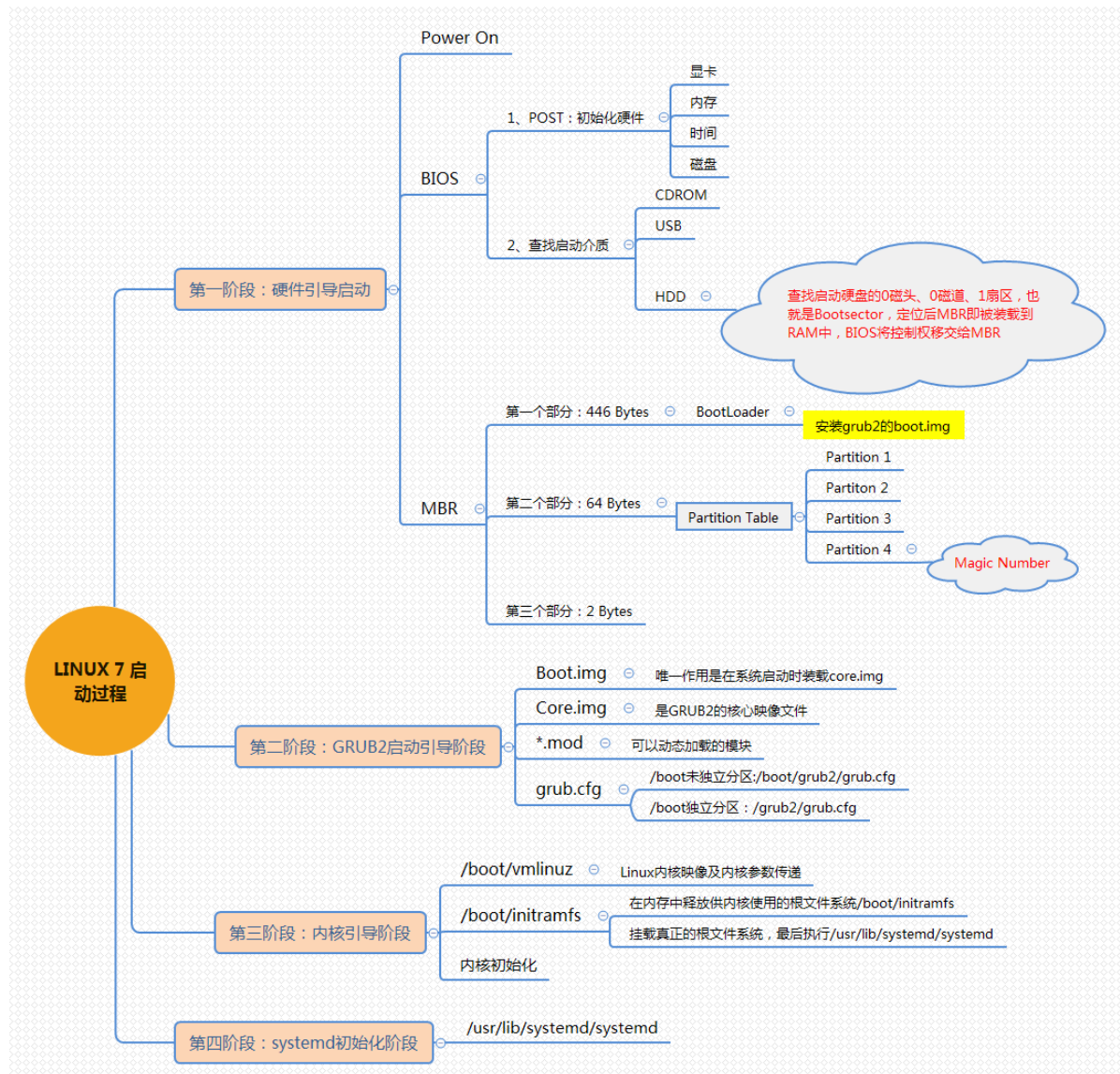


centos7 系统启动过程及相关配置文件

centos7 系统启动过程 CentOS7 引导顺序

1. UEFI 或 BIOS 初始化, 运行 POST 开机自检
2. 选择启动设备
3. 引导装载程序, centos7 是 grub2
4. 加载装载程序的配置文件: /etc/grub.d/ /etc/default/grub
/boot/grub2/grub.cfg
5. 加载内核选项 vmlinuz-3.10.0-693.el7.x86_64
6. 加载 initramfs 初始化伪文件系统
7. 内核初始化, centos7 使用 systemd 代替 init
8. 执行 initrd.target 所有单元, 包括挂载/etc/fstab
9. 从 initramfs 根文件系统切换到磁盘根目录
10. systemd 执行默认 target 配置, 配置文件/etc/systemd/system/default.target
11. systemd 执行 sysinit.target 初始化系统及 basic.target 准备操作系统
12. systemd 启动 multi-user.target 下的本机与服务器服务
13. systemd 执行 multi-user.target 下的/etc/rc.d/rc.local
14. Systemd 执行 multi-user.target 下的 getty.target 及登录服务
15. systemd 执行 graphical 需要的服务

centos7 启动过程:



```
1 [root@exercise1 ~]# find /boot/ -name *img*
2 /boot/grub2/i386-pc/core.img
3 /boot/grub2/i386-pc/boot.img
4 /boot/initrd-plymouth.img
5 /boot/initramfs-0-rescue-
   fff9bf659e8a49d7b721bafff48235b5.img
6 /boot/initramfs-3.10.0-693.el7.x86_64.img
7 [root@exercise1 ~]#
```

Systemd 运行原理-了解一下

Systemd 概述：systemd即为 system daemon ['di:mən] 守护进程，是 linux 下的一种 init 软件,开发目标是提供更优秀的框架以表示系统服务间的依赖关系，并依此实现系统初始化时服务的并行启动，同时达到降低 Shell的系统开销的效果，最终代替现在常用的 System V 不 BSD 风格 init 程序。

不多数发行版使用的 System V 风格 init 相比，systemd 采用了以下新技术：

(1) 采用Socket 激活式与总线激活式服务，以提高相互依赖的各服务的并行运行性能；

(2) 用 Cgroups 代替 PID 来追踪进程，以此即使是两次 fork 之后生成的守护进程也不会脱离 systemd的控制。

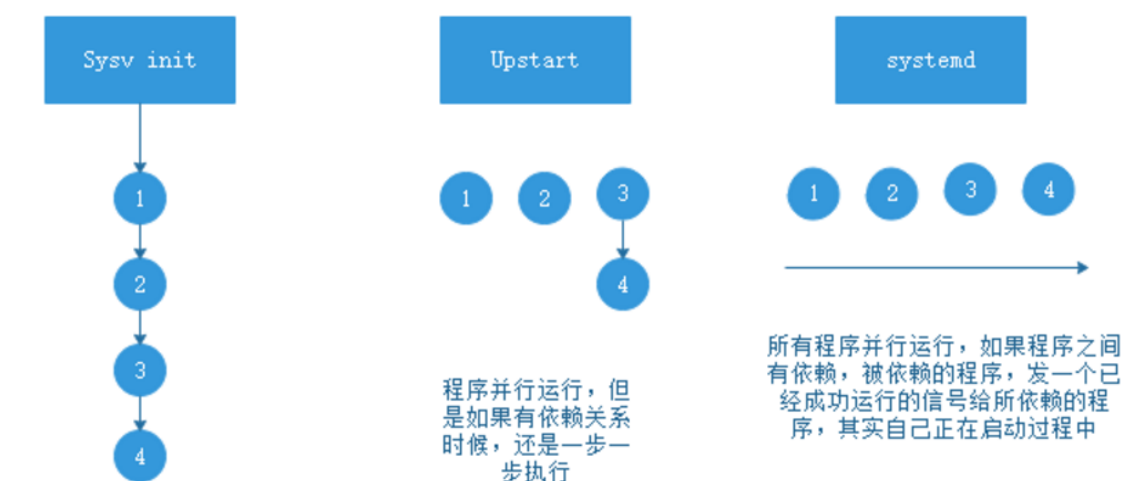
unit 对象：unit 表示不同类型的 systemd 对象，通过配置文件进行标识和配置；文件中主要 包含了系统服务、监听 socket、保存的系统快照以及其它不 init 相关的信息

Systemd 配置文件：

- /usr/lib/systemd/system/#这个目录存储每个服务的启动脚本，类似于之前的/etc/init.d/(centos6)
- /run/systemd/system/#系统执行过程中所产生的服务脚本，比上面目录优先运行
- /etc/systemd/system/#管理员建立的执行脚本，类似于/etc/rc.d/rcN.d/Sxx 类的功能， 比上面目录优先运行

总结：centos5-6-7 3 个系统版本启动过程：

CentOS 5: SysV init ; CentOS 6: Upstart ; CentOS 7: Systemd



程序一步一步运行，即便他们之间并没有依赖关系

程序并行运行，但是如果有依赖关系时候，还是一步一步执行

所有程序并行运行，如果程序之间有依赖，被依赖的程序，发一个已经成功运行的信号给所依赖的程序，其实自己正在启动过程中

管理系统服务

命令： **systemctl COMMAND name.service**

	centOS6	CentOS7
启动	service name start	systemctl start name.service
停止	service name stop	systemctl stop name.service
重启	service name restart	systemctl restart name.service
状态	service name status	systemctl status name.service
重载或重启服务(先加载，再启动)	-	systemctl reload-or-restart name.service

chkconfig 命令的对应关系

	centOS6	CentOS7
设定某服务开机自启	chkconfig name on	systemctl enable name.service
设定某服务开机禁止启动	chkconfig name off	systemctl disable name.service
查看所有服务的开机自启状态	chkconfig --list	systemctl list-unit-files --type service
用来列出该服务在		

哪些运行级别下启用和禁用	centOS6 chkconfig sshd -list	CentOS7 ls /etc/systemd/system/*.wants/ssh.service
查看服务是否开机自启	-	systemctl is-enabled name.service

服务状态

```
1 [root@exercise1 ~]# systemctl list-unit-files #显示状态
```

- **loaded:Unit** 配置文件已处理
- **active(running):**一次或多次持续处理的运行
- **active(exited):**成功完成一次性的配置
- **active(waiting):**运行中，等待一个事件
- **inactive:**不运行
- **enabled:开机启动**
- **disabled:开机不启动**
- **static:开机不启动，但可被另一个启用的服务激活**

实战：修改第三方软件的启动方式

方法1：修改nginx为systemctl启动

```

1 #灵活变动
2 [root@exercise1 ~]# vim
   /usr/lib/systemd/system/nginx.service #配置文件内容
3
4 [Unit]
5 Description=nginx
6 After=network.target
7
8 [Service]
9 Type=forking
10 ExecStart=/usr/local/nginx/sbin/nginx
11 ExecReload=/usr/local/nginx/sbin/nginx -s reload
12 ExecStop=/usr/local/nginx/sbin/nginx -s stop
13 PrivateTmp=true
14
```

```
15 [Install]
16 WantedBy=multi-user.target
```

[Unit]:服务的说明

Description:描述服务

After:描述服务类别

[Service]服务运行参数的设置

Type=forking是后台运行的形式

ExecStart为服务的具体运行命令

ExecReload为重启命令

ExecStop为停止命令

PrivateTmp=True表示给服务分配独立的临时空间

注意: [Service]的启动、重启、停止命令全部要求使用绝对路径

[Install]运行级别下服务安装的相关设置, 可设置为多用户, 即系统运行级别为3

注意: 对于新创建的service 文件, 或者修改了的service文件, 要通知systemd 重载此配置文件,而后可以选择重启

```
1 [root@exercise1 ~]# systemctl daemon-reload
```

方法2:

```
1 [root@exercise1 ~]# cd /usr/sbin 或 cd /usr/bin
2 [root@exercise1 ~]# ln -s /usr/local/nginx/sbin/nginx
  nginx
```

方法3:

```
[root@home sbin]# cd /etc/init.d
```

```
[root@home init.d]# ln -s /usr/local/nginx/sbin/nginx nginx
```

方法4:

```
[root@master opt]# alias abc=/usr/local/nginx/sbin/nginx
```

```
[root@master opt]# abc
```

```
[root@master opt]# ps -ef |grep nginx
```

方法5:

临时设置:

```
[root@master opt]# PATH=/usr/local/nginx/sbin/nginx/:
```

```
Error: You can't use 'macro parameter character #' in math mode
```

```
PATH" >>/etc/profile
```

```
[root@master opt]# nginx
```

```
[root@master opt]# ps -ef |grep nginx
```

方法6:

chkconfig命令主要用来更新（启动或停止）和查询系统服务的运行级信息。谨记chkconfig不是立即自动禁止或激活一个服务，它只是简单的改变了符号连接。

```
chkconfig [--add][--del][--list][系统服务] 或 chkconfig [--level <等级代号>][系统服务][on/off/reset]
```

参数:

-add 增加所指定的系统服务，让chkconfig指令得以管理它，并同时在系统启动的叙述文件内增加相关数据。

-del 删除所指定的系统服务，不再由chkconfig指令管理，并同时在系统启动的叙述文件内删除相关数据。

-level<等级代号> 指定读系统服务要在哪一个执行等级中开启或关毕。

等级0表示：表示关机

等级1表示：单用户模式

等级2表示：无网络连接的多用户命令行模式

等级3表示：有网络连接的多用户命令行模式

等级4表示：不可用

等级5表示：带图形界面的多用户模式

等级6表示：重新启动

chkconfig -list [name]: 显示所有运行级系统服务的运行状态信息（on或off）。如果指定了name，那么只显示指定的服务在不同运行级的状态。

chkconfig -add name: 增加一项新的服务。chkconfig确保每个运行级有一项启动(S)或者杀死(K)入口。如有缺少，则会从缺省的init脚本自动建立。

chkconfig -del name: 删除服务，并把相关符号连接从/etc/rc[0-6].d删除。

chkconfig [--level levels] name: 设置某一服务在指定的运行级是被启动, 停止还是重置

例:

chkconfig --list #列出所有的系统服务启动情况

chkconfig --list httpd #列出httpd服务设置情况

chkconfig --add httpd #增加httpd服务

chkconfig --del httpd #删除httpd服务

chkconfig --level 35 httpd on #设定httpd在等级3和5为开机运行服务, -level 35表示操作只在等级3和5执行, on表示启动, off表示关闭

chkconfig httpd on #设定mysqld在各等级为on, “各等级”包括2、3、4、5等级

chkconfig httpd off #设定mysqld在各等级为off

如何增加一个服务:

1.服务脚本必须存放在/etc/init.d/目录下;

每个被chkconfig管理的服务需要在对应的init.d下的脚本加上两行或者更多行的注释。

第一行告诉chkconfig默认启动的运行级以及启动和停止的优先级。如果某服务默认不在任何运行级启动, 那么使用 - 代替运行级。

第二行对服务进行描述, 可以用\ 跨行注释。

例:

file:///c:/users/admini~1/appdata/local/temp/tmp189rkl\1.png

2.添加执行权限

2.chkconfig -add servicename

在chkconfig工具服务列表中增加此服务, 此时服务会被在/etc/rc.d/rcN.d中赋予K/S入口了;

3.chkconfig -level 35 httpd on

修改服务的默认启动等级

例：

`vim /etc/init.d/nginx`

官方脚本如下链接：

<https://www.nginx.com/resources/wiki/start/topics/examples/redhatnginxinit/>

注：

`nginx="/usr/sbin/nginx"` 修改成nginx执行主程序的路径。

`NGINX_CONF_FILE="/etc/nginx/nginx.conf"` 修改成配置文件的路径。

运行级别

centos6 下 Linux 运行级别 0-6 的各自含义

0：关机模式

1：单用户模式，用于破解 root 密码

2：无网络，支持的多用户模式

3：有网络支持的多用户模式（一般叫字符界面，工作中最长使用的模式）

4：保留，未使用

5：有网络支持，支持图形界面，支持的多用户模式（图形界面）

6：重新引导系统，及重启

可以在不同级别下，设置服务是否随系统启动运行。在 centOS7 上运行级别的含义已经和之前不同了，已由.target 来代替运行级别，我们可以称 target 为目标态，我们可以通过 target 定制更符合我们工作运行环境。

```
1 [root@exercise1 ~]# ls /usr/lib/systemd/system/*.target
#查看我们的机器上有多少个 target
```

```
/usr/lib/systemd/system/remote-fs-pre.target
/usr/lib/systemd/system/remote-fs.target
/usr/lib/systemd/system/rescue.target
/usr/lib/systemd/system/rpcbind.target
/usr/lib/systemd/system/runlevel0.target
/usr/lib/systemd/system/runlevel1.target
/usr/lib/systemd/system/runlevel2.target
/usr/lib/systemd/system/runlevel3.target
/usr/lib/systemd/system/runlevel4.target
/usr/lib/systemd/system/runlevel5.target
/usr/lib/systemd/system/runlevel6.target
/usr/lib/systemd/system/shutdown.target
/usr/lib/systemd/system/sigpwr.target
```

```
1 | [root@exercise1 ~]# ls /usr/lib/systemd/system/*.target
    | grep runlevel
```

```
[root@exercise1 ~]# ls /usr/lib/systemd/system/*.target | grep runlevel
/usr/lib/systemd/system/runlevel0.target
/usr/lib/systemd/system/runlevel1.target
/usr/lib/systemd/system/runlevel2.target
/usr/lib/systemd/system/runlevel3.target
/usr/lib/systemd/system/runlevel4.target
/usr/lib/systemd/system/runlevel5.target
/usr/lib/systemd/system/runlevel6.target
[root@exercise1 ~]#
```

注：发现在 runlevel2-4 都是调用 **multi-user.target** 这个 unit。所以在 **centos7** 上 runlevel2-4 是一个意思

67

init systemd

=====

Traditional **runlevel** | New target name Symbolically linked to...

Runlevel0 | **runlevel0.target** -> **poweroff.target**

Runlevel1 | **runlevel1.target** -> **rescue.target**

Runlevel2 | **runlevel2.target** -> **multi-user.target**

Runlevel3 | **runlevel3.target** -> **multi-user.target**

Runlevel4 | **runlevel4.target** -> **multi-user.target**

Runlevel5 | **runlevel5.target** -> **graphical.target**

Runlevel6 | **runlevel6.target** -> **reboot.target**

=====

Init0 -> systemctl poweroff 关机

Init1 -> systemctl isolate rescue.target 单用户

Init3 -> systemctl isolate multi-user.target 字符界面

Init5 -> systemctl isolate graphical.target 图形化

Init6 -> systemctl reboot 重启

```
1 [root@exercise1 ~]# systemctl list-unit-files --type
  service    # 查看所有 service 的状态
```

[root@exercise1 ~]# systemctl list-unit-files --type target #查看所有 target 的状态

[root@exercise1 ~]# systemctl list-dependencies runlevel3.target #查看 3 级别Unit 的所有依赖

Unit 之间存在依赖关系：A 依赖于 B，就意味着 Systemd 在启动 A 的时候，同时会去启动 B。也可以理解也 3 运行级别下都开启哪些服务在 CentOS7 上所谓的目标态，其实就是由各种指定的服务和基础 target 组合而成的。

总结：centos6 和 7 运行级别的变化

运行级别的切换

1、在 CentOS6 上，我们切换级别使用 init，在 CentOS7 上虽然也能使用，但是调用的不再是原来的程序了。

CentOS7 使用 systemctl isolate name.target 来切换 target

例 1：在 CentOS6/7 下切换到字符界面

```
1 [root@exercise1 ~]# init 3      #切换到字符界面
2 [root@exercise1 ~]# init 5      #切换到图形界面
```

例 2：CentOS7 切换到字符界面

```
1 [root@exercise1 ~]# systemctl isolate multi-user.target
2
3 或:
4
5 [root@exercise1 ~]# systemctl isolate runlevel3.target
```

2、centos7 设置默认系统默认启动级别

systemctl set-default name.target来修改我们的目标态
看一下我们的默认目标态究竟为何物

```
1 [root@exercise1 ~]# ll
   /etc/systemd/system/default.target
2 lrwxrwxrwx. 1 root root 37 1月  9 09:31
   /etc/systemd/system/default.target ->
   /lib/systemd/system/multi-user.target
3 [root@exercise1 ~]#
```

```
[root@exercise1 ~]# ll /etc/systemd/system/default.target
lrwxrwxrwx. 1 root root 37 1月  9 09:31 /etc/systemd/system/default.target -> /lib/systemd/system/multi-user.target
```

注：它其实就是创建了一个软链接到指定的 target 上去了

例 1：默认系统启动使用 3 级别字符界面

```
1 [root@exercise1 ~]# systemctl set-default multi-
   user.target
2 Removed symlink /etc/systemd/system/default.target.
3 Created symlink from /etc/systemd/system/default.target
   to
4 /usr/lib/systemd/system/multi-user.target.
```

```
[root@exercise1 ~]# ll /etc/systemd/system/default.target
lrwxrwxrwx. 1 root root 37 1月  9 09:31
/etc/systemd/system/default.target -> /lib/systemd/system/multi-
user.target
```

例 2：默认系统启动使用 5 级别图形界面

```
1 [root@exercise1 ~]# systemctl set-default
   graphical.target
```

grub2 和 grub 区别-了解

在 centOS6 上, 我们的 grub 文件是/boot/grub/grub.conf

在 centOS7 使用 grub2, 配置文件改成/boot/grub2/grub.cfg了, 但是功能还是大致一样的都是用于加载 内核的, 不过在 centOS7 上设置默认启动项发生了一些变化。

互动: 如果我们的系统中有两个内核? 改变默认启动的内核顺序? (了解)

例 1: centos7 修改内核启动顺序

```
[root@exercise1 ~]# vim /etc/default/grub
```

GRUB_TIMEOUT=5#开机时 grub 默认 5 秒后启动内核

GRUB_DISTRIBUTOR="(sed's,release.*,,g' /etc/system-release)"

改: GRUB_DEFAULT= saved

为: GRUB_DEFAULT= 1#这里我们改成 1, 0 代表第一个内核, 1 代表第二个, 以此类推。

UB_DISABLE_SUBMENU=true

GRUB_TERMINAL_OUTPUT="console"

GRUB_CMDLINE_LINUX="crashkernel=auto rhgb quiet

net.ifnames=0"

GRUB_DISABLE_RECOVERY="true"

[root@exercise1 ~]# grub2-mkconfig -o /boot/grub2/grub.cfg #修改完成后, 并没有立即生效, 使用此命令来生成 grub.cfg 文件, 我们在下次启动的时候就会默认选择新的默认内核。

[root@exercise1 ~]# uname -r#查当前系统内核

3.10.0-693.2.2.el7.x86_64

[root@exercise1 ~]# reboot

[root@exercise1 ~]# uname -r#重启成功后, 发现加载的内核变了

3.10.0-693.el7.x86_64

[root@exercise1 ~]# cat /etc/redhat-release

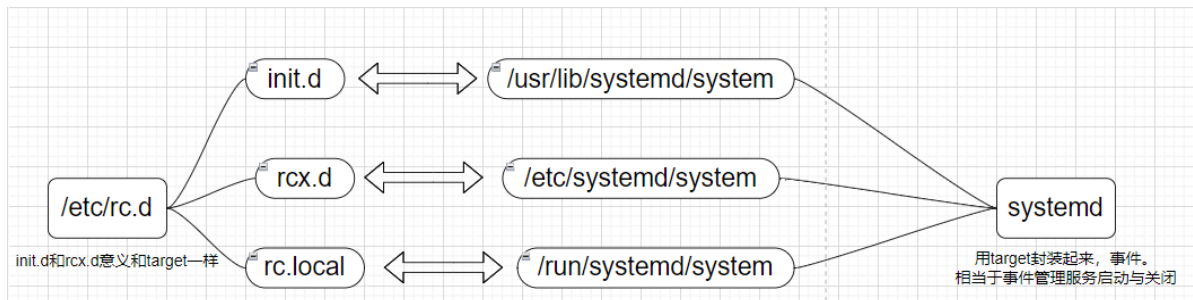
CentOS Linux release 7.4.1708 (Core)#查看linux版本

例 2: centos6 修改内核启动顺序-了解

```

1 [root@exercise1 ~]# vim /boot/grub/grub.conf
2 改: 10 default=0
3 为: 10 default=1
4 [root@exercise1 ~]# reboot

```



破译root密码<进入到单用户模式>:

进入到grub2启动屏显时, 按e进入编辑模式

在rootUUID号最后加init=/bin/sh

ctrl+x ----按照修改的参数启动服务

启动到单用户模式

mount -o remount, rw / 挂载根

passwd root ---修改root密码

如果系统启动了selinux限制, 那么需要执行

touch /.autorelabel

运行exec /sbin/init ---来正常启动

或者exec /sbin/reboot

或者使用rd.break重置密码

1、启动时按e

2、找到以linux16开头的在结尾处输入 rd.break

3、ctrl+x

4、mount -o remount, rw /sysroot/

- 5、chroot /sysroot/
- 6、echo redhat | passwd --stdin root
- 7、touch /.autorelabel (如果开启selinux)
- 8、ctrl+d 退出
- 9、reboot

17.3 实战-加密 grub 防止黑客通过单用户系统破解 root 密码

实战场景：如何防止别人恶意通过单用户系统破解 root 密码，进入系统窃取数据？

给 grub 加密，不让别人通过 grub 进入单用户。

17.3.1 基于 centos6 进行 grub 加密

```
[root@base ~]# grub-md5-crypt Password: 123456
```

```
Retype password: 123456
```

```
1oaqo5
```

Error: You can't use 'macro parameter character #' in math mode

```
1oaqo53d/cmTosm68jTw6o1wCu31
```

```
title Red Hat Enterprise Linux (2.6.32-220.el6.x86_64)
```

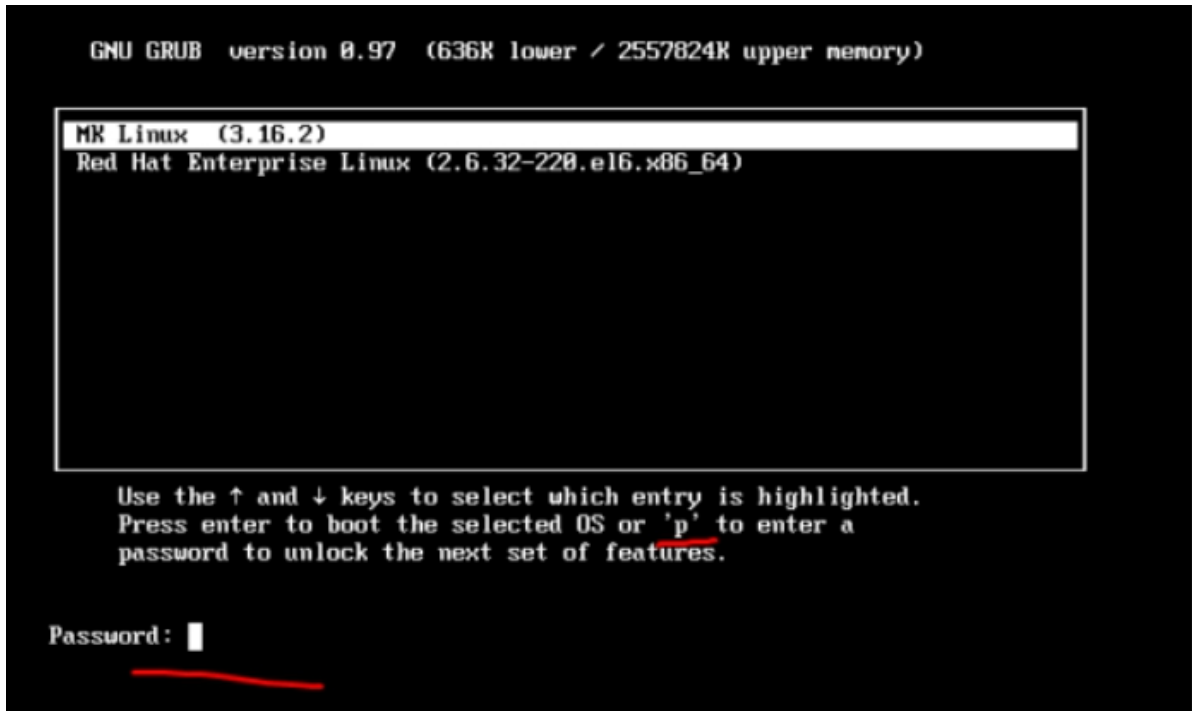
```
root (hd0,0)
```

如图：

```
#boot=/dev/sda
default=0
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
password --md5 $1$oaqo5$3d/cmTosm68jTw6o1wCu31
title Red Hat Enterprise Linux (2.6.32-220.el6.x86_64)
```

重启测试:

编辑 grub 时, 需要按下 p 键, 然后输入密码: 123456



17.3.2 基于 centos7 进行 grub 加密 生成密码

```
[root@base ~]# grub2-mkpasswd-pbkdf2
```

输入口令: 123456

Reenter password: 123456

PBKDF2 hash of your password is

grub.pbkdf2.sha512.10000.8F355BAB512AFB7B8C990A1FEB887B8F2F

3F1C54467E9B9F0535F2268E

1FFC5F4E8D33F7633D7FBEC25B2039C6D8B3226A90528D4883AB9B9

9E391A4965D069F.DDE99269

3BE2C09FFEEC1149120B6B84DBAB933DE6CF7BFF718E1DDC858AB73

EE32CFF45EB7F06AC45AA679

2E91C4CD09E2B445FC288C47E79F537DBBABAD756

```
[root@base ~]# vim /etc/grub.d/00_header    #在最后后面添加如下内容,
```

注 mk 这个用户名可

以换成自己的用户名

```
cat <<EOF
```

```
set superusers='mk'
```

```
password_pbkdf2 mk
```

```
grub.pbkdf2.sha512.10000.8F355BAB512AFB7B8C990A1FEB887B8F2F
```

```
3F1C54467E9B9F0535F2268E
```

```
1FFC5F4E8D33F7633D7FBEC25B2039C6D8B3226A90528D4883AB9B9
```


9E391A4965D069F.DDE99269

3BE2C09FFEEC1149120B6B84DBAB933DE6CF7BFF718E1DDC858AB73

EE32CFF45EB7F06AC45AA679

2E91C4CD09E2B445FC288C47E79F537DBBABAD756

EOF

如下图：

```
cat <<EOF
set superusers='mk'
password_pbkdf2 mk grub.pbkdf2.sha512.10000.8F355BAB512AFB7B8C990A1FEB887B8F2F3F1C5
4467E9B9F0535F2268E1FFC5F4E8D33F7633D7FBEC25B2039C6D8B3226A90528D4883AB9B99E391A4965
D069F.DDE992693BE2C09FFEEC1149120B6B84DBAB933DE6CF7BFF718E1DDC858AB73EE32CFF45EB7F06
AC45AA6792E91C4CD09E2B445FC288C47E79F537DBBABAD756
EOF
```

[root@base ~]# grub2-mkconfig -o /boot/grub2/grub.cfg #更新 grub
信息

重启验证：

```
Red Hat Enterprise Linux Server (3.10.0-327.el7.x86_64) 7.2 (Maipo)
Red Hat Enterprise Linux Server (0-rescue-889865d1faa54423ab571e2597a8ec)

Use the ↑ and ↓ keys to change the selection.
Press 'e' to edit the selected item, or 'c' for a command prompt.
```

输入用户名和密码

```
Enter username:
mk
Enter password:
_
```

看到可以进入 GRUB 菜单，就证明你加密成功了

```
setparams 'Red Hat Enterprise Linux Server (3.10.0-327.el7.x86_64) 7.2 (Maipo)\n\

load_video
set gfxpayload=keep
insmod gzio
insmod part_msdos
insmod xfs
set root='hd0,msdos1'
if [ x$feature_platform_search_hint = xy ]; then
    search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hin\
t-efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1 --hint='hd0,msdos1' b00399dc-6\
868-4295-837a-a3ad42e6f901
else
    search --no-floppy --fs-uuid --set=root b00399dc-6868-4295-837a-a3ad\4

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

按 ctrl-x 开始启动

17.4 实战-通过 liveCD 进入救援模式-重装 grub 修复损坏的系统

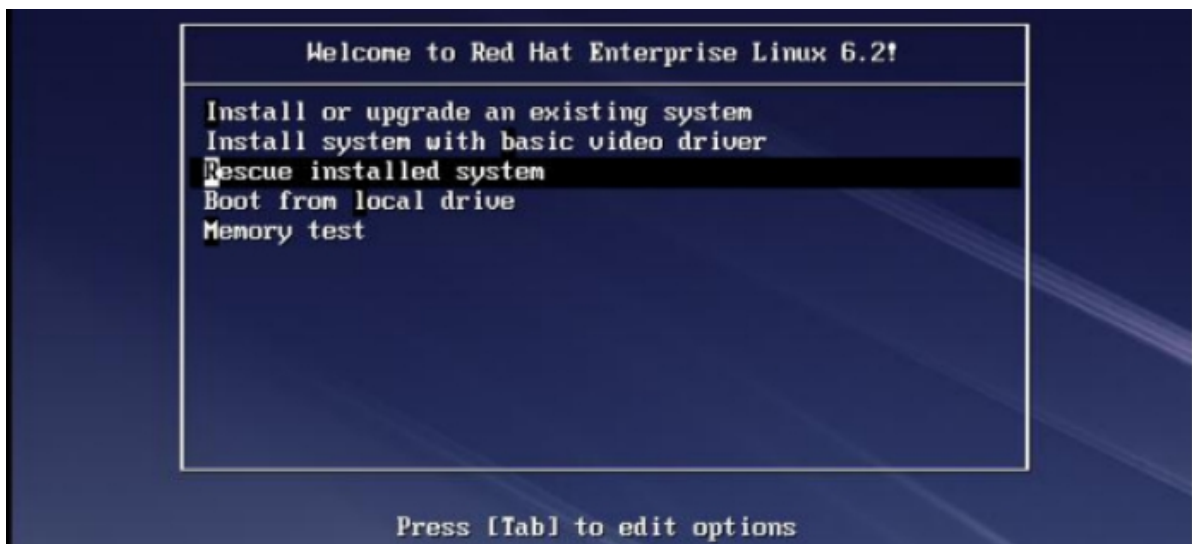
实战：使用系统光盘进入救援模式拯救坏掉的系统

实战场景：当系统坏了，进不去了，还需要把里面的数据复制出来，怎么办？

可以进入救援模式拷贝数据

17.4.1 基于 6 版本系统进入救援模式

修改 BIOS 启动顺序，直接以光盘引导系统



Choose a Language

What language would you like to use during the installation process?

Catalan
Chinese(Simplified)
Chinese(Traditional)
Croatian
Czech
Danish
Dutch
English

OK

Keyboard Type

What type of keyboard do you have?

slovene
sr-cy
sr-latin
sv-latin1
trq
ua-utf
uk
us

OK

Back

What type of media contains the rescue image?

Local CD/DVD

Hard drive

NFS directory

URL

OK

Back

Setup Networking

Do you want to start the network interfaces on this system?

Yes

No

Rescue

The rescue environment will now attempt to find your Linux installation and mount it under the directory /mnt/susimage. You can then make any changes required to your system. If you want to proceed with this step choose 'Continue'. You can also choose to mount your file systems read-only instead of read-write by choosing 'Read-Only'. If you need to activate SAN devices choose 'Advanced'.

If for some reason this process fails you can choose 'Skip' and this step will be skipped and you will go directly to a command shell.

Continue

Read-Only

Skip

Advanced

Rescue

Your system has been mounted under /mnt/sysimage.

Press <return> to get a shell. If you would like to make your system the

root environment, run the command:

```
chroot /mnt/sysimage
```

The system will reboot automatically
when you exit from the shell.

OK

Rescue

Your system is mounted under the
/mnt/sysimage directory.

OK

shell Start shell
fakd Run diagnostic
reboot Reboot

Ok

Cancel

```

bash-4.1#
bash-4.1#
bash-4.1# pwd
/
bash-4.1# head /etc/shadow
root::14438:0:99999:7:::
install::14438:0:99999:7:::
bash-4.1# _

```

ramfs : 内存文件系统

chroot /mnt/sysimage # 切换文件系统根

```

bash-4.1# chroot /mnt/sysimage/
sh-4.1# head /etc/shadow
root:$6$MPqI3IBGhnLo/G0T$4/UjwWT7SUB1kINh1Sna5nj8hwvmcDFJdgc5AR26fARTnUNUbXZ0JxY
xG.d30T4WTyfffL.KpZhDKrS0AWISl/:15691:0:99999:7:::
bin:!:15155:0:99999:7:::
daemon:!:15155:0:99999:7:::
adm:!:15155:0:99999:7:::
lp:!:15155:0:99999:7:::
sync:!:15155:0:99999:7:::
shutdown:!:15155:0:99999:7:::
halt:!:15155:0:99999:7:::
mail:!:15155:0:99999:7:::
uucp:!:15155:0:99999:7:::

```

17.4.2 实战-当 MBR 引导记录损坏后-重装 grub 进行修复

使用场景：修复 MBR，主要出现在安装双系统时，后安装的系统把原来系统的 MBR 删除了，需要修复。

第一步：在 centOS7 下破坏硬盘的前 446 字节：

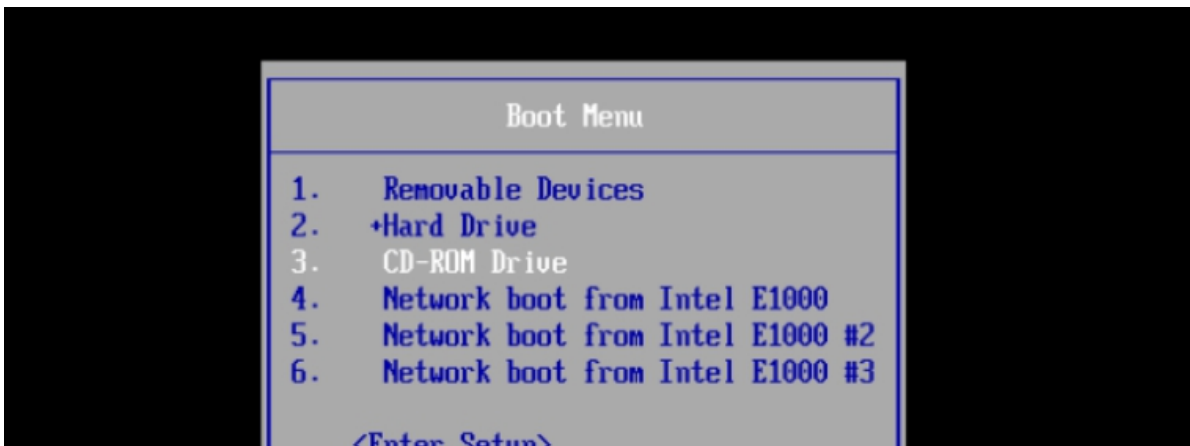
```
[root@CT731 ~]#dd if=/dev/zero of=/dev/sda bs=1 count=446 446+0
records in
```

```
446+0 records in
```

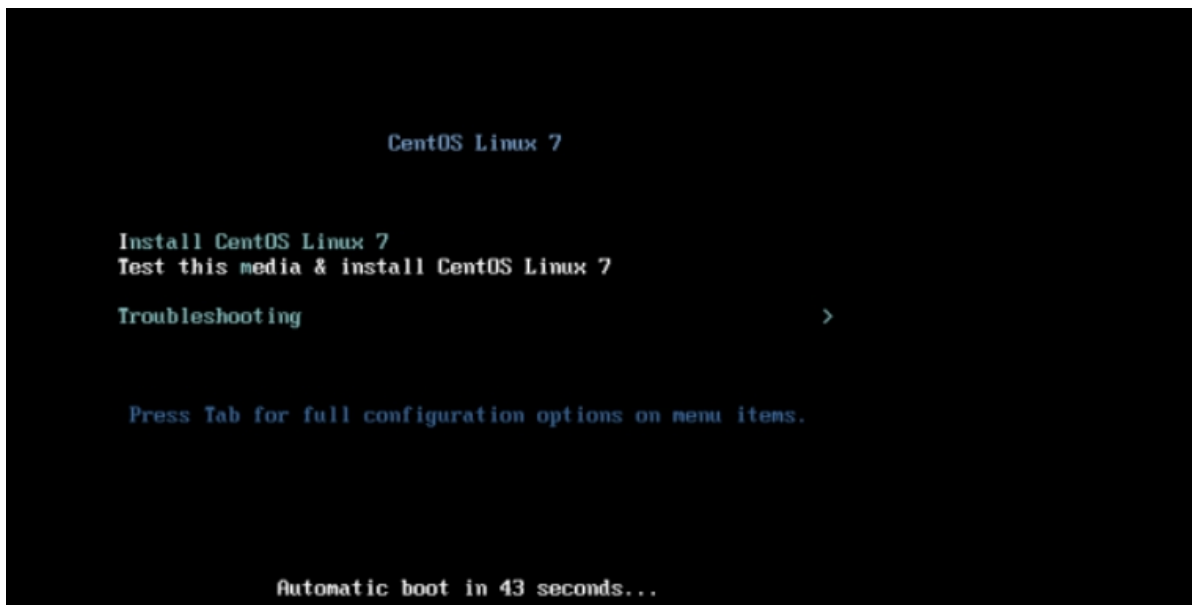
```
446+0 records out
```

```
446 bytes (446 B) copied,0.000758682 s,588 kB/s
```

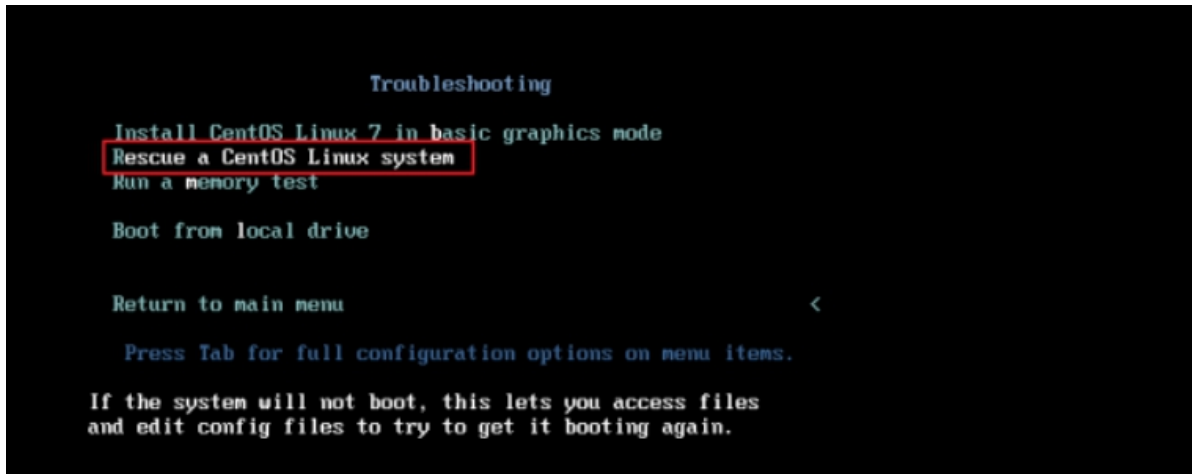
第二步：将 centos7 系统光盘挂载到虚拟机光驱上，，重启计算机，修改 BIOS 引导顺序，让光盘启动。



进入启动的界面



上面有三项，我们选择第三项进入 troubleshooting 故障排除界面，进入第三项后，点击第二项，进入救援模式的 centos 的系统



然后我们进入如下模式，选择 1，继续进行，接下来，我们会进入到一个 shell 模式中，需要切换根目录，进行系统修复：

```
1) Continue
2) Read-only mount
3) Skip to shell
4) Quit (Reboot)

Please make a selection from the above: 1
=====
Rescue Mount

Your system has been mounted under /mnt/sysimage.

If you would like to make your system the root environment, run the command:

    chroot /mnt/sysimage
Please press <return> to get a shell.
When finished, please exit from the shell and your system will reboot.
sh-4.2# chroot /mnt/sysimage
bash-4.2# grub2-install /dev/sda
Installing for i386-pc platform.
Installation finished. No error reported.
bash-4.2# reboot
```

先退一下，再重启，修复完成

```
bash-4.2# exit
exit
sh-4.2# reboot
```

17.4.3 实战-在 centOS7 下误删除 grub 文件进行修复

第一步：删除 grub2

[root@base ~]# rm -rf /boot/grub2

第二步，重启计算机

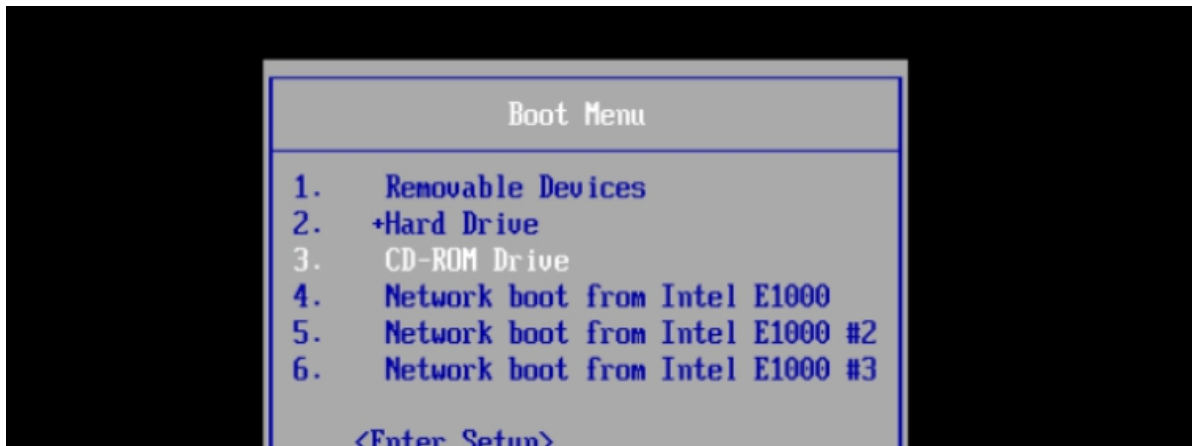
[root@base ~]# reboot

进入如下界面：

```
error: file '/grub2/i386-pc/normal.mod' not found.
Entering rescue mode...
grub rescue> _
```

现在开始解决 grub

重启系统，按 Esc，进入光盘救援模式，选择第三项，进入光盘救援（前提是挂载光盘）



使用 live cd 进入救援模式后：

第一步：切根

```
bash-4.2# chroot /mnt/sysimage/  
bash-4.2# ls  
app  backup  boot  etc  lib  media  mnt  opt  root  sbin  sys  testdir  usr  
bin  dev  home  lib64  misc  net  proc  run  srv  system  tmp  var  
bash-4.2#
```

然后执行命令

grub2-install

```
bash-4.2# grub2-install /dev/sda  
Installing for i386-pc platform.  
Installation finished. No error reported.  
bash-4.2# ls /boot/grub  
grub/  grub2/  
bash-4.2# ls /boot/grub2/  
fonts  grubenv  i386-pc  locale  
bash-4.2# _
```

下图中，我们可以看到在 grub2 文件夹中，还没有 grub.cfg 文件，接下来，我们需要生成配置文件：

进入到 grub2 下，

```
bash-4.2# grub2-mkconfig -o /boot/grub2/grub.cfg  
Generating grub configuration file ...  
Found linux image: /boot/vmlinuz-3.10.0-693.2.2.el7.x86_64  
Found initrd image: /boot/initramfs-3.10.0-693.2.2.el7.x86_64.img  
Found linux image: /boot/vmlinuz-3.10.0-693.el7.x86_64  
Found initrd image: /boot/initramfs-3.10.0-693.el7.x86_64.img  
Found linux image: /boot/vmlinuz-0-rescue-a5268a686d41459d8e5a8e699c8a9f43  
Found initrd image: /boot/initramfs-0-rescue-a5268a686d41459d8e5a8e699c8a9f43.img  
done  
bash-4.2# ls /boot/grub2/  
fonts  grub.cfg  grubenv  i386-pc  locale  
bash-4.2# _
```

exit 然后，重启电脑：

```
bash-4.2#  
bash-4.2# exit  
exit  
sh-4.2# reboot
```

修改 BIOS 引导，让硬盘做第一引导

注： centos 系统下载链接 <http://vault.centos.org/> 大家可以在这里
下载 centos6 相关的系统。

实例一之MBR挂了

\1. 开机启动，发现MBR挂了，但分区表还在

下面将MBR弄挂，然后重启<模拟故障>

-----if in file 从哪个文件读取数据

-----of out file 写入到哪个文件

-----bs 数据大小 默认单位byte

-----count 指定bs的数量

#破坏mbr引导 MBR:448+64分区表数据=512bytes

#dd if=/dev/zero of=/dev/sda bs=100 count=1

注意:配置selinux不生效

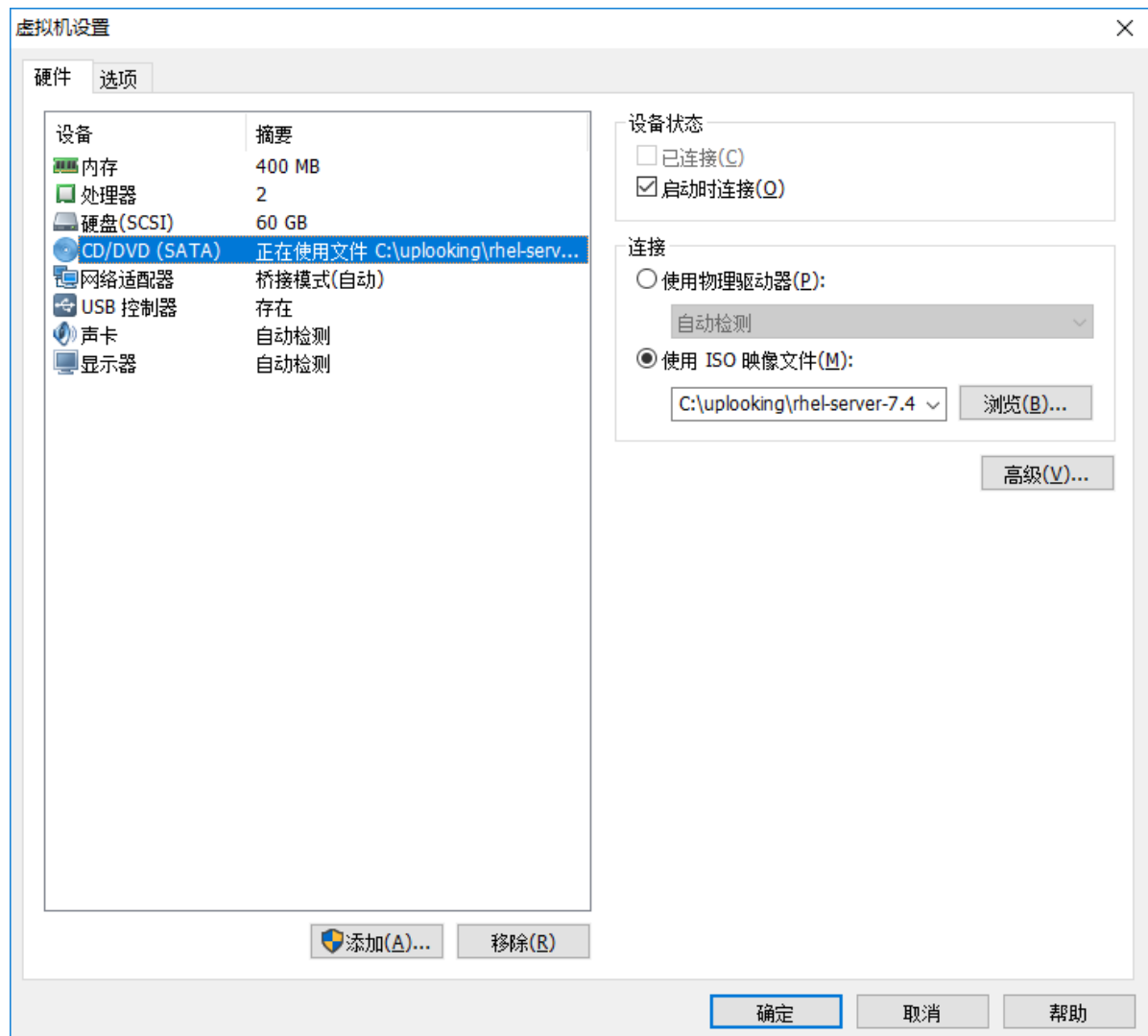
[root@uplooking system]# setenforce 0

setenforce: SELinux is disabled

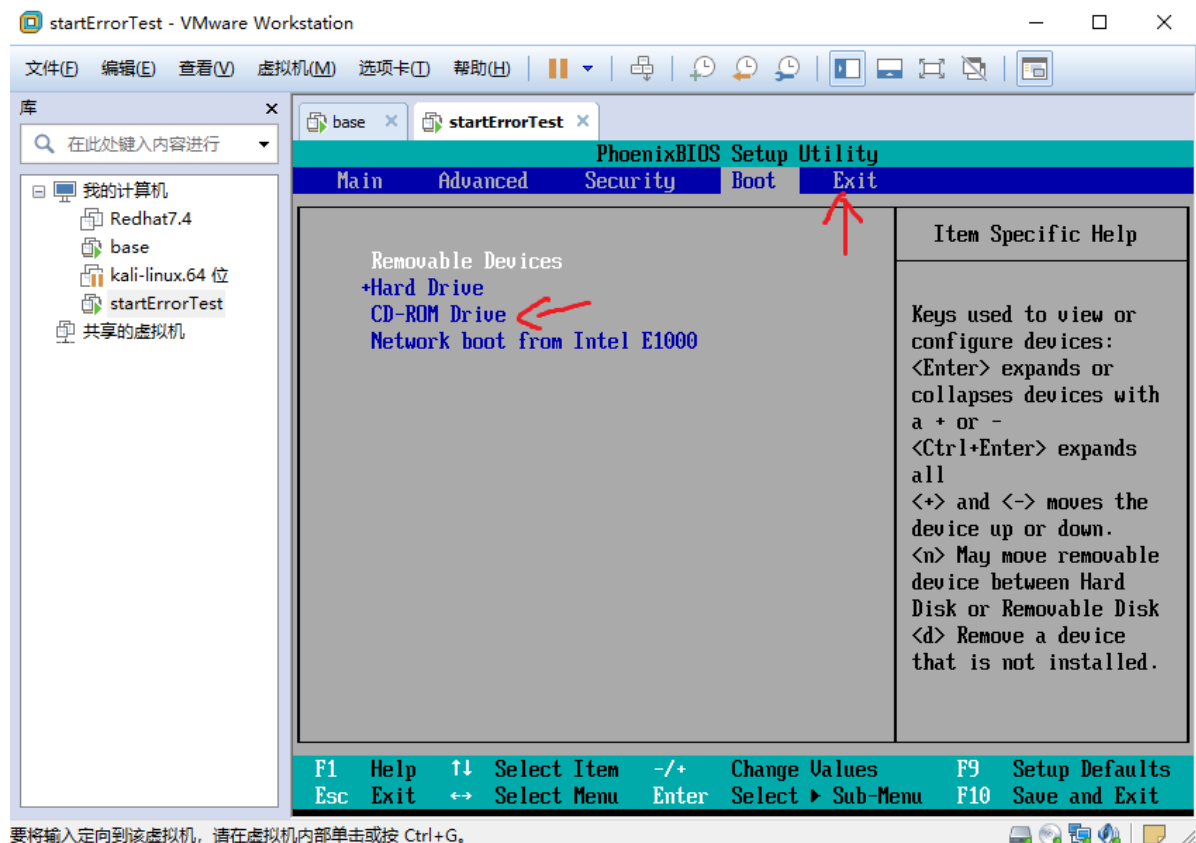
[root@uplooking system]# getenforce

Disabled

下面配置虚拟机cd/dvd启用ISO镜像



在虚拟启动的下拉菜单选择---打开电源进入固件设置---进入BIOS设置

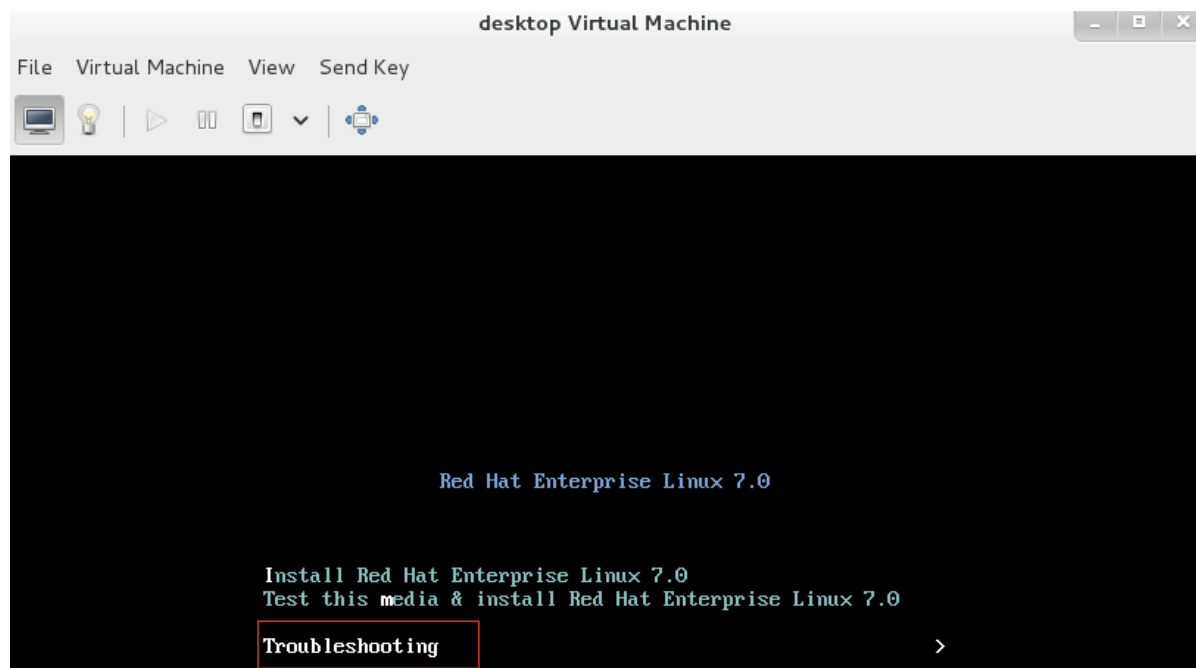


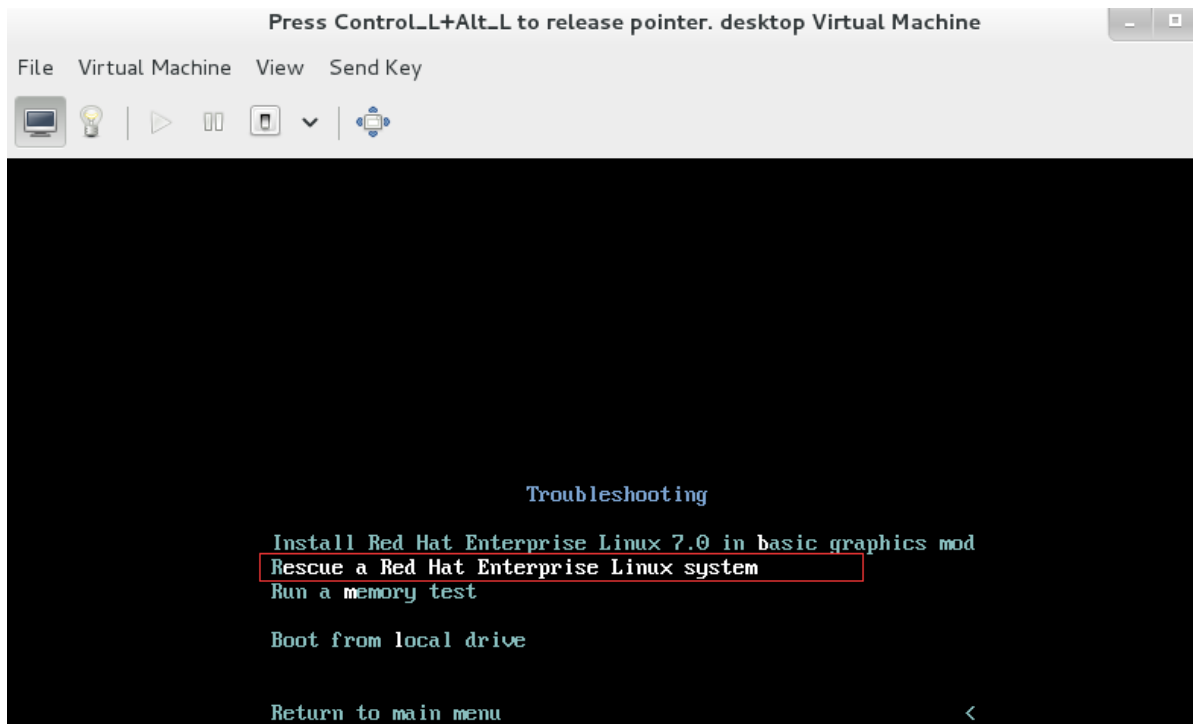
读/boot - 读内核 - 读驱动 - 读根/

\\2. 于是开机进rescue模式用grub2-install找回mbr

用光盘或者网络引导进入救援模式

选择Troubleshooting，然后 选择Rescue a Red Hat Enterprise Linux system





```
* if the graphical installation interface fails to start, try again with the
inst.text bootoption to start text installation
* when reporting a bug add logs from /tmp as separate text/plain attachments
=====
Rescue

The rescue environment will now attempt to find your Linux installation and
mount it under the directory : /mnt/sysimage. You can then make any changes
required to your system. Choose '1' to proceed with this step.
You can choose to mount your file systems read-only instead of read-write by
choosing '2'.
If for some reason this process does not work choose '3' to skip directly to a
shell.

1) Continue
2) Read-only mount
3) Skip to shell
4) Quit (Reboot)

Please make a selection from the above: 1
=====
Rescue Mount

Your system has been mounted under /mnt/sysimage.

If you would like to make your system the root environment, run the command:

    chroot /mnt/sysimage
Please press <return> to get a shell.
When finished, please exit from the shell and your system will reboot.
sh-4.2#
[anaconda] 1:main* 2:shell 3:log 4:storage-log 5:program-log Switch tab: Alt+Tab | Help: F1
```

这里是提示，你原来的linux系统的硬盘被挂载到了/mnt/sysimage下面

```
server Virtual Machine
File Virtual Machine View Send Key

Starting installer, one moment...
anaconda 19.31.79-1 for Red Hat Enterprise Linux 7.0 started.

Your system is mounted under the /mnt/sysimage directory.
When finished please exit from the shell and your system will reboot.

sh-4.2# df -TH
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/mapper/live-rw ext4       2.1G  1.1G  1.1G  49% /
devtmpfs        devtmpfs  1.1G    0  1.1G   0% /dev
tmpfs           tmpfs     1.1G    0  1.1G   0% /dev/shm
tmpfs           tmpfs     1.1G  8.7M  1.1G   1% /run
tmpfs           tmpfs     1.1G    0  1.1G   0% /sys/fs/cgroup
/dev/sda        iso9660    3.8G  3.8G    0 100% /run/install/repo
tmpfs           tmpfs     1.1G 144k  1.1G   1% /tmp
/dev/vda1       xfs       11G   3.4G  7.5G  31% /mnt/sysimage
devtmpfs        devtmpfs  1.1G    0  1.1G   0% /mnt/sysimage/dev
tmpfs           tmpfs     1.1G    0  1.1G   0% /mnt/sysimage/dev/shm
tmpfs           tmpfs     1.1G    0  1.1G   0% /dev/shm
tmpfs           tmpfs     1.1G  8.7M  1.1G   1% /mnt/sysimage/run
```

```
File Virtual Machine View Send Key

Starting installer, one moment...
anaconda 19.31.79-1 for Red Hat Enterprise Linux 7.0 started.

Your system is mounted under the /mnt/sysimage directory.
When finished please exit from the shell and your system will reboot.

sh-4.2# df -TH
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/mapper/live-rw ext4       2.1G  1.1G  1.1G  49% /
devtmpfs        devtmpfs  1.1G    0  1.1G   0% /dev
tmpfs           tmpfs     1.1G    0  1.1G   0% /dev/shm
tmpfs           tmpfs     1.1G  8.7M  1.1G   1% /run
tmpfs           tmpfs     1.1G    0  1.1G   0% /sys/fs/cgroup
/dev/sda        iso9660    3.8G  3.8G    0 100% /run/install/repo
tmpfs           tmpfs     1.1G 144k  1.1G   1% /tmp
/dev/vda1       xfs       11G   3.4G  7.5G  31% /mnt/sysimage
devtmpfs        devtmpfs  1.1G    0  1.1G   0% /mnt/sysimage/dev
tmpfs           tmpfs     1.1G    0  1.1G   0% /mnt/sysimage/dev/shm
tmpfs           tmpfs     1.1G    0  1.1G   0% /dev/shm
tmpfs           tmpfs     1.1G  8.7M  1.1G   1% /mnt/sysimage/run

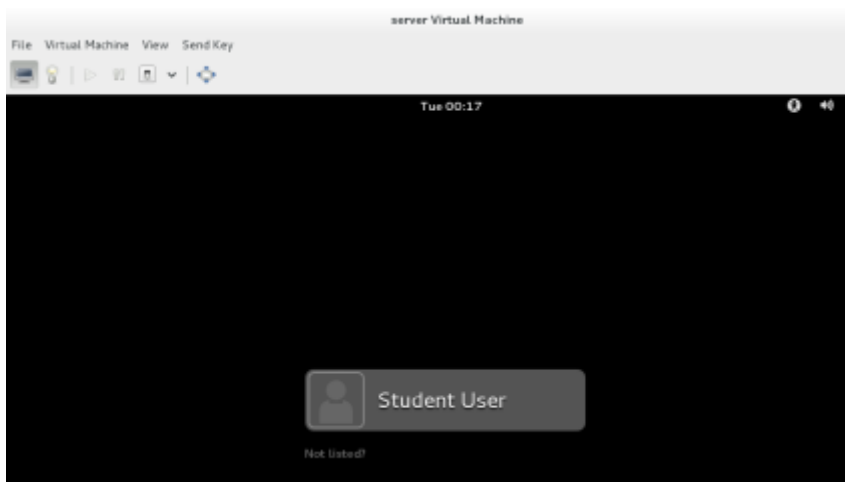
sh-4.2# chroot /mnt/sysimage/
bash-4.2# grub2-install /dev/vda
Installing for i386-pc platform.
Installation finished. No error reported.
```

Sh-4.2#chroot /mnt/sysimage/ #将根目录切换到硬盘

Bash-4.2#grub2-install /dev/vda #使用grub2-install恢复，可以看到sh变成了bash

2次exit重启(注意取消从光盘启动)

\3. 因此正常开机



#####

实例二之MBR挂了2

网络配置参照实例一

\1. 使用DD将MBR备份出来

```
#dd if=/dev/sda of=mbr.bak bs=512 count=1
```

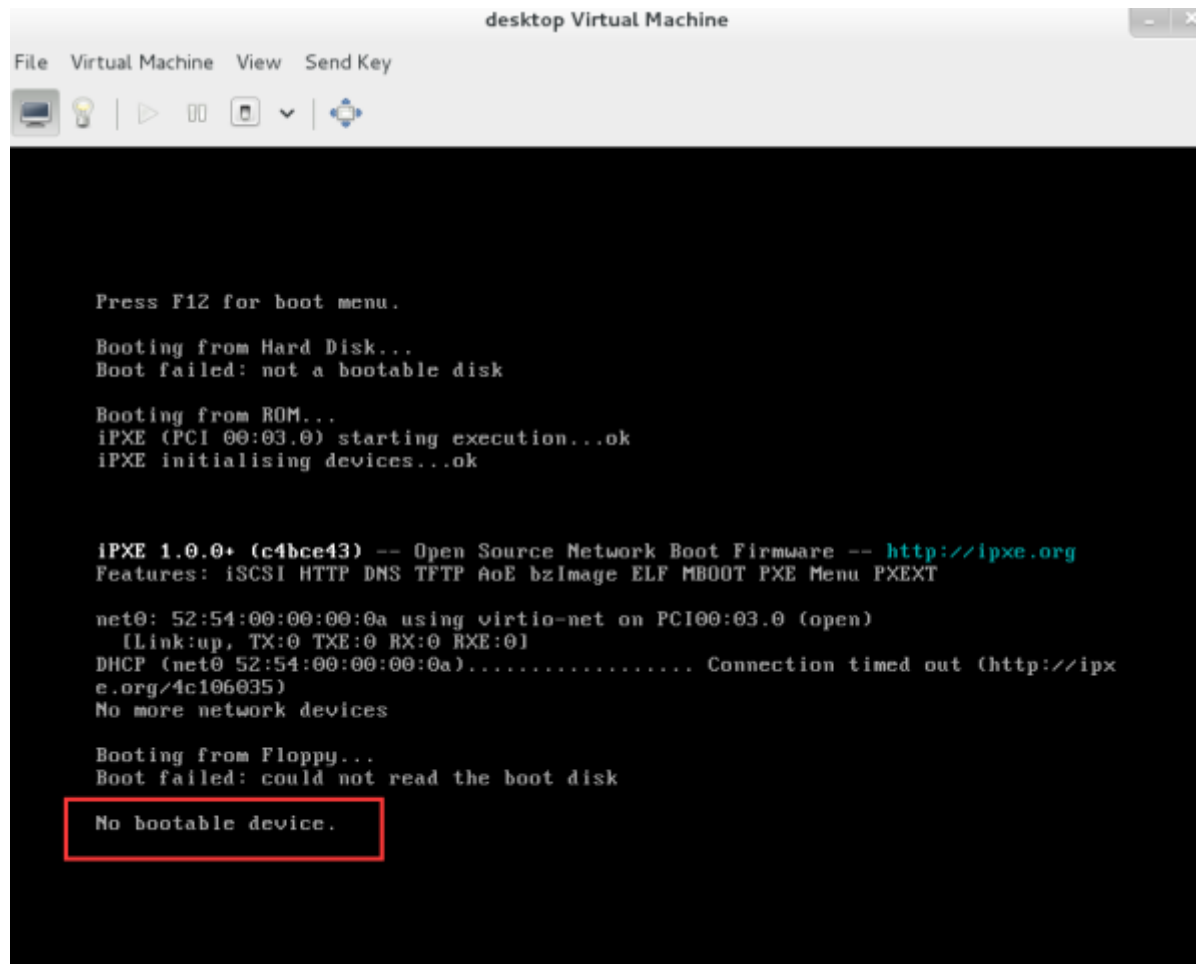
#复制到远程主机

```
#scp mbr.bak 172.19.18.151:/tmp
```

使用dd将mbr弄挂

```
#dd if=/dev/zero of=/dev/sda bs=512 count=1
```

\2. 开机启动，发现MBR挂了，分区表也没了



\3. 于是进rescue模式，恢复以前备份的MBR文件

增加光盘，请查看实例一

```
sh-4.2# df
Filesystem      1K-blocks    Used Available Use% Mounted on
/dev/mapper/live-rw 2030899 1116124    910679  56% /
devtmpfs         159292      0    159292   0% /dev
tmpfs            184476      4    184472   1% /dev/shm
tmpfs            184476    8672    175804   5% /run
tmpfs            184476      0    184476   0% /sys/fs/cgroup
/dev/sr0         3963760 3963760      0 100% /run/install/repo
tmpfs            524288     308    523980   1% /tmp
sh-4.2# ifconfig ens33 172.19.18.153 ^C
sh-4.2# ping 172.19.18.151
PING 172.19.18.151 (172.19.18.151) 56(84) bytes of data:
64 bytes from 172.19.18.151: icmp_seq=1 ttl=64 time=0.325 ms
64 bytes from 172.19.18.151: icmp_seq=2 ttl=64 time=0.527 ms
^C
--- 172.19.18.151 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.325/0.426/0.527/0.101 ms
sh-4.2#
```

读/boot - 读内核 - 读驱动 - 读根/

上图可得，分区表也没了

因此直接配置IP地址，然后从备份服务器下载以前备份的mbr

#ifconfig ens33 172.19.18.153

#ping 172.19.18.151（备份服务器）可ping通,于是通过scp获取

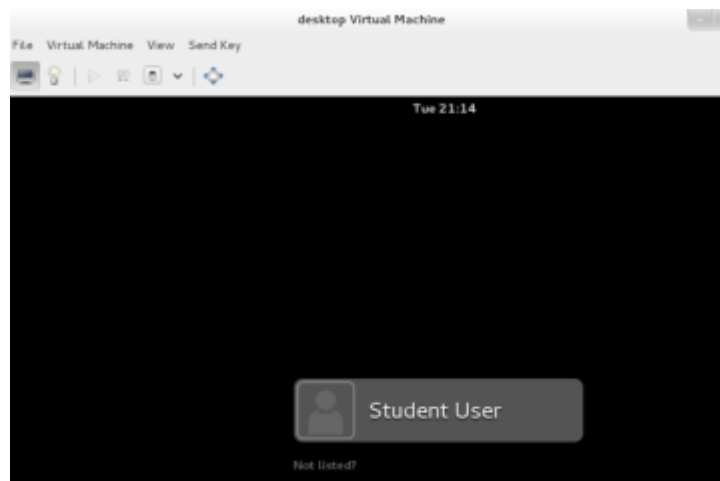
#scp 172.19.18.151:/tmp/mbr.bak /

#dd if=/mbr.bak of=/dev/sda #恢复mbr

2次exit重启

记得将光盘取消（请看实例一）

\4. 因此正常开机



####

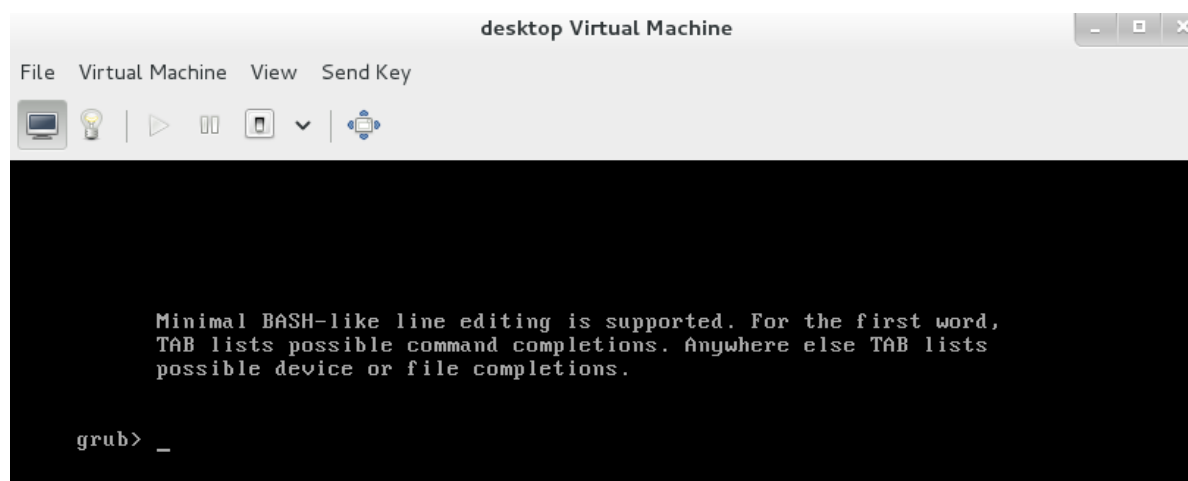
实例三之GRUB2挂了1

挂载光盘请看实例一

1.将/boot/grub2/grub.cfg删除

```
[root@uplooking system]# cd /boot/grub2/  
[root@uplooking grub2]# ls  
device.map  fonts  grub.cfg  grubenv  i386-pc  locale  
[root@uplooking grub2]# mv grub.cfg grub.cfg.bak
```

2只有grub界面，手动指定，然后进入系统



root=(hd0,msdos1)指定boot分区

#指定内核 与 "/" 文件系统 selinux=0 关闭还原模式

```
Minimal BASH-like line editing is supported. For the first word,
TAB lists possible command completions. Anywhere else TAB lists
possible device or file completions.

grub> root=(hd0,msdos1) 指定 boot 分区
grub> linux16 /vmlinuz-3.10.0-693.el7.x86_64 root=/dev/mapper/rhel-root ro selin
ux=0 指定内核 及根文件系统
grub> initrd16 /initra
Possible files are:

  initramfs-0-rescue-657add8ff3fa4fff9b0f14b974e31ed9.img
initramfs-3.10.0-693.el7.x86_64.img initramfs-3.10.0-693.el7.x86_64kdump.img
grub> initrd16 /initramfs-3.
Possible files are:

  initramfs-3.10.0-693.el7.x86_64.img initramfs-3.10.0-693.el7.x86_64kdump.img
grub> initrd16 /initramfs-3.10.0-693.el7.x86_64.img 指定 内核驱动文件
grub> boot_ 执行引导
```

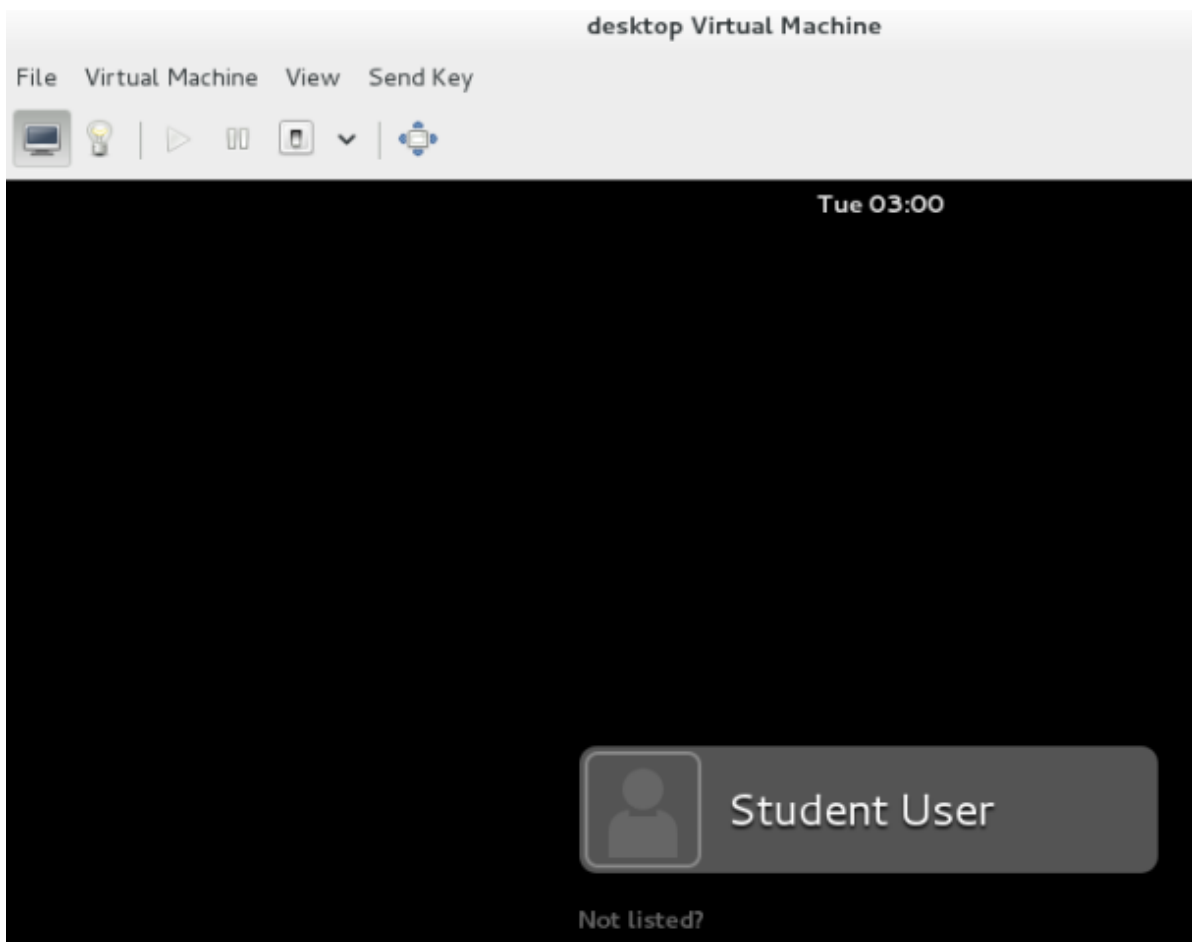
可用tab

3.因此正常开机

一定要记得把selinux关了，不然是这个后果

```
desktop Virtual Machine
File Virtual Machine View Send Key
[ OK ] Started Tell Plymouth To Write Out Runtime Data.
[ 11.664564] type=1305 audit(1439233040.398:4): audit_pid=447 old=0 auid=42949
67295 ses=4294967295 subj=system_u:system_r:auditd_t:s0 res=1
[ OK ] Started Show Plymouth Boot Screen.
[ OK ] Started udev Wait for Complete Device Initialization.
Starting Activation of DM RAID sets...
[ OK ] Started Activation of DM RAID sets.
[ OK ] Reached target Local File Systems.
Starting Relabel all filesystems, if necessary...
Starting Trigger Flushing of Journal to Persistent Storage...
[ 11.791820] [drm] main mem slot 1 [f4000000,3ffe000]
[ 11.805799] [drm] fb mappable at 0xf4000000, size 3145728
[ 11.810798] [drm] fb: depth 24, pitch 4096, width 1024, height 768
[ 11.815341] fbcon: qxldrmfb (fb0) is primary device
[ 11.927061] Console: switching to colour frame buffer device 128x48
[ 11.943942] qxl 0000:00:02.0: fb0: qxldrmfb frame buffer device
[ 11.943944] qxl 0000:00:02.0: registered panic notifier
[ 11.944063] [drm] Initialized qxl 0.1.0 20120117 for 0000:00:02.0 on minor 0
Starting Create Volatile Files and Directories...
Starting Security Auditing Service...
Starting Tell Plymouth To Write Out Runtime Data...
[ OK ] Reached target Encrypted Volumes.
[ OK ] Started Trigger Flushing of Journal to Persistent Storage.
[ OK ] Started Tell Plymouth To Write Out Runtime Data.

*** Warning -- SELinux targeted policy relabel is required.
*** Relabeling could take a very long time, depending on file
*** system size and speed of hard drives.
[ OK ] Started Update UTMP about System Reboot/Shutdown.
33.9%^\
```



4./boot下没有grub.cfg于是生成grub2配置文件

#通过命令生成新的grub.cfg文件

#grub2-mkconfig -o /boot/grub2/grub.cfg 或者 grub2-mkconfig >
/boot/grub2/grub.cfg #前提/etc/grub.d没有被删

```
[root@desktop0 ~]# cd /boot/
[root@desktop0 boot]# cd grub2
[root@desktop0 grub2]# ls
device.map  fonts  grubenv  i386-pc  locale  themes
[root@desktop0 grub2]# grub2-mk
grub2-mkconfig      grub2-mklayout      grub2-mkrelpath
grub2-mkfont         grub2-mknetdir      grub2-mkrescue
grub2-mkimage        grub2-mkpasswd-pbkdf2  grub2-mkstandalone
[root@desktop0 grub2]# grub2-mkconfig >/boot/grub2/grub.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-3.10.0-123.el7.x86_64
Found initrd image: /boot/initramfs-3.10.0-123.el7.x86_64.img
Found linux image: /boot/vmlinuz-0-rescue-946cb0e817ea4adb916183df8c4fc817
Found initrd image: /boot/initramfs-0-rescue-946cb0e817ea4adb916183df8c4fc817.im
g
done
[root@desktop0 grub2]#
```


#####

实例四之GRUB2挂了2

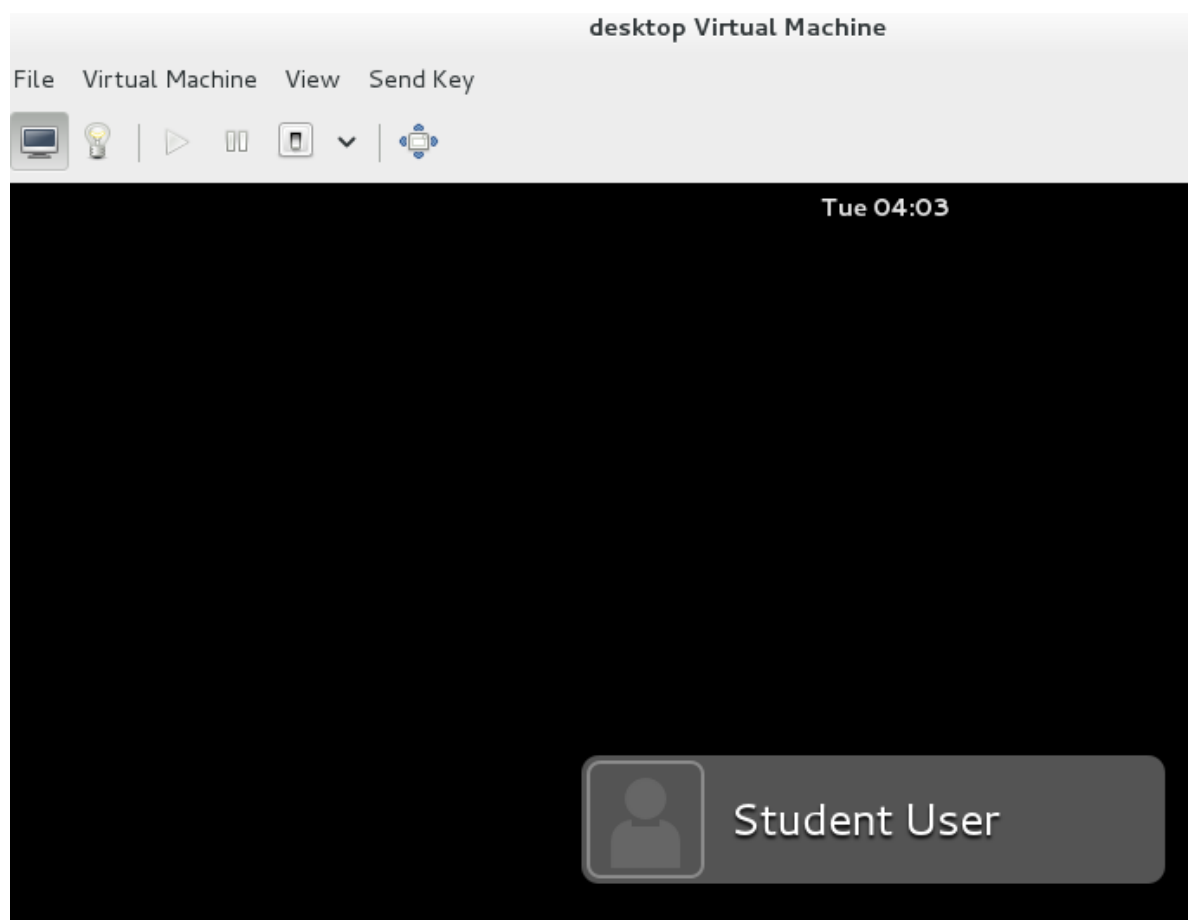
1.把/boot和/etc下的grub相关文件删除，无法生成grub配置文件

```
[root@localhost ~]#  
[root@localhost ~]#  
[root@localhost ~]# rm -rf /etc/grub.d/ 删除arub.cfa  
[root@localhost ~]# rm -rf /boot/grub2/grub.cfg _  
删除grub2-mkconfig 生成grub.cfg的配置文件
```

2.于是手动指定

请查看实例三

3.因此正常开机



4./boot和/etc没有grub相关文件，手动写一个grub配置文件

```
[root@desktop0 ~]# cd /boot/grub2/
[root@desktop0 grub2]# ls
device.map  fonts  grubenv  i386-pc  locale  themes
[root@desktop0 grub2]# grub2-mkconfig >/boot/grub2/grub.cfg
Generating grub configuration file ...
done
[root@desktop0 grub2]# █
```

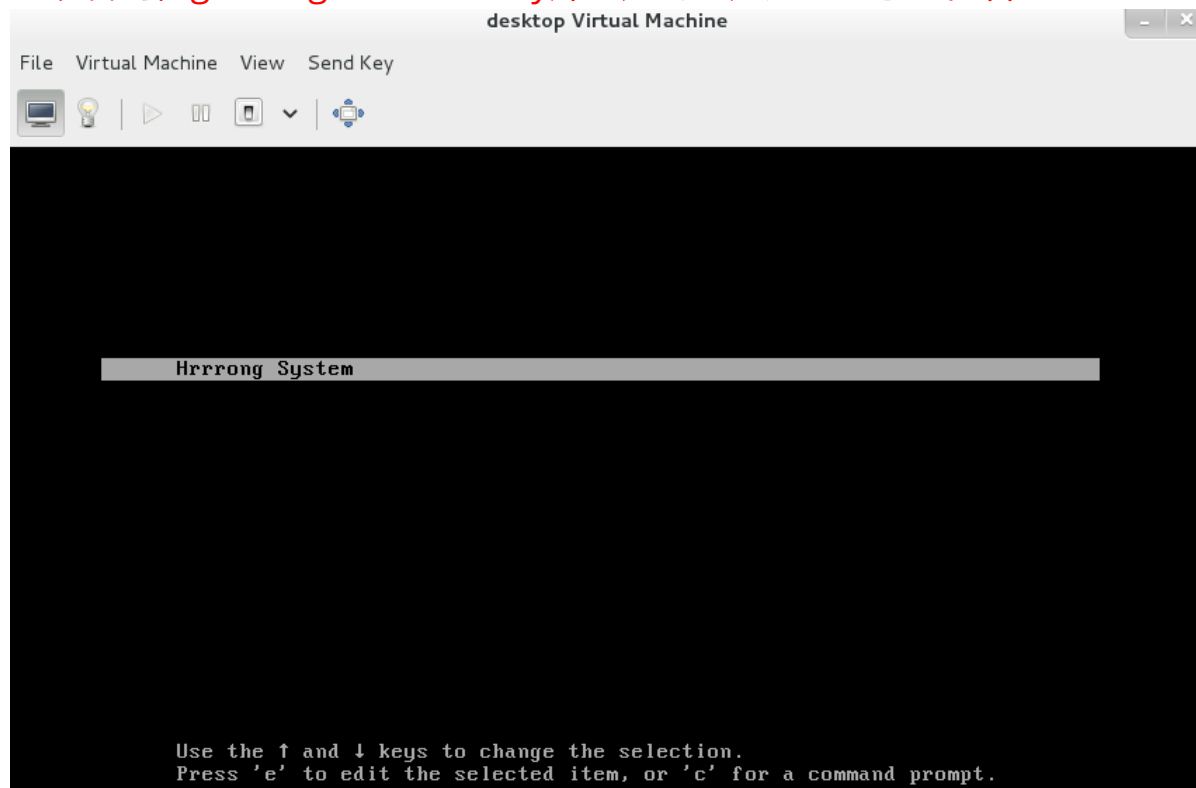
#vim /boot/grub2/grub.cfg #手动写一个

```
default=0
timeout=5

menuentry 'my rhel7.4' {
    root=(hd0,msdos1)
    linuxx16 /vmlinuz-3.10.0-693.el7.x86_64 root=/dev/mapper/rhel-root ro selinux=0
    initrd16 /initramfs-3.10.0-693.el7.x86_64.img
}
```

:wq保存退出，下面重启

可以看到，grub.cfg里menuentry那里是写的是这里显示的内容



实例五之内核失踪

1.将内核删了

```
desktop Virtual Machine
File Virtual Machine View Send Key
Applications Places Terminal
root@localhost:/boot
File Edit View Search Terminal Help
[root@localhost Desktop]# cd /boot/
[root@localhost boot]# ls
config-3.10.0-123.el7.x86_64
grub
grub2
initramfs-0-rescue-946cb0e817ea4adb916183df8c4fc817.img
initramfs-3.10.0-123.el7.x86_64.img
initrd-plymouth.img
symvers-3.10.0-123.el7.x86_64.gz
System.map-3.10.0-123.el7.x86_64
vmlinuz-0-rescue-946cb0e817ea4adb916183df8c4fc817
vmlinuz-3.10.0-123.el7.x86_64
[root@localhost boot]# mv vmlinuz-3.10.0-123.el7.x86_64 vmlinuz-3.10.0-123.el7.x
86_64.bk
[root@localhost boot]#
```

挂载光盘请看实例一

2.于是进rescue模式，安装内核

```
desktop Virtual Machine
File Virtual Machine View Send Key
Starting installer, one moment...
anaconda 19.31.79-1 for Red Hat Enterprise Linux 7.0 started.
Your system is mounted under the /mnt/sysimage directory.
When finished please exit from the shell and your system will reboot.
sh-4.2# ls
sh-4.2# df -TH
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/mapper/live-rw ext4       2.1G  1.1G  1.1G  49% /
devtmpfs         devtmpfs  1.1G    0  1.1G   0% /dev
tmpfs            tmpfs     1.1G    0  1.1G   0% /dev/shm
tmpfs            tmpfs     1.1G  8.7M  1.1G   1% /run
tmpfs            tmpfs     1.1G    0  1.1G   0% /sys/fs/cgroup
/dev/sda         iso9660    3.8G  3.8G    0 100% /run/install/repo
tmpfs            tmpfs     1.1G 148k  1.1G   1% /tmp
/dev/uda1        xfs       11G   3.2G   7.6G  30% /mnt/sysimage
devtmpfs         devtmpfs  1.1G    0  1.1G   0% /mnt/sysimage/dev
tmpfs            tmpfs     1.1G    0  1.1G   0% /mnt/sysimage/dev/shm
tmpfs            tmpfs     1.1G    0  1.1G   0% /dev/shm
tmpfs            tmpfs     1.1G  8.7M  1.1G   1% /mnt/sysimage/run
sh-4.2#
```

光盘

硬盘

将光盘里的内核包复制到硬盘并强制安装

```
Press Control+L+Alt+L to release pointer. desktop Virtual Machine

File Virtual Machine View Send Key

Starting installer, one moment...
anaconda 19.31.79-1 for Red Hat Enterprise Linux 7.0 started.

Your system is mounted under the /mnt/sysimage directory.
When finished please exit from the shell and your system will reboot.

sh-4.2# df -TH
Filesystem                Type      Size  Used Avail Use% Mounted on
/dev/mapper/live-rw       ext4       2.1G   1.1G   1.1G  49% /
devtmpfs                  devtmpfs   1.1G    0   1.1G   0% /dev
tmpfs                     tmpfs      1.1G    0   1.1G   0% /dev/shm
tmpfs                     tmpfs      1.1G   8.7M   1.1G   1% /run
tmpfs                     tmpfs      1.1G    0   1.1G   0% /sys/fs/cgroup
/dev/sda                   iso9660    3.8G   3.8G    0 100% /run/install/repo
tmpfs                     tmpfs      1.1G  144k   1.1G   1% /tmp
/dev/uda1                  xfs        11G   3.2G   7.6G  30% /mnt/sysimage
devtmpfs                  devtmpfs   1.1G    0   1.1G   0% /mnt/sysimage/dev
tmpfs                     tmpfs      1.1G    0   1.1G   0% /mnt/sysimage/dev/shm
tmpfs                     tmpfs      1.1G    0   1.1G   0% /dev/shm
tmpfs                     tmpfs      1.1G   8.7M   1.1G   1% /mnt/sysimage/run

sh-4.2# cp -rf /run/install/repo/
addons/                    GPL                      media.repo              RPM-GPG-KEY-redhat-be
discinfo                  images/                  Packages/                RPM-GPG-KEY-redhat-re
EFI/                      isolinux/                release-notes/           TRANS.TBL
EULA                      LiveOS/                  repodata/                tpsinfo

sh-4.2# cp -rf /run/install/repo/Packages/kernel-3.10.0-123.el7.x86_64.rpm /mnt/sysimage/boot/
sh-4.2#
```

```
desktop Virtual Machine

File Virtual Machine View Send Key

sh-4.2# chroot /mnt/sysimage/
bash-4.2# df -TH
df: /sys/fs/cgroup/systemd: No such file or directory
df: /sys/fs/cgroup/cpuset: No such file or directory
df: /sys/fs/cgroup/cpu,cpuacct: No such file or directory
df: /sys/fs/cgroup/memory: No such file or directory
df: /sys/fs/cgroup/devices: No such file or directory
df: /sys/fs/cgroup/freezer: No such file or directory
df: /sys/fs/cgroup/net_cls: No such file or directory
df: /sys/fs/cgroup/blkio: No such file or directory
df: /sys/fs/cgroup/perf_event: No such file or directory
df: /sys/fs/cgroup/hugetlb: No such file or directory
Filesystem                Type      Size  Used Avail Use% Mounted on
/dev/mapper/live-rw       ext4       11G   3.2G   7.6G  30% /
devtmpfs                  devtmpfs   1.1G    0   1.1G   0% /dev
tmpfs                     tmpfs      1.1G    0   1.1G   0% /dev/shm
/dev/sda                   iso9660    1.1G   8.7M   1.1G   1% /run/install/repo
hugetlbfs                 hugetlbfs  1.1G    0   1.1G   0% /dev/hugepages
/dev/uda1                  xfs        11G   3.2G   7.6G  30% /
devtmpfs                  devtmpfs   1.1G    0   1.1G   0% /dev
tmpfs                     tmpfs      1.1G    0   1.1G   0% /dev/shm
tmpfs                     tmpfs      1.1G    0   1.1G   0% /dev/shm
tmpfs                     tmpfs      1.1G   8.7M   1.1G   1% /run

bash-4.2#
```

切换到硬盘目录下

```

bash-4.2# cd boot/
bash-4.2# ls
config-3.10.0-123.el7.x86_64      kernel-3.10.0-123.el7.x86_64.rpm
grub                               symvers-3.10.0-123.el7.x86_64.gz
grub2                             System.map-3.10.0-123.el7.x86_64
initramfs-0-rescue-946cb0e817ea4adb916183df8c4fc817.img  vmlinuz-0-rescue-946cb0e817ea4adb916183df8c4fc817
initramfs-3.10.0-123.el7.x86_64.img  vmlinuz-3.10.0-123.el7.x86_64.bk
initrd-plymouth.img
bash-4.2# ls -l
total 102656
-rw-r--r--. 1 root root 122059 May 5 2014 config-3.10.0-123.el7.x86_64
drwxr-xr-x. 2 root root 37 May 7 2014 grub
drwxr-xr-x. 6 root root 4096 Aug 11 03:59 grub2
-rw-r--r--. 1 root root 30676364 May 7 2014 initramfs-0-rescue-946cb0e817ea4adb916183df8c4fc817.img
-rw-r--r--. 1 root root 30299761 May 7 2014 initramfs-3.10.0-123.el7.x86_64.img
-rw-r--r--. 1 root root 866987 Jun 7 2014 initrd-plymouth.img
-r--r--r--. 1 root root 30264588 Aug 11 04:36 kernel-3.10.0-123.el7.x86_64.rpm
-rw-r--r--. 1 root root 228562 May 5 2014 symvers-3.10.0-123.el7.x86_64.gz
-rw-----. 1 root root 2840084 May 5 2014 System.map-3.10.0-123.el7.x86_64
-rwxr-xr-x. 1 root root 4902000 May 7 2014 vmlinuz-0-rescue-946cb0e817ea4adb916183df8c4fc817
-rwxr-xr-x. 1 root root 4902000 May 5 2014 vmlinuz-3.10.0-123.el7.x86_64.bk
bash-4.2# rpm -ivh kernel-3.10.0-123.el7.x86_64.rpm --force
warning: kernel-3.10.0-123.el7.x86_64.rpm: Header V3 RSA/SHA256 Signature, key ID fd431d51: NOKEY
Preparing...
Updating / installing...
 1:kernel-3.10.0-123.el7

```

强制安装kernel

#rpm -ivh kernel-3.10.0-123.el7.x86_64.rpm -force #强制安装

```

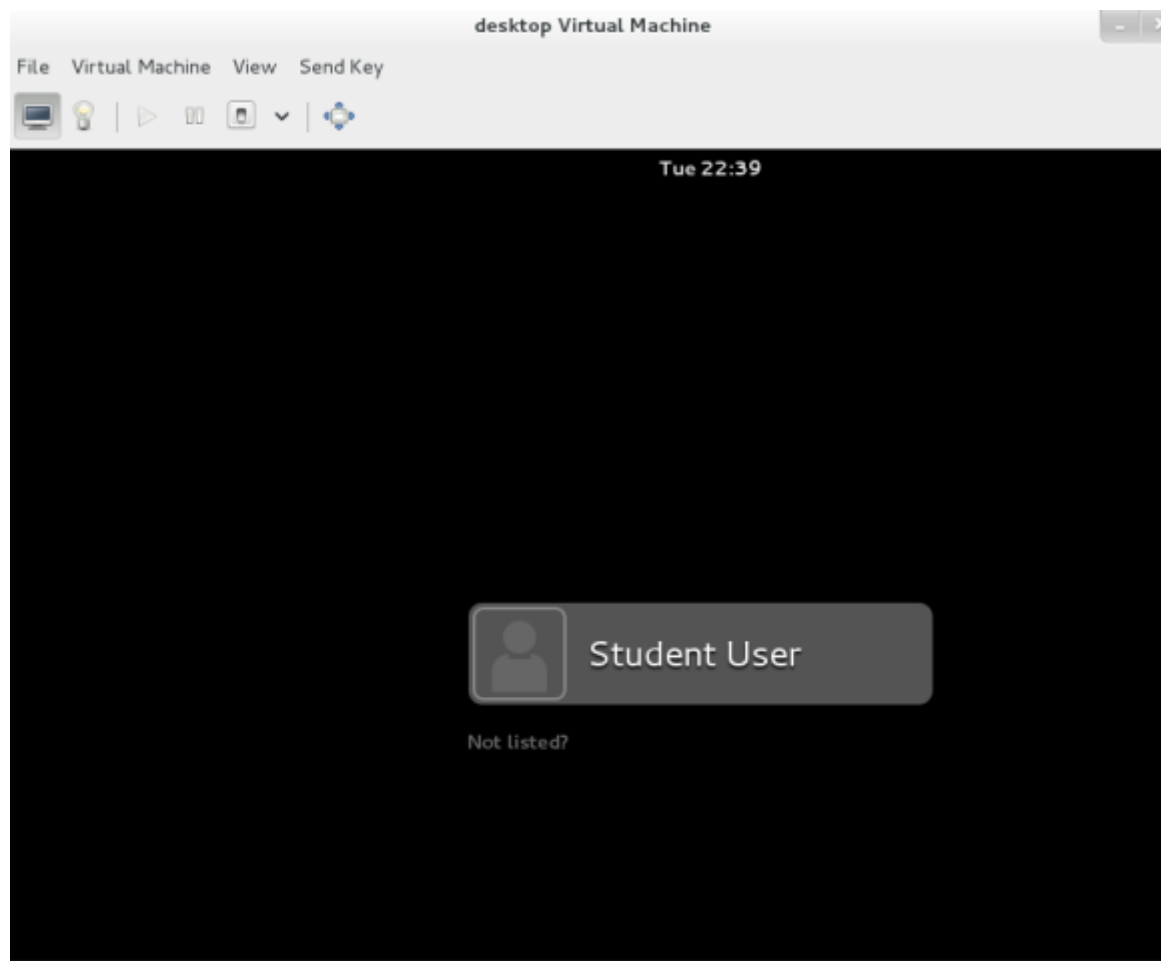
bash-4.2# rpm -ivh kernel-3.10.0-123.el7.x86_64.rpm --force
warning: kernel-3.10.0-123.el7.x86_64.rpm: Header V3 RSA/SHA256 Signature, key ID fd431d51: NOKEY
Preparing...
Updating / installing...
 1:kernel-3.10.0-123.el7
/usr/sbin/new-kernel-pkg: line 137: 1213 Segmentation fault $grubby --grub2 -c $grub2Config --add-k
TRD --copy-default $mkdefdefault --title "$title" $fmbkernel:++add-multiboot="$fmbkernel") $fmbargs:++mba
root=$rootdevice $kernargs" --remove-kernel="TITLE=$title"
/usr/sbin/new-kernel-pkg: line 385: 13145 Segmentation fault $grubby --grub2 -c $grub2Config --updat
INITRD $kernargs:++args="$kernargs"} $fremoveargs:++remove-args="$fremoveargs"}
bash-4.2#

```

2次exit重启

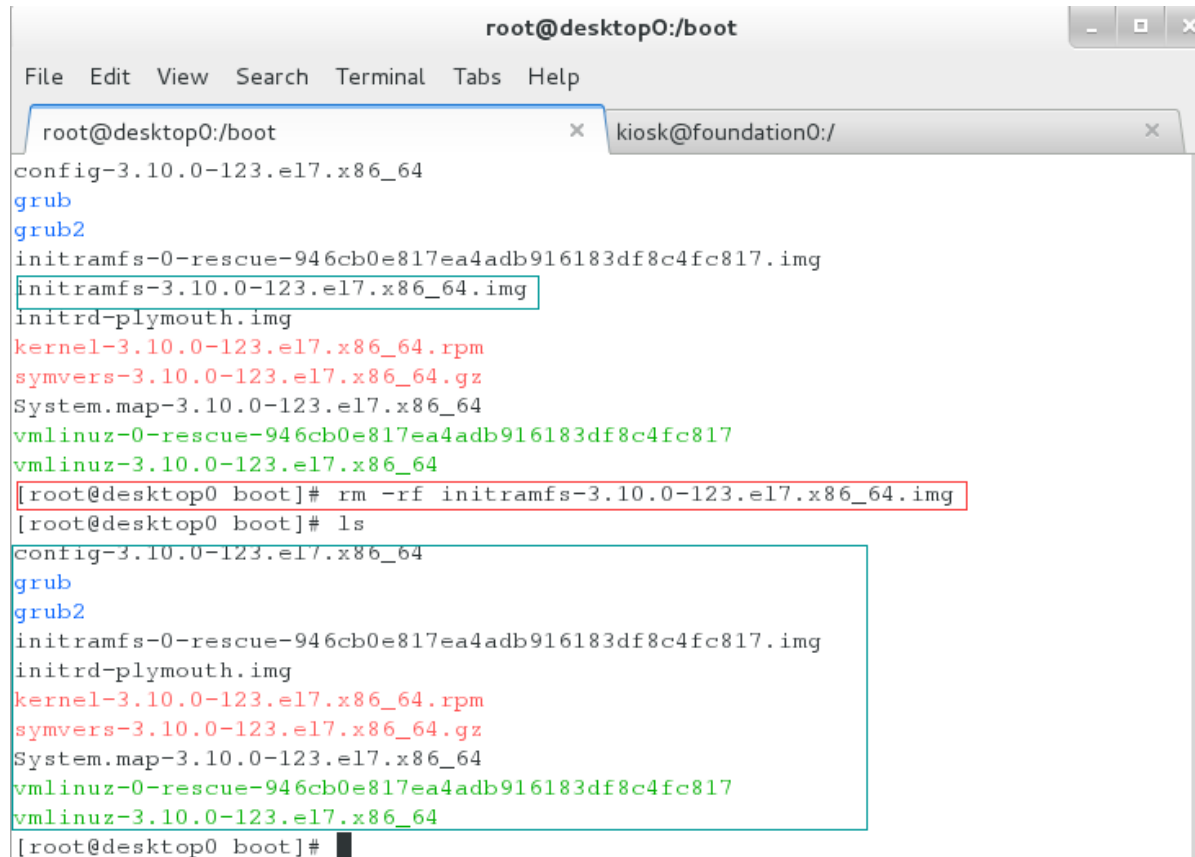
记得将光盘取消（请看实例一）

3.因此正常开机



实例六之驱动失踪

1. 将initramfs-3.10.0-123.el7.x86_64.img删除



A terminal window titled 'root@desktop0:/boot' with a menu bar (File, Edit, View, Search, Terminal, Tabs, Help). It shows a directory listing of the /boot directory. The file 'initramfs-3.10.0-123.el7.x86_64.img' is highlighted with a green box. Below it, the command 'rm -rf initramfs-3.10.0-123.el7.x86_64.img' is entered and highlighted with a red box. The subsequent 'ls' command shows that the file has been removed.

```
root@desktop0:/boot
File Edit View Search Terminal Tabs Help

root@desktop0:/boot x kiosk@foundation0:/
config-3.10.0-123.el7.x86_64
grub
grub2
initramfs-0-rescue-946cb0e817ea4adb916183df8c4fc817.img
initramfs-3.10.0-123.el7.x86_64.img
initrd-plymouth.img
kernel-3.10.0-123.el7.x86_64.rpm
symvers-3.10.0-123.el7.x86_64.gz
System.map-3.10.0-123.el7.x86_64
vmlinuz-0-rescue-946cb0e817ea4adb916183df8c4fc817
vmlinuz-3.10.0-123.el7.x86_64
[root@desktop0 boot]# rm -rf initramfs-3.10.0-123.el7.x86_64.img
[root@desktop0 boot]# ls
config-3.10.0-123.el7.x86_64
grub
grub2
initramfs-0-rescue-946cb0e817ea4adb916183df8c4fc817.img
initrd-plymouth.img
kernel-3.10.0-123.el7.x86_64.rpm
symvers-3.10.0-123.el7.x86_64.gz
System.map-3.10.0-123.el7.x86_64
vmlinuz-0-rescue-946cb0e817ea4adb916183df8c4fc817
vmlinuz-3.10.0-123.el7.x86_64
[root@desktop0 boot]#
```

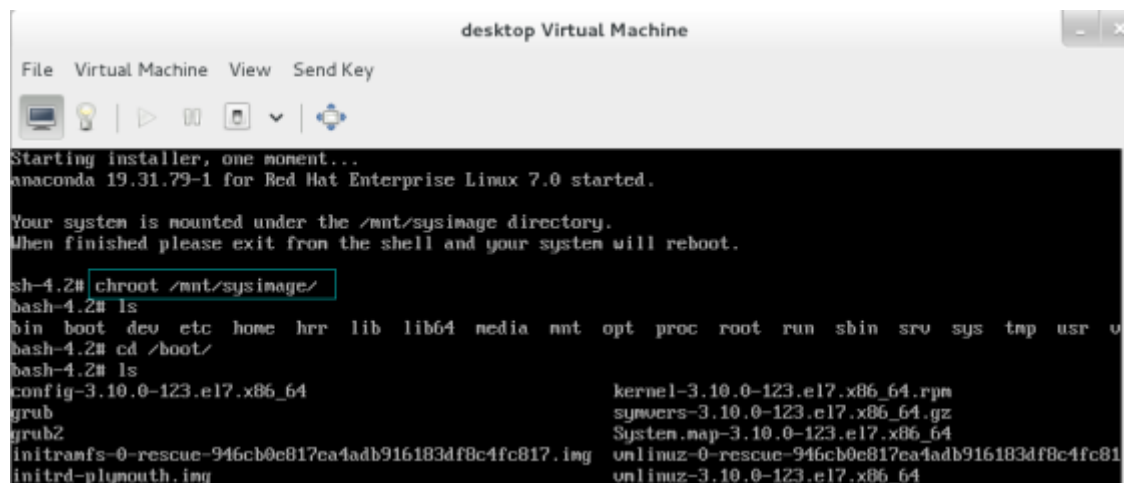
2. 于是进入rescue模式，生成initramfs驱动



A screenshot of a desktop virtual machine window titled 'desktop Virtual Machine'. The console shows an error message: 'error: file '/boot/initramfs-3.10.0-123.el7.x86_64.img' not found. Press any key to continue..._'. The window has a menu bar (File, Virtual Machine, View, Send Key) and a toolbar with icons for power, settings, and other VM controls.

```
desktop Virtual Machine
File Virtual Machine View Send Key

error: file '/boot/initramfs-3.10.0-123.el7.x86_64.img' not found.
Press any key to continue..._
```



A screenshot of a desktop virtual machine window titled 'desktop Virtual Machine'. The console shows the installation of anaconda. It starts with 'Starting installer, one moment...' and 'anaconda 19.31.79-1 for Red Hat Enterprise Linux 7.0 started.'. It then shows the system is mounted under '/mnt/sysimage' and prompts to exit the shell to reboot. The user enters 'chroot /mnt/sysimage/' and runs 'ls' in the chroot environment, showing the contents of the root directory.

```
desktop Virtual Machine
File Virtual Machine View Send Key

Starting installer, one moment...
anaconda 19.31.79-1 for Red Hat Enterprise Linux 7.0 started.

Your system is mounted under the /mnt/sysimage directory.
When finished please exit from the shell and your system will reboot.

sh-4.2# chroot /mnt/sysimage/
bash-4.2# ls
bin boot dev etc home hrr lib lib64 media mnt opt proc root run sbin srv sys tmp usr v
bash-4.2# cd /boot/
bash-4.2# ls
config-3.10.0-123.el7.x86_64          kernel-3.10.0-123.el7.x86_64.rpm
grub                                symvers-3.10.0-123.el7.x86_64.gz
grub2                              System.map-3.10.0-123.el7.x86_64
initramfs-0-rescue-946cb0e817ea4adb916183df8c4fc817  vmlinuz-0-rescue-946cb0e817ea4adb916183df8c4fc81
initrd-plymouth.img                vmlinuz-3.10.0-123.el7.x86_64
```

#mkinitrd -v initramfs-*(uname-r)*.img *(uname -r)* #生成
initramfs驱动

```
Skipping udev rule: 56-lvm.rules
Skipping udev rule: 60-persistent-storage-lvm.rules
*** Including module: ndraid ***
Skipping udev rule: 63-md-raid-arrays.rules
Skipping udev rule: 64-md-raid-assembly.rules
*** Including module: multipath ***
Skipping udev rule: 40-multipath.rules
*** Including module: qemu ***
*** Including module: qemu-net ***
*** Including module: fcoe ***
*** Including module: fcoe-uefi ***
*** Including module: iscsi ***
*** Including module: nfs ***
*** Including module: resume ***
*** Including module: rootfs-block ***
*** Including module: terminfo ***
*** Including module: udev-rules ***
Skipping udev rule: 91-permissions.rules
*** Including module: virtfs ***
*** Including module: biosdevname ***
*** Including module: systemd ***
*** Including module: usrmount ***
*** Including module: base ***
*** Including module: fs-lib ***
*** Including module: shutdown ***
*** Including module: uefi-lib ***
*** Including modules done ***
*** Installing kernel module dependencies and firmware ***
*** Installing kernel module dependencies and firmware done ***
*** Resolving executable dependencies ***
*** Resolving executable dependencies done ***
*** Hardlinking files ***
*** Hardlinking files done ***
*** Stripping files ***
*** Stripping files done ***
*** Generating early-microcode cpio image ***
*** Constructing GenuineIntel.bin ****
```

desktop Virtual Machine

File Virtual Machine View Send Key

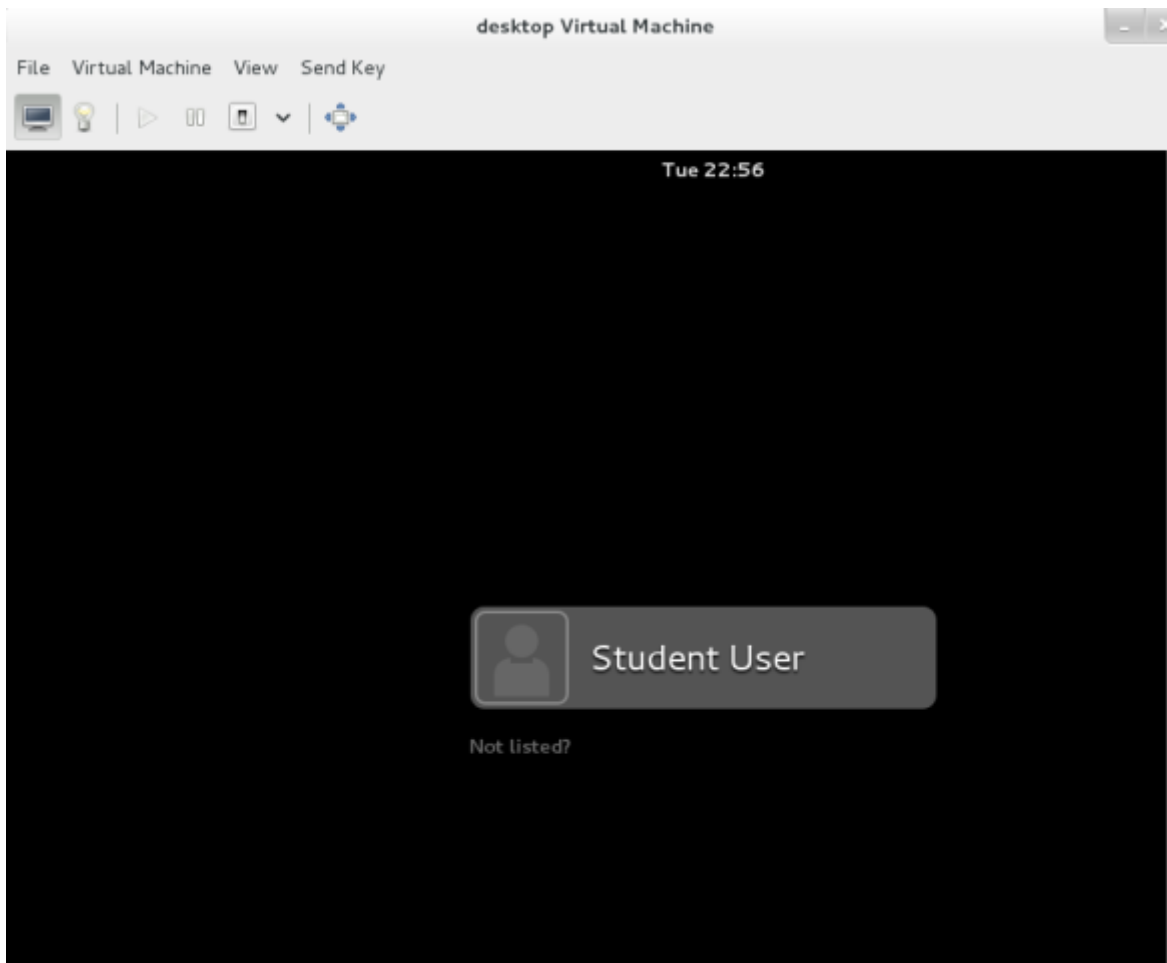
bash-4.2# ls
config-3.10.0-123.el7.x86_64
grub
grub2
initramfs-0-rescue-946cb0e817ea4adb916183df8c4fc817.img
initramfs-3.10.0-123.el7.x86_64.img
initrd-plymouth.img
bash-4.2#

kernel-3.10.0-123.el7.x86_64.rpm
syndevs-3.10.0-123.el7.x86_64.gz
System.map-3.10.0-123.el7.x86_64
vmlinuz-0-rescue-946cb0e817ea4adb916183df8c4fc81
vmlinuz-3.10.0-123.el7.x86_64

2次exit重启

记得将光盘取消（请看实例一）

\3. 因此正常开机



如果确保内核与驱动模块文件都存在，但是引导依然存在问题，则很有可能时grub.cfg文件问题,grub2-mkconfig -o /boot/grub2/grub.cfg解决或者重新正确的手写

实例七之ROOT密码被遗忘

<

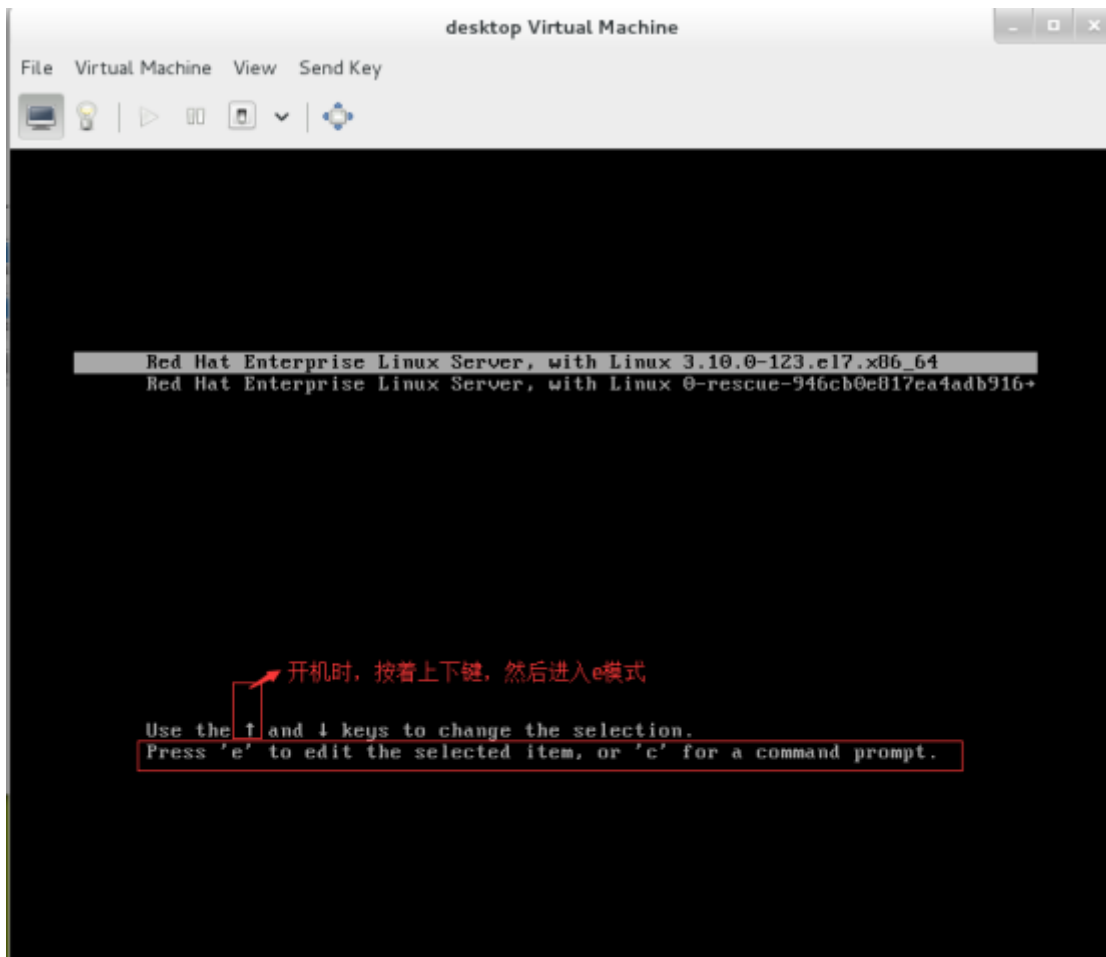
单用户模式:进启动系统核心部分，不启动第三方服务程序 basic.target
rescue.target

多用户模式:除图形界面之外,启动完整系统包括服务自启项. multi-
user.target

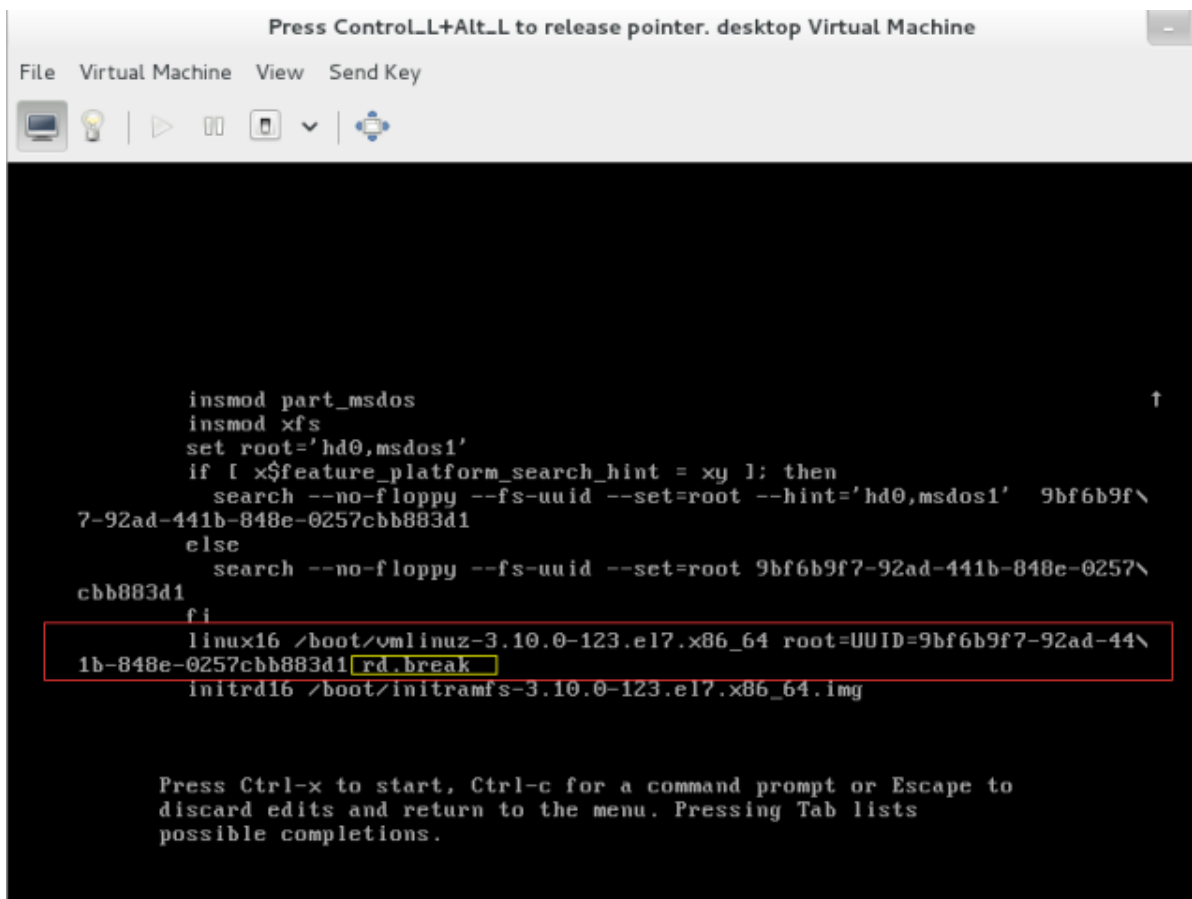
图形模式 :完整系统包括图形界面 graphical.target
systemctl set-default multi-user.target

>

1. 进入e模式, rd.break

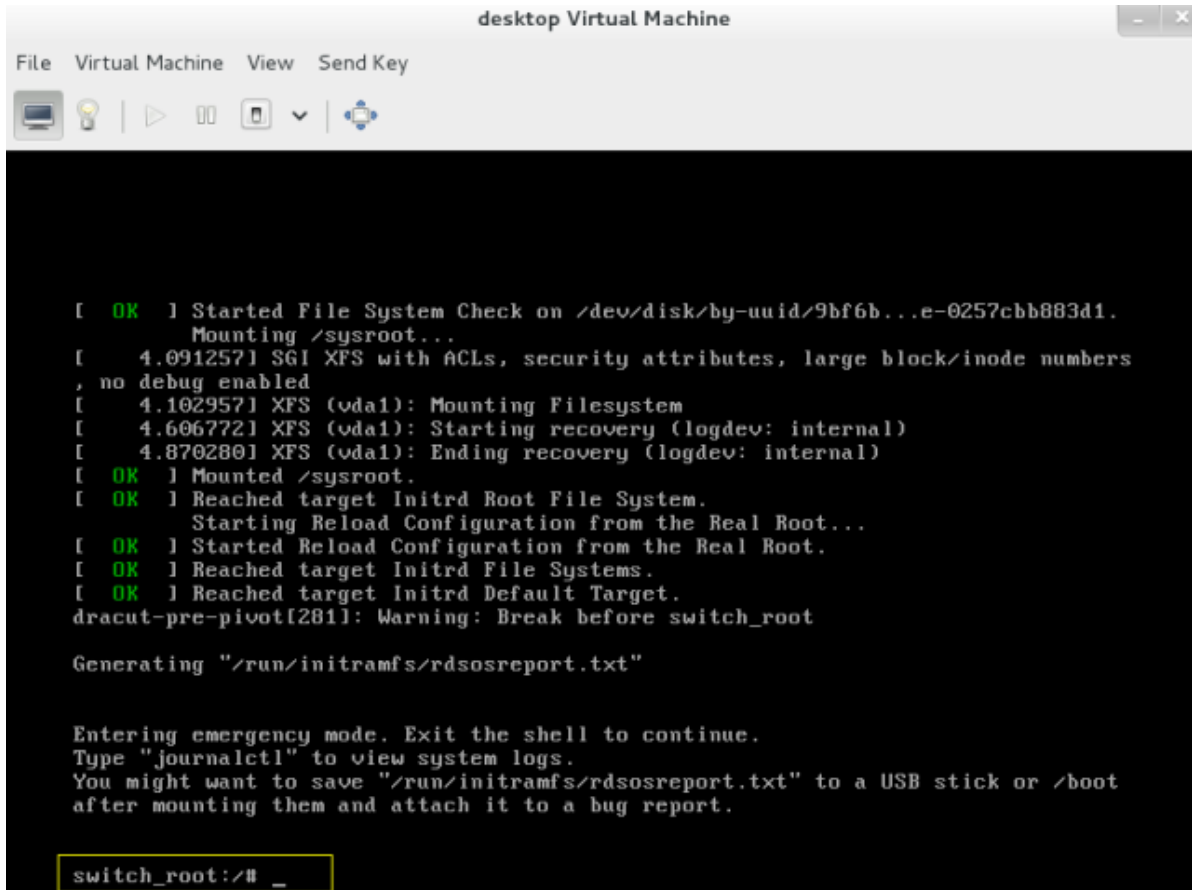


把rhgb quiet删除, 加上rd.break enforcing=0 (!!!!enforcing 关闭 selinux)



然后ctrl+x启动

\2. 进入switch-root#



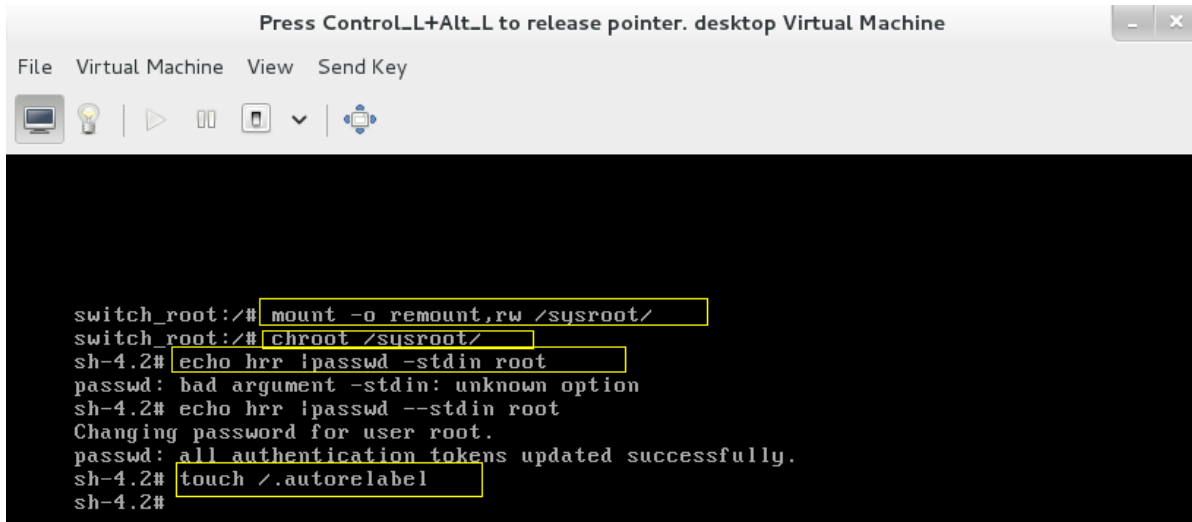
```
[ OK ] Started File System Check on /dev/disk/by-uuid/9bf6b...e-0257cbb883d1.
Mounting /sysroot...
[ 4.091257] SGI XFS with ACLs, security attributes, large block/inode numbers
, no debug enabled
[ 4.102957] XFS (vda1): Mounting Filesystem
[ 4.606772] XFS (vda1): Starting recovery (logdev: internal)
[ 4.870280] XFS (vda1): Ending recovery (logdev: internal)
[ OK ] Mounted /sysroot.
[ OK ] Reached target Initrd Root File System.
Starting Reload Configuration from the Real Root...
[ OK ] Started Reload Configuration from the Real Root.
[ OK ] Reached target Initrd File Systems.
[ OK ] Reached target Initrd Default Target.
dracut-pre-pivot[281]: Warning: Break before switch_root

Generating "/run/initramfs/rdsosreport.txt"

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

switch_root:/# _
```

\3. 改密码



```
switch_root:/# mount -o remount,rw /sysroot/
switch_root:/# chroot /sysroot/
sh-4.2# echo hrr | passwd --stdin root
passwd: bad argument --stdin: unknown option
sh-4.2# echo hrr | passwd --stdin root
Changing password for user root.
passwd: all authentication tokens updated successfully.
sh-4.2# touch /.autorelabel
sh-4.2#
```

Switch_root:/#mount -o remount,rw /sysroot/ #使可读可写

Switch_root:/#chroot /sysroot/ #切换根目录

Sh-4.2#echo hrr | passwd --stdin root #改root密码

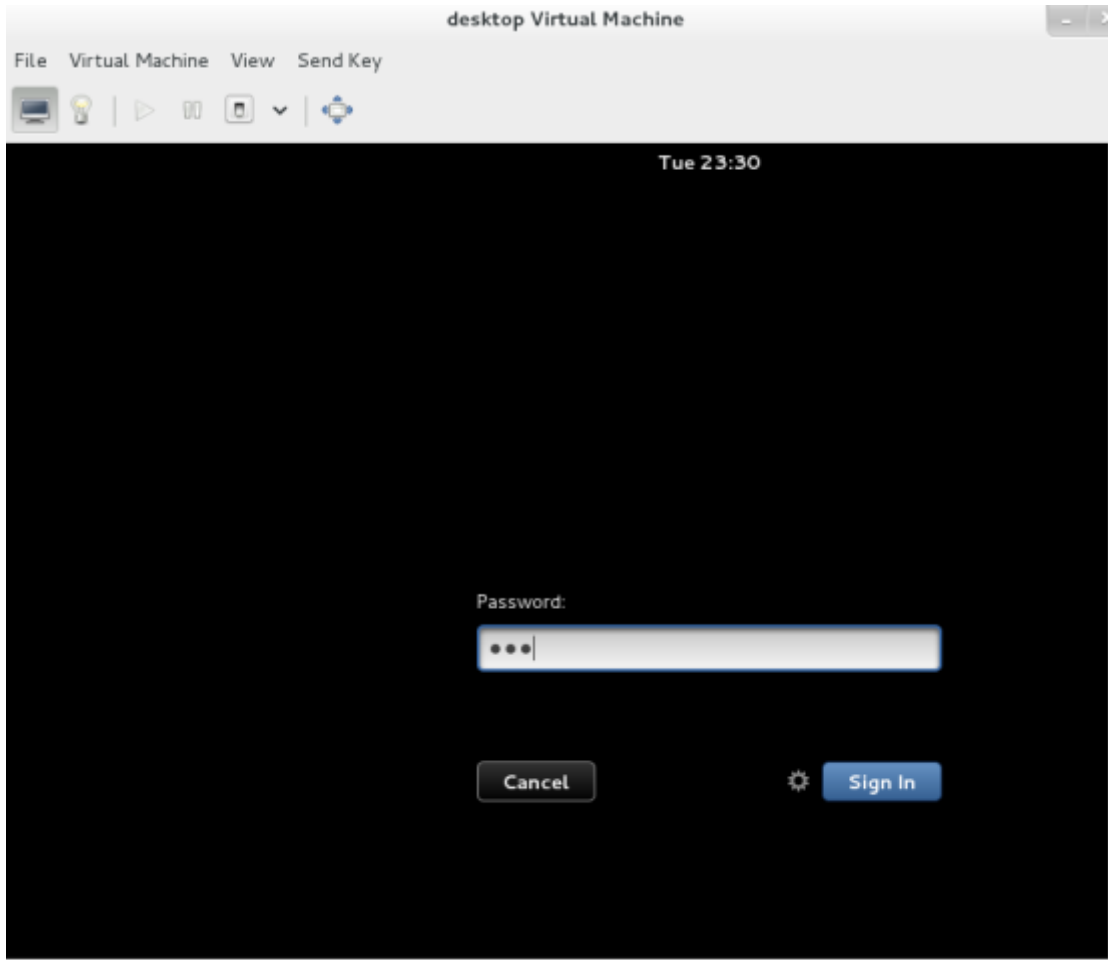
**Sh-4.2#touch /.autorelabel #重打标签，！！一定要有这个！！ <避免
selinux重新检查文件系统的更从而还原>**

Sh-4.2#exit

Switch_root:/#exit

2次exit进入系统

\4. 因此成功进入系统



=====

#在配置文件配置grub 用户及密码

```
[root@localhost ~]# vim /etc/grub.d/10_linux
```

```
cat <<EOF
```

```
set superusers="admin"
```

#明文密码

```
password admin 123456
```

```
EOF
```

#用命令生成密文密码grub2-mkpasswd-pbkdf2

```
cat <<EOF
```

```
set superusers="admin"
```

```
password_pbkdf2 admin
```

```
grub.pbkdf2.sha512.10000.2B3F99FEC1152CB38B1F958206FEB57BE4  
6CC747363DF1FA401B73D13F2645E9CAB50BF4A21AFE98A7ECD38545  
2EF830903DADC8E2DBA34AB98E5E521B24E3FA.A4FDA961327E16B1F
```

7B744E17C79688A8E17EF1B846AF9B90A25B73A6A76B8B385070A0EB
067E49B503B6861C7B7F9B8329C0DBA62F40DC3D527C88E875B23FA
EOF

#根据10_linux配置生成新的grub2/grub.cfg文件

[root@localhost ~]# grub2-mkconfig -o /boot/grub2/grub.cfg

