

GRUB工作原理及安装使用

可以正常工作的grub包括的文件：

```
1 [localhost@root ~]tree /boot/grub/
2 /boot/grub/
3 |-- device.map
4 |-- e2fs_stage1_5
5 |-- fat_stage1_5
6 |-- ffs_stage1_5
7 |-- grub.conf
8 |-- iso9660_stage1_5
9 |-- jfs_stage1_5
10 |-- menu.lst -> ./grub.conf
11 |-- minix_stage1_5
12 |-- reiserfs_stage1_5
13 |-- splash.xpm.gz
14 |-- stage1
15 |-- stage2
16 |-- ufs2_stage1_5
17 |-- vstafs_stage1_5
18 `-- xfs_stage1_5
19
20 0 directories, 16 files
```

GRUB是两段式的引导，其中stage1要被装入某个硬盘的RBM区域存储，stage1的作用就是找到stage2文件存放的位置，来完成后续工作。stage2文件存放的硬盘必定是某种文件系统。因为RBM区域容量限制，stage1无法识别文件系统，所有需要调用stage1_5文件，比如把stage2存在ext2格式的文件系统，则需要调用e2fs_stage1_5文件。这时候找到stage2文件并把控制权交给stage2。它会读取配置文件/boot/grub/grub.conf或menu.lst等，根据其中的配置加载操作系统。

GRUB的安装

通常需要重新安装grub的情况有：

- 给硬盘安装grub
- grub.conf文件损坏
- grub被破坏掉

给硬盘安装grub

安装grub

```
1 #添加三个分区，并调整交换分区类型为82
2 fdisk /dev/sdb
3 #格式化分区
4 mke2fs /dev/sdb1
5 mke2fs /dev/sdb2
6 mkswap /dev/sdb3
7 mkdir /mnt/boot #创建目录
8 mount /dev/sdb1 /mnt/boot #挂载设备
9 grub-install --root-directory=/mnt /dev/sdb #安装
```

为硬盘的grub添加配置文件（测试）

```
1 vim /mnt/boot/grub/grub.conf
2 default=0
3 timeout=10
4 title test Linux
5     root (hd0,0)
6     kernel /test
7     initrd /test
8
9 sync #写入内存
10 umont /mnt/boot #卸载硬盘
```

测试grub是否可用

- 新建一个虚拟机，将硬盘添加至虚拟机，即可测试是否可用。

当grub.conf文件损坏

grub.conf损坏或丢失时，机器启动后，会看到grub提示界面，因为grub.conf损坏，所以并不能正常找到内核的位置并加载内核启动机器。如下图：

```
GNU GRUB version 0.97 (635K lower / 1046400K upper memory)
[ Minimal BASH-like line editing is supported.  For the first word, TAB
  lists possible command completions.  Anywhere else TAB lists the possible
  completions of a device/filename.]

grub> _
```

因为我们并不知道内核所在分区，所以我们用命令来试着找一下内核的位置：

```
1 | grub> find (hdN,N)/ #按tab键显示如下
```

```
grub> find (hd0,0)/
Possible files are: lost+found grub efi .vmlinuz-2.6.32-431.el6.x86_64.hmac Sy
stem.map-2.6.32-431.el6.x86_64 config-2.6.32-431.el6.x86_64 symvers-2.6.32-431.
el6.x86_64.gz vmlinuz-2.6.32-431.el6.x86_64 initramfs-2.6.32-431.el6.x86_64.img

grub> find (hd0,0)/_
```

找到内核所在位置后进行如下操作：

```
1 | grub> root (hd0,0)
2 | grub> kernel /vmlinuz-2.6.18-64.el5 ro
   root=/dev/mapper/volGroup-lv_root rd_NO_LUKS rd_NO_MD
   rd_LVM_LV=volGroup/lv_swap
   crashkernel=auto LANG=zh_CN.UTF-8
   rd_LVM_LV=volGroup/lv_root
3 |     KEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM rhgb quiet
4 | grub> initrd /initrd #按tab键
5 | grub> boot
```

这样grub就安装成功了。

注意：当你指定内核后，但未指定内核后面的参数（ro root=(此处未指定的话)）将无法启动，请给root参数，一般情况下是系统是可以自动探测到，但这个功能并不靠谱，那么只能靠备份或你的记忆将参数补上，所以定期备份/etc/fstab、与grub.conf是非常重要的工作。

当grub损坏

如MBR数据损坏（仅是bootloader损坏，分区表是好的），如果没有重新启动还可能修复，但是如果是重启后发现grub损坏，那么只能挂载光盘进入紧急救援模式。（以下将以挂载光盘说明）。

```
1 #模拟grub损坏
2 [localhost@root ~]#dd if=/dev/zero of=/dev/sda
   count=1 bs=100
3 [localhost@root ~]#sync
```

如果系统未重启

方式一：

```
1 [localhost@root ~]grub #进入greb命令行
2 >grub root(hd0,0) #指定root所在分区
3 >grub setup (hd0) #安装grub,并指定硬盘
4 >grub reboot #退出
5 [sum:4 localhost@root ~]#shutdow -r now #重启
```

方式二：

```
1 [localhost@root ~]grub-install --root-
   directory=/path/to/boot_part_dir /path/to/device
2 #指定grub所在的分区，以及所要安装grub的设备
```

如果系统已重启

模拟grub损坏之后重启电脑，发现已经无法正常进入系统，如图：

```
Network boot from Intel E1000
Copyright (C) 2003-2014 VMware, Inc.
Copyright (C) 1997-2000 Intel Corporation

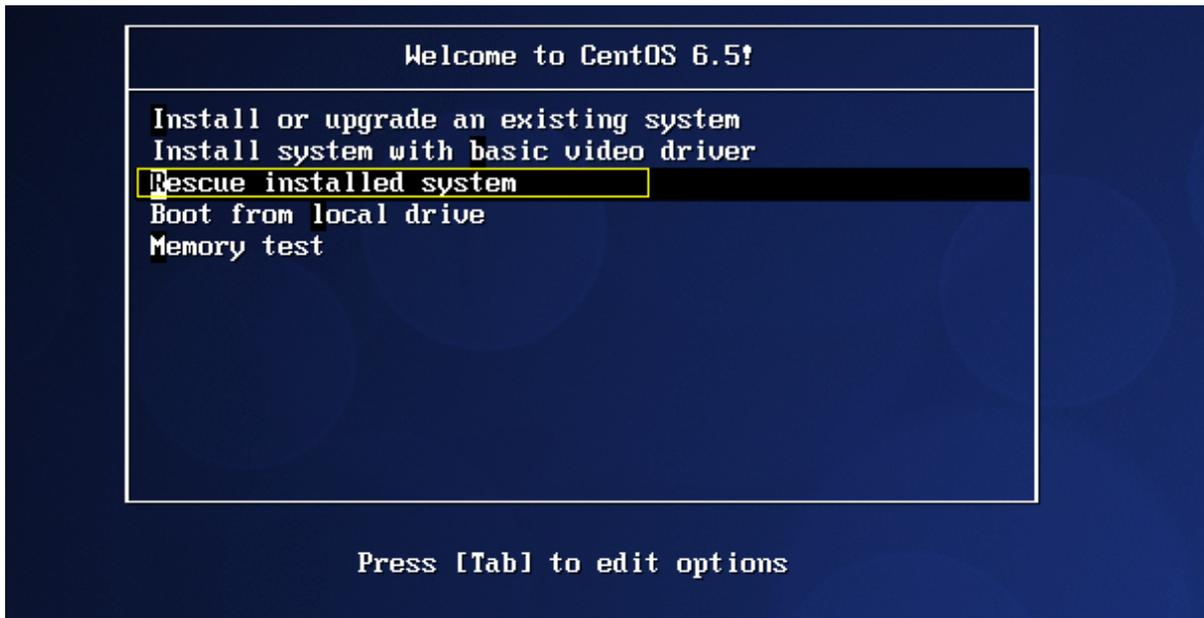
CLIENT MAC ADDR: 00 0C 29 66 9B F8  GUID: 564DF126-65A1-4046-5C3F-8E87CD669BF8
PXE-E53: No boot filename received

PXE-M0F: Exiting Intel PXE ROM.
Operating System not found

-
```

拯救模式修复grub

挂载光盘并进入BIOS界面，选择BOOT选项，将CD-ROM Drive设置为第一启动项，保存退出。重启后，选择 `Rescue installed system`



选择选项之后，会开启一个新的shell如图：



然后如下操作：

```
1 bash-4.1# chroot /mnt/sysimage #切换到原系统
2 sh-4.1# root (hd0,0)
3 sh-4.1# setup (hd0)
```

最后下载光盘，重启机器，修复完毕。

GRUB的配置文件

/boot/grub/grub.comf 或者 /etc/grub.conf

```
1 default=0 #设定默认启动的title编号，即系统
2 timeout=5 #等待用户选择的超时时长，单位是秒
3 splashimage=(hd0,0)/grub/splash.xpm.gz #指定背景图片
4 hiddenmenu #隐藏菜单
5 password 密码字符
6 password --md5 #加密的密码使用grub-md5-crypt命令生成
7 title CentOS (2.6.32-431.el6.x86_64) #引导的内核的标题
  或操作系统标题，可自由修改
8     root (hd0,0) #内核文件所在的设备：对于grub硬盘为hd，
  第一个数字表示第一块磁盘，第二个数字表示对应磁盘的分区
9     kernel /vmlinuz-2.6.32-431.el6.x86_64 ro
  root=/dev/mapper/VolGroup-lv_root rd_NO_LUKS
  rd_NO_MD rd_LVM_LV=VolGroup/lv_swap
  crashkernel=auto LANG=zh_CN.UTF-8
  rd_LVM_LV=VolGroup/lv_root KEYBOARDTYPE=pc
  KEYTABLE=us rd_NO_DM rhgb quiet #内核文件路径，及传递
  给内核的参数 /proc/cmdline
10     initrd /initramfs-2.6.32-431.el6.x86_64.img #虚
  根
11     password --md5 #加密的密码使用grub-md5-crypt命令生
  成。启动内核需要密码
```