
link: null
title: 珠峰架构师成长计划
description: 分为内核版本和发型版本
keywords: null
author: null
date: null
publisher: 珠峰架构师成长计划
stats: paragraph=317 sentences=361, words=2161

1. Linux

- Linux是一套免费使用和自由传播的类Unix操作系统
- 在服务器端领域和嵌入式领域有非常广泛的应用

2.版本

分为内核版本和发型版本

- [kernel \(https://www.kernel.org/\)](https://www.kernel.org/)
- 各个厂商会制作自己的发行版本
 - redhat
 - CentOS
 - ubuntu
 - fedora

3. Linux与Windows的不同

- Linux严格区分大小写
- Linux中所有内容以文件形式保存，包括硬件、用户和文件。
- Linux不靠扩展名区分文件类型，是靠权限来区分，但是有一些约定的扩展名，是给管理员看的
 - 压缩包 .gz .bz2 .tar.bz2 .tgz
 - 二进制文件 .rpm
 - 网页文件 .html .php
 - 脚本文件 .sh
 - 配置文件 .conf
- Windows下的程序不能直接在Linux中安装和运行
- Linux更多使用字符界面
 - 占用的系统资源更少
 - 减少了出错和被攻击的可能性，会让系统更稳定

4. VMware安装

4.1 什么是虚拟机

- 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个人电脑
- 可以实现在一台机器上真正同时运行两个独立的操作系统
- [VMware \(http://www.vmware.com\)](http://www.vmware.com)

4.2 虚拟机的主要特点

- 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
- 本机系统可以与虚拟机系统网络通信
- 可以设定并且随时修改虚拟机操作系统的硬件环境
- 系统快照可以方便备份和回滚

4.3 建议VMWare配置

- CPU 建议主频1GHz以上
- 内存 建议2GB以上
- 硬盘 建议分区空闲空间8GB以上

4.4 虚拟机的安装

- [VMware8.0 \(http://www.vmware.com\)](http://www.vmware.com)
- [4m2g \(https://pan.baidu.com/s/1kiOXFDYq4bp8jPqYspBg\)](https://pan.baidu.com/s/1kiOXFDYq4bp8jPqYspBg)

5.linux 系统启动

5.1 BIOS

- 计算机通电后，第一件事就是读取刷入ROM芯片的开机程序，这个程序叫做基本输入输出系统(Basic Input/Output System)

5.2 硬件自检

- BIOS程序首先检查，计算机硬件能否满足运行的基本条件，这叫做“硬件自检” (Power-On Self-Test)
- 如果硬件出现问题，主板会发出不同含义的蜂鸣，启动中止。如果没有问题，屏幕就会显示出CPU、内存、硬盘等信息。



5.3 启动顺序 #

- 硬件自检完成后，BIOS把控制权转交给下一阶段的启动程序。
- 这时，BIOS需要知道 6#x4E0B; 6#x4E00; 6#x9636; 6#x6BB5; 6#x7684; 6#x542F; 6#x52A8; 6#x7A0B; 6#x5E8F; 具体存放在哪一个设备
- BIOS需要有一个外部储存设备的排序，排在前面的设备就是优先转交控制权的设备。这种排序叫做“启动顺序”（Boot Sequence）
- BIOS按照“启动顺序”，把控制权转交给排在第一位的储存设备



5.4 主引导记录 #

- 计算机读取该设备的第一个扇区，也就是读取最前面的 512 个字节。如果这 512 个字节的最后两个字节是 0x55 和 0xAA，表明这个设备可以用于启动；如果不是，表明设备不能用于启动，控制权于是被转交给 6#x542F; 6#x52A8; 6#x987A; 6#x5E8F; 中的下一个设备
- 这最前面的 512 个字节，就叫做 6#x4E3B; 6#x5F15; 6#x5BFC; 6#x8BB0; 6#x5F55; （Master boot record，缩写为MBR）

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
33	C0	8E	D0	BC	00	7C	FB	50	07	50	1F	FC	BE	1B	7C
BF	1B	06	50	57	B9	E5	01	F3	A4	CB	BD	BE	07	B1	04
38	6E	00	7C	09	75	13	83	C5	10	E2	F4	CD	18	8B	F5
83	C6	10	49	74	19	38	2C	74	F6	A0	B5	07	B4	07	8B
F0	AC	3C	00	74	FC	BB	07	00	B4	0E	CD	10	EB	F2	88
4E	10	E8	46	00	73	2A	FE	46	10	80	7E	04	0B	74	0B
80	7E	04	0C	74	05	A0	B6	07	75	D2	80	46	02	06	83
46	08	06	83	56	0A	00	E8	21	00	73	05	A0	B6	07	EB
BC	81	3E	FE	7D	55	AA	74	0B	80	7E	10	00	74	C8	A0
B7	07	EB	A9	8B	FC	1E	57	8B	F5	CB	BF	05	00	8A	56
00	B4	08	CD	13	72	23	8A	C1	24	3F	98	8A	DE	8A	FC
43	F7	E3	8B	D1	86	D6	B1	06	D2	EE	42	F7	E2	39	56
0A	77	23	72	05	39	46	08	73	1C	B8	01	02	BB	00	7C
8B	4E	02	8B	56	00	CD	13	73	51	4F	74	4E	32	E4	8A
56	00	CD	13	EB	E4	8A	56	00	60	BB	AA	55	B4	41	CD
13	72	36	81	FB	55	AA	56	C1	01	74	2B	61	60		
6A	00	6A	00	FF	76	0A	FF	76	08	6A	00	68	00	7C	6A
01	6A	10	B4	42	8B	F4	CD	13	61	61	73	0E	4F	74	0B
32	E4	8A	56	00	CD	13	EB	D6	61	F9	C3	49	6E	76	61
6C	69	64	20	70	61	72	74	69	74	69	6F	6E	20	74	61
62	6C	65	00	45	72	72	6F	72	20	6C	6F	61	64	69	6E
67	20	6F	70	65	72	61	74	69	6E	67	20	73	79	73	74
65	6D	00	4D	69	73	73	69	6E	67	20	6F	70	65	72	61
74	69	6E	67	20	73	79	73	74	65	6D	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	2C	44	63	51	59	88	80	26	26	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	55	AA

5.5 主引导记录的结构

- 主引导记录只有512个字节。它的主要作用是告诉计算机到硬盘的哪一个位置去找操作系统。
 - (1) 第1-446字节：是用来记录系统的启动信息的,调用操作系统的机器码
 - (2) 第447-510字节(64个字节)：分区表（Partition table），分区表的作用，是将硬盘分成若干个区
 - (3) 第511-512字节：主引导记录签名（0x55和0xAA）

0000-0088	Master Boot Record主引导程序	主引导程序	0-136	137
0089-01BD	出错信息数据区	数据区	137-445	309
01BE-01CD	分区项1（16字节）	分区表	446-461	16
01CE-01DD	分区项2（16字节）		462-477	16
01DE-01ED	分区项3（16字节）		478-493	16
01EE-01FD	分区项4（16字节）		494-509	16
01FE	55	结束标志	510	1
01FF	AA		511	1
				512

5.6 分区表

- 磁盘分区是使用分区编辑器在磁盘上划分几个逻辑部分
- 磁盘一旦划分成多个分区，不同类的目录与文件可以存储进不同的分区内
- 主引导记录因此必须知道将控制权转交给哪个区
- 分区表的长度只有64个字节，里面又分成四项，每项16个字节。所以，一个硬盘最多只能分四个一级分区，又叫做 0x4E3B ； 0x5206 ； 0x533A ；
 - (1) 第1个字节：如果为 0x80 ，就表示该主分区是激活分区，控制权要转交给这个分区。四个主分区里面只能有一个是激活的。
 - (2) 第2-4个字节：主分区第一个扇区的物理位置(柱面(Cylinder)、磁头(Heads)、扇区号(Sector)等等)
 - (3) 第5个字节：主分区类型，比如FAT32、NTFS等
 - (4) 第6-8个字节：主分区最后一个扇区的物理位置
 - (5) 第9-12字节：主分区第一个扇区的逻辑地址
 - (6) 第13-16字节：主分区的扇区总数

偏移	描述	大小/字节	地址	数据
0x000	引导代码	446	0x1BE	0x80
0x1BE	第1分区入口	16	0x1BF	0x03
0x1CE	第2分区入口	16	0x1C1	0x2D
0x1DE	第3分区入口	16	0x1C2	0x00
0x1EE	第4分区入口	16	0x1C3	0x06
0x1FE	结束标志(0x55 AA)	2	0x1C4	0x07
			0x1C6	0xE7
			0x1C5	0xE2
			0x1C6	0xE9
			0x1C7	0x00
			0x1C8	0x00
			0x1C9	0x00
			0x1CA	0x17
			0x1CB	0x4B
			0x1CC	0x0F
			0x1CD	0x00

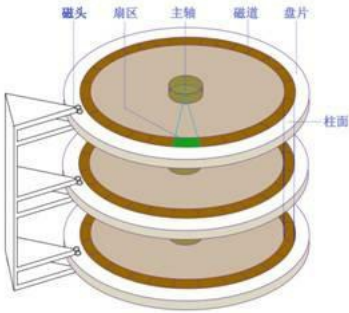
偏移	描述	大小/字节
0x00	分区状态	1
0x01	分区开始磁头	1
0x02	分区开始簇/扇区	2
0x04	分区类型	1
0x05	分区结束磁头	1
0x06	分区结束簇/扇区	2
0x08	分区相对起始扇区	4
0x0C	分区总扇区数	4

图 2 主引导记录

WeeQoo 维库

5.6.1 扇区 #

- 扇区是硬盘存储上的概念，机械硬盘的内部是金属盘片，将圆形的盘片划分成若干个扇形区域，这就是扇区，若干个扇区就组成整个盘片
- 扇区是硬盘上最小的读写单位，这个是硬盘决定的，不是操作系统决定的
- 对现在的硬盘来说，逻辑扇区的大小等于物理扇区的大小，所以也并没有严格区分物理扇区和逻辑扇区
- 物理扇区就是底层硬件意义上的扇区
- 而在物理扇区之上，操作系统划分的逻辑扇区，是为了方便操作系统读写入硬盘数据而设置的，其大小与具体地址，都可以通过一定的公式与物理地址对应
- 同时如果出现坏扇区，系统可以通过逻辑扇区，将物理上的坏扇区地址，重新定位到硬盘上备用的好扇区上，这样也就延长了硬盘的使用寿命



5.7 硬盘启动 #

- 计算机的控制权就要转交给硬盘的某个分区了
- 四个主分区里面，只有一个激活的。计算机读取激活分区的第一个扇区，叫做“卷引导记录”(Volume boot record，缩写为VBR)

5.8 操作系统 #

- 控制权转交给操作系统后，操作系统的内核首先被载入内存。
- 以Linux系统为例，先载入 /boot目录下面的 kernel。内核加载成功后，第一个运行的程序是 /sbin/init。它根据配置文件 (Debian系统是/etc/initab) 产生init进程。这是Linux启动后的第一个进程，pid 进程编号为1，其他进程都是它的后代
- 然后，init线程加载系统的各个模块，比如窗口程序和网络程序，直至执行 /bin/login程序，跳出登录界面，等待用户输入用户名和密码。

6. 硬件设备文件名 #

- 只要插入硬盘，Linux会自动检测和分配名称
- 一个硬盘可以分成多个分区，每个分区都会有一个系统分配的名称
- 第一块SCSI硬盘名称叫 sda,它的第一个分区叫 sda1

硬件 设备文件名 IDE硬盘 /dev/hd[a-d] SCSI/SATA/USB硬盘 /dev/sd[a-p] 光驱 /dev/cdrom或/dev/hdc 软盘 /dev/fd[0-1] 打印机(25针) /dev/lp[0-2] 打印机(USB) /dev/usb/lp[0-15] 鼠标 /dev/mouse

6.1 IDE硬盘接口 #



6.2 SCSI硬盘接口 #



6.3 SATA硬盘接口 #



7. 分区 #

衣柜(硬盘)



- 磁盘分区是使用分区编辑器在磁盘上划分几个逻辑部分
- 磁盘一旦划分成多个分区，不同类的目录与文件可以存储进不同的分区内
- 分区表的长度只有64个字节，里面又分成四项，每项16个字节。所以，一个硬盘最多只能分四个一级分区，又叫做 6#x4E3B; 6#x5206; 6#x533A;

7.1 扩展分区 #

- 随着硬盘越来越大,四个主分区已经不够了,需要更多的分区. 但是,分区表只有四项, 因此规定有且仅有一个区可以被定义成扩展分区(Extended partition)
- 所谓扩展分区, 就是指这个区里面又分成多个区. 这种分区里面的分区, 就叫做逻辑分区(logical partition)
- 为了突破4个分区的限制, 就取出一个分区作为 6#x6269; 6#x5C55; 6#x5206; 6#x533A;
 - 扩展分区最多只能有 1个
 - 主分区加扩展分区最多有 4个
 - 不能写入数据, 只能包含逻辑分区, 逻辑分区最多是23个

8. 格式化 #

- 格式化是指根据用户选定的文件系统(如FAT16(2G),FAT32(4G)、NTFS、EXT2、EXT3、EXT4)对分区进行划分
- 目的是为了更的写入和读取数据
- 主要是把整个分区切分成等大小的数据块, 每个数据块是4KB, 10K需要使用2个半的数据块。是存放文件的最小空间。
- 微软操作系统（DOS、WINDOWS等）中磁盘文件存储管理的最小单位叫做 6#x7C07;
- 簇(cluster)的本意就是 6#x4E00; 6#x7EC4;, 即一组扇区(一个磁道可以分割成若干大小相等的圆弧, 叫扇区)的意思。因为扇区的单位太小, 因此把它捆在一起, 组成一个更大的单位簇更方便进行灵活管理
- 在分区中划出一片用于存放文件分配表, 目录表等用户文件管理的磁盘空间。
 - ID
 - 修改时间
 - 权限
 - 数据块位置
- 格式化会清空数据

表1 FAT32短文件目录项32个字节的表示定义			
字节偏移(16进制)	字节数	定义	
0x0~0x7	8	文件名	
0x8~0xA	3	扩展名	
0xB*	1	属性字节	00000000(读写) 00000001(只读) 00000010(隐藏) 00000100(系统) 00001000(卷标) 00010000(子目录) 00100000(归档)
0xC	1	系统保留	
0xD	1	创建时间的10毫秒位	
0xE~0xF	2	文件创建时间	
0x10~0x11	2	文件创建日期	
0x12~0x13	2	文件最后访问日期	
0x14~0x15	2	文件起始簇号的高16位	
0x16~0x17	2	文件的最近修改时间	
0x18~0x19	2	文件的最近修改日期	
0x1A~0x1B	2	文件起始簇号的低16	
0x1C~0x1F	4	表示文件的长度	

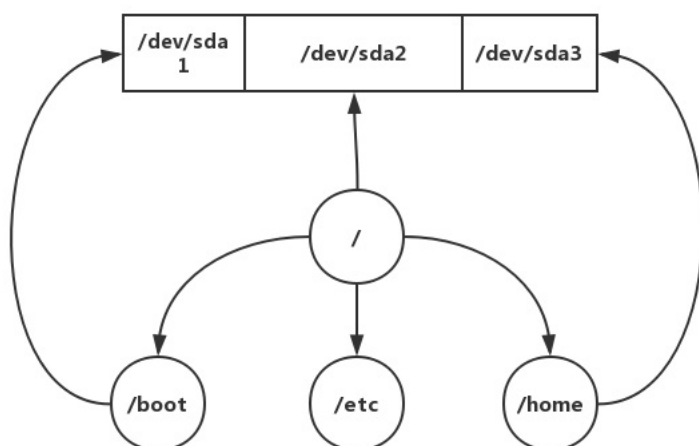
*此字段在短文件目录项中不可取值 0FH,如果设值为 0FH, 目录段为长文件名目录段

9. 挂载点

- 为了让Linux系统中可以访问这些分区, 需要把这些分区挂载到对应的目录上
- 在Linux中是把目录称为 6#x6302; 6#x8F7D; 6#x70B9;
- 把目录和分区链接在一起的过程成为 6#x6302; 6#x8F7D;
- /为根目录, 必须挂载到一个分区上, 默认所有子目录都会写入这个分区
- 同一级目录下面的所有子目录可以有自己的独立存储空间
- 必须有的分区
 - / 根分区
 - swap分区(交换分区, 虚拟内存, 一般为内存的2倍, 不要超过2G)
- 推荐分区
 - /boot (启动分区,200M) 单独分区, 避免分区写满造成系统无法启动

9.1 挂载示例

- /dev/sd2挂载到了 /目录上,也就是说向 /目录下在写文件就是往 /dev/sd2分区里写文件
- /dev/sd1挂载到了 /boot目录上,也就是说向 /boot目录下在写文件就是往 /dev/sd1分区里写文件
- /dev/sd3挂载到了 //home目录上,也就是说向 //home目录下在写文件就是往 /dev/sd3分区里写文件



10. 虚拟机使用

10.1 新建虚拟机

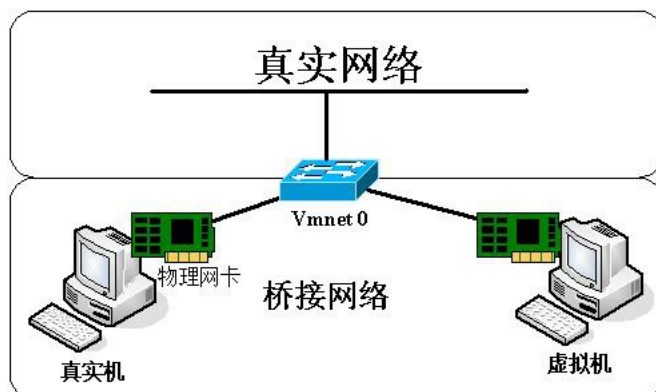
- 1. Create a New Virtual Machine开始新建虚拟机向导
- 1. 我以后再安装操作系统
- 1. Linux CentOS 32位
- 1. 20G硬盘

10.2 网络链接

- VMWare提供了三种工作模式，它们是 bridged(桥接模式)、NAT(网络地址转换模式)和 host-only(主机模式)

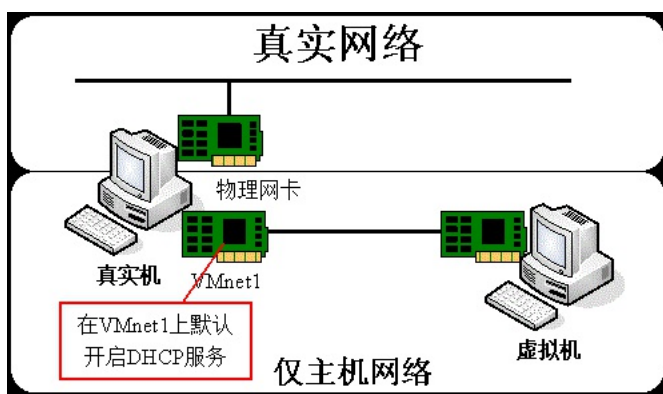
10.2.1 bridged(桥接模式)

- 在这种模式下，VMWare虚拟出来的操作系统就像是局域网中的一台独立的主机，它可以访问网内任何一台机器。
- 在桥接模式下，你需要手工为虚拟系统配置IP地址、子网掩码，而且还要和宿主机器处于同一网段，这样虚拟系统才能和宿主机进行通信
- 如果你想利用VMWare在局域网内新建一个虚拟服务器，为局域网用户提供网络服务，就应该选择桥接模式
- bridged模式下的 VMnet0虚拟网络不提供DHCP服务
- vmnet0，实际上就是一个虚拟的网桥，这个网桥有若干个端口，一个端口用于连接你的Host，一个端口用于连接你的虚拟机，他们的位置是对等的，谁也不是谁的网关



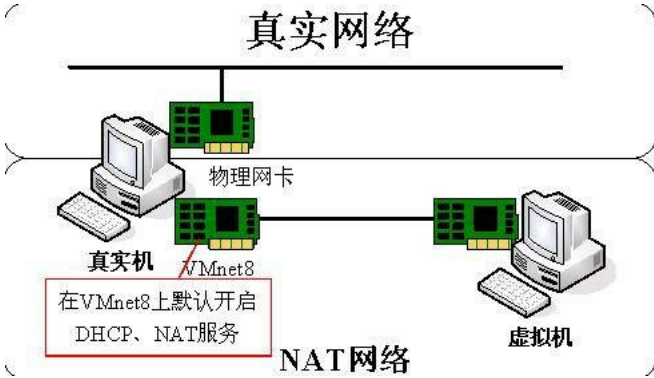
10.2.2 host-only(主机模式)

- 所有的虚拟系统是可以相互通信的，但虚拟系统和真实的网络是被隔离开的
- 虚拟系统和宿主机系统是可以相互通信的
- 虚拟系统的TCP/IP配置信息(如IP地址、网关地址、DNS服务器等)，都是由VMnet1(host-only)虚拟网络的DHCP服务器来动态分配的,IP地址是随机生成的



10.2.3 NAT(网络地址转换模式)

- 使用NAT模式，就是让虚拟系统借助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
- 使用NAT模式可以在虚拟系统里访问互联网。NAT模式下的虚拟系统的TCP/IP配置信息是由VMnet8(NAT)虚拟网络的DHCP服务器提供的，无法进行手工修改
- 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工作站，其他工作站不能访问虚拟机



10.3 使用快照

- 可以使用快照
- 在合适的时间恢复快照

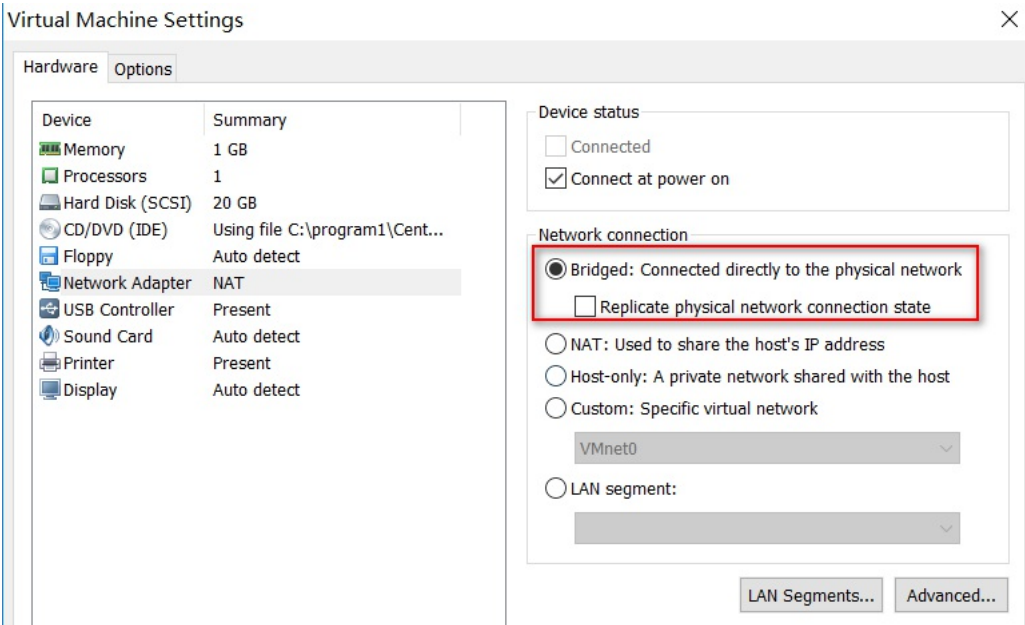
10.4 克隆

从当前的虚拟机克隆出一个虚拟机

- 可以克隆当前或者快照
- 克隆方式可以选择链接克隆或者完整克隆

11. linux系统安装

11.1 设置硬件环境



11.2 设置安装类型

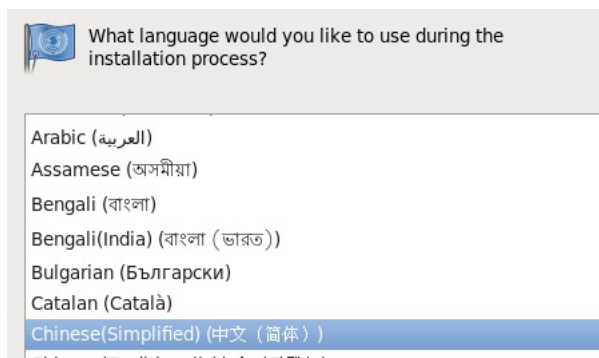
- Install or upgrade an existing system 安装过程采用基本的显卡驱动
- Install system with basic video driver 安装过程采用基本的显卡驱动
- Rescue installed system 进入系统修复模式
- Boot from local drive 退出安装从硬盘启动
- Memory test 存储介质检测

11.3 跳过检查和警告

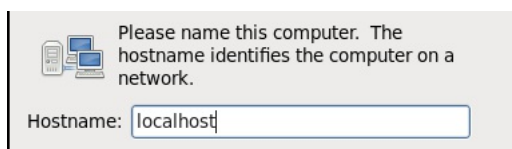




11.4 选择语言



11.5 主机名




11.6 设置主机名和网络



11.7 设置密码 #

- 复杂性
 - 八位字符以上、大小写字母、数字、符号
 - 不能是英文单词
 - 不能是和用户相关的内容
- 易记忆性
- 实效性




根帐号被用来管理系统。请为根用户输入一个密码。

根密码 (P) :


确认 (C) :

11.8 使用存储空间 #


您要进行哪种类型的安装？

☐ 


使用所有空间
删除所选设备中的所有分区。其中包含其它操作系统创建的分区。
提示：这个选项将删除所选设备中的所有数据。确定您进行了备份。

☐ 


替换现有 Linux 系统
只删除 Linux 分区（由之前的 Linux 安装创建的）。这样就不会删除您存储设备中的其它分区（比如 VFAT 或者 FAT32）。
提示：这个选项将删除您所选设备中的所有数据。确定您进行了备份。

☐ 

缩小现有系统
缩小现有分区以便为默认布局生成剩余空间。

☐ 

使用剩余空间
保留您的现有数据和分区且只使用所选设备中的未分区空间，假设您有足够的空间可用。


☒ 

创建自定义布局
使用分区工具手动在所选设备中创建自定义布局。

11.9 分区 #

请选择源驱动器

设备	大小 (MB)	挂载点/ RAID/卷	类型	格式
▼ 硬盘驱动器				
▼ sda (/dev/sda)				
sda1	200	/boot	ext4	✓
sda2	2000	/home	ext4	✓
sda3	1000		swap	✓
▼ sda4	17279		扩展分区	
sda5	17278	/	ext4	✓



已选择要格式化以下已存在的设备，破坏所有数据。

/dev/sda partition table (MSDOS)

取消(C)

格式化(F)

☒ 在 /dev/sda 中安装引导装载程序 (I)。

更换设备 (C)

☐ 使用引导装载程序密码(U)

改变密码(p)

引导装载程序操作系统列表

默认	标签	设备
<input checked="" type="radio"/>	CentOS	/dev/sda5

11.10 服务器的类型 #

- Desktop (桌面)
- Minimal Desktop (最小化桌面)
- Minimal (最小化)
- Basic Server (基本服务器,推荐)
- Database Server (数据库服务器)
- Web Server (网页服务器)
- Virtual Host (虚拟主机)
- software development workstation(软件开发工作站)

CentOS 默认安装是最小安装。您现在可以选择一些另外的软件。

☐ Desktop

☐ Minimal Desktop

☐ Minimal

☒ Basic Server

☐ Database Server

☐ Web Server

☐ Virtual Host

☐ Software Development Workstation

请选择您的软件安装所需要的存储库。

☒ CentOS

+ (A) 添加额外的存储库

修改库 (M)

或者.

☒ 以后自定义 (I)

☐ 现在自定义 (C)

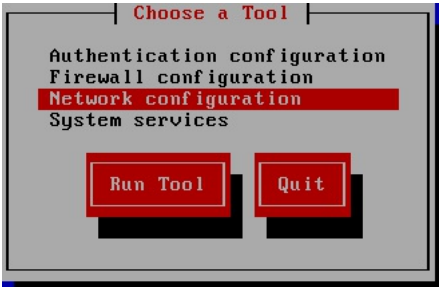
11.11 网络配置 #

11.11.1 安装日志 #

- /root/install.log 存储了安装在系统中的软件包及其版本信息
- /root/install.log.syslog 存储了安装过程中留下的事件记录
- /root/anaconda-ks.cfg 记录了安装过程中设置的选项信息，可以做为安装的模版文件

11.11.2 setup #

- 网络配置





11.11.3 ifcfg-eth0

cat /etc/sysconfig/network-scripts/ifcfg-eth0

参数 含义 TYPE=Ethernet #网卡类型 DEVICE=eth0 #网卡接口名称 ONBOOT=yes #系统启动时是否自动加载 BOOTPROTO=static #启用地址协议 -static:静态协议 -bootp协议 -dhcp协议 IPADDR=192.168.1.11 #网卡IP地址 NETMASK=255.255.255.0 #网卡网络地址 GATEWAY=192.168.1.1 #网卡网关地址 DNS1=10.203.104.41 #网卡DNS地址 HWADDR=00:0C:29:13:5D:74 #网卡设备MAC地址 BROADCAST=192.168.1.255 #网卡广播地址 NM_CONTROLLED=yes Network manager服务

11.11.4 网卡接口关闭与激活

```
ifdown eth0      #关闭网络
ifup eth0        #启动网络
```

11.11.5 网络服务启动与关闭

```
service network restart #重启网络服务
```

12.linux常用命令

12.1 常见目录

- / 根目录
- /boot 启动目录，启动相关文件
- /dev 设备文件
- /etc 配置文件
- /home 普通用户的家目录,可以操作
- /lib 系统库保存目录
- /mnt 移动设备挂载目录
- /media 光盘挂载目录
- /misc 磁带机挂载目录
- /root 超级用户的家目录,可以操作
- /tmp 临时目录,可以操作
- /proc 正在运行的内核信息映射, 主要输出进程信息、内存资源信息和磁盘分区信息等等
- /sys 硬件设备的驱动程序信息
- /var 变量
- /bin 普通的基本命令，如ls,chmod等,一般的用户也都可以使用
- /sbin 基本的系统命令，如shutdown, reboot, 用于启动系统，修复系统,只有管理员才可以运行
- /usr/bin 是在后期安装的一些软件的运行脚本
- /usr/sbin 置一些用户安装的系统管理的必备程序

12.2 命令基本格式

12.2.1 命令提示符

```
[root@zhangrenyang ~]
```

- root 当前登录用户
- localhost 主机名
- ~ 当前工作目录,默认是当前用户的家目录，root就是/root,普通用户是 /home/用户名
- 提示符 超级用户是 #,普通用户是 \$

12.2.2 命令格式

- 命令 [选项] [参数]
- 当有多个选项时，可以写在一起
- 一般参数有简化和完整写法两种 -a 与 --all等效

12.2.3 ls

- 查询目录中的内容
- ls [选项] [文件或者目录]
- 选项
 - -a 显示所有文件，包括隐藏文件
 - -l 显示详细信息
 - -d 查看目录本身的属性而非子文件 ls /etc/
 - -h 人性化的方式显示文件大小
 - -i 显示inode,也就是i节点，每个节点都有ID号
- 默认当前目录下的文件列表

** 12.2.3.1 -l #**

显示详细信息

```
drwxr-xr-x . 1 root root 800 Sep 16 00:19 logs
```

drwxr-xr-x . 1 root root 800 Sep 16 00:19 logs 文件类型和权限 ACL权限 硬链接引用计数 所有者 所属组 文件大小 最后修改时间 文件名

12.3 文件处理命令

12.3.1 mkdir

- 建立目录 make directory
- mkdir -p [目录名]
 - -p 递归创建

12.3.2 cd

- 切换所在目录 `change directory`
- `cd` [目录]
 - `~` 家目录
 - 家目录
 - `-` 上次目录
 - `.` 当前目录
 - `..` 上级目录
- 相对路径是参照当前所在目录
- 绝对路径是从根目录开始
- 按TAB键可以补全命令和目录

12.3.3 `pwd` <#>

- 显示当前目录 `pwd`

12.3.4 `rmdir` <#>

- 删除目录 `remove empty directory`
- `rmdir` [目录名]

12.3.5 `rm` <#>

- 删除文件或者目录 `remove`
- `rm` [文件或者目录]
 - `-r` 删除目录
 - `-f` 强制删除
- `rm -rf` 文件或者目录] 递归强制删除所有目录

12.3.6 `cp` <#>

- `copy` 复制命令
- `copy` [源文件或者目录] [目标文件]
 - `-r` 复制目录,默认是复制文件
 - `-p` 连带文件属性复制
 - `-d` 若源文件是链接文件,则复制连接属性
 - `-a` 相当于 `-rpd`

12.3.7 `mv` <#>

- 移动文件或者改名 `move`
- `mv` [源文件或者目录] [目标文件]

12.3.8 `ln` <#>

- 链接命令,生成链接文件 `link`

** 12.3.8.1 硬链接特征 <#> **

- 拥有相同的节点和存储block块,可以看作是同一个文件
- 可以通过节点访问
- 不能跨分区
- 不能针对目录使用
- 一般不使用

** 12.3.8.2 软链接特征 <#> **

- `ln -s` [源文件] [目标文件]
 - `-s` 创建软链接
- 类似Windows快捷方式
- 软链接拥有自己的节点和Block块,但是数据块中只保存源文件的文件名和节点号,并没有实际的文件数据
- `lwxrwxrwx` | 软链接 软链接的文件权限都是 `777`
- 修改任意一个文件,另一个都会改变
- 删除源文件,软链接不能使用
- 软链接源文件必须写绝对路径

12.4 文件搜索命令 <#>

12.4.1 `locate` <#>

- 在后台数据库中按文件名搜索,速度比较快
- 数据保存在 `/var/lib/mlocate` 后台数据库,每天更新一次
- 可以 `updatedb` 命令立刻更新数据库
- 只能搜索文件名
- `/etc/updatedb.conf` 建立索引的配置文件
 - `PRUNE_BIND_MOUNTS = "yes"` 全部生效,开启搜索限制
 - `PRUNEFS` 不搜索的文件系统
 - `PRUNENAMES` 忽略的文件类型
 - `PRUNEPATHS` 忽略的路径 `/tmp`

12.4.2 `whereis` <#>

- 搜索命令所在路径以及帮助文档所在位置
- `whereis` 命令名 `whereis ls`
 - `-b` 只查找可执行文件
 - `-m` 只查找帮助文件

12.4.3 `which` <#>

- 可以看到别名 `which ls`
- 能看到的都是外部安装的命令
- 无法查看Shell自带的命令,如 `which cd`

12.4.4 环境变量 <#>

```
/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin
```

- 定义的是系统搜索命令的路径
- `echo $PATH`

12.4.5 find #

- 文件搜索命令
- find [搜索范围] [搜索条件]

** 12.4.5.1 按名称搜索 #**

- 避免大范围的搜索，会非常消耗系统资源

```
find / -name aaa.log
```

** 12.4.5.2 通配符 #**

- find是在系统当中搜索符合条件的文件名，如果需要匹配，使用通配符匹配，通配符是完全匹配
- 通配符
 - * 匹配任意内容
 - ? 匹配任意一个字符
 - [] 匹配任意一个中括号内的字符

```
find . -name "ab[cd ef]"
```

** 12.4.5.3 -i #**

不区分大小写

```
find / -iname A.log
```

** 12.4.5.4 -user #**

按所有者进行搜索

```
find /root -user root
find /root -nouser
```

** 12.4.5.5 按时间搜索 #**

```
find /nginx/access.log -mtime +5
```

参数 含义 **atime** 文件访问时间 **ctime** 改变文件属性 **mtime** 修改文件内容 参数 含义 **-5** 5天内修改的文件 **5** 5天前当前修改的文件 **+5** 5天前修改的文件

** 12.4.5.6 按大小搜索 #**

- k小写,M大写

```
find . -size 100k
```

参数 含义 **-8k** 小于8K **8k** 等于8K **+8k** 大于8K **+8M** 小于8M

** 12.4.5.7 按节点搜索 #**

```
find . -inum 123456
```

** 12.4.5.8 综合应用 #**

```
find /tmp -size +10k -a -size -20k
```

- 查找/etc目录下，大于10KB并且小于20KB的文件
- **-a** and 逻辑与，两个条件都满足
- **-o** or 逻辑或，两个条件满足一个就可以

```
find /tmp -size +10k -a -size ~20k -exec ls -lh {} \;
```

- **exec** 对上个命令的结果进行操作

** 12.4.5.9 grep #**

- 在文件当中匹配符合条件的字符串
- **grep "10" access.log**
 - **-i** 忽略大小写
 - **-v** 排除指定字符串
- **find**命令，在系统当中搜索符合条件的文件名，如果需要匹配，使用通配符匹配，通配符是完全匹配
- **grep**命令 在文件当中搜索符合条件的字符串，如果需要匹配，使用正则表达式进行匹配，正则表达式时包含匹配

12.5 帮助命令 #

12.5.1 基本用法 #

- **man** 命令 获取指定命令的帮助
- **man ls** 查看ls的帮助

12.5.2 man的级别 #

- 1 查看命令的帮助
- 2 查看可被内核调用的函数的帮助
- 3 查看函数和函数库的帮助
- 4 查看特殊文件的帮助
- 5 查看配置文件的帮助
- 6 查看游戏的帮助
- 7 查看其它的帮助
- 8 查看系统管理员可用命令的帮助
- 9 查看和内核相关文件的帮助

12.5.3 查看命令级别 #

- 查看命令级别
- **1p**: POSIX utilities
- POSIX表示可移植操作系统接口（Portable Operating System Interface of UNIX，缩写为 POSIX），POSIX标准定义了操作系统应该为应用程序提供的接口标准。

```
man -f ls
what is ls
man 1 ls
man 1p ls
```

12.5.4 关键字搜索 #

```
- man -k passwd
```

12.5.5 shell 内部帮助 #

- **whereis**找到就是外部，找不到就是内部

```
help cd
```

12.6 压缩与解压缩命令 #

.zip .gz .bz2 .tar.gz .tar.bz2

12.6.1 zip格式 #

压缩文件或目录,是一种压缩格式

- 压缩文件 zip 6#x538B;6#x7F29;6#x6587;6#x4EF6;6#x540D;.zip 6#x6E90;6#x6587;6#x4EF6;
- 压缩目录 zip -r 6#x538B;6#x7F29;6#x76EE;6#x5F55;6#x540D;.zip 6#x6E90;6#x76EE;6#x5F55;
- 解压 unzip 6#x538B;6#x7F29;6#x76EE;6#x5F55;6#x540D;.zip

```
mkdir book
touch book/1.txt
touch book/2.txt
zip -r book.zip book
unzip book.zip
```

12.6.2 gzip #

gzip为高压，可以把文件压缩得更小

命令 示例 含义 gzip 源文件 gzip a.txt 压缩为.gz格式的压缩文件，源文件会消失 gzip -c 源文件 > 压缩文件 gzip -c yum.txt > yum.txt.gz 压缩为.gz格式的压缩文件，源文件不会消失 gzip -r 目录 gzip -r xx 把目录下的每个子文件都变成压缩包，并删除原文件，当前目录无变化 gzip -d 压缩文件名 gzip -d yum.txt.gz 解压缩文件,不保留压缩包 gunzip 压缩文件 gunzip yum.txt.gz 解压缩文件,也不保留压缩包

- 压缩是压缩目录下的文件

12.6.3 .bz2格式压缩 #

bzip2是一个压缩能力更强的压缩程序

命令 示例 含义 bzip2 源文件 bzip2 1.txt 压缩为.bz2格式的文件，不保留源文件 bzip2 -k 源文件 bzip2 -k 1.txt 压缩为.bz2格式的文件，保留源文件 bzip2 -d 压缩文件名 bzip2 -d 1.txt.bz2 解压缩压缩包,不保留压缩包 bunzip2 压缩文件名 bunzip2 1.txt.bz2 解压缩压缩包,也不保留压缩包

- bzip2 不能压缩目录

12.6.4 tar #

- 打包命令,只打包并不压缩
- tar -cvf 打包文件名 源文件
 - -c 打包
 - -v 显示过程
 - -f 指定打包后的文件名

```
tar -cvf book.tar book 会找出一个book.tar文件
```

- x 解打包

```
tar -xvf book.tar
```

12.6.4 tar.gz压缩格式 #

- zip可以压缩目录但压缩效率不高，gzip和bzip2压缩效率高但不支持目录
- 可以先打包为 .tar格式，再压缩为 .gz格式 -z 压缩为.tar.gz格式 -x 解压缩.tar.gz格式

命令 示例 含义 tar -zcvf 压缩包名 .tar.gz

源文件 tar -zcvf book.gargz book可以先打包为 .tar

格式，再压缩为 .gz

格式 tar -zxvf 压缩包名.tar.gz tar -zxvf book.tar.gz 解压tar.gz压缩包 tar -jcvf 压缩包名 .tar.bz2

源文件 tar -jcvf book.tar.bz2 book可以先打包为 .tar

格式，再压缩为 .bz2

格式 tar -jxvf 压缩包名.tar.bz2 tar -jxvf book.tar.bz2 解压tar.bz2压缩包

12.7 关机和重启命令 #

12.7.1 shutdown #

shutdown 关机命令

- -c 取消前一个关机命令
- -h 关机
- -r 重启

```
shutdown -r 06:00
shutdown -c
```

12.7.2 init #

关机

```
init 0
```

重启

```
init 6
```

12.7.3 logout #

退出登录

```
logout
```

12.8 查看登录用户信息 #

12.8.1 w #

查看登录用户信息

- USER 登录的用户名
- TTY 登录的终端 tty1 本地终端 pts/0远程终端
- FROM 登录的IP

- LOGIN 登录时间
- IDLE 用户闲置时间
- JCPU 该终端所有进程占用的时间
- PCPU 当前进程所占用的时间
- WHAT 正在执行的命令

12.8.2 who

查看登录用户信息

- USER 登录的用户名
- TTY 登录的终端 tty1 本地终端 pts/0 远程终端
- LOGIN 登录时间（登录的IP）

12.8.3 last

查看当前登录和过去登录的用户信息 默认读取 /var/log/wtmp 文件

- 用户名
- 登录终端
- 登录IP
- 登录时间
- 退出时间(在线时间)

12.8.4 lastlog

查看所有用户的最后一次登录时间

- 用户名
- 登录终端
- 登录IP
- 最后一次登录时间

12.9 磁盘管理

12.9.1 df

- 查看磁盘分区使用情况

参数 描述 -l 仅显示本地磁盘(默认) -a 显示所有文件系统的使用情况 -h 以1024进制计算最合适的单位显示磁盘容量 -H 以1000进制计算最合适的单位显示磁盘容量 -T 显示磁盘分区类型 -t 显示指定类型文件系统的磁盘分区 -x 不显示指定类型文件系统的磁盘分区

12.9.2 du

- 统计以磁盘上的文件大小

参数 描述 -b 以byte为单位统计文件 -k 以KB为单位统计文件 -m 以MB为单位统计文件 -h 以1024为单位统计文件 -H 以1000为单位统计文件 -s 指定统计目标

```
du -s /etc
du -sH /etc
```

12.9.3 添加新硬盘后的分区和格式化

- 硬件设备是由linux系统自动识别并以文件的形式存在于根目录下的 dev目录下
- 1-4分区编号是留给主分区和扩展分区的,逻辑分区只能从5开始

命令 含义 m print this menu 打印菜单 n add a new partition 添加一个分区 d delete a partition 删除一个分区 p print the partition table 打印分区表 q quit without saving changes 退出不保存 w write table to disk and exit 写入分区表并保存

```
fdisk -l
Disk /dev/sda: 21.5 GB, 21474836480 bytes
Device      Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1           26       204800    83  Linux

Disk /dev/sdb: 8589 MB, 8589934592 bytes
fdisk /dev/sdb 开始对这块硬盘进行分区
m 打印命令
n 创建一个分区
Partition number (1-4): 1 选择分区编号
First cylinder (1-1044, default 1): 1 输入开始扇区
Last cylinder, +cylinders or +size(K,M,G) (1-1044, default 1044): +3000M 输入结束扇区
p 查看当前分区
n 创建分区
e 扩展分区
l 创建逻辑分区
d 删除分区
w 分区表写入磁盘
```

12.9.4 GPT

- MBR下主分区最多4个, GPT可达128个
- MBR下主分区容量最大2TB, GPT模式下容量可达18EB(1EB=1024PB,1PB=1024TB,1TB=1024GB)

```
parted 开始分区,默认是对第一块硬盘分区
mklabel gpt 指定分区表的类型为gpt
print 查看分区表的类型
mkpart 开始分区
分区名称? []? system
文件系统类型? [ext2]?

起始点? 0
结束点? 2000
警告: The resulting partition is not properly aligned for best performance.

忽略/Ignore/放弃/Cancel? c
(parted) 1
结束点? 2000
mkpart 2th 2000 3000
quit 退出编辑
```

12.9.5 格式化

```
ls -l /dev/sdb*
mkfs.ext3 /dev/sdb1
mkfs -t ext4 /dev/sdb2
```

12.9.6 挂载 #

```
mkdir /mnt/zhufeng
mount /dev/sdb1 /mnt/zhufeng
umount /mnt/zhufeng
vim + /etc/fstab

/dev/sdb1 /mnt/zhufeng ext3 defaults 0 0
分区名称 挂载点 文件系统类型
```

12.9.7 添加swap交换分区 #

- 建立普通的linux分区
- 修改分区类型的16进制编码
- 格式化成交换分类
- 启动交换分区

```
fdisk /dev/sdb
p 查看当前的分区
Command (m for help): t 修改分区的系统ID
Partition number (1-4): 3 修改分区编号为3的分区
Hex code (type L to list codes): L 列出所有编号
Hex code (type L to list codes): 82 把编号修改为16进制的82
Changed system type of partition 3 to 82 (Linux swap / Solaris)
p
/dev/sdb3 767 1044 2233035 82 Linux swap / Solaris

free 查看剩余内存
mkswap /dev/sdb3 把sdb3设置为交换分区
swapon /dev/sdb3 挂载sdb3成交换区
free
swapoff /dev/sdb3
```

12.9.8 挂载 #

** 12.9.8.1 挂载命令格式 #**

- mount [-t 文件系统] [-o 特殊选项] 设备文件名 挂载点
- 选项
 - -t 文件系统 ext4 iso9660
 - -o 特殊选项

** 12.9.8.2 挂载光驱 #**

```
mkdir /mnt/cd
mount -t iso9660 /dev/sr0 /mnt/cdrom
```

** 12.9.8.3 卸载光驱 #**

```
umount /mnt/cdrom
```

** 12.9.8.4 挂载U盘 #**

```
fdisk -l 查看硬盘及分区信息
mount -t vfat /dev/sdb1 /mnt/usb
```

- linux默认不支持NTFS格式

12.10 文件查看命令 #

12.10.1 cat #

- cat 命令用于连接文件并打印到标准输出设备上。
- cat [-AbeEnSTuv] [-help] [--version] fileName
- 参数
 - -n 或 --number: 由 1 开始对所有输出的行数编号。

```
cat -n textfile1
```

12.10.2 more #

- Linux more 命令类似 cat，不过会以一页一页的形式显示，更方便使用者逐页阅读，而最基本的指令就是按空白键（space）就往下一页显示，按 b 键就会往回（back）一页显示，而且还有搜寻字串的功能（与 vi 相似），使用中的说明文件，请按 h。
- more fileName

```
more testfile
.
```

12.10.3 head #

- 用来显示开头某个数量的文字区块

```
head -5 readme.txt
```

12.10.4 tail #

- tail 命令可用于查看文件的内容
- 有一个常用的参数 -f 常用于查阅正在改变的日志文件。
- tail [参数] [文件]
- 参数
 - -f 循环读取
 - -n

12.10.5 第二页 #

```
head -10 file | tail -5
```