 网关就是路由器的IP

6 DNS: 域名解析服务器

119.75.217.109 www.baidu.com

1.5.2 互联网DNS服务器的域名访问机制



1.5.3 设备介绍

1 路由器

路由器是连接两个或多个网络的硬件设备，在网络间起网关的作用，是读取每一个数据包中的地址然后决定如何传送的专用智能性的网络设备。它能够理解不同的协议，例如某个局域网使用的以太网协议，因特网使用的TCP/IP协议。这样，路由器可以分析各种不同类型网络传来的数据包的目的地址，把非TCP/IP网络的地址转换成TCP/IP地址，或者反之；再根据选定的路由算法把各数据包按最佳路线传送到指定位置。所以路由器可以把非TCP/ IP网络连接到因特网上。

2 交换机

交换机 (Switch) 意为“开关”是一种用于电（光）信号转发的网络设备。它可以为接入交换机的任意两个网络节点提供独享的电信号通路。最常见的交换机是以太网交换机。其他常见的还有电话语音交换机、光纤交换机等。

3 Modem

调制解调器（英文名Modem），俗称“猫”，是一种计算机硬件。它能把计算机的数字信号翻译成可沿普通电话线传送的脉冲信号，而这些脉冲信号又可被线路另一端的另一个调制解调器接收，并译成计算机可懂的语言。

计算机内的信息是由“0”和“1”组成数字信号，而在电话线上传递的却只能是模拟电信号。于是，当两台计算机要通过电话线进行数据传输时，就需要一个设备负责数模的转换。

4 虚拟机

虚拟机（Virtual Machine）指通过软件模拟的具有完整硬件系统功能的、运行在一个完全隔离环境中的完整计算机系统。在实体计算机中能够完成的工作在虚拟机中都能够实现。在计算机中创建虚拟机时，需要将实体机的部分硬盘和内存容量作为虚拟机的硬盘和内存容量。每个虚拟机都有独立的CMOS、硬盘和操作系统，可以像使用实体机一样对虚拟机进行操作。

1.5.4 虚拟机网络连接方式

安装完虚拟机后，默认安装了如下图的两块虚拟网卡——VMnet1和VMnet8，其中VMnet1是host网卡，用于host方式连接网络；VMnet8是NAT网卡，用于NAT方式连接网络的，并且它们的IP地址是随机生成的。

VMware提供了三种工作模式，分别是bridged（桥接模式）、NAT（网络地址转换模式）和host-only（仅主机模式）。

VMware所在主机的设置，见下图



1.5.4.1 bridged (桥接模式)

在这种模式下，使用VMnet0虚拟交换机，虚拟操作系统就像是局域网中的一台独立的主机，与宿主计算机一样，它可以访问网内任何一台机器。在桥接模式下，可以手工配置它的TCP/IP配置信息（IP、子网掩码等，而且还要和宿主机器处于同一网段），以实现通过局域网的网关或路由器访问互联网，还可以将IP地址和DNS设置成“自动获取”。如果你想利用VMWare在局域网内新建一个服务器，为局域网用户提供Web或网络服务，就应该选择桥接模式。

在桥接模式中，使用VMnet0虚拟交换机，此时虚拟机相当于网络上的一台独立计算机与主机一样，拥有一个独立的IP地址。

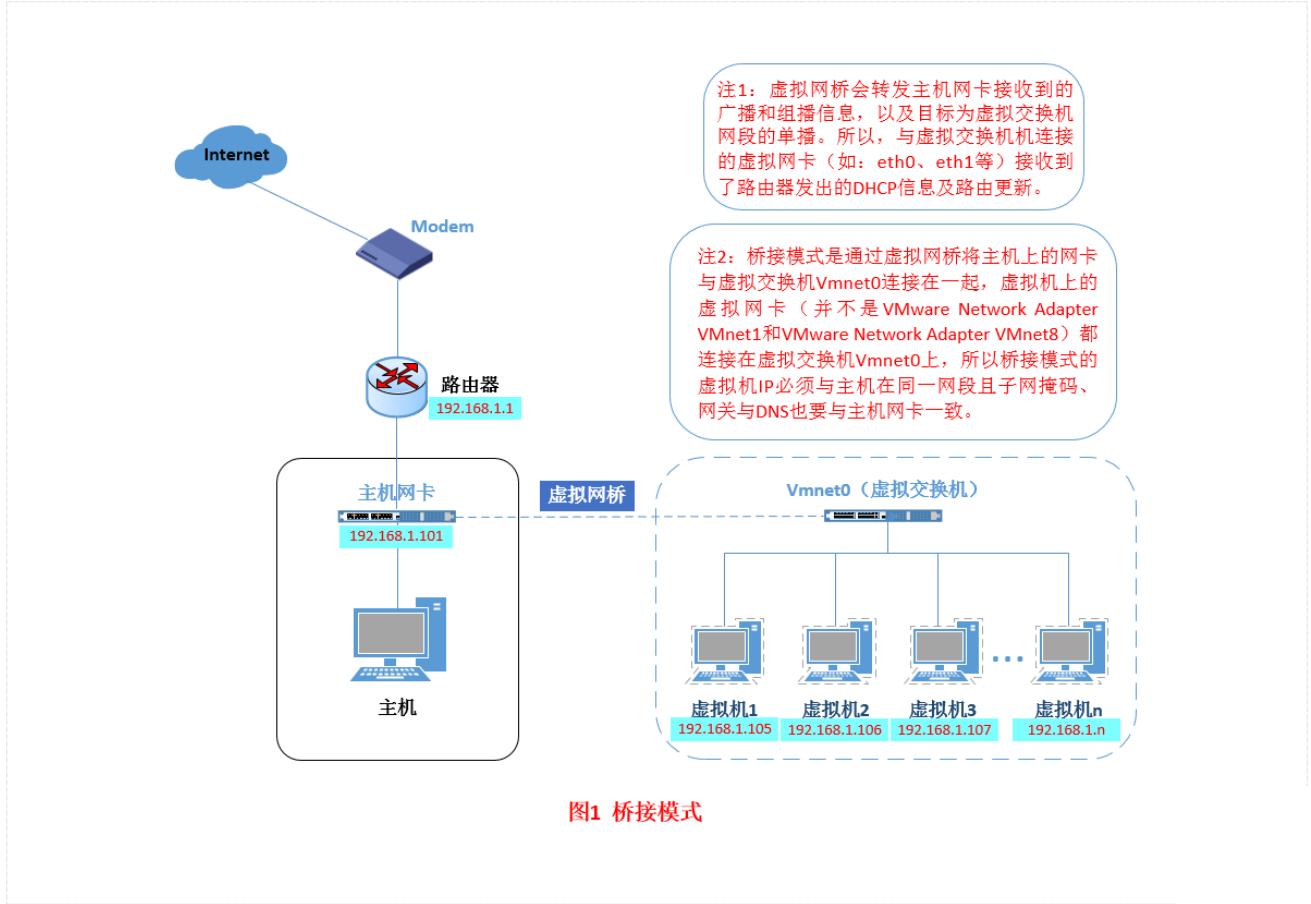


图1 桥接模式

1.5.4.2 NAT (网络地址转换模式)

使用NAT模式，就是让虚拟机借助NAT（网络地址转换）功能，通过宿主机器所在的网络来访问公网。也就是说，使用NAT模式可以实现在虚拟系统里访问互联网。NAT模式下的虚拟机的TCP/IP配置信息是由VMnet8虚拟网络的DHCP服务器提供的，因此IP和DNS一般设置为“自动获取”，因此虚拟系统也就无法和本局域网中的其他真实主机进行通讯。采用NAT模式最大的优势是虚拟系统接入互联网非常简单，你不需要进行任何其他的配置，只需要宿主机器能访问互联网即可。如果你想利用VMWare安装一个新的虚拟系统，在虚拟系统中不用进行任何手工配置就能直接访问互联网，建议你采用NAT模式。

NAT模式中使用Vmnet8虚拟交换机，此时虚拟机可以通过主机“单向访问”网络上的其他主机，其他主机不能访问虚拟机。

注意：主机处的192.168.62.1指的是主机的连接V8交换机的地址，与虚拟交换机处的网关必须是同一个地址

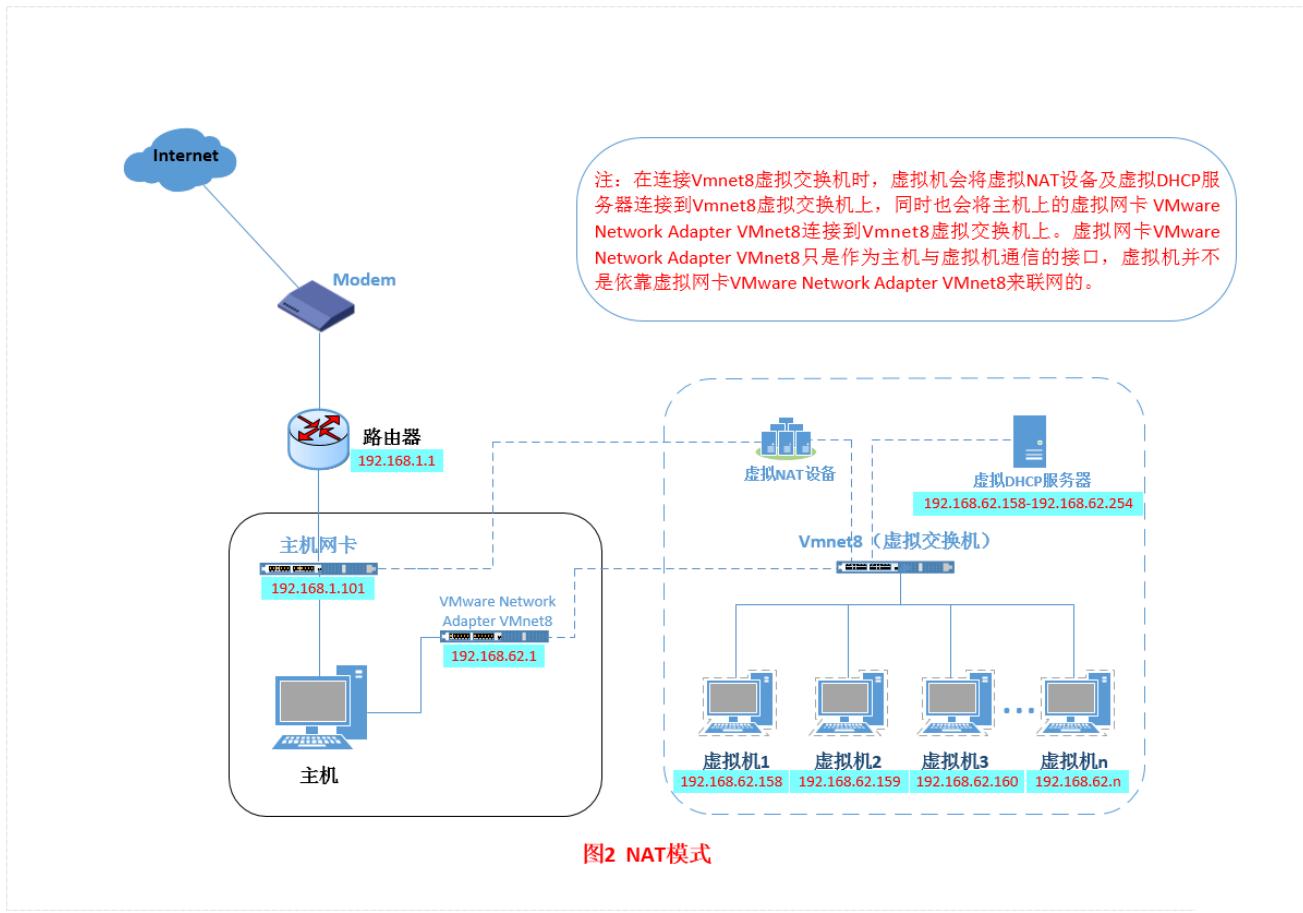


图2 NAT模式

1.5.4.3 host-only (主机模式)

Host-Only模式其实就是NAT模式去除了虚拟NAT设备，然后使用VMware Network Adapter VMnet1虚拟网卡连接VMnet1虚拟交换机来与虚拟机通信的，Host-Only模式将虚拟机与外网隔开，使得虚拟机成为一个独立的系统，只与主机相互通讯。其网络结构如下图所示：

如果要使得虚拟机能联网，我们可以将主机网卡共享给VMware Network Adapter VMnet1网卡，从而达到虚拟机联网的目的。

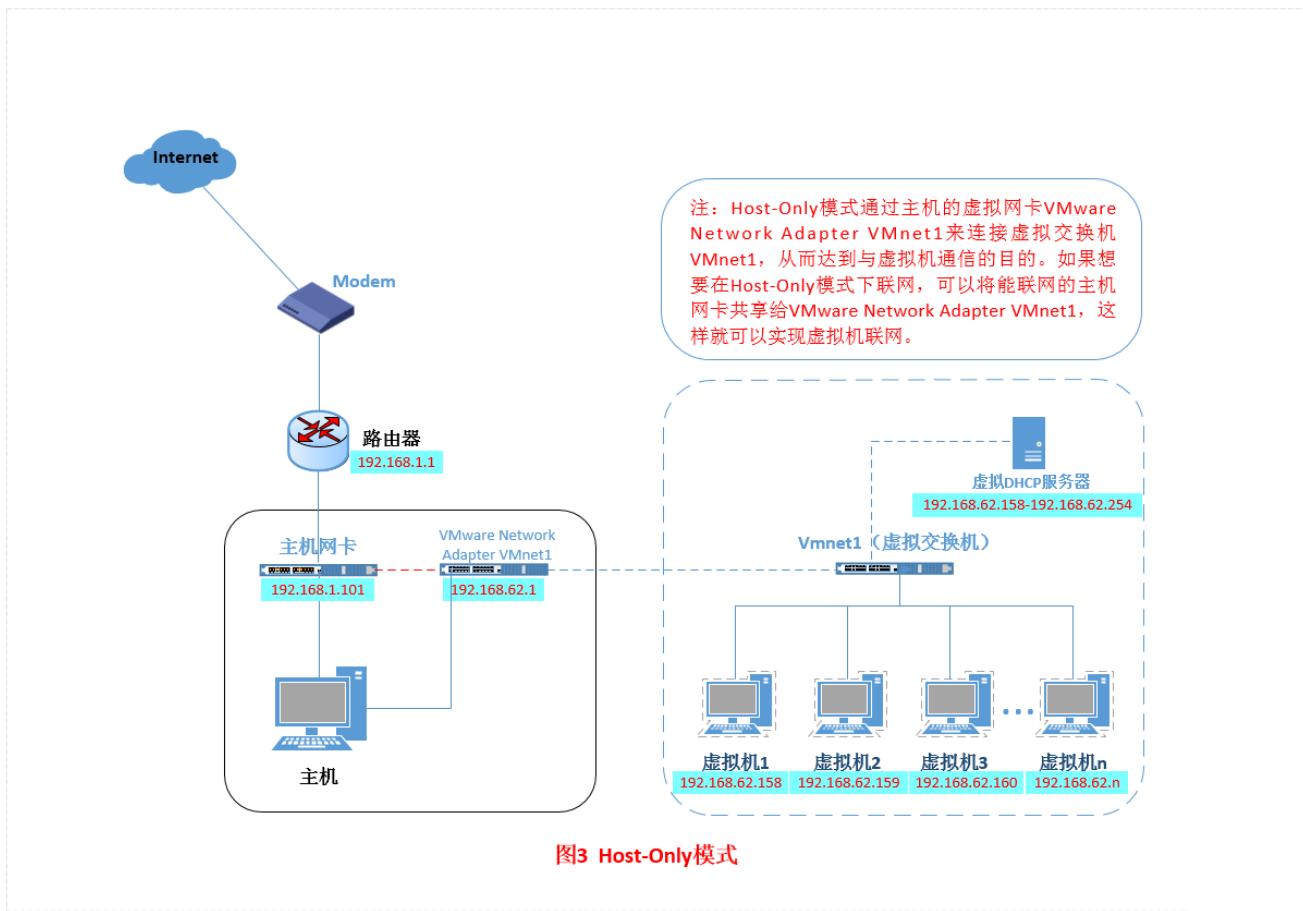
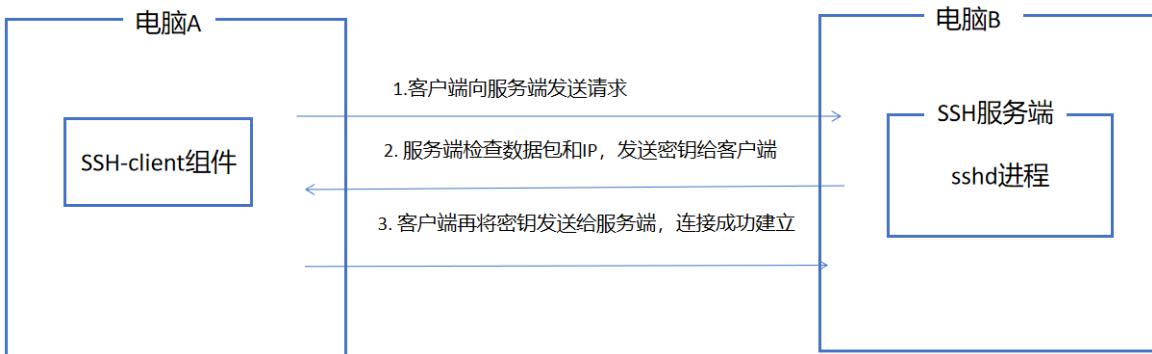


图3 Host-Only模式

1.6 远程工具的使用

1.6.1 SSH协议介绍

- 1 SSH 为 Secure Shell 的缩写, 由 IETF 的网络小组 (Network Working Group) 所制定
- 2 SSH 为建立在应用层基础上的安全协议。
- 3 SSH 专为远程登录会话和其他网络服务提供安全性的协议, 可以有效防止远程管理过程中的信息泄露问题。
- 4 SSH 由客户端和服务端两部分组成。服务端是一个sshd进程, 提供密钥认证、密钥交换等处理。客户端包含 SSH程序以及scp(远程拷贝)、slogin(远程登陆)、sftp(安全文件传输)

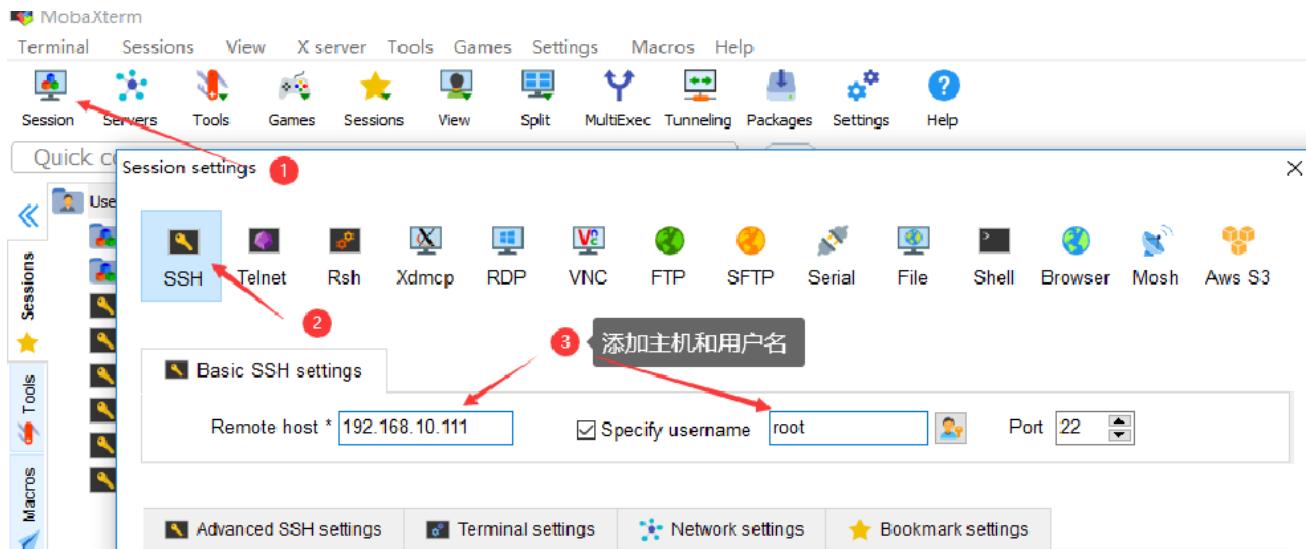


1.6.2 远程工具的使用

常用的连接工具有putty、MobaXterm、SmarTTY、SecureCRT、XShell、finalshell等

我们选择MobaXterm，进行安装和使用

安装包：MobaXterm_Installer_v12.4.zip 下一步下一步即可
使用：启动MobaXterm后，参考下图得1, 2, 3步，进行填写。



第二章 常用命令

2.1 命令格式的说明

命令格式：命令 [-选项][参数]

eg: [root@qianfeng01 ~]# ls -la /usr

说明：

大部分命令遵从该格式

多个选项时，可以一起写

eg: [root@qianfeng01 ~]# ls -l -a ls -la

简化选项与完整选项(注：并非所有选项都可使用完整选项)

eg: [root@qianfeng01 ~]# ls -all ls -a

帮助命令：（相当于命令说明书）

2.2 帮助命令

2.2.1 man

英文：manual 命令路径：/usr/bin/man 执行权限：所有用户

作用：获取命令或配置文件的帮助信息

语法：man [命令/配置文件]

eg:[root@qianfeng01 ~]# man ls man services

(查看配置文件时，不需要配置文件的绝对路径，只需要文件名即可)

调用的是more命令来浏览帮助文档，按空格翻下一页，按回车翻下一行，按q退出。

2.2.2 help

help 查看shell内置命令的帮助信息

eg: [root@qianfeng01 ~]#help cd

命令名 --help

命令名 --help 列举该命令的常用选项

eg: [root@qianfeng01 ~]#cp --help

2.3 文件处理命令

2.3.1 cd

英文: change directory 命令路径: 内部命令 执行权限: 所有用户

cd : 切换工作目录

cd -: 回到上一次的操作所在位置

cd ~: 回到用户家目录

cd ./ : 当前目录

cd ../ : 回到上一级目录

cd : 回到用户家目录

2.3.2 ls

查看目录 ls + 选项 + 目录名

英文: list 命令路径: /bin/ls 执行权限: 所有用户

ll 等价于ls -l

-l 列表形式显示

```
eg: [root@qianfeng01 ~]# ls -l /bin 查看根目录下的bin目录下的所有文件或目录以列表的形式查看
```

-a 所有文件或目录包括隐藏文件

```
eg: [root@qianfeng01 ~]# ll -a /bin 查看根目录下的bin目录下的所有文件或目录包含隐藏文件
```

-h 以可读的方式显示文件大小,配合-l使用

```
eg: [root@qianfeng01 ~]# ll -h /bin 查看根目录下的bin目录下的所有文件或目录 (显示文件大小单位, 如KB等)
```

2.3.3 pwd

英文: print working directory 命令路径: /bin/pwd 执行权限: 所有用户

作用: 显示当前工作目录

linux下的路径

pwd: 打印当前工作目录 (全目录)

绝对路径: 以根目录为开始的目录

相对路径: 相对当前目录的路径

语法: pwd [-P]

```
eg: [root@qianfeng01 ~]# cd /etc/init.d
```

```
pwd [-P]
```

2.3.4 mkdir

英文: `make directories` 命令路径: `/bin/mkdir` 执行权限: 所有用户

作用: 创建新目录

语法: `mkdir [-p] 目录名`

`-p` 父目录不存在情况下先生成父目录 (parents)

eg: [root@qianfeng01 ~]# `mkdir linux/test` 如果目录linux不存在, 则报错, 使用参数-p即可自动创建父目录

2.3.5 touch

命令路径: `/bin/touch` 执行权限: 所有用户

作用: 创建空文件或更新已存在文件的时间

语法: `touch 文件名`

eg: [root@qianfeng01 ~]# `touch a.txt b.txt` `touch {a.txt,b.txt}` 同时创建多个文件, 注意: 后面{}中, 中间不能有空格

eg: [root@qianfeng01 ~]# `touch "program files"` 创建带空格的文件

注意: 生产环境中, 文件名, 一定不要加空格

2.3.6 echo

命令路径：/bin/echo 执行权限：所有用户

作用：查看某些环境变量/给文件增加内容（将文件创建出来）

语法：echo '字符串' > 文件名

eg：

```
[root@qianfeng01 ~]# echo $PWD      查看当前的路径
```

```
[root@qianfeng01 ~]# echo 'haha' > test1.txt      将字符串haha添加到test1.txt  
里面，注意；会将文件原来的内容覆盖
```

```
[root@qianfeng01 ~]# echo 'haha' >> test1.txt      将字符串haha追加到test1.txt  
中，不会覆盖原来的内容
```

2.3.7 cp

英文：copy 命令路径：/bin/cp 执行权限：所有用户

作用：复制文件或目录

语法：cp [-rp] 源文件或目录 目的目录

-r -R recursive 递归处理，复制目录

-p 保留文件属性（原文件的时间不变）

eg：

1. 相对路径 [root@qianfeng01 ~]# cp -R /etc/* . 将etc下的全部内容递归拷贝到当前的目录下

2. 绝对路径 [root@qianfeng01 ~]# cp -R /*ect/service /root/test/aa/bb

2.3.8 mv

英文: move 命令路径: /bin/mv 执行权限: 所有用户

作用: 移动文件或目录、文件或目录改名

语法: mv 源文件或目录 目的目录

2.3.9 rm

英文: remove 命令路径: /bin/rm 执行权限: 所有用户

作用: 删除文件

语法: rm [-rf] 文件或目录

-r (recursive) 删除目录, 同时删除该目录下的所有文件

-f (force) 强制删除文件或目录 即使原档案属性设为唯读, 亦直接删除, 无需逐一确认

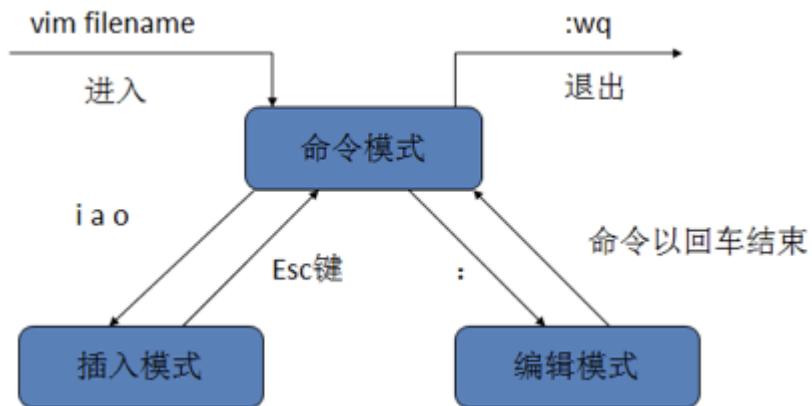
注意: 工作中, 谨慎使用rm -rf 命令。

2.3.10 vim编辑器

vim/vi是Unix / Linux上最常用的文本编辑器而且功能非常强大。

只有命令, 没有菜单。

VIM工作模式



命令模式：又称一般模式

编辑模式：又称底行模式，命令行模式

2.3.10.1 插入命令

命令	作用
a	在光标后附加文本
A (shift + a)	在本行行末附加文本 行尾
i	在光标前插入文本
I(shift+i)	在本行开始插入文本 行首
o	在光标下插入新行
O(shift+o)	在光标上插入新行

2.3.10.2 定位命令

命令	作用
:set nu	设置行号
:set nonu	取消行号
gg G	到第一行 到最后一行
nG	到第n行
:n	到第n行

2.3.10.3 保存和退出命令

命令	作用
:w	保存修改
:w new_filename	另存为指定文件,如果文件不存在,会直接创建一个新的文件
:w >> a.txt	内容追加到a.txt文件中 文件需存在
:wq	保存修改并退出
shift+zz (ZZ)	快捷键, 保存修改并退出
:q!	不保存修改退出
:wq!	保存修改并退出 (文件所有者可忽略文件的只读属性)

eg:

- 1 不保存并退出 :q
- 2 有修改, 但是修改后的内容是不保存的 :q!
- 3 有突发情况, 导致窗口退出。修改文件之后, 直接删除同名的.swp文件

2.3.10.4 删除命令

命令	作用
x	删除光标所在处字符 nx 删除光标所在处后n个字符
dd	删除光标所在行, ndd删除n行
:n1,n2d	删除指定范围的行 (eg :1,3d 删除了123这三行)
dG	删除光标所在行到末尾的内容
D	删除从光标所在处到行尾

2.3.10.5 复制和剪切命令

命令	作用
yy、Y	复制当前行
nyy、nY	复制当前行以下n行
dd	剪切当前行
ndd	剪切当前行以下n行
p、P	粘贴在当前光标所在行下 或行上

2.3.10.6 替换和取消命令

命令	作用
r	取代光标所在处字符
R(shift + r)	从光标所在处开始替换字符, 按Esc结束
u	undo,取消上一步操作
ctrl+r	redo,返回到undo之前

2.3.10.7 搜索和替换命令

命令	作用
/string	向后搜索指定字符串
?string	向前搜索指定字符串
n	搜索字符串的下一个出现位置,与搜索顺序相同
N (Shift + n)	搜索字符串的上一个出现位置,与搜索顺序相反
:%s/old/new/g	全文替换指定字符串
:n1,n2s/old/new/g	在一定范围内替换指定字符串

注意:当遇到特殊字符时,需要进行转义处理:比如 . 写成 \. * 写成 * ^ 写成 \^

% 指全文, s 指开始, g 指结束

起始行, 终止行s/要替换的字符串/替换的新的字符串/g 将/g换成/c 询问确认

eg: : %s/ftp/yang/g 全局替换, 把ftp替换为yang

eg: : 41,44.yang/lee/c 从41行到44行, 把yang替换为lee, 询问是否替换

eg: : 41,44.yang/lee/g 同上, 不询问, 直接替换

eg: : %s/\root/\ROOT/g 把/root替换为/ROOT, 遇到特殊字符需要进行转义处理

eg: : %s#/bin/bash#/bin/ksh/g 把/bin/bash全部替换为/bin/ksh

2.3.10.8 使用替换命令添删注释

```
:% s/^/#/g 来在全部内容的行首添加 # 号注释      ^代表行首
```

```
:1,10 s/^/#/g 在1~10 行首添加 # 号注释
```

vi里面查命令

```
: ! which cp
```

vi里面导入命令的结果

```
: r !which cp
```

I或者O进入插入模式。

2.3.11 ln

1 英文： link 命令路径： /bin/ln 执行权限：所有用户

2 作用：产生链接文件

3 语法：

```
ln -s [源文件][目标文件]    创建软链接    相当于建立快捷方式    注意：源文件 使用 绝对路径
```

```
ln [源文件][目标文件]    创建硬链接    相当于深拷贝
```

```
eg: [root@qianfeng01 ~]# ln -s /etc/service ./service.soft
```

创建文件/etc/service的软链接service.soft

```
eg: [root@qianfeng01 ~]# ln /etc/service /service.hard
```

创建文件/etc/service的硬链接/service.hard

4 软连接类似于windows下的快捷方式

```
软连接文件格式： lrwxrwxrwx. 1 root root 13 Jul 20 07:50 service -> /etc/services
```

5 格式解析（特征）：

- 1 1代表硬链接数量，如果该文件没有硬链接，就只有本身一个硬链接。
 - 2 13代表 链接文件的长度
 - 3 软连接的文件类型是 l (软连接)，软连接文件的权限 都是 lrwxrwxrwx
 - 4 -> 箭头指向到源文件
- 真正的权限取决于对源文件的权限
- 时间值为创建软连接的时间
- 软连接可以跨文件系统生成

6 硬链接特征

- 1 相当于 cp -p +同步更新
- 2 通过i节点识别，与源文件有相同的inode节点
当修改其中的任意一个硬链接文件时，其他的文件内容也会同步更新
- 3 硬链接不能跨分区， [root@qianfeng01 ~]# ln /home/test/issues /boot/test (错误)
- 4 不能针对目录使用 [root@qianfeng01 ~]# ln /tmp/ aa.hard (无法将目录/tmp 生成硬链接)

7 删除软连接,源文件还在

```
[root@qianfeng01 ~]# rm -rf symbolic_name
```

8 inode说明

Linux文件系统中的inode主要用于存放档案和目录的基本信息，当系统创建文件系统的同时会创建大量的inode

2.4 查看命令

2.4.1 cat

英文: concatenate

命令路径: /bin/cat

执行权限: 所有用户

作用: 显示文件内容, 直接显示全部内容

语法: cat [-n][文件名]

-A 显示所有内容, 包括隐藏的字符

-n 显示行号

eg: [root@qianfeng01 ~]# cat /etc/services

2.4.2 more

命令路径: /bin/more

执行权限: 所有用户

作用: 分页显示文件内容

语法: more [文件名]

空格或f 显示下一页

enter键 显示下一行

q或Q 退出

相关指令:less

2.4.3 head

命令路径: /usr/bin/head

执行权限: 所有用户

作用: 查看文件前几行 (默认10行)

语法: head [文件名]

-n 指定行数

```
eg: [root@qianfeng01 ~]#  
head -20 /etc/services  
head -n 3 /etc/services
```

2.4.4 tail

命令路径: /usr/bin/tail 执行权限: 所有用户

作用: 查看文件的后几行

语法: tail [文件名]

-n 指定行数

获取一个大文件的部分文件, 可使用head或tail命令

eg:

```
[root@qianfeng01 ~]# head -n 100 /etc/services >config.log      会直接用  
/etc/services里面前100行的数据导入config.log文件中,并覆盖其中的内容
```

```
[root@qianfeng01 ~]# head -n 100 /etc/services >>config.log      会直接用  
/etc/services里面前100行的数据导入config.log文件中,并进行内容的追加
```

2.5 搜索查找命令

2.5.1 find

命令路径：/bin/find

执行权限：所有用户

作用：查找文件或目录

语法：find [搜索路径][匹配条件]

如果没有指定搜索路径，默认从当前目录查找

find命令选项

1) -name按名称区分大小写查找

说明：精准查找

eg: [root@qianfeng01 ~]# find /etc -name "init" 在目录/etc中查找文件init

2) -iname按名称不区分大小写查找

说明：不区分大小写

eg: [root@qianfeng01 ~]# find /etc -iname "init" 在目录/etc中查找文件init

模糊查询：

find查找字符匹配：

*：匹配所有

?：匹配单个字符

eg: [root@qianfeng01 ~]# find /etc -name "init???" 在目录/etc中查找以init开头的，且后面有三位的文件

模糊匹配的条件，建议使用单引号或双引号括起来。

如果*被转义，可使用单双引号括住查询条件，或者使用*。

eg: [root@qianfeng01 ~]# find -name "*g"

3) -size按文件大小查找

以block为单位，一个block是512B，1K=2block +大于 -小于 不写是等于

eg: [root@qianfeng01 ~]# find /etc -size -204800 在etc目录下找出小于100MB的文件

100MB=102400KB=204800block

4) -type按文件类型查找

f 二进制文件 l 软连接文件 d 目录 c 字符文件

eg:[root@qianfeng01 ~]# find /dev -type c

find查找的基本原则：

占用最少的系统资源，即查询范围最小，查询条件最精准

注意：如果明确知道查找的文件在哪一个目录，就直接对指定目录查找，不查找根目录/

2.5.2 grep

命令路径：/bin/grep 执行权限：所有用户

作用：在文件中搜寻字符串匹配的行并输出

语法：grep [-cinv] '搜寻字符串' filename

选项与参数：

-c : 输出匹配行的次数（是以行为单位，不是以出现次数为单位）

-i : 忽略大小写，所以大小写视为相同

-n : 显示匹配行及行号

-v : 反向选择，显示不包含匹配文本的所有行。

eg: [root@qianfeng01 ~]# grep ftp /etc/services

```
eg: [root@qianfeng01 ~]# grep -v ^# /etc/inittab          去掉文件行首的#号
```

```
eg: [root@qianfeng01 ~]# grep -n "init"/etc/inittab  显示在inittab文件中, init匹  
配行及行号
```

```
eg: [root@qianfeng01 ~]# grep -c"init"/etc/inittab  显示在inittab文件中, init匹  
配了多少次
```

管道的功能(过滤)

```
eg: [root@qianfeng01 ~]# find /root | grep a.txt
```

2.5.3 which

命令路径: /usr/bin/which 执行权限: 所有用户

作用: 显示系统命令所在目录 (绝对路径及别名)

which命令的作用是, 在PATH变量指定的路径中, 搜索某个系统命令的位置, 并且返回第一个搜索结果。也就是说, 使用which命令, 就可以看到某个系统命令是否存在, 以及执行的到底是哪一个位置的命令

```
eg: [root@qianfeng01 ~]# which ls      出现下面的信息, 是ls指令的详情, 说明ls指令是存  
在的
```

```
alias ls='ls --color=auto'
```

```
/bin/ls
```

```
eg: [root@qianfeng01 ~]# which zs      出现下面的信息, 说明zs指令是不存在的
```

```
/usr/bin/which: no zs in  
(/usr/local/bin:/usr/bin:/bin:/usr/local/sbin:/usr/sbin:/sbin:/home/ch/bin)
```

2.5.4 whereis

命令路径： /usr/bin/whereis

执行权限： 所有用户

作用： 搜索命令所在目录 配置文件所在目录 及帮助文档路径

eg： [root@qianfeng01 ~]# which passwd 和 [root@qianfeng01 ~]#whereis passwd

eg： 查看/etc/passwd配置文件的帮助，就用 man 5 passwd

2.6 磁盘的操作命令

2.6.1 df命令

作用： 用于查看Linux文件系统的状态信息，显示各个分区的容量、已使用量、未使用量及挂载点等信息。看剩余空间

语法： df [-hkm][挂载点]

-h (human-readable) 根据磁盘空间和使用情况 以易读的方式显示 KB, MB, GB等

-k 以KB 为单位显示各分区的信息， 默认

-m 以MB为单位显示信息

-a 显示所有分区包括大小为0 的分区

2.6.2 du命令

作用： 用于查看文件或目录的大小（磁盘使用空间）

语法： du [-ahs][文件名目录]

-a 显示子文件的大小

-h 以易读的方式显示 KB, MB, GB等

-s summarize 统计总占有量

```
eg: [root@qianfeng01 ~]# du -a(all) /home 显示/home 目录下每个子文件的大小,默认单位为kb
```

```
eg: [root@qianfeng01 ~]# du -h /home 以K, M,G为单位显示/home 文件夹下各个子目录的大小
```

```
eg: [root@qianfeng01 ~]# du -sh /home 以常用单位 (K,M,G) 为单位显示/home 目录的总大小 -s summarize
```

df命令和du命令的区别：

df命令是从文件系统考虑的，不仅考虑文件占用的空间，还要统计被命令或者程序占用的空间。

du命令面向文件，只计算文件或目录占用的空间。

eg：执行下面的命令查看区别

```
[root@qianfeng01 ~]# df -h /
```

```
[root@qianfeng01 ~]# du -sh /
```

2.6.3 free

作用：显示系统内存的使用情况，包括物理内存、交换内存(swap)和内核缓冲区内存。

语法： free [-kmg]

选项：

-k： 以KB为单位显示，默认就是以KB为单位显示

-m： 以MB为单位显示

-g： 以GB为单位显示

清理缓存命令：

```
eg: [root@qianfeng01 ~]# echo 1 > /proc/sys/vm/drop_caches
```

2.7 打包压缩命令

2.7.1 gzip

英文: GNU zip 命令路径: /bin/gzip

执行权限: 所有用户

作用: 压缩(解压)文件, 压缩文件后缀为.gz

gzip 只能压缩文件, 不能压缩目录; 不保留原文件

语法: gzip 文件

-d 将压缩文件解压 (decompress)

解压使用gzip -d或者 gunzip

2.7.2 bzip2

命令路径: /usr/bin/bzip2

执行权限: 所有用户

作用: 压缩(解压)文件, 压缩文件后缀为.bz2

语法: bzip2 [-k][文件]

注意: 在纯命令行系统中没有默认安装bzip2命令, 需要手动安装 yum -y install bzip2.x86_64

-k: 产生压缩文件后保留原文件 (压缩比高)

-d: 解压缩的参数 (decompress)

解压使用bzip2 -d或者 bunzip2

2.7.3 zip

命令路径: /usr/bin/zip

执行权限: 所有用户

作用: 压缩(解压)文件, 压缩文件后缀为.zip, 保留源文件

语法: zip 选项 [压缩后文件名称] [文件或目录]

注意: 在纯命令行系统中没有默认安装zip命令,需要手动安装 yum -y install zip

-r 压缩目录

eg:

```
[root@qianfeng01 ~]# zip services.zip /etc/services 压缩文件
```

```
[root@qianfeng01 ~]# zip -r test.zip /test 压缩目录
```

如果不加-r选项, 压缩后的文件没有数据。

解压使用unzip ,注意如果解压时,已经存在与压缩文件同名的目录名,默认不会进行覆盖,如果选择Y,会进行覆盖

2.7.4 tar

命令路径: /bin/tar 执行权限: 所有用户

作用: 文件、目录打(解)包

语法: tar [-zcf] 压缩后文件名 文件或目录

-c 建立一个压缩文件的参数指令 (create) , 后缀是.tar

-x 解开一个压缩文件的参数指令 (extract)

-z 以gzip命令压缩/解压缩

-j 以bzip2命令压缩/解压缩

-v 压缩的过程中显示文件 (verbose)

-f file 指定文件名,必选项

2.7.5 示例

1) 简单理解

```
[root@qianfeng01 ~]# tar -cf    tar -xf  单独的打包 , 解包  
  
[root@qianfeng01 ~]# gzip    bzip2      打包之后, 进行压缩  
  
[root@qianfeng01 ~]# tar -zcvf  
  
[root@qianfeng01 ~]# tar -zxvf 一步到位  
  
[root@qianfeng01 ~]# tar -z 以gzip打包目录并压缩 文件格式.tar.gz (.tgz)  
  
[root@qianfeng01 ~]# tar -j 以bzip2打包目录并压缩 文件格式.tar.bz2
```

2) 示例

```
eg: [root@qianfeng01 ~]# tar -zcvf dir1.tar.gz  dir1  使用gzip将目录dir1压缩成  
一个打包并压缩文件dir1.tar.gz  
  
eg: [root@qianfeng01 ~]# tar -zxvf  dir1.tar.gz -C(大写的) /root      解压到指定  
的/root目录  
  
eg: [root@qianfeng01 ~]# tar -cvf bak.tar .      将当前目录的文件打包为bak.tar  
  
eg: [root@qianfeng01 ~]# tar -xvPf bak.tar      解压到当前目录, 默认是相对路径, P意思  
是支持绝对路径  
  
eg: [root@qianfeng01 ~]# tar -cvPf bak.tar /a/b    将/a/b路径压缩成bak.tar  
  
eg: [root@qianfeng01 ~]# tar -zcvf bak.tar.gz  bak.tar 或 [root@qianfeng01  
~]# gzip  bak.tar    使用gzip将打包文件bak.tar压缩为bak.tar.gz  
  
eg: [root@qianfeng01 ~]# tar -jcvf bak.tar.bz2 bak.tar 或 [root@qianfeng01  
~]# bzip2 bak.tar    使用bzip2将打包文件bak.tar压缩为bak.tar.bz2  
  
eg: [root@qianfeng01 ~]# tar -rvf bak.tar /etc/password    将/etc/password追加  
文件到bak.tar中  
  
eg: [root@qianfeng01 ~]# tar -cJvf test.tar.bz2 test  生成test.tar.bz2的压缩文
```

件

```
eg: [root@qianfeng01 ~]# tar -xjf test.tar.bz2      解压
```

3) 最常用: tar + gzip

```
[root@qianfeng01 ~]# tar -zcvf 压缩
```

```
[root@qianfeng01 ~]# tar -zxvf  解压
```

补充: 源文件是保留的, 不会被删除

2.8 date

1 date命令是显示或设置系统时间与日期。

2 查看系统时间直接使用 date

3 以自定义的方式显示系统时间: date +'%Y-%m-%d %H:%M:%S'

注意:+与字符串之间不能有空格,与date之间要有空格

4 设置时间(注意:最好不要随意的设置,特别是不要随意的同步bios,造成系统的时间混乱)

```
eg: [root@qianfeng01 ~]# date -s "2015-5-8 19:48:00"
```

5 同步到bios, 重启之后才能继续生效

```
eg: [root@qianfeng01 ~]# hwclock -w
```

2.9 系统关机命令

2.9.1 关机命令: shutdown

语法: shutdown [选项] 时间

选项:

-c: 取消前一个关机命令

-h: 关机

-r: 重启

eg:

```
[root@qianfeng01 ~]# shutdown -h now 立即关机
```

```
[root@qianfeng01 ~]# shutdown -h 20:30 定时关机
```

其他关机命令：

halt

poweroff

init 0

2.9.2 重启命令：reboot

eg: reboot 重启系统

```
[root@qianfeng01 ~]# reboot -h now 立即重启
```

2.9.3 重启命令：init 6

注意：生产环境中，关机命令和重启命令谨慎执行。

2.9.4 执行等级

linux有七种执行等级：

init 0: 停机

init 1: 单用户形式，只root进行维护

init 2: 多用户，不能使用net file system

init 3: 完全多用户

init 5: 图形化

init 4: 安全模式

init 6: 重启

2.9.5 其他命令

```
uname -r          查看系统版本信息  
[root@qianfeng01 ~]# basename      a/b/c/test.txt          显示:test.txt  
查看命令中的名字  
[root@qianfeng01 ~]# dirname      a/b/c/test.txt          显示: a/b/c/    查看命  
令中的路径
```

2.10 Linux的快捷键命令

善于查看man help等帮助文档

利用好Tab键 自动补全

掌握好一些快捷键

ctrl + c (停止当前进程)

ctrl + z 挂起当前进程，放后台

ctrl + r (查看命令历史) history

ctrl + l (清屏，与clear命令作用相同)

对当前命令行的操作：

方向箭头 上 下 可以查看执行过的命令并再次使用

ctrl + a 行首 ctrl + e 行尾 ctrl+d 清除当前的字符

ctrl + k 清除光标后面的内容 ctrl+ w 清除光标前面的单词

小技巧：linux 不进去系统更改root密码

- 1.开机时按'e'键进入grub
- 2.在grub选项菜单按e进入编辑模式
- 3.编辑kernel那行 输入“1”（空格1），然后按enter键。
- 4.按b重启
- 5.进入后执行下列命令
passwd root
passwd root (配置root的密码)→Enter new unix password:输入新的密码→init 6

2.11 网络相关操作

2.11.1 修改ip地址

说明: (以网络方式为NAT示例)

1 图形化界面

2 使用命令行修改配置文件

```
[root@qianfeng01 ~]# vi /etc/sysconfig/network-scripts/ifcfg-ens33

BOOTPROTO=static          static 静态ip  dhcp 动态ip  none 不设置,默认
静态ip

HWADDR=00:0C:29:3D:B0:9B  电脑的mac地址,建议去掉

UUID=6f89eb5e-bff4-4f6b-a045-b7722bd188fc  电脑的UUID  建议去掉

ONBOOT=yes                是否开机自动使用

nm_controlled=yes         设置network manager的参数,实时生效,修改后无需要重启网卡立即生
效。建议删掉,在CentOS7的命令行版(minimal)中没有这个设置

IPADDR=192.168.10.111

NETMASK=255.255.255.0    或者  PREFIX=24

GATEWAY=192.168.10.2     #网段2任意, IP地址2固定, 网段为vmnet8的设置的IP网段

DNS1=114.114.114.114     国内的DNS

DNS2=8.8.8.8   谷歌的DNS

defroute=yes  默认的路由  建议删掉

IPV4_FAILURE_FATAL=yes  如果为yes, 则ipv4配置失败禁用设备  建议删掉
```

3 重启网络生效: service network restart 或者 systemctl restart network

2.11.2 关闭防火墙

1 及时生效,重启后复原

```
eg: 关闭:[root@qianfeng01 ~]# systemctl stop firewalld
eg: 开启:[root@qianfeng01 ~]# systemctl start firewalld
eg: 查看状态:[root@qianfeng01 ~]# systemctl status firewalld
```

2 非及时性生效,重启后永久性生效

```
eg: 关闭:[root@qianfeng01 ~]# systemctl disable firewalld  
eg: 开启:[root@qianfeng01 ~]# systemctl enable firewalld  
eg: 查看状态[root@qianfeng01 ~]#: systemctl status firewalld
```

2.11.3 关闭NetworkManager

NetworkManager服务是管理和监控网络设置的守护进程，CentOS7更加注重使用NetworkManager服务来实现网络的配置和管理它是一个动态的，事件驱动的网络管理服务。

常用管理命令：

```
[root@qianfeng01 ~]# systemctl status NetworkManager  
[root@qianfeng01 ~]# systemctl start NetworkManager  
[root@qianfeng01 ~]# systemctl stop NetworkManager  
[root@qianfeng01 ~]# systemctl disable NetworkManager  
[root@qianfeng01 ~]# systemctl enable NetworkManager
```

2.11.4 网络通信命令

2.11.4.1 ping

命令路径：/bin/ping 执行权限：所有用户

作用：测试网络的连通性

语法：ping 选项 IP地址

-c 指定发送次数

ping 命令使用的是icmp协议，不占用端口

eg： [root@qianfeng01 ~]#ping -c 3 127.0.0.1

2.11.4.2 ip addr

英文：interface configure 命令路径：/sbin/ifconfig

执行权限：root

作用：查看和设置网卡网络配置

语法：ifconfig [-a][网卡设备标识]

-a： 显示所有网卡信息

ifconfig 网卡名字 查看单个的网卡信息

2.11.4.3 netstat

英文: network statistics 命令路径: /bin/netstat 执行权限: 所有用户

注意: 如果在系统中默认没有这个命令: 直接安装net-tools包即可

作用: 主要用于检测主机的网络配置和状况

-a (all)显示所有连接和监听端口

-t (tcp)仅显示tcp相关选项

-u (udp)仅显示udp相关选项

-n 使用数字方式显示地址和端口号

-p 显示socket的PID和进程的名字

-l (listening)显示监控中的服务器的socket

eg: [root@qianfeng01 ~]# netstat -tlnu 查看本机监听的端口

eg: [root@qianfeng01 ~]# netstat -tlnu | grep httpd

协议 待收数据包 待发送数据包 本地ip地址: 端口 远程IP地址: 端口

netstat -antpl

eg: [root@qianfeng01 ~]# netstat -antpl | grep httpd 查看某一个程序的端口信息

eg: [root@qianfeng01 ~]# netstat -atnp | grep httpd

eg: [root@qianfeng01 ~]# netstat -atnp | grep 80 查看端口

2.11.5 修改主机名

临时生效:

hostname 主机名

```
[root@localhost ~]# hostname  
localhost.localdomain  
[root@localhost ~]#  
[root@localhost ~]# hostname node1.edu  
[root@localhost ~]# hostname  
node1.edu
```

注意：通过调用logout先退出，再登录—写入内存

永久生效：修改配置文件

```
[root@qianfeng01 ~]# hostnamectl set-hostname qianfeng01
```

2.11.6 修改映射关系

修改主机名和ip地址之间的映射关系

执行 [root@qianfeng01 ~]# vim /etc/hosts 进入hosts文件,加入代码

主机ip 主机名

192.168.10.111 qianfeng01

192.168.10.112 qianfeng02

192.168.10.113 qianfeng03

最好让主机名字,虚拟机名字,网络连接名字保持一致

2.12 进程管理命令

2.12.1 进程和程序的区别

1 程序是静态概念，本身作为一种软件资源长期保存；而进程是程序的执行过程，它是动态概念，有一定的生命期，是动态产生和消亡的。

2 程序和进程一一对应关系。一个进程在活动中可有顺序地执行若干个程序。

2.12.2 父进程与子进程

- 1 子进程是由一个进程所产生的进程，产生这个子进程的进程称为父进程。
- 2 在Linux系统中，使用系统调用fork创建进程。fork复制的内容包括父进程的数据和堆栈段以及父进程的进程环境。
- 3 父进程终止子进程自然终止。

2.12.3 进程和线程的区别

进程：就是正在执行的程序或命令，每一个进程都是一个运行的实体，都有自己的地址空间，并占用一定的系统资源。

线程：轻量级的进程；进程有独立的地址空间，线程没有；线程不能独立存在，它由进程创建；相对讲，线程耗费的cpu和内存要小于进程。

进程管理的作用：判断服务器的健康状态；查看系统所有的进程；杀死进程。

2.12.4 前台进程和后台进程

前台进程：

在Shell提示处打入命令后，创建一个子进程，运行命令，Shell等待命令退出，然后返回到对用户给出提示符。这条命令与Shell异步运行，即在前台运行，用户在它完成之前不能执行另一个命令。

后台进程：

在Shell提示处打入命令，若后随一个&，Shell创建的子进程运行此命令，但不等待命令退出，而直接返回到对

用户给出提示。这条命令与Shell同步运行，即在后台运行。后台进程必须是非交互式的。

2.12.5 ps命令

作用：查看系统中的进程信息

语法：ps [-auxle]

常用选项

a：显示所有用户的进程

u：显示用户名和启动时间

x：显示没有控制终端的进程

e：显示所有进程，包括没有控制终端的进程

l：长格式显示

2.12.6 查看系统中所有进程

```
ps aux      #查看系统中所有进程，使用BSD操作系统格式， unix
```

```
ps -le      #查看系统中所有进程，使用Linux标准命令格式
```

```
eg: [root@qianfeng01 ~]# ps -u or ps -l    查看隶属于自己进程详细信息
```

```
eg: [root@qianfeng01 ~]# ps aux | grep sam    查看用户sam执行的进程
```

```
eg: [root@qianfeng01 ~]# ps -ef | grep init      查看指定进程信息
```

2.12.7 pstree

需要安装软件包:psmisc.x86_64 0:22.20-16.el7

作用：查看当前进程树

语法：pstree [选项]

-p 显示进程PID

-u 显示进程的所属用户

2.12.8 top

作用：查看系统健康状态

显示当前系统中耗费资源最多的进程，以及系统的一些负载情况。

语法：top [选项]

-d 秒数，指定几秒刷新一次，默认3秒（动态显示）

2.12.9 kill

作用：关闭进程

语法：kill [-选项] pId

eg: [root@qianfeng01 ~]# kill -9 进程号 (强行关闭) 常用

eg: [root@qianfeng01 ~]# kill -1 进程号 (重启进程)

eg: [root@qianfeng01 ~]# killall -1 关闭所有进程(忽略进程名的大小写)

2.12.10 w

作用：查看用户信息

语法: w 用户名

eg: [root@qianfeng01 ~]# w root

显示:

USER	TTY	FROM	LOGIN@	IDLE	JCPU	PCPU	WHAT
root	pts/0	10.0.158.3	06:02	0.00s	0.13s	0.00s	w root

解释:

JCPU: 以终端代号来区分, 该终端所有相关的进程执行时, 所消耗的CPU时间会显示在这里

PCPU: CPU执行程序耗费的时间

WHAT: 用户正在执行的操作

2.12.11 nohup

作用: 使进程在用户退出登陆后仍旧继续执行, nohup命令将执行后的数据信息和错误信息默认储存到文件nohup.out中

语法: nohup program & &在后台运行的意思

eg: [root@qianfeng01 ~]# nohup ping www.baidu.com &

使用: [root@qianfeng01 ~]# ps -ef | grep ping 查看当前的ping进程

2.12.12 uptime

作用：查看负载

例如：使用uptime确定是服务器还是网络出了问题。如果网络应用程序运行，运行uptime来了解系统负载是否很高。如果负载不高，这个问题很有可能是由于网络引起的而非服务器。

语法： uptime

eg：执行： [root@qianfeng01 ~]# uptime

显示： 06:32:04 up 19:57, 4 users, load average: 0.00, 0.01, 0.00

解释： 系统时间 用户数量 服务器在过去的1分钟、5分钟、15分钟的系统平均负载值

2.13 用户与组的创建和删除

2.13.1 su

语法： su 用户名

作用： 切换用户

eg： [root@qianfeng01 ~]# su root 切换回root用户,注意:如果是root用户,每次都会要求输入密码.普通的用户可以直接切换

2.13.2 useradd

添加用户

语法： useradd [选项] 用户名

2.13.3 passwd

修改密码命令

语法：passwd [选项] [用户名]

用户密码：生产环境中，用户密码长度8位以上，设置大小写加数字加特殊字符，要定期更换密码。

ys^h_L9t

2.13.4 userdel

删除用户

-r 删除账号时同时删除宿主目录 (remove)

注意：一个用户已经打开，需要删除

做法：先跳到当前用户下面，执行exit 或 exit 用户名。这时会自动跳回root界面。

如果还是无法删除，可以将系统重启

2.13.5 groupadd

用于添加组

-g 指定gid

2.13.6 groupmod

用于修改组

-n 更改组名 (new group)

```
[root@qianfeng01 ~]# groupmod -n new_gname old_gname
```

2.13.7 groupdel

用于删除组

如果要删除的组归属于某一个用户的所属组，则不能删除该组

```
[root@qianfeng01 ~]# useradd -g hadoop hdfs
```

```
[root@qianfeng01 ~]# groupdel hadoop
```

2.14 文件权限命令

2.14.1 三种基本权限

r 读权限 (read)

w 写权限 (write)

x 执行权限 (execute)

2.14.2 权限说明



所有者 所属组 其他人

第1位：文件类型 (d 目录, - 普通文件, l 链接文件)

第2-4位：所属用户(所有者)权限，用u (user) 表示

第5-7位：所属组权限，用g (group) 表示

第8-10位：其他用户（其他人）权限，用o (other) 表示

第2-10位：表示所有的权限，用a (all) 表示

eg: -rw-r--r--. 1 root root 3664 Nov 30 17:42 CentOS-Vault.repo.bak

第十一位的1代表硬链接数

字符	权限	对文件的含义	对目录的含义
r	读权限	可以查看文件内容	可以列出目录的内容 (ls)
w	写权限	可以修改文件内容	可以在目录中创建删除文件(mkdir,rm)
x	执行权限	可以执行文件	可以进入目录(cd)

对于文件，我们有执行权限的命令：

r-cat,more,head,tail,less

w-echo,vi

x-命令，脚本

对于目录，我们有执行权限的命令：

r-ls

w-touch,mkdir,rm,rmdir

x-cd

能删除文件的权限是必须对该文件所在的目录有wx权限。

2.14.3 chmod

用于权限更改

英文: change mode (change the permissions mode of a file)

作用: 改变文件或目录权限

语法:

chmod [{ugoa}{+-=}{rwx}][文件名或目录]

chmod [mode=421][文件或目录]

参数: -R 下面的文件和子目录做相同权限操作 (Recursive递归的)

可以有修改一个文件的权限:

1.root

2.文件所有者

eg: [root@qianfeng01 ~]# chmod u+x a.txt

eg: [root@qianfeng01 ~]# chmod u+x,o-x a.txt

用数字来表示权限 (r=4, w=2, x=1, -=0)

eg: [root@qianfeng01 ~]# chmod 750 b.txt

rwx和数字表示方式能随意切换

注意: root用户是超级用户, 不管有没有权限, root都能进行更改。用普通用户测试权限。

不能用一个普通用户去修改另一个普通用户的权限。

2.14.4 chown

用于更改所有者

英文: change file ownership

作用：更改文件或者目录的所有者

语法 : chown user[:group] file...

-R : 递归修改

参数格式 : user : 新的档案拥有者的使用者 ID

group : 新的档案拥有者的使用者群体(group)

eg: [root@qianfeng01 ~]# chown lee file1 把file1文件的所有者改为用户lee

eg: [root@qianfeng01 ~]# chown lee:test file1 把file1文件的所有者改为用户lee,所属组改为test

eg: [root@qianfeng01 ~]# chown -R lee:test dir 修改dir及其子目录的所有者和所属组

2.14.5 chgrp

用于改变所属组

英文: change file group ownership

作用：改变文件或目录的所属组

语法 : chgrp [group] file...

eg: [root@qianfeng01 ~]# chgrp root test.log 把test.log的所属组修改为root

2.15 sudo权限的配置

2.15.1 作用

root把本来只能超级用户执行的命令赋予普通用户执行。

sudo的操作对象是系统命令

2.15.2 修改sudoers文件

执行 [root@qianfeng01 ~]# visudo

这里实际修改的是/etc/sudoers文件

sudoers文件内部的信息解释

root ALL= (ALL) ALL
#用户名 被管理主机的地址 (不是访问地址) = (可使用的身份) 授权命令 (绝对路径)

%wheel ALL= (ALL) ALL
#%组名 被管理主机的地址= (可使用的身份) 授权命令 (绝对路径)

[root@qianfeng01 ~]# sudo -l 查看可用的sudo命令

注意：

sudo命令用来以其他身份来执行命令，预设的身份为root。在/etc/sudoers中设置了可执行sudo指令的用户。

若其未经授权的用户企图使用sudo，则会发出警告的邮件给管理员。

用户使用sudo时，必须先输入密码，之后有5分钟的有效期限，超过期限则必须重新输入密码。

2.15.3 语法

sudo(选项)(参数)

选项

-b：在后台执行指令；

-h：显示帮助；

-H：将HOME环境变量设为新身份的HOME环境变量；

-k：结束密码的有效期限，也就是下次再执行sudo时便需要输入密码；

-l：列出目前用户可执行与无法执行的指令；

-p：改变询问密码的提示符号；
-s：执行指定的shell；
-u<用户>：以指定的用户作为新的身份。若不加上此参数，则预设以root作为新的身份；
-v：延长密码有效期限5分钟；
-V：显示版本信息。

2.15.4 实例

2.15.4.1 说明信息

1 配置sudo必须通过编辑/etc/sudoers文件，而且只有超级用户才可以修改它，还必须使用visudo编辑。

2 之所以使用visudo有两个原因

一是它能够防止两个用户同时修改它

二是它也能进行有限的语法检查

3 所以，即使只有你一个超级用户，你也最好用visudo来检查一下语法。visudo默认的是在vi里打开配置文件，用vi来修改文件。我们可以在编译时修改这个默认项。

4 visudo不会擅自保存带有语法错误的配置文件，它会提示你出现的问题，并询问该如何处理，错误提示：sudoers file:syntax error, line 22<<

此时我们有三种选择：

键入“e”是重新编辑

键入“x”是不保存退出

键入“Q”是退出并保存。如果真选择Q，那么sudo将不会再运行，直到错误被纠正。

2.15.4.2 实例1

让用户hd1可以通过sudo执行所有root可执行的命令。以root身份用visudo打开配置文件，可以看到类似下面几行：

```
ALL=(ALL)ALL
```

这里root有所有权限，只要仿照现有root的例子就行，我们在下面加一行（最好用tab作为空白）

```
hd1 ALL=(ALL) ALL
```

第一个ALL是指网络中的主机，它指明hd1可以在此主机上执行后面的命令。

第二个括号里的ALL是指目标用户，也就是以谁的身份去执行命令，通常我们指定为all即可。

最后一个ALL当然就是指命令名了。

保存退出后，切换到hd1用户，我们用它的身份执行命令：

```
[hd1@qianfeng01 ~]# sudo ls /root 在ls前面添加sudo可以让hd1拥有root的权限，查看root目录
```

2.15.4.3 实例2

我们限制一下hd1的权利。比如我们只想让他像root那样使用ls，继续修改sudoers文件

```
db1 ALL=(ALL) /bin/ls,/usr/bin/sudo
```

再次切换到hd1用户，执行命令：

```
[root@qianfeng01 ~]# sudo head -5 /etc/shadow 这是提示hd1没有权限执行这个命令。
```

2.15.4.4 免密配置

我们这样再添加一行配置文件

```
hd1 ALL=NOPASSWD: /bin/ls, /usr/bin/sudo
```

再来sudo一下：不再需要密码

第三章 软件管理机制

3.1 软件安装介绍

学软件开发，各种台的软件熟练安装是必须要熟练掌握。大家都知道，Windows下安装软件时，只需用鼠标双击软件的安装程序，或者用Zip等解压缩软件解压缩即可安装；在android或者apple中安装软件时，只需要在手机应用商店点击安装即可。而在Linux下安装软件难度高于Windows、Android、ios和windows phone下软件安装。下面我就详细讲解Linux下如

何安装软件。

Linux下软件的安装大概可以分为如下几种：

- 1 二进制程序的安装(最简单安装)
- 2 rpm程序安装(后缀名为*.rpm)
- 3 yum源安装(本质也是rpm安装，它只是rpm的一种安装方式而已)
- 4 源码安装

3.2 二进制安装

Linux下二进制格式的软件是指事先已经在各种平台编译安装好相关软件，然后压缩打包，在安装时只需解压或者执行安装可执行文件即可。这种软件发行格式类似与windows系统，这样做的原因是保护源代码程序不对外泄露，保护知识产权。二进制软件包的优点是安装简单、容易，缺点是缺乏灵活性，相应的软件包执行在对应平台下安装，离开这个环境软件就无法运行。

二进制软件包提供了很多类型的打包方式，最常见的就是我们RPM格式的包，还有以“*.tar.gz、 *.tgz、 *.bz2”等形式的二进制软件包，最后还有一个就是提供安装程序进行安装的二进制软件包。下面分别介绍：

3.2.1 安装*.tar.gz、 *.bz2二进制软件包

这种格式的软件包，安装其实就是简单的解压过程，根据不同的软件打包格式，我们用相应的解压命令解压即可。

对于*.tar.gz软件格式解压：tar -zxvf xxxxxxx.tar.gz 对于*.bz2软件格式解压：tar -jxvf xxxxxxx.tar.gz这类软件的卸载，对于解压后只有单一目录的软件，可以直接删除对应软件目录即可，如果解压后文件分散在几个目录中，需要一一手动删除目录。

例如：我们常用的应用服务器容器tomcat软件就是基于这种打包压缩格式发行的。我们只需下载后解压即可完成安装，如果需要卸载，直接删除对应的tomcat目录即可。

3.2.2 提供安装程序的软件包

这种软件包都提供了安装脚本或者安装向导程序，只需在下载此类软件包后解压，然后进入安装目录，找到类似的setup、install、install.sh之类的可执行文件运行即可，然后根据提示（比如安装路径，参数设置等等）进行相应的设置，接着安装就自动完成。

这类软件的卸载也提供了相应的卸载脚本或者卸载向导，根据提示即可完成软件卸载。

例如：经常使用的sun jdk就是这种安装方式，从网络下载下来的jdk是个二进制可执行文件，首先设置文件的可执行权限（例如chmod 755 jdk1.6.0_07.bin，权限操作本书后面会有详细讲述），然后运行软件（也即是./ jdk1.6.0_07.bin）即可完成安装。安装完成会在当前目录产生一个对应的jdk程序目录。

3.3 rpm安装

RPM软件包管理

RPM软件包也称为二进制软件包

RPM是RedHat Package Manager（RedHat软件包管理工具）的缩写，这一文件格式名称虽然打上了RedHat的标志，但是其原始设计理念是开放式的，现在包括RedHat、CentOS、SUSE等Linux的发行版本都有采用，可以算是公认的行业标准了。RPM文件在Linux系统中的安装最为简便

RPM命令使用

rpm的常用参数

i: 安装应用程序 (install)

vh: 显示安装进度 (verbose hash)

u: 升级软件包 (update)

qa: 显示所有已安装软件包 (query all)

e: 卸载应用程序 (erase)

注意：如果其它软件包有依赖关系，卸载时会产生提示信息，可使用--nodeps强行卸载。

查询所有安装的rpm包： [root@qianfeng01 ~]# rpm -qa

查询mysql相关的包： [root@qianfeng01 ~]# rpm -qa | grep mysql

安装： [root@qianfeng01 ~]# rpm -ivh jdk.rpm

卸载： [root@qianfeng01 ~]# rpm -e mysql*

```
强行卸载: [root@qianfeng01 ~]# rpm -e mysql* --nodeps
```

3.4 yum日常操作

图形化界面，手动挂载

yum(yellowdog updater modified):软件包管理工具

3.4.1 应用yum的好处

1 自动解决软件包依赖关系

2 方便的软件包升级

3.4.2 查询

```
yum list      查询所有可用软件包列表
```

```
yum search 关键字      搜索服务器上所有和关键字相关的包
```

```
yum info 关键字 来查找包名
```

3.4.3 安装

```
yum -y install 包名      -y 自动回答yes
```

3.4.4 升级

```
yum -y update 包名
```

注意：如果不加包名，就升级所有的，包括内核。必须加包名升级单个软件包，慎用升级所有的

```
检测升级 yum check-update
```

3.4.5 卸载

```
yum -y remove 包名
```

3.4.6 其他操作

yum --help、man yum 帮助

yum clean all 清除缓存和旧的包

yum repolist 查看当前可用的yum源

yum deplist httpd 列出一个包所有依赖的包

3.5 本地yum源安装

3.5.1 挂载光盘

1 自己创建mnt/cdrom文件夹:

```
[root@qianfeng01 ~]# mkdir /mnt/cdrom
```

2 相关内容说明

挂/卸载

命令: `mount umount`

格式: `mount [参数] 存储设备 挂载点`

说明: 存储设备对应分区的设备文件(如“`/dev/sdb1`”)或网络资源路径,
挂载点为用户指定用于挂载的目录。挂载点必须是一个已经存在的目录

参数	说明			
<code>-t</code> 文件系统类型	指定文件系统的类型，通常不必指定。 <code>mount</code> 会自动选择正确的类型	常用类型	光盘或光盘镜像	iso9660
			fat16文件系统	msdos
			fat32文件系统	vfat
			ntfs文件系统	ntfs
<code>-o options</code>	主要用来描述设备或档案的挂接方式	常用参数	采用只读方式挂接设备	ro
			用来把一个文件当成硬盘分区挂接上系统	loop
			采用读写方式挂接设备	rw
			指定访问文件系统所用字符集	iocharset
<code>-f</code>	用与除错。它会使 <code>mount</code> 并不执行实际挂上的动作，而是模拟整个挂上的过程。			
<code>-v</code>	显示较详细信息，通常会和 <code>-f</code> 一起使用。			
<code>-F</code>	使每一个 <code>mount</code> 的动作产生一个行程负责执行。挂上大量NFS档案系统时可以加快挂上的动作。			
<code>-a</code>	将 <code>/etc/fstab</code> 中定义的所有档案系统挂上。通常和 <code>-F</code> 一起使用			

##命令格式: `mount [-t vfstype] [-o options] device dir`

其中:

#1.-t vfstype 指定文件系统的类型，通常不必指定。`mount`会自动选择正确的类型。常用类型有:

光盘或光盘镜像: iso9660

DOS fat16文件系统: msdos

Windows 9x fat32文件系统: vfat

Windows NT ntfs文件系统: ntfs

Mount Windows文件网络共享: smbfs

UNIX(LINUX) 文件网络共享: nfs

#2.-o options 主要用来描述设备或档案的挂接方式。常用的参数有:

loop: 用来把一个文件当成硬盘分区挂接上系统

ro: 采用只读方式挂接设备

rw: 采用读写方式挂接设备

iocharset: 指定访问文件系统所用字符集

#3.device 要挂接(mount)的设备。

#4.dir设备在系统上的挂接点(mount point)。

##将镜像文件挂载在/mnt/cdrom下

```
[root@qianfeng01 ~]# mkdir /mnt/cdrom #在根目录下的mnt目录下创建目录cdrom
```

```
[root@qianfeng01 ~]# mount -t iso9660 -o loop /dev/cdrom /mnt/cdrom ##挂载
```

```
[root@qianfeng01 ~]# umount /mnt/dvd
```

##卸载

3.5.2 让网络yum源文件失效

```
[root@qianfeng01 yum.repos.d]# cd /etc/yum.repos.d/  
  
[root@qianfeng01 yum.repos.d]# rename .repo .repo.bak * #重命名所有的.repo文件  
  
[root@qianfeng01 yum.repos.d]# cp CentOS-Media.repo.bak CentOS-Media.repo #配置一个.repo文件
```

3.5.3 对CentOS-Base.repo的解释

```
[root@qianfeng01 yum.repos.d]# cat /etc/yum.repos.d/CentOS-Base.repo
```

```
[base]  
  
name=CentOS-$releasever - Base  
  
mirrorlist=http://mirrorlist.centos.org/?  
release=$releasever&arch=$basearch&repo=os  
  
baseurl=http://mirror.centos.org/centos/$releasever/os/$basearch/  
  
gpgcheck=1  
  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6
```

解释：

[base]	容器名称，一定要放在[]中
name	容器说明，这个可以自己随便写
mirrorlist	镜像站点，这个可以注释掉
baseurl	yum源服务器的地址。默认是CentOS官方的yum源服务器
enabled	此容器是否生效，如果不写或写enable=1都是生效，写成enable=0就是不生效
gpgcheck	如果是1是指RPM的数字证书生效，如果是0则不生效
pgpkey	数字证书的公匙文件保存位置，不用修改

3.5.4 修改光盘yum源文件

```
[root@qianfeng01 yum.repos.d]# vi CentOS-Media.repo

[c6-media]
name=CentOS-$releasever - Media
baseurl=file:///mnt/cdrom
#这里的地址为自己光盘挂载地址，并把不存在的地址注释掉，在行首注释
# file:///media/cdrom/
# file:///media/cdrecorder/
gpgcheck=1
enabled=1          #把原来的0改为1，让这个yum源配资文件生效
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6
```

可使用 yum repolist 查看 当前可用yum源

```
yum clean all

yum repolist
```

注意点：

- 1 如果一旦重启服务器，需要重新挂载
- 2 需要检查当前的虚拟机的镜像文件挂载处，将已经连接前面打勾

3 如果镜像文件有两个DVD,我们可以这样做

1)再添加一个光驱,将DVD2连接进来

2)将服务器重启,发现在/dev下有了两个镜像文件(cdrom和cdrom1)

3)在/mnt下再去创建一个文件夹(cdrom1)

4)分别进行挂载操作

```
[root@qianfeng01 ~]# mount -t iso9660 -o loop /dev/cdrom /mnt/cdrom
```

```
[root@qianfeng01 ~]# mount -t iso9660 -o loop /dev/cdrom1 /mnt/cdrom1
```

5)到/etc/yum.repos.d里面,同时添加路径

```
file:///mnt/cdrom
```

```
file:///mnt/cdrom1
```

3.6 搭建局域网YUM源

3.6.1 选择局域网内的一台机器作为YUM源服务器, 安装HTTP服务进程

```
[root@qianfeng01 ~]# yum list | grep http
[root@qianfeng01 ~]# yum -y install httpd.x86_64
```

3.6.2 开启HTTP服务进程并设置开机自启动

```
[root@qianfeng01 ~]# service httpd start
[root@qianfeng01 ~]# chkconfig httpd on
```

3.6.3 配置HTTP服务管理YUM源目录

```
# HTTP服务进程管理的目录是/var/www/html  
  
#在管理目录下，创建软连接，指向YUM源目录  
[root@qianfeng01 ~]# ln -s /mnt/dvd /var/www/html/repo      <====都是用绝对路  
径，保没错
```

3.6.4 其他机器只需要把本地YUM源的路径指向局域网YUM源地址即可

```
[root@qianfeng01 yum.repos.d]# vi local.repo          #编辑文件，后缀  
必须是.repo  
  
[local]  
<==YUM源仓库名  
name=localrepo                                     <==昵称  
baseurl=http://局域网IP/repo                      <==局域网YUM仓库位置  
enabled=1                                           <==1  
表示启用，0表示禁用  
gpgcheck=1                                         <==1  
表示 校验  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6 <==校验钥匙
```

3.7 源码编译安装

由于linux操作系统开放源代码，因而在其上安装的软件大部分也都是开源软件，例如 apache、tomcat、php等软件。开源软件基本都提供源码下载，源码安装的方式；

3.7.1 源码安装的优点

- 1 用户可以自己定制软件功能，安装需要的模块，不需要的功能可以不用安装。
- 2 用户还可以自己选择安装路径，方便管理。
- 3 卸载软件也很方便，只需删除对应的安装目录即可。
- 4 能最大程度和服务器平台融合，效率稍微比其他方式高。
- 5 没有windows所谓的注册表之说。

3.7.2 源码安装的缺点

- 1 安装较为繁琐，需要自己配置
- 2 安装较为耗时，需要自己编译源码
- 3 安装较为容易出错，出错也难以解决

3.7.3 安装步骤

源码安装软件一般有以下几个步骤组成：下载解压源码、分析安装平台环境（ifconfig） 、编译安装软件（make,make install）。下面我们分别介绍。

3.7.3.1 下载解压源码

linux下软件的源码一般都是C或者C++语言编写的，并且都会在软件的官方网站上提供源码包下载，例如apache开源项目的官方网站为<http://www.apache.org>，此外在网上的很多开源社区一般也提供一些常用软件的源码下载，源码都打包成压缩文件，常见的源码打包压缩格式有“.tar.gz”、“.tar.baz2”等。我们可以从网站上首先下载源码文件，然后在传到所在的linux系统下，如果你的linux系统处于联网状态下，也可以直接在系统内通过wget之类的下载命令将源码包直接下载到linux系统下。

下载完成，把相应的软件解包即可，针对下载软件包的不同，使用不同的方法进行解压，解压完毕后进入解压生成的目录中，在这个目录下，一般都存在一个README文件，这个文本文件非常重
要，它详细的介绍了这个软件所能完成的功能、授权许可、安装需求、安装注意事项、安装方式等等，由于linux各个版本的差异，以及安装环境的不同，软件的安装方式也不尽相同，所以在安装软件前，一定要事先阅读这个README文件，确保安装的正确性。

3.7.3.2 分析安装平台环境

在软件包解压完毕之后，我们进入源码目录，一般都会存在configure和README这么两个文件， README就是上面我们讲到的，是对软件的介绍和安装说明；linux下软件的安装受到操作系统安装环境的影响，比如某些软件在安装或者运行过程中需要调用操作系统本身的库文件，或者需要运行系统的某个工具等等，但是系统中不存在这个库文件或者这个工具没有安装，那么安装或者运行软件就会失败。

为了避免这个问题的出现，在安装软件的时候就需要分析操作系统环境，检测当前系统是否拥有安装软件所需的所有文件和工具，如果系统缺少某个文件，就给出提示，直到满足软件的所有需求为止。这个就是configure文件的功能，configure文件一般是个可执行文件，可以在当前目录下直接输入“./configure”进行软件安装的环境测试，如果提示缺少某些安装包，就需要进行安装，直到测试通过。通常的，源码安装都需要GCC或者CC编译器，这些编译器一般在安装系统时定制安装包中的开发工具选项下，其实这也是我们在第二章安装linux系统中要求选择开发工具包的原因所在。

此外，在执行configure分析软件需求时，还可以在“./configure”后面加上软件的安装路径以及安装所需的模块等等一些选项，来定制用户需要的软件功能。

3.7.3.3 编译安装软件

在验证软件安装环境后，正式进入软件的编译步骤，在进行编译前，首先了解下关于编译的一些基础知识。

在linux系统下，make是我们经常用到的编译命令，无论是安装软件还是项目开发，都会经常用到编译安装命令，也就是make和make install，对于一个包含很多源文件的应用程序，使用make和makefile工具可以简单快速的解决各个源文件之间复杂的依赖关系，同时，make工具还可以自动完成所有源码文件的编译工作，并且可以只对上次编译后修改过的文件进行增量编译，因此，熟练掌握了make和makefile工具之后，源码安装软件就变的像windows下安装软件一样简单。

3.7.3.4 Makefile文件介绍

make工具最主要的功能就是通过makefile文件来实现的，makefile文件是按照某种语法来进行编写的，文件中定义了各个源文件之间的依赖关系，并说明了如何编译源文件并生成可执行文件，它通过描述各个源程序之间的关系让make工具自动完成编译工作。

在linux系统下，习惯用Makefile替代makefile文件，我们在执行configure通过后，就会在当前目录下生成这个文件，一般情况下在命令行输入make时，make命令会默认去寻找当前目录下的Makefile文件，如果我们用其它文件作为makefile，就要在make命令选项后指定makefile文件，例如我们设定编译规则文件为make_file，就需要使用以下命令指定：[root@WEBServer ~]#make -f make_file

3.7.3.5 Makefile举例

下面我们通过举例了解下Makefile的编写原则和内容。

例如有一个test程序，有三个源文件file1.c、file2.c、file3.c以及库文件LS编译生成，这三个源文件还分别包含了自己的头文件1.h、2.h、3.h。通常，C编译器将会输出三个目标文件file1.o、file2.o、file3.o。假定file1.c和file3.c都要声明用到一个apcs的文件，但是file2.c不用，也就是在file1.c和file3.c文件中都有类似“#include “apcs””的声明。

下面这段代码就是对test程序的描述：

```
1 #It is a example for describing makefile
2 test: file1.o file2.o file3.o
3 cc file1.o file2.o file3.o -LS -o test
4 file1.o: file1.c 1.h apcs
5 gcc -c file1.c
6 file3.o: file3.c 3.h apcs
7 gcc -c file3.c
8 file2.o: file2.c c.h
9 gcc -c file2.c
```

这个描述文档其实就是一个makefile文件，第一行为注释行，第二行指定test程序有file1.o、file2.o、file3.o链接生成，第三行描述了如何从test所依赖的文件建立可执行文件，接下来的第四、六、八行定义了三个目标文件和它们依赖的“.c”、“.h”和“apcs”文件。剩下的第五、七、九行定义了如何从目标依赖的文件生成可执行文件。

在了解Makefile文件结构后，我们就可以控制编译的选项，定制自己所需的软件功能。

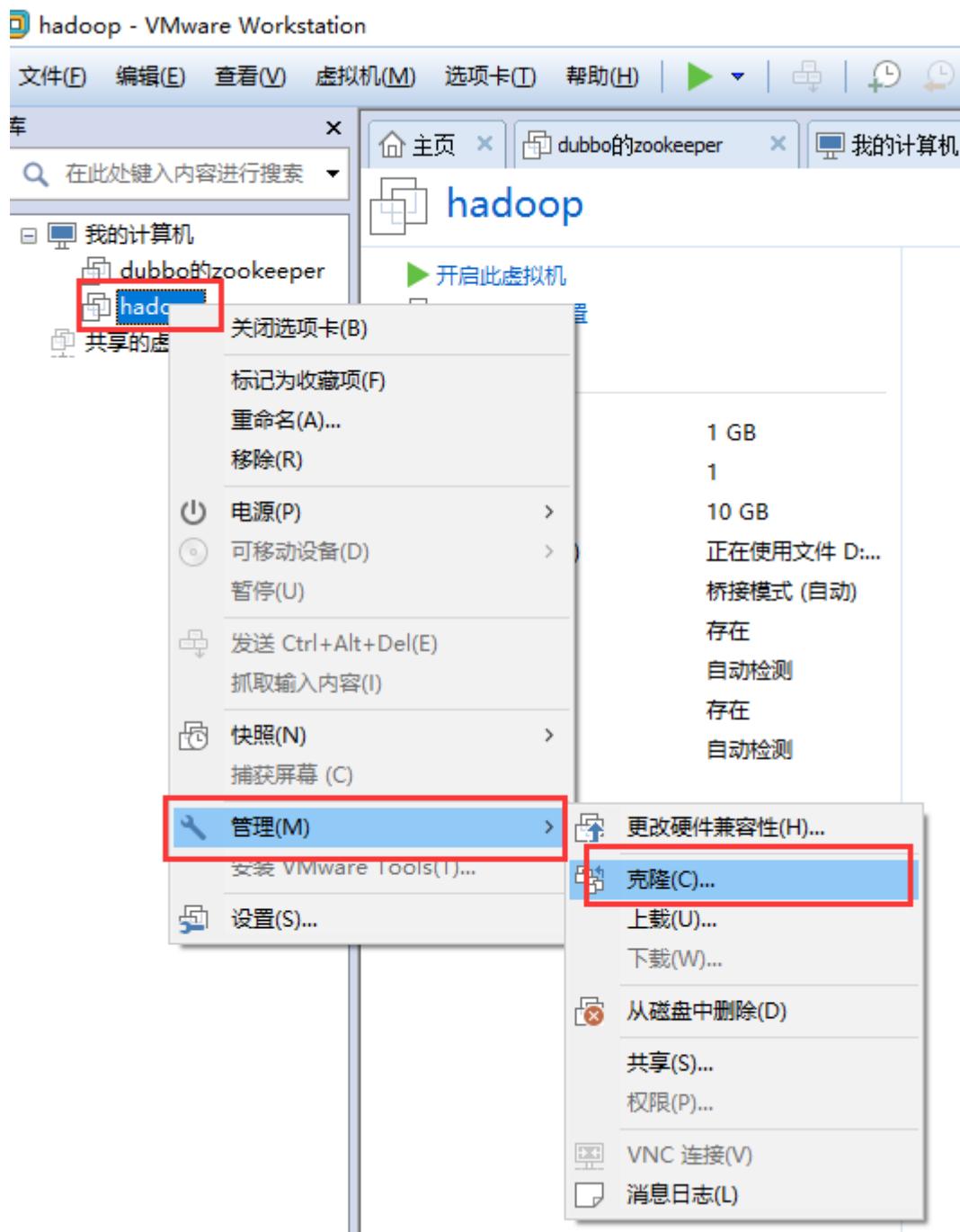
接下来，我们只需在命令行输入make命令，即可进入编译阶段，根据软件源程序的大小和系统的硬

件配置，编译时间不定，编译完毕，会在当前目录以及子目录下生成相应的可执行文件，接下来就进入最后的安装软件阶段，在命令行输入make install命令，开始安装软件，安装进程会首先创建安装目录，如果我们没有指定安装的目录，安装程序默认会安装在系统的/usr/local目录下创建安装目录，然后将相应的文件和可执行程序从源码目录拷贝到安装目录下，这样安装就完毕了。

3.8 克隆

3.8.1 克隆前的准备工作

安装minimal版本虚拟机(自行安装)



3.8.2 克隆后所需要进行的操作

3.8.2.1 修改IP地址

```
[root@qianfeng01 ~]# vi /etc/sysconfig/network-scripts/ifcfg-ens33
```

3.8.2.2 修改主机名

改主机名

```
[root@qianfeng01 ~]# hostnamectl set-hostname xxx
```

3.8.2.3 修改域名映射文件/etc/hosts

```
[root@qianfeng01 ~]# vi /etc/hosts
```

3.8.2.4 使用nat模式实现两台虚拟机局域网通讯

3.8.2.5 操作新linux系统流程

1 更改主机名临时+永久 2 检查网卡是否启动 3 关闭NetworkManager网络管理服务 4 关闭防火墙 5 更改ip地址 6 更改主机名称和主机映射 7 使用连接工具（finalshell）进行连接 8 如果出现问题按照以上流程检查

3.9 scp命令

集群模式下，各节点之间如何方便的拷贝文件？

命令格式：

scp file 远程用户名@远程服务器IP:~/ (注意：冒号和目录之间不能有空格)

如果拷贝目录，需要加-r 选项。

使用root用户

eg: [root@qianfeng01 ~]# scp /etc/profile root@qianfeng02:/etc

eg: [root@qianfeng01 ~]# scp -r /usr/jdk1.8 qianfeng02:/usr/java 使用
root用户可以将远程用户名省略

```
eg: [root@qianfeng01 ~]# scp /root/a/ qianfeng02:$PWD
```

注：使用\$PWD 默认到当前正在操作的目录(qianfeng02的etc下)。 注意：目标主机要有与当前正在操作的目录相同的目录，前面的文件或文件夹会直接拷贝到后面的目录中

~:到当前用户的宿主目录

```
eg: [root@qianfeng01 ~]# scp /etc/services qianfeng02:/root/service.hard
```

可以通过这种方式修改拷贝的文件名。

3.10 ssh免密登录

3.10.1 ssh免密登录作用

方便的进行节点之间的切换 ssh node2

3.10.2 免密登录实现流程

1 在第一台机器上生成一对钥匙，公钥和私钥

```
[root@node1 ~]# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): 
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase): 
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:
bd:8c:39:36:1c:e2:f8:c1:a6:53:98:09:e4:0b:63:e5 root@node1
The key's randomart image is:
+---[ RSA 2048]----+
          .. 
         ooE 
        .o o +. S . 
        . ++.o = . 
        ..= B o 
        .+ o o 
        ...
+---[ RSA 2048]----+
[root@node1 ~]#
```

输入三次回车

2 当前用户的宿主目录下的.ssh目录多了两个文件

```
[root@node1 ~]# cd .ssh/
[root@node1 .ssh]# ll
total 8
-rw-----. 1 root root 1675 Jul 23 18:12 id_rsa
-rw-r--r--. 1 root root 392 Jul 23 18:12 id_rsa.pub
[root@node1 .ssh]#
```

3 将公钥拷贝给要免密码登录的机器

注意：

1 主机名和ip都可以（确保配置了ip的映射）

2 如果出现-bash: ssh-copy-id: command not found,说明ssh-copy-id这个指令没有找到，自己安装下即可

安装命令：

```
[root@qianfeng01 ~]# yum -y install openssh-clients
```

还需要输入密码

```
[root@node1 .ssh]# ssh-copy-id root@node2          主机名或ip地址都可以
root@node2's password:
Now try logging into the machine, with "ssh 'root@node2'", and check in:
  .ssh/authorized_keys
to make sure we haven't added extra keys that you weren't expecting.
[root@node1 .ssh]#
```

4 拷贝完成之后，会在要免密登录的机器上生成授权密码文件

```
[root@node2 .ssh]#
[root@node2 .ssh]# pwd
/root/.ssh
[root@node2 .ssh]# ll
total 4
-rw----- 1 root root 392 Jul 23 18:18 authorized_keys
[root@node2 .ssh]#
```

注意：免密码登录是单向的

3.10.2 ssh免密登录原理

1 node1 使用 ssh-keygen -t rsa 生成公钥和私钥

2 ssh-copy-id root@node2 将公钥拷贝到node2上，实际上是把公钥内容追加到 authorized_keys 文件中。

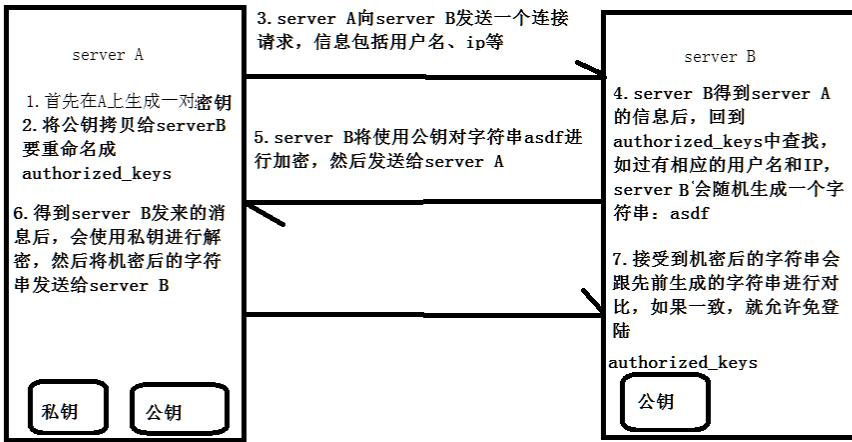
3 请求时：node1向node2发送连接请求时，附带主机，ip地址等信息

4 node2收到请求后，去授权文件里查找node1的公钥，找到之后，随机生成一个字符串，并用公钥加密，并发送给node1

5 node1接收到密文之后，用私钥解密，并把解密结果返回给node2

6 node2拿到解密结果之后，与之前的字符串进行比较，如果相同，则可以登录成功。

原理图



3.11 定时器crontab

3.11.1 crontab的简介

- 在Linux中,周期性的执行任务一般由cron这个守护进程来处理,它是一个linux下的定时任务执行工具,可以在无需人工干预的情况下运行作业。

- [ps -ef|grep cron]
- cron读取一个或多个配置文件,这些配置文件中包含了命令行及其调用时间。
- cron的配置文件称为“crontab”,是“cron table”的简写。

3.11.2 cron服务

```
[root@qianfeng01 ~]# service crond start      //启动服务
[root@qianfeng01 ~]# service crond stop       //关闭服务
[root@qianfeng01 ~]# service crond restart    //重启服务
[root@qianfeng01 ~]# service crond reload    //重新载入配置
[root@qianfeng01 ~]# service crond status     //查看服务状态
```

3.11.3 cron的配置文件位置

- 1 /var/spool/cron/
- 2 说明这个目录下存放的是每个用户包括root的crontab任务,每个任务以创建者的名字命名,比如tom建的crontab任务对应的文件就是/var/spool/cron/tom。一般一个用户最多只有一个crontab文件。

3.11.4 crontab命令格式

作用：用于生成cron进程所需要的crontab文件

格式：crontab [-u username] -e

3.11.5 文件内容格式

格式如下：

```
* * * * * user-name command to be executed
```

共有六部分组成，分别表示：

分 时 日 月 星期 要运行的命令

解析：

minute:	一小时中的哪一分钟 [0 ~ 59]
hour:	一天中的哪个小时 [0 ~ 23]
day:	一月中的哪一天 [1 ~ 31]
month:	一年中的哪一月 [1 ~ 12]
week:	一周中的哪一天 [0 ~ 6] 0表示星期天
commands:	执行的命令

书写注意事项

- 1 全都不能为空，必须填入，不知道的值使用通配符*表示任何时间
- 2 每个时间字段都可以指定多个值，不连续的值用,间隔，连续的值用-间隔。
- 3 命令应该给出绝对路径
- 4 用户必须具有运行所对应的命令或程序的权限
- 5 */num 表示频率

3.11.6 应用案例

1 每天早上6点

```
0 6 * * * echo "Good morning." >> /tmp/test.txt
```

//注意 如果不进行追加，从屏幕上看不到任何输出，因为cron把任何输出都email到root的信箱了。

2 每两个小时

```
0 */2 * * * echo "Have a break now." >> /tmp/test.txt
```

3 晚上11点到早上8点之间每两个小时和早上八点

```
0 23-7/2, 8 * * * echo "Have a good dream" >> /tmp/test.txt
```

4 周一到周五下午，5点半提醒学生15分钟后关机

```
30 17 * * 1-5 /usr/bin/wall < /etc/issue
```

```
45 17 * * 1-5 /sbin/shutdown -h now
```

5 学校的计划任务，12点14点，检查apache服务是否启动

```
*/2 12-14 * 3-6,9-12 1-5
```

6 每月 1、1 0、2 2日的4:45运行/apps/bin目录下的backup.sh

```
45 4 1,10,22 * * /apps/bin/backup.sh
```

7 每周六、周日的 1 : 10运行一个find命令

```
10 1 * * 6,0 /bin/find -name "core" -exec rm {} \;
```

8 在每天 18:00至23 :00之间每隔30分钟运行/apps/bin目录下的dbcheck.sh

```
0,30 18-23 * * * /apps/bin/dbcheck.sh
```

9 每星期六的 11:00 pm运行/apps/bin目录下的qtrend.sh

```
0 23 * * 6 /apps/bin/qtrend.sh
```

3.12 时间同步服务器的搭建

在实际生产环境中，很多软件或者很多任务对集群上的时间是否一致要求是很严格的。有的要求集群中的所有机器上的时间差不能在10分钟以外，有的要求所有机器上的时间差不能在30秒以外。这样，在技术上，就要涉及到怎么将时间同步的问题了。

3.12.1 同步网络上的时间

在每一台机器上，都可以使用定时任务crontab同步网络上的时间服务器，如time.windows.com

前提：安装ntpdate.x86_64

然后：使用root用户，进行定时

```
0 * * * * /usr/sbin/ntpdate -u time.windows.com
```

3.12.2 自定义时间服务器

如果集群没有联网，那么可以自己搭建一个时间服务器，让集群中的所有机器都同步局域网内的时间服务器。

1 选择集群中的某一台机器master作为时间服务器

2 保证这台服务器安装了ntp.x86_64。

3 保证ntpd 服务运行.....

```
[root@qianfeng01 ~]# sudo service ntpd start
```

开机自启动：

```
[root@qianfeng01 ~]# chkconfig ntpd on
```

4 配置相应文件： [root@qianfeng01 ~]# vi /etc/ntp.conf

```
# Hosts on local network are less restricted.
```

```
#restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap
```

```
restrict 192.168.81.0 mask 255.255.255.0 nomodify notrap
```

```
// 添加集群中的网络段位
```

```
# Use public servers from the pool.ntp.org project.  
# Please consider joining the pool  
(http://www.pool.ntp.org/join.html).  
#server 0.centos.pool.ntp.org iburst    注释掉  
#server 1.centos.pool.ntp.org iburst    注释掉  
#server 2.centos.pool.ntp.org iburst    注释掉  
#server 3.centos.pool.ntp.org iburst    注释掉  
server 127.127.1.0      -master作为服务器
```

5 其他机器要保证安装ntpdate.x86_64

6 其他机器要使用root定义定时器

```
*/1 * * * * /usr/sbin/ntpdate -u master
```