**Linux\_cloud\_Linuxbase\_day01笔记**

**= linux发展历史的详细故事 =**

1. **简单的说：**

一个名叫Linus Torvalds 的芬兰大学生想要了解Intel的新CPU386。他认为比较好的学习方法是自己编写一个操作系统的内核。出于这种目的，加上他对当时Unix 变种版本（即Minix）对于80386类机器的脆弱支持十分不满，他决定要开发出一个全功能的、支持POSIX标准的、类Unix的操作系统内核，该系统吸收了BSD和System V的优点，同时摒弃了它们的缺点。Linus独立把这个内核开发到0.02 版，这个版本已经可以运行gcc、bash 和很少的一些应用程序。这些就是他开始的全部工作了。后来，他又开始在因特网上寻求广泛的帮助。   
  
**二、具体的分析：** Linux 操作系统的诞生、发展和成长过程依赖于以下五个重要支柱：

  UNIX 操作系统-->MINIX操作系统-->GNU 计划-->POSIX 标准-->nternet 网络-->UNIX操作系统

  Linux 操作系统是UNIX 操作系统的一个克隆版本。UNIX 操作系统是美国贝尔实验室的Ken.Thompson和Dennis Ritchie 于1969 年夏在DEC PDP-7 小型计算机上开发的一个分时操作系统。Ken Thompson 为了能在闲置不用的PDP-7 计算机上运行他非常喜欢的星际旅行（Space travel）游戏，于是在1969 年夏天乘他夫人回家乡加利福尼亚渡假期间，在一个月内开发出了UNIX 操作系统的原型。当时使用的是BCPL 语言（基本组合编程语言），后经Dennis Ritchie 于1972 年用移植性很强的C语言进行了改写，使得UNIX 系统在大专院校得到了推广。

MINIX操作系统

  MINIX 系统是由Andrew S. Tanenbaum（AST）开发的。AST 是在荷兰Amsterdam 的Vrije 大学数学与计算机科学系统工作，是ACM 和IEEE 的资深会员(全世界也只有很少人是两会的资深会员)。MINIX 是他1987 年编制的，主要用于学生学习操作系统原理。到1991 年时版本是1.5。目前主要有两个版本在使用：1.5 版和2.0 版。当然目前MINIX 系统已经是免费的，可以从许多FTP 上下载。  
  对于Linux 系统，他后来曾表示对其开发者Linus的称赞。但他认为Linux的发展很大原因是由于他为了保持MINIX 的小型化，能让学生在一个学期内就能学完，因而没有接纳全世界许多人对MINIX的扩展要求。因此在这样的前提下激发了Linus 编写Linux 系统。当然Linus 也正好抓住了这个好时机。  
  作为一个操作系统，MINIX 并不是优秀者，但它同时提供了用C 语言和汇编语言编写的系统源代码。这是第一次使得有抱负的程序员或hacker 能够阅读操作系统的源代码。在当时，这种源代码是软件商们一直小心守护着的秘密。

GNU计划

  软件产业在70年代成就了两位针锋相对的领袖人物，来自哈佛大学的比尔·盖茨和Richard M.Stallman。前者宣布了Copyright(版权)时代的到来，并构建了微软帝国的辉煌；后者于1984年创立自由软件体系GNU，拟定普遍公用版权协议（GeneralPublicLicense，简称GPL），今天Linux的成功就得益于GPL协议。  
  所有GPL协议下的自由软件都遵循着Richard M. Stallman的"Copyleft"(非版权)原则：即自由软件允许用户自由拷贝、修改和销售，但是对其源代码的任何修改都必须向所有用户公开。  
  GNU 计划和自由软件基金会FSF(the Free Software Foundation)是由Richard M. Stallman 于1984 年一手创办的。旨在开发一个类似UNIX 并且是自由软件的完整操作系统：GNU 系统（GNU 是"GNU's Not Unix"的递归缩写，它的发音为"guh-NEW"）。  
  各种使用Linux 作为核心的GNU 操作系统正在被广泛的使用。虽然这些系统通常被称作"Linux"，但是Stallman 认为，严格地说，它们应该被称为GNU/Linux系统。  
  到上世纪90 年代初，GNU 项目已经开发出许多高质量的免费软件，其中包括有名的emacs 编辑系统、bash shell 程序、gcc 系列编译程序、gdb 调试程序等等。这些软件为Linux 操作系统的开发创造了一个合适的环境。这是Linux 能够诞生的基础之一，以至于目前许多人都将Linux 操作系统称为“GNU/Linux”操作系统。

POSIX标准

  POSIX（Portable Operating System Interface for Computing Systems）是由IEEE 和ISO/IEC 开发的一簇标准。该标准是基于现有的UNIX 实践和经验，描述了操作系统的调用服务接口。用于保证编制的应用程序可以在源代码一级上在多种操作系统上移植和运行。它是在1980 年早期一个UNIX 用户组(usr/group)的早期工作基础上取得的。该UNIX 用户组原来试图将AT&T 的System V 操作系统和BerkeleyCSRG 的BSD 操作系统的调用接口之间的区别重新调和集成。并于1984 年定制出了/usr/group 标准。  
  在90 年代初，POSIX 标准的制定正处在最后投票敲定的时候，那是1991-1993 年间。此时正是Linux刚刚起步的时候，这个UNIX 标准为Linux 提供了极为重要的信息，使得Linux 能够在标准的指导下进行开发，并能够与绝大多数UNIX 操作系统兼容。在最初的Linux 内核源代码中（0.01 版、0.11 版）就已经为Linux 系统与POSIX 标准的兼容做好了准备工作。在Linux 0.01 版内核的/include/unistd.h 文件中就已经定义了几个有关POSXI 标准要求的符号常数，而且Linus 在注释中已写道：“OK，这也许是个玩笑，但我正在着手研究它呢”。

Internet 网络

  Internet是linux开发、交流、发展的信息和资源的交流平台。  
 **三、Linux的诞生**  
  在1981 年，IBM 公司推出了享誉全球的微型计算机IBM PC。在1981-1991 年间，MS-DOS 操作系统 一直是微型计算机操作系统的主宰。此时计算机硬件价格虽然逐年下降，但软件价格仍然居高不下【a】。  
  当时Apple 的MACs 操作系统可以说是性能最好的，但是其天价使得没人能够轻易靠近 【b】。  
  当时的另一个计算机技术阵营就是UNIX 世界。但是UNIX 操作系统就不仅是价格昂贵的问题 了。为了寻求高利润率，UNIX 经销商们把价格抬得极高，PC 小用户根本不能靠近它。曾经一度收到Bell Labs许可而能在大学中用于教学的UNIX 源代码也一直被小心地守卫着不许公开 【c】。  
  对于广大的PC 用户，软件行业的大型供应商们始终没有给出有效的解决这个问题的手段。  
  
  正在此时，出现了MINIX 操作系统 【d】，并且有一本描述其设计实现原理的书 同时发行。由于AST的这本书写的非常详细，并且叙述得有条有理，于是几乎全世界的计算机爱好者都开始看这本书，以期能理解操作系统的工作原理。  
  其中也包括Linux 系统的创始者Linus Benedict Torvalds。当时(1991 年)，Linus Benedict Torvalds 是赫尔辛基大学计算机科学系的二年级学生，也是一个自学的计算机hacker。这个21岁的芬兰年轻人喜欢鼓捣他的计算机，测试计算机的性能和限制。但当时他所缺乏的就是一个专业级的操作系统。  
  在同一年间，GNU 计划已经开发出了许多工具软件。其中最受期盼的GNU C 编译器已经出现，但还没有开发出免费的GNU 操作系统。 即使是教学使用的MINIX 操作系统也开始有了版权，需要购买才能得到源代码。虽然GNU 的操作系统HURD一直在开发之中【e】，但在当时看来不能在几年内完成。  
  
（已经有了【a－e】，但是还没有一个os）  
  为了能更好地学习计算机知识（或许也只是为了兴趣），Linus 使用圣诞节的压岁钱和贷款购买了一台386 兼容电脑 ，并从美国邮购了一套MINIX 系统软件 。就在等待MINIX 软件期间，Linus 认真学习了有关Intel 80386 的硬件知识。为了能通过Modem 拨号连接到学校的主机上，他使用汇编语言并利用80386 CPU的多任务特性编制出一个终端仿真程序。此后为了将自己一台老式电脑上的软件复制到新电脑上，他还为软盘驱动器、键盘等硬件设备编制出相应的驱动程序。（他编写了一些os相关的程序）  
  通过编程实践，并在学习过程中认识到MINIX 系统的诸多限制（MINIX 虽然很好，但只是一个用于教学目的简单操作系统，而不是一个强有力的实用操作系统），而且通过上述实践Linus 已经有了一些类似于操作系统硬件设备驱动程序的代码，于是他开始有了编制一个新操作系统的想法。（一个影响深远的想法诞生了）  
  从1991年4月份起（let take this time as thestart），他通过修改终端仿真程序和硬件驱动程序，开始编制起自己的操作系统来。开始，他的目的很简单，只是为了学习Intel 386 体系结构保护模式运行方式下的编程技术。但后来Linux发展却完全改变了初衷。（起始的时间）  
  根据Linus 在comp.os.minix 新闻组上发布的消息，我们可以知道他逐步从学MINIX 系统阶段发展到开发自己的Linux 系统的过程。（具体了解的来源）  
  Linus第1次向comp.os.minix 投递消息是在1991年3月29日。所发帖子的题目是“gcc on minix-386doesn't optimize”,是有关gcc 编译器在MINIX-386 系统上运行优化的问题（由此可知，Linus在1991 年初期就已经开始深入究了MINIX 系统，并在这段时间有了改进MINIX 操作系统的思想。在进一步学习MINIX 系统之后，这个想法逐步演变成想重新设计一个基于Intel 80386 体系结构的新操作系统的构思。（思想的转变）  
  他在回答有人提出MINIX 上的一个问题时，所说的第一句话就是“阅读源代码”（“RTFSC (Read the F\*\*ing Source Code :-)”）。他认为答案就在源程序中。这也说明了对于学习系统软件来说，我们不光需要懂得系统的工作基本原理，还需要结合实际系统，学习实际系统的实现方法。因为理论毕竟是理论，其中省略了许多枝节，而这些枝节问题虽然没有太多的理论含量，但却是一个系统必要的组成部分，就象麻雀身上的一根羽毛。（实践的重要意义）  
  
（以下是按照时间顺序进行介绍）  
  从1991 年4 月 份开始，Linus 几乎花费了全部时间研究MINIX-386 系统(Hacking the kernel)，并且尝试着移植GNU 的软件到该系统上(GNU gcc、bash、gdb 等)。并于4 月13 日在comp.os.minix 上发布说自己已经成功地将bash 移植到了MINIX 上，而且已经爱不释手、不能离开这个shell 软件了。  
  第一个与Linux 有关的消息是在1991 年7 月3日 在comp.os.minix 上发布的（当然，那时还不存在Linux 这个名称，当时Linus 脑子里想的名称可能是FREAX，FREAX 的英文含义是怪诞的、怪物、异想天开等）。其中透露了他正在进行Linux 系统的开发，并且已经想到要实现与POSIX 兼容的问题了。  
  在Linus 另一个发布的消息中(1991 年8 月25日 comp.os.minix)，他向所有MINIX 用户询问“Whatwould you like to see in minix?”(“你最想在MINIX系统中见到什么？”)，在该消息中他首次透露出正在开发一个(免费的)386(486)操作系统，并且说只是兴趣而已，代码不会很大，也不会象GNU 的那样专业。希望大家反馈一些对于MINIX 系统中喜欢哪些特色不喜欢什么等信息，并且说明由于实际和其它一些原因，新开发的系统刚开始MINIX 很象（并且使用了MINIX的文件系统）。并且已经成功地将bash(1.08版)和gcc(1.40 版)移植到了新系统上，而且在过几个月就可以实用了。最后，Linus 申明他开发的操作系统没有使用一行MINIX 的源代码；而且由于使用了386 的任务切换特性，所以该操作系统不好移植（没有可移植性），并且只能使用AT 硬盘。对于Linux 的移植性问题，Linus当时并没有考虑。但是目前Linux 几乎可以运行在任何一种硬件体系结构上。  
   到了1991 年的10 月5 日 ，Linus 在comp.os.minix 新闻组上发布消息，正式向外宣布Linux 内核系统的诞生（Free minix-like kernel sources for 386-AT）。这段消息可以称为Linux 的诞生宣言，并且一直广为流传。因此10 月5 日对Linux社区来说是一个特殊的日子，许多后来Linux 的新版本发布时都选择了这个日子。所以RedHat 公司选择这个日子发布它的新系统也不是偶然的。（产生的时间）

**= Linux系统安装步骤 =**

**一、虚拟机基本设备介绍**

classroom 服务站，为其他机器提供软件提供服务。

desktop 客户站，用来做实验，测试。

server 服务站，搭建服务端软件。

所有虚拟机密码 redhat 账号是root

#rht-vmctl reset classroom/desktop/server

1. Linux系统安装

**系统安装要求**

硬盘200G，内存2G，CPU 1核。

/boot 200M 启动分区，存放内核，引导文件的分区

/data 50G 自定义的一个挂载点。

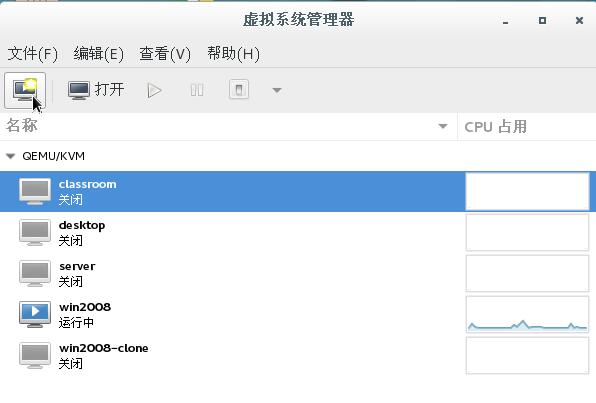
swap 4096M 虚拟交换分区。

/ 所有空间 （/ 读根），给我们最大老大分配所有的空间。

安装GUI桌面环境

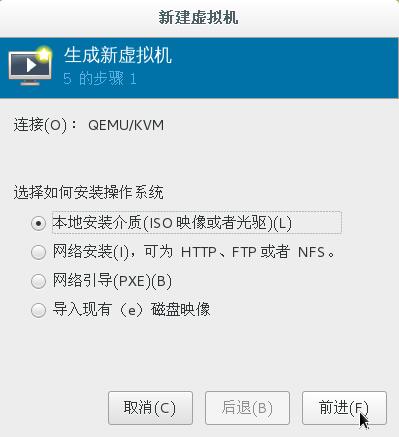
**开始安装**

1. 打开虚拟机，选择左上角“新建虚拟机”（电脑小灯泡）



创建信虚拟机

1. 四种安装源，选择本地安装介质，点击前进



1. 镜像选择，“浏览”，选择“rhel-server-7.2-x86\_64-dvd.iso”镜像，点击前进



1. 确定分配内存2G，CPU 1核，点击前进



1. 确定硬盘200G，点击前进后正式进入安装图形界面

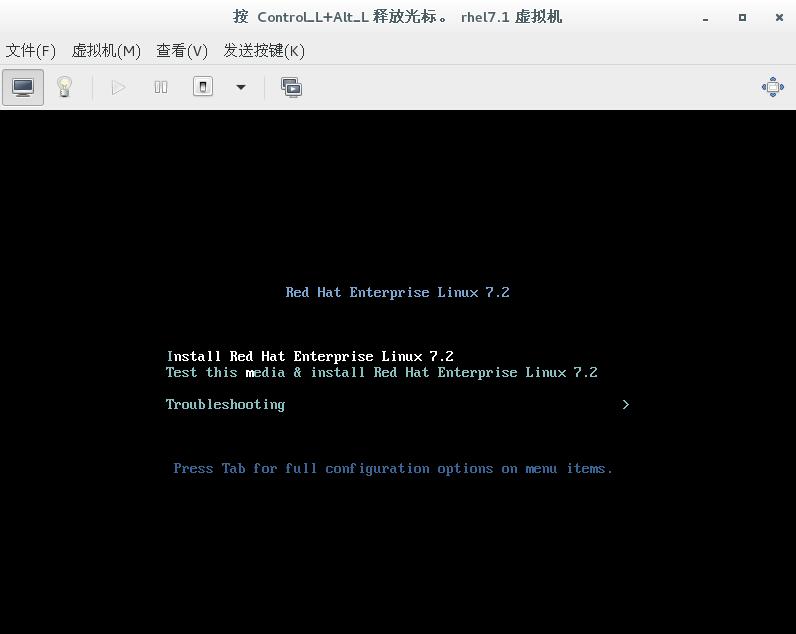


1. 确定配置，点击“完成”



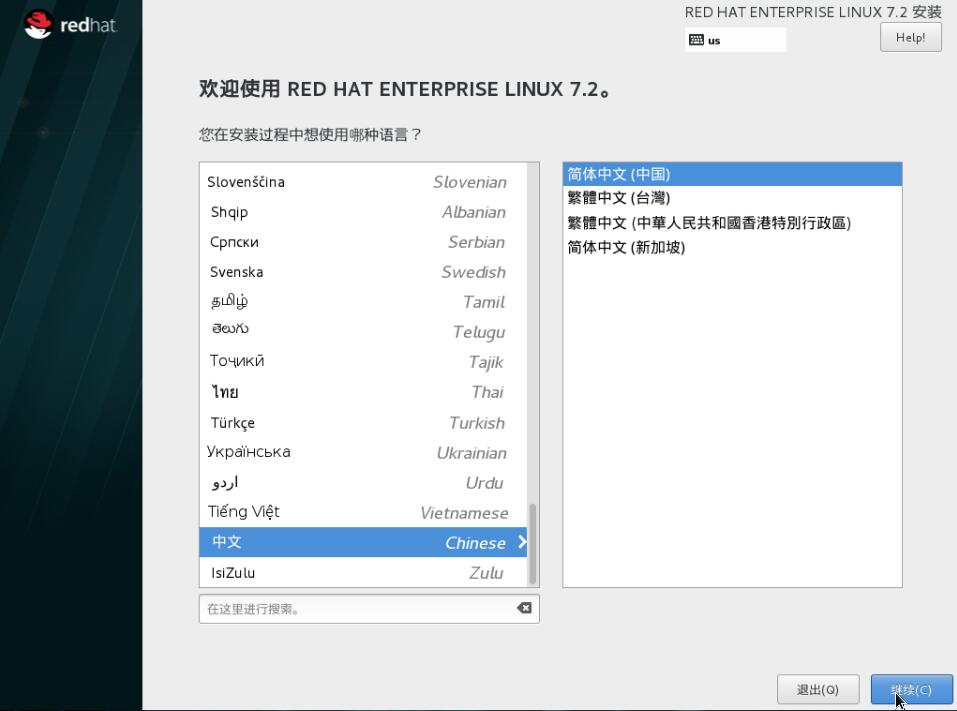
1. 进入到安装界面，选择第一个

|  |  |
| --- | --- |
| |  | | --- | |  | |



1） 第一行直接进入安装系统；   
2） 第二行是要对我们的安装源进行检测后，确保没有问题后进入安装界面（不建议选择，因为太过消耗时间，而且iso一般情况下没有问题）；   
3） “Troubleshooting”是进行故障排错(当系统重要文件被删除影响系统运行时可以通过此方法进行系统挽救)。

1. 进行语言的选择（系统语言和键盘语言），拉到最下面，选择中文，点击继续



1. 进入安装信息摘要界面



**本地化**

时间：上海

键盘：汉语

语言支持：简体中文（中国）

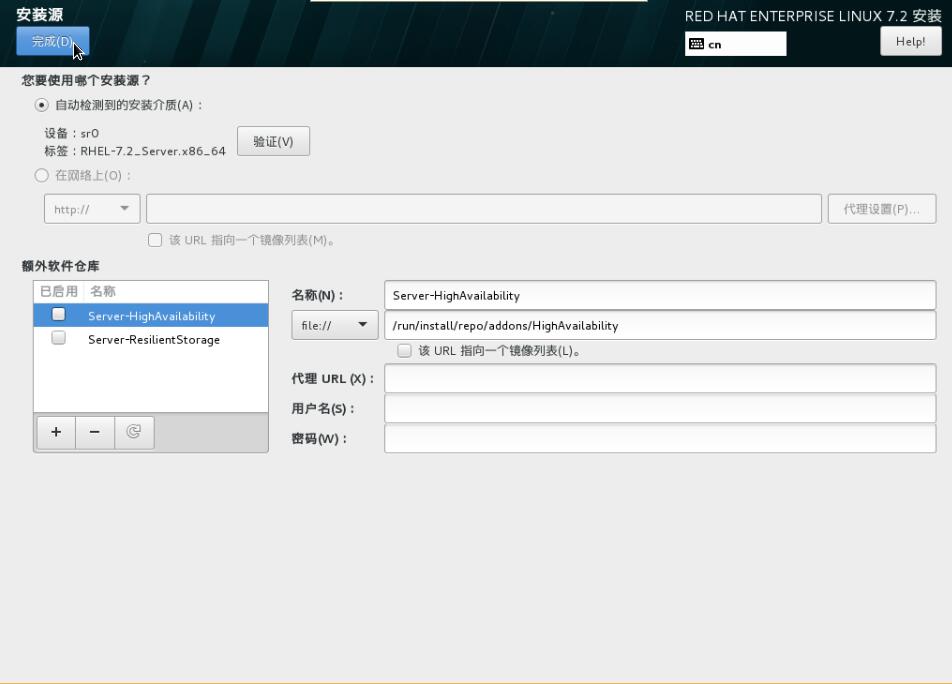
**软件**

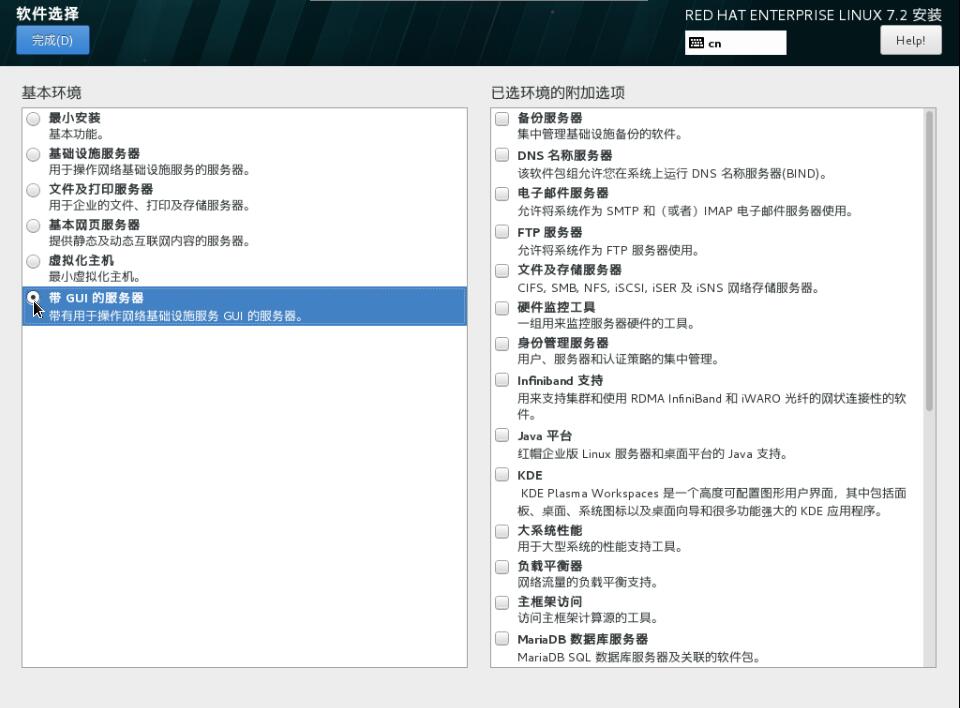
安装源：本地介质

设备:sr0

标签：rhel-server-7.2-x86\_64

软件选择：最小安装--基本环境：带GUI的服务器



****

**系统**

安装位置：

其他存储选项-我要配置分区-完成

手动分区-“点击这里自动创建他们”-

/boot 200M

swap 4096M

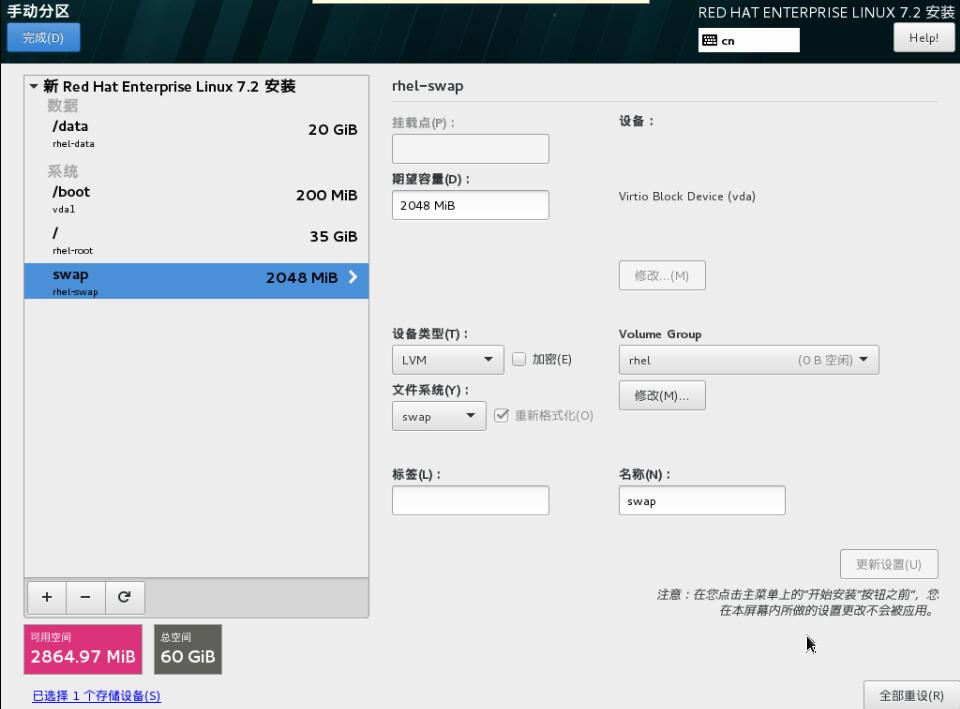
/data 50G（添加新挂载点）

/ 所有剩余空间

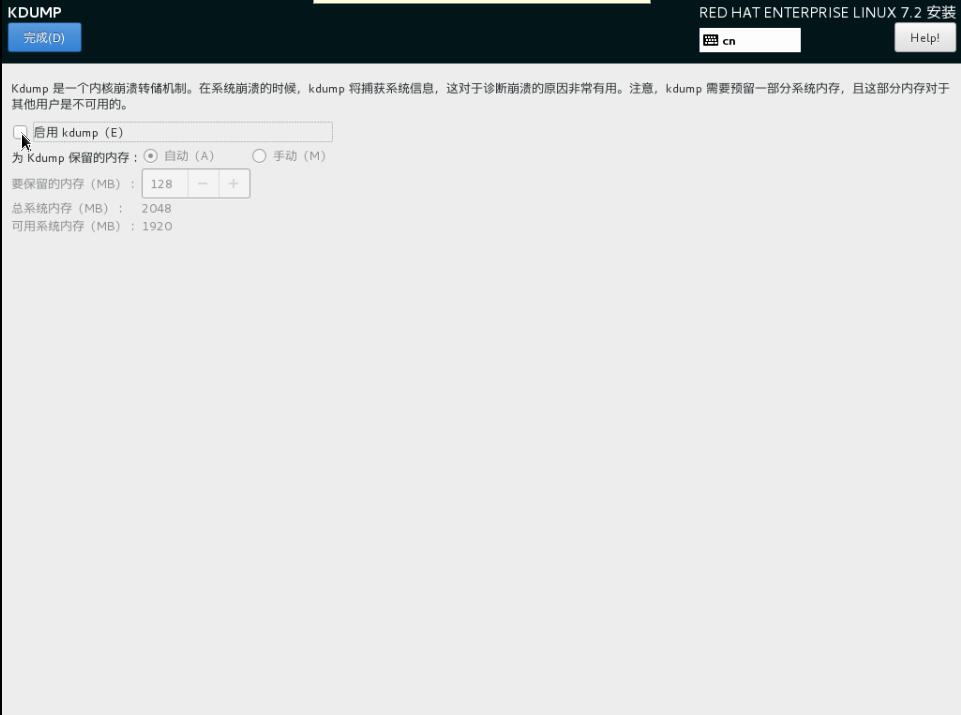
完成-更改摘要-接受更改

kdump：一般不启用（占内存）



****

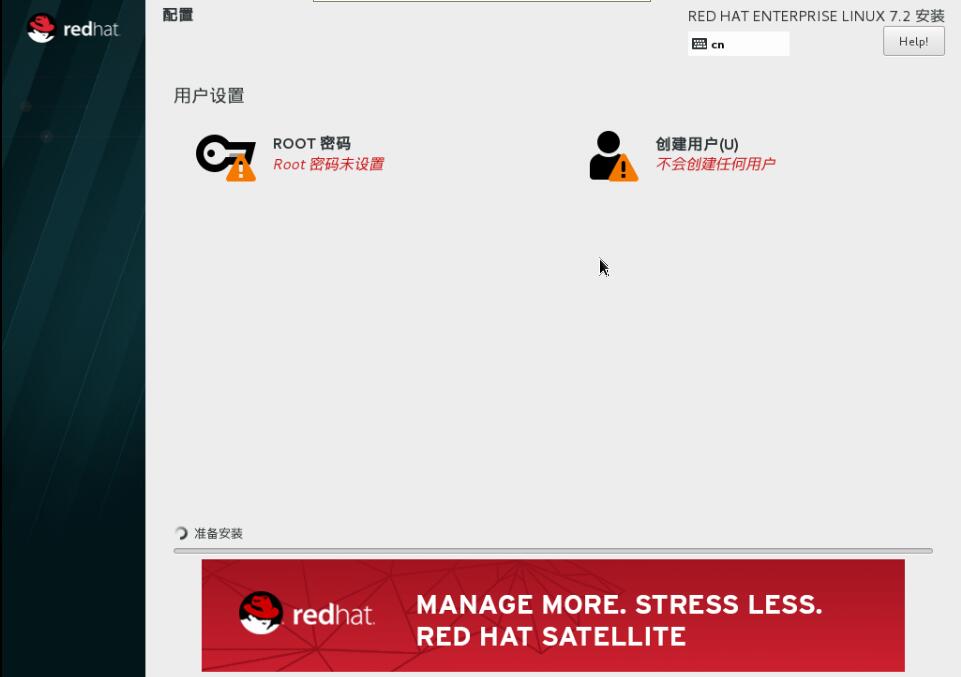
添加或删除挂载点

****

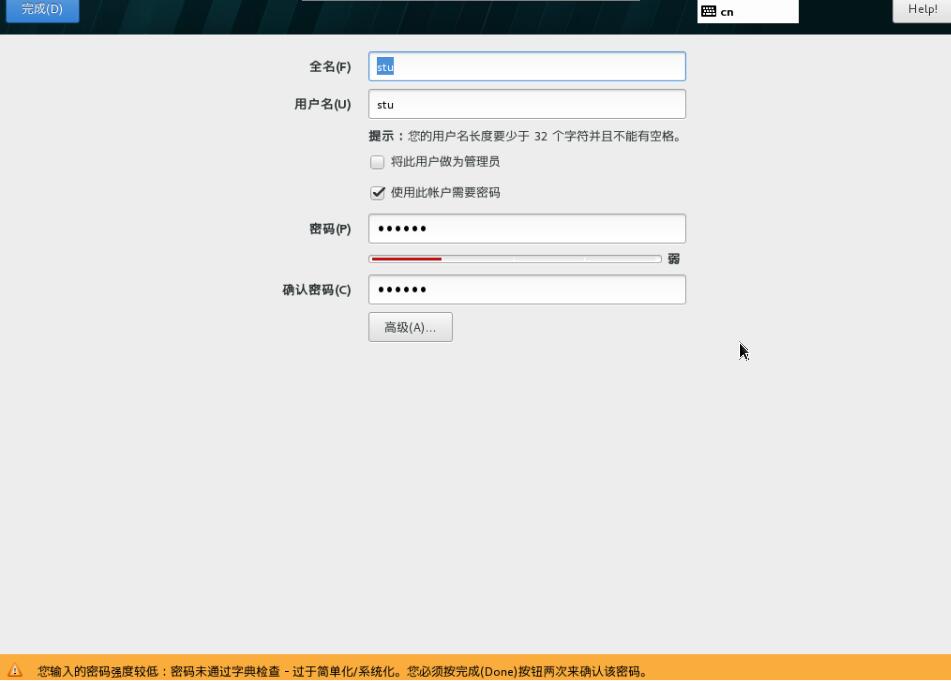
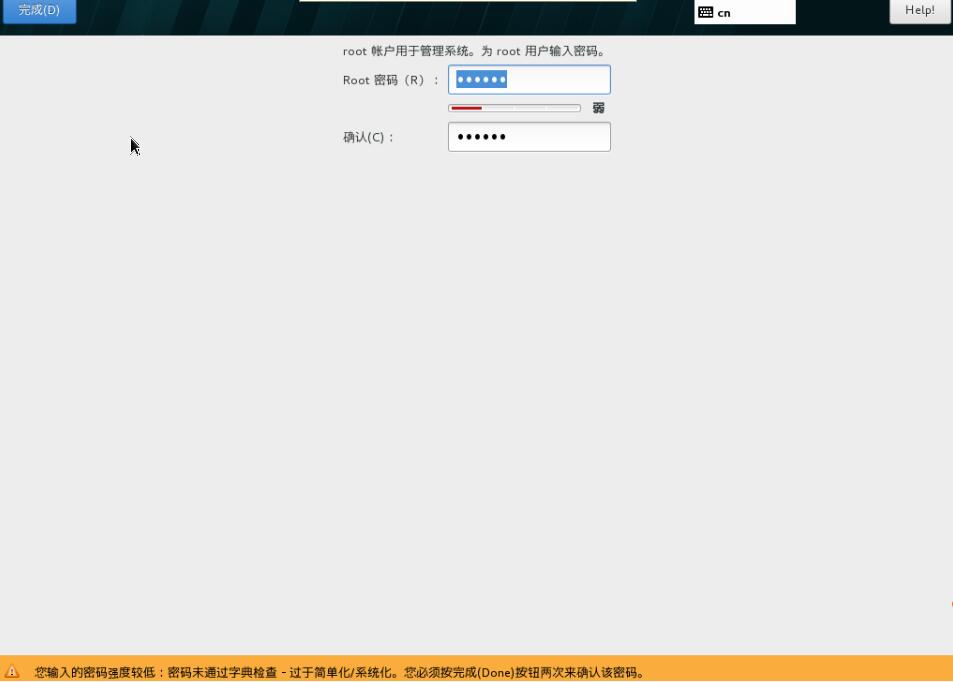
**开始安装**

ROOT密码：Taren1（受密码策略限制，点击两下完成即可）

创建用户：用户名、密码（受密码策略限制，点击两下完成即可）

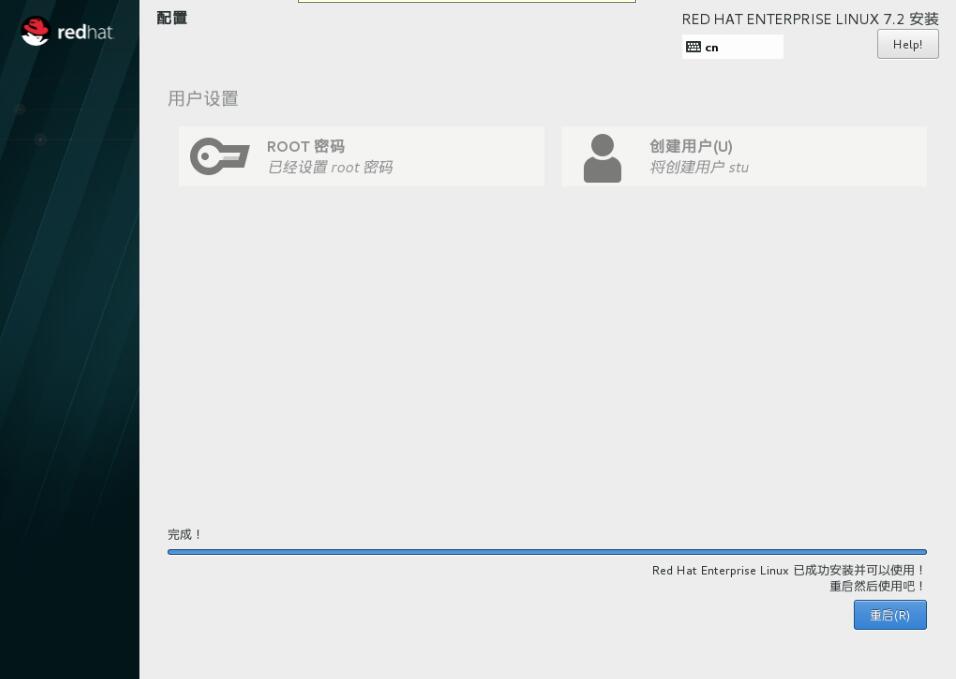


安装的同时可以进行管理员和新用户设置。



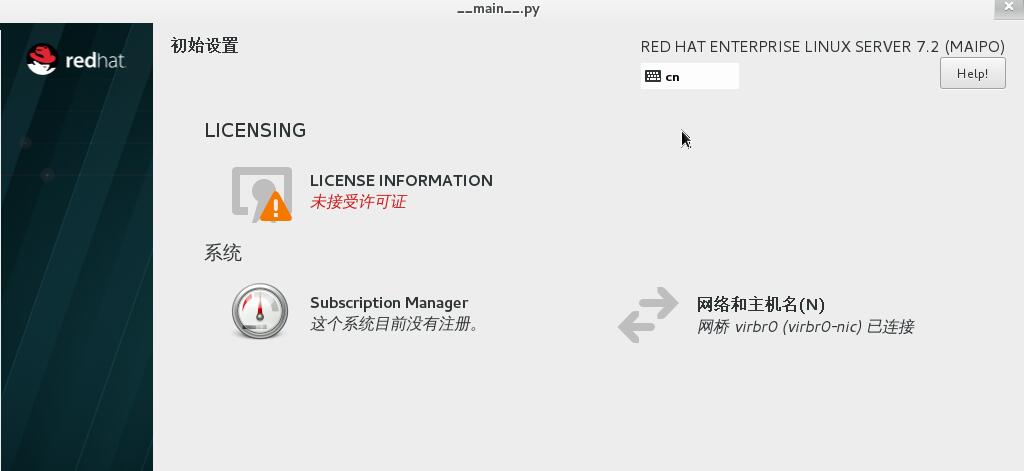
简单密码，点击两下完成即可强制修改密码，因为管理员权限最大

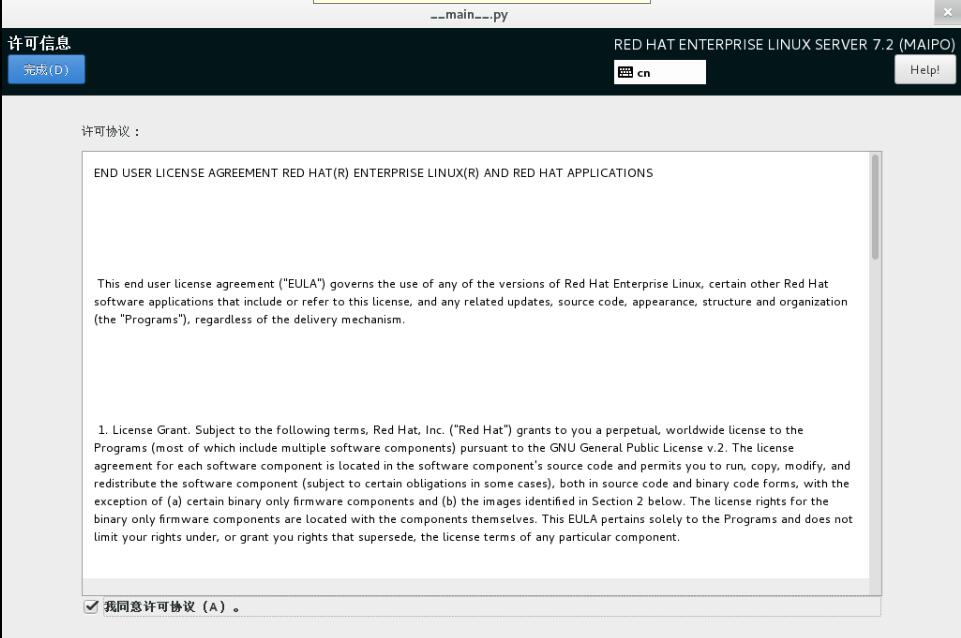
**安装完成后按“重启”**



**剩余安装**

初始设置-许可信息-我同意许可协议-完成配置-即可





## = linux系统基本命令操作 =

### 1、命令的格式

命令字 [选项] **[参数]**

命令字表示完成相应操作的具体命令

选项表示命令的要求

**参数表示命令作用的对象**

**[ ]表示可选项**

**Linux下面的命令严格区分大小写。**

**命令提示符**

[登录用户@主机名 工作目录]#

# 代表管理员

$ 代表普通用户

### 2、Linux下文件颜色意义

蓝色 --> 目录

绿色 --> 可执行文件

红色 --> 压缩文件

浅蓝色 --> 链接文件

白色 --> 其他文件

黄色 --> 设备文件

### 3、目录相关命令

#### 1) 查看当前所在的目录

pwd

#### 2) 创建目录

mkdir 目录名

#### 3) 查看当前目录的内容

ls

ls -l 以长格式显示文件信息

ls -la 以长格式显示所有文件（隐藏文件）

ls -lh 以长格式显示，以易读的单位显示文件信息

#### 4) 切换目录

cd 目录名（进入目录）

相对路径：相对某一个参考点

绝对路径：以/开始的路径

cd ..返回上一级目录

cd ../..

cd ~进入用户宿主目录

#### 5) 在当前目录建立两个目录

mkdir abc 123

#### 6) 递归创建目录

一次性创建多个层级目录，mkdir命令默认只创建路径中的最后那个目录，所以如果创建/root/abc/a/b/c目录，该命令成功的前提是目录abc、b、c已经存在。如果需要一次性创建成功，加-p选项

mkdir –p /root/abc/a/b/c

rmdir 删除空目录

### 4、文件类操作命令

#### 1） 创建文件

touch 01.txt（创建文件01.txt）

#### 2）复制文件或目录

cp 文件名 目录名(将文件复制到指定的目录)

注：cp -r 目录名 目录名（复制目录到指定的目录）

#### 3）移动文件或目录

mv 文件名 目录名（将文件移动到指定的目录）

在linux下，没有单独重命名命令，直接使用mv 原文件名 新文件名

#### 4） 删除文件或目录

rm 文件名（删除文件）

rm -f 文件名（删除文件并不提示）

rm -r 目录名（删除目录）

rm -rf 目录名（强制删除目录）

### 5、快捷键

Ctrl+u清空至行首

Ctrl+k清空至行尾

Ctrl+l清空整个屏幕

Ctrl+c结束当前编辑的命令

Ctrl+a光标移至行首

Ctrl+e光标移至行尾

Ctrl+左或右方向键一次移动一个单词

Ctrl+w删除前一个单词

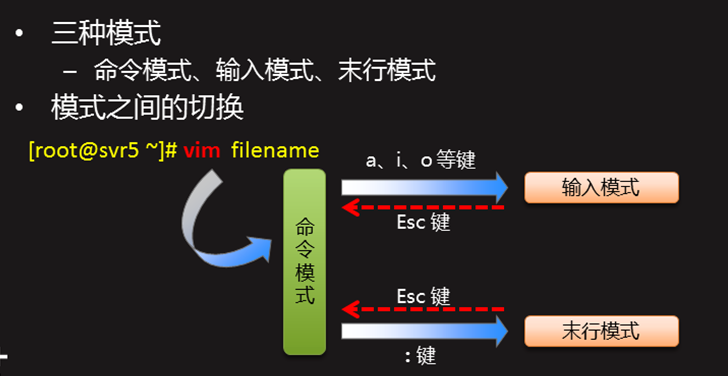
Ctrl+d相当于exit

ctrl+ shift + t 在当前终端打开新的终端。

alt + . 获取上一个命令的参数

**= vim的使用 =**

### 1、三种模式



命令模式： 按字母 u 可以撤销上一步操作

末行模式： set nu #显示每一行行号

命令模式： gg 、 G 、 3gg

#### 从命令模式到输入模式：

a 从当前光标后

i 从当前光标前插入一个空字符

o 在当前光标下新加一空行

#### 从命令模式到末行模式：

:

?

/

#### 从输入模式和末行模式返回命令模式：

esc键

### 2，vim的打开与退出

### （1）vim file 若file存在则打开file

若file不存在则新建file

### （2）翻页

PgUp 向上翻动一整页内容 = Ctrl + B

PgDn 向下翻动一整页内容 = Ctrl + F

### （3）行内的跳转

^ 将光标跳转到本行的行首字符 = Home键和数字0

$ 将光标跳转到本行的行尾字符 = End键

### （4）文件行内的跳转

1G=gg 跳转到文件的首行

G 跳转到文件的尾行

#G=#gg 跳转到文件的#行（命令模式）

:# 跳转到文件的#行（末行模式）

G或等于 shift+g

Ctrl+v+左右方向 选择内容

Vim中 大写字母=shift+小写字母

### （5）复制粘贴

yy 复制当前光标所在行

#yy 复制当前光标向下#行

p 粘贴

在末行模式中复制 :1,3y 复制1-3行

### （6）删除操作（在命令模式中）

x 删除光标处的单个字符 = Del

dd 删除光标所在行

#dd 删除#行（从光标所在处删除#行）

d^ 从光标处之前删除至行首

d$ 从光标处删除到行尾

### （7）字符串的查找（末行模式）

/world 向下查找world

？world 向上查找world

n 定位下一个匹配的字串

N 定位上一个匹配的字串

### （8）撤销编辑

u 取消最近一次的操作，能多次使用

U 取消当前行所有的操作

ctrl +r 对使用u命令撤销操作进行恢复

J 合并当前行和下一行

dj 删除当前行和下一行。

### （9）存盘及退出（在末行模式中按）

:q 退出

:w 保存

:wq 保存退出 = ZZ = :x

:X 加密

:wq! 强制保存退出

:! 命令 在vim中执行外面命令

:w file 另存为file

:r file 读入file

:e file 打开其他文件编辑

### （10）字符串的替换（末行模式）

: s/old/new 将当前行中查找到的第一个字符“old” 串替换为“new”

: s/old/new/g 将当前行中查找到的所有字符串“old” 替换为“new”

:#,# s/old/new/g 在行号“#,#”范围内替换所有的字符串“old”为“new”

:% s/old/new/gc 在整个文件范围内替换所有的字符串“old”为“new并对每个替换动作提醒，如果不加c就直接全部替换

### （11）末行模式基本操作

:set nu|nonu 显示/不显示行号

:syntax on|off 启用/关闭语法高亮

:set hlsearch|nohlsearch 开启/关闭查询结果高亮显示

:set autoindent|noautoindent 启用/关闭自动缩进

在vim ~/.vimrc文件中

