



南昌大学实验报告

学生姓名：__丁俊__ 学 号：__8003119100__ 专业班级：__信息安全 193 班__
实验类型：☐ 验证 ☒ 综合 ☐ 设计 ☐ 创新 实验日期：__2021.11.12__ 实验成绩：_____

一、 实验项目名称

Linux 期中测试报告

二、 实验目的

1. 了解常用的 shell 命令。
2. 熟悉远程管理工具 Xshell 的基本使用。
3. 熟悉 Linux 系统的引导步骤。
4. 掌握在 GRUB 设置启动项的操作。
5. 掌握关机命令的使用。
6. 掌握目录和文件的操作。
7. 掌握文件权限的设置。
8. 了解用户和组的概念。
9. 掌握目录和文件的操作。
10. 掌握文件权限的设置。
11. 掌握创建用户和组的操作命令。
12. 了解 Linux 下软件包的概念。
13. 掌握 YUM 软件包的管理。
14. 掌握 Linux 网络配置方法。
15. 掌握网络通信管理操作。
16. 掌握进程管理的操作。
17. 了解 Linux 系统服务进程的概念。
18. 熟悉 Linux 磁盘操作管理的相关内容。
19. 掌握进程管理的操作。

三、 实验要求

通过上机实际操作，总结验证相关知识，要求按照先说明实验原理，再通过实验过程验证，最后总结反思的方式进行。

四、主要仪器设备及耗材

计算机、VMware、CentOS 7、word

五、实验原理

1、组和用户

如果 root 工作比较忙，没有时间来管理组，可以交由组管理来进行管理
gpasswd 命令是 Linux 下工作组文件/etc/group 和/etc/gshadow 管理工具。

参数说明

无参数：gpasswd 给组创建一个密码，用户加入该组需要输入密码

-a: 添加用户到组；

-d: 从组删除用户；

-A: 指定管理员；

-M: 指定组成员和-A 的用途差不多；

-r: 删除密码；

-R: 限制用户登入组，只有组中的成员才可以用 newgrp 加入该组。

配置实例：

1) 给 test 组创建一个密码

```
[root@localhost tmp]# gpasswd test
```

Changing the password for group test

New Password:

Re-enter new password:

2) 添加 user1，让 user1 来管理 test 组

```
[root@localhost tmp]# useradd user1
```

```
[root@localhost tmp]# gpasswd -A user1 test
```

3) 创建 user2，user3 用户，并且让 user1 添加这两个账户到 test 组中

```
[root@localhost tmp]# useradd user2
```

```
[root@localhost tmp]# useradd user
```

```
[root@localhost tmp]# su - user1
```

```
[user1@localhost ~]$ gpasswd -a user2 test
```

Adding user user2 to group test

```
[user1@localhost ~]$ gpasswd -a user3 test
```

Adding user user3 to group test

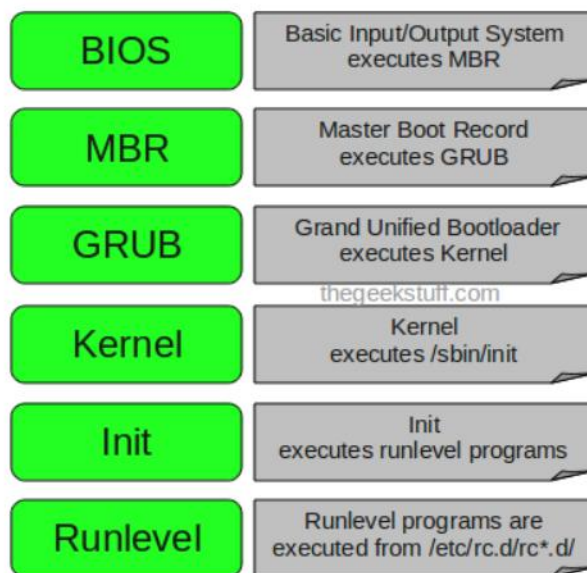
4) 查看/etc/group 文件进行验证

```
[root@localhost ~]# tail -n 10 /etc/group | grep test
```

```
test:x:1001:user2,user3
```

2、Grub 免密登录

为了跳过系统正常启动过程中的某些步骤，必须知道大致的系统启动流程是什么样的。大概情况如下：



加电---->BIOS—>启动介质—>先读第一个扇区，引导 MBR—>GRUB—>找到 kernel 加载到内存—>执行第一个程序 init 单。用户模式是从 GRUB 中跳过了用户认证，直接运行 bin/sh,进而改 root 用户密码或者去掉密码。救援模式是从 BIOS 中更改启动首选项，从系统光盘中引导系统启动，再回来更改 root 用户文件，达到更改密码或者去掉密码的目的。

3、SGID 有以下特点

SGID 只针对可执行文件有效，换句话说，只有可执行文件才可以被赋予 SGID 权限，普通文件赋予 SGID 没有意义。用户需要对此可执行文件有 x 权限；用户在执行具有 SGID 权限的可执行文件时，用户的群组身份会变为文件所属群组；SGID 权限赋予用

户改变组身份的效果，只在可执行文件运行过程中有效；当一个目录被赋予 SGID 权限后，进入此目录的普通用户，其有效群组会变为该目录的所属组，会就使得用户在创建文件（或目录）时，该文件（或目录）的所属组将不再是用户的所属组，而使用的是目录的所属组。

SetGID 对目录的作用：

事实上，SGID 也能作用于目录，且这种用法很常见。

当一个目录被赋予 SGID 权限后，进入此目录的普通用户，其有效群组会变为该目录的所属组，会就使得用户在创建文件（或目录）时，该文件（或目录）的所属组将不再是用户的所属组，而使用的是目录的所属组。

也就是说，只有当普通用户对具有 SGID 权限的目录有 rwx 权限时，SGID 的功能才能完全发挥。比如说，如果用户对该目录仅有 rx 权限，则用户进入此目录后，虽然其有效群组变为此目录的所属组，但由于没有 x 权限，用户无法在目录中创建文件或目录，SGID 权限也就无法发挥它的作用。

举个例子：

```
[root@localhost ~]# cd /tmp
```

#进入临时目录做此实验。因为只有临时目录才允许普通用户修改

```
[root@localhost tmp]# mkdir dtest
```

#建立测试目录

```
[root@localhost tmp]# chmod g+s dtest
```

#给测试目录赋予 SetGID 权限

```
[root@localhost tmp]# ll -d dtest
```

```
drwxr-sr-x 2 root root 4096 Apr 19 06:04 dtest
```

#SetGID 权限已经生效

```
[root@localhost tmp]# chmod 777 dtest
```

#给测试目录赋予 777 权限，让普通用户可以写

```
[root@localhost tmp]# su - lamp
```

```
[lamp@localhost ~]# grep lamp /etc/passwd /etc/group
```

```
/etc/passwd:lamp:x:501:501::/home/lamp:/bin/bash
```

```
/etc/group:lamp:x:501:
```

#切换成普通用户 lamp，此用户的所属群组为 lamp

```
[lamp@localhost ~]$ cd /tmp/dtest/
```

#普通用户进入测试目录

```
[lamp@localhost dtest]$ touch abc
[lamp@localhost dtest]$ mkdir zimulu
#在此目录中创建新的文件 abc 和子目录 zimulu
[lamp@localhost dtest]$ ll
total 0
-rw-rw-r--. 1 lamp root 0 Apr 19 06:07 abc
drwxrwsr-x. 2 lamp root 40 Apr 19 06:07 zimulu
```

六、实验过程

1、添加一个用户和组，并把这个用户设置为该组的管理员。

```
[root@localhost ~]# useradd dk
[root@localhost ~]# groupadd dg
[root@localhost ~]# gpasswd -a dk dg // 将 dk 加入 dg 组
正在将用户“dk”加入到“dg”组中
[root@localhost ~]# gpasswd -A dk dg // 将 dk 设置为 dg 的管理员
```

2、通过设置 grub 使得 root 用户可以免密登录。

设置单用户模式

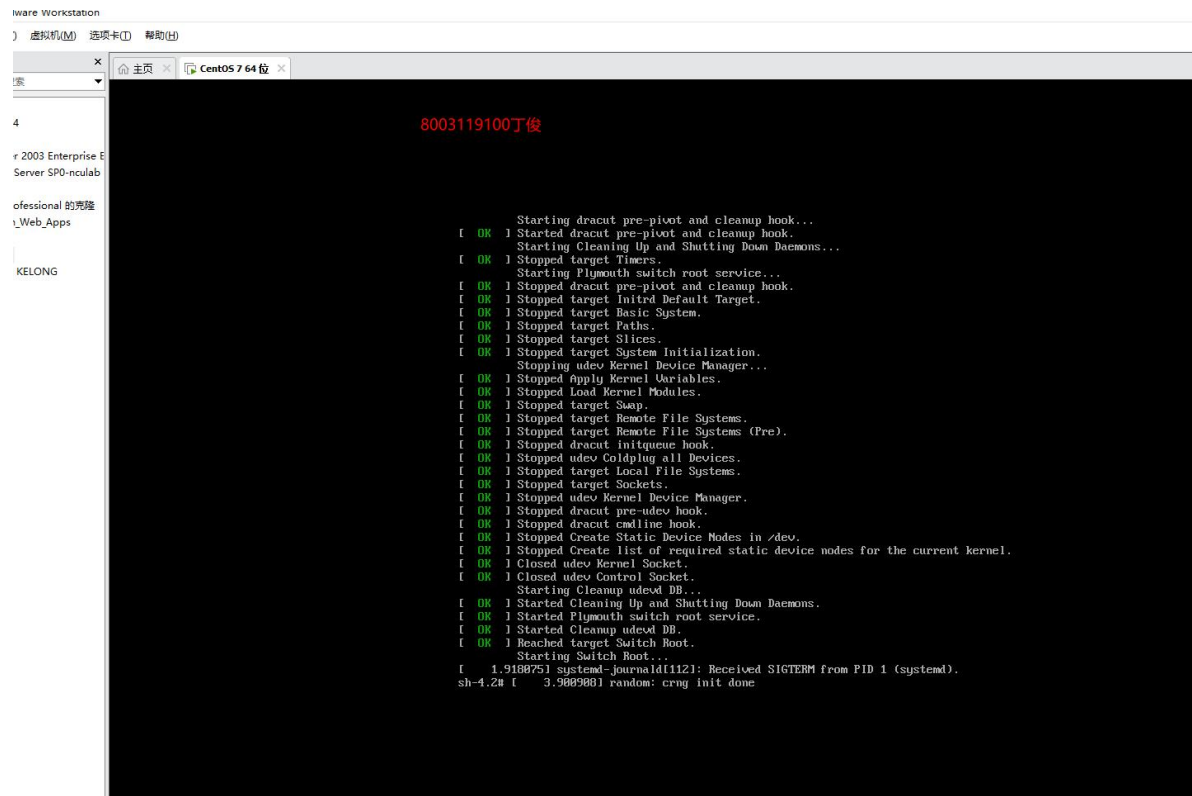
将 ro 改为 r\w,

在 UTF-8 后面添加在 UTF-8 后面添加 init=/bin/shinit=/bin/sh

8003119100丁俊

```
setparams 'CentOS Linux (3.10.0-1127.el7.x86_64) 7 (Core)'\n\nload_video\nset gfxpayload=keep\ninsmod gzio\ninsmod part_msdos\ninsmod xfs\nset root='hd0,msdos1'\nif [ x${feature_platform_search_hint} = xy ]; then\n    search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hin\nt-efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1 --hint='hd0,msdos1' 3c271069-4\nddb-435a-acd3-fb0a5201d20a\nelse\n    search --no-floppy --fs-uuid --set=root 3c271069-4ddb-435a-acd3-fb0a\5201d20a\n\nPress Ctrl-x to start, Ctrl-c for a command prompt or Escape to\ndiscard edits and return to the menu. Pressing Tab lists\npossible completions.
```

按 ctrl + x 启动



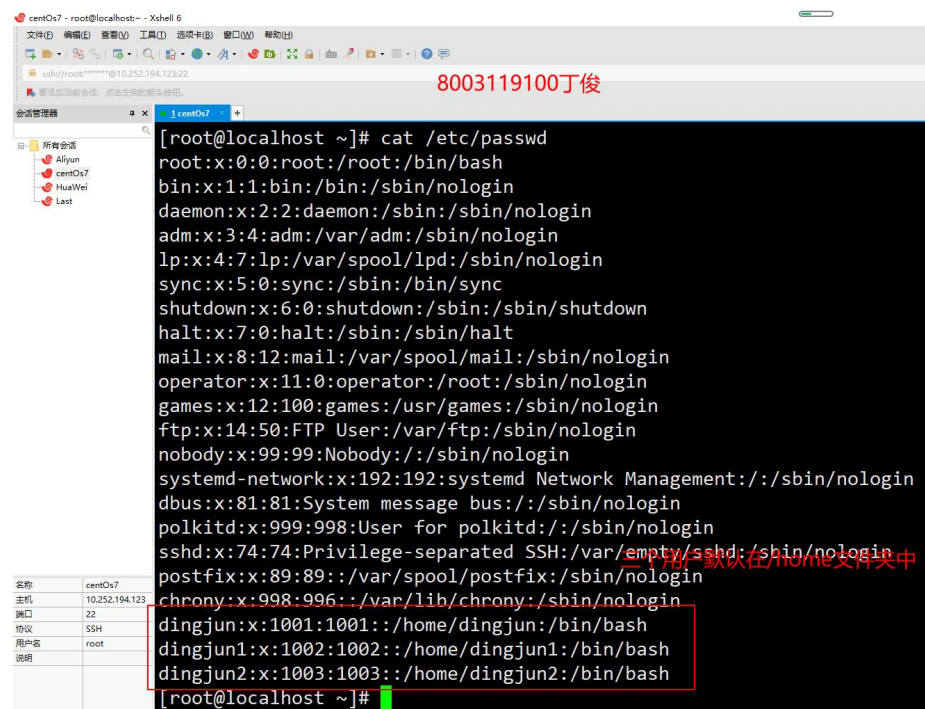
3、以自己姓名全拼的普通用户的身份创建一个共享目录 share，利用文件的特殊权限实现同一部门内的所有人都能读取共享目录的内容（至少需要 3 个不同用户进行验证）。

步骤一： 首先创建好三个用户 dingjun、dingjun1、dingjun2(默认文件夹)

```
[root@localhost ~]# useradd dingjun1
```

```
[root@localhost ~]# useradd dingjun2
```

```
[root@localhost ~]# useradd dingjun3
```



```
[root@localhost ~]# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
chrony:x:998:996::/var/lib/chrony:/sbin/nologin
dingjun:x:1001:1001:./home/dingjun:/bin/bash
dingjun1:x:1002:1002:./home/dingjun1:/bin/bash
dingjun2:x:1003:1003:./home/dingjun2:/bin/bash
[root@localhost ~]#
```

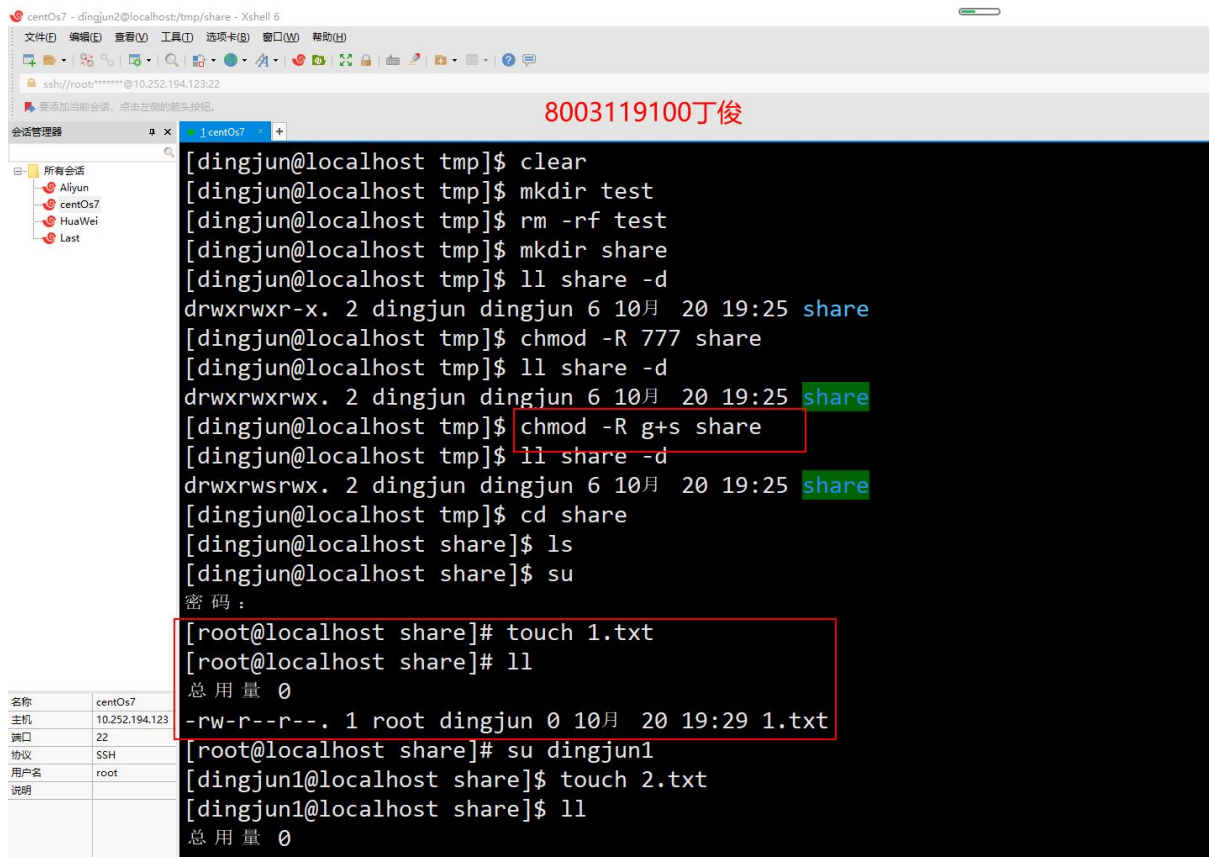
三个用户默认在/home文件夹中

步骤二： 创建 Dj 文件夹，设置 sgid 权限

```
[dingjun1@localhost tmp]$ mkdir Dj
```

```
[dingjun1@localhost tmp]$ chmod -R 777 Dj
```

```
[dingjun1@localhost tmp]$ chmod -R g+s Dj
```



```
centOs7 - dingjun2@localhost/tmp/share - Xshell 6
文件(F) 编辑(E) 查看(V) 工具(T) 选项卡(O) 窗口(W) 帮助(H)
ssh://root@10.252.194.123:22
会话管理器
所有会话
Aliyun
centOs7
HuaWei
Last
[dingjun@localhost tmp]$ clear
[dingjun@localhost tmp]$ mkdir test
[dingjun@localhost tmp]$ rm -rf test
[dingjun@localhost tmp]$ mkdir share
[dingjun@localhost tmp]$ ll share -d
drwxrwxr-x. 2 dingjun dingjun 6 10月 20 19:25 share
[dingjun@localhost tmp]$ chmod -R 777 share
[dingjun@localhost tmp]$ ll share -d
drwxrwxrwx. 2 dingjun dingjun 6 10月 20 19:25 share
[dingjun@localhost tmp]$ chmod -R g+s share
[dingjun@localhost tmp]$ ll share -d
drwxrwsrwx. 2 dingjun dingjun 6 10月 20 19:25 share
[dingjun@localhost tmp]$ cd share
[dingjun@localhost share]$ ls
[dingjun@localhost share]$ su
密码:
[root@localhost share]# touch 1.txt
[root@localhost share]# ll
总用量 0
-rw-r--r--. 1 root dingjun 0 10月 20 19:29 1.txt
[root@localhost share]# su dingjun1
[dingjun1@localhost share]$ touch 2.txt
[dingjun1@localhost share]$ ll
总用量 0
```

名称	centOs7
主机	10.252.194.123
端口	22
协议	SSH
用户名	root
说明	

步骤三: dingjun2 用户在共享文件夹 Dj 中新建文件

```
[root@localhost tmp]# su dingjun2
```

```
[dingjun2@localhost tmp]$ cd Dj
```

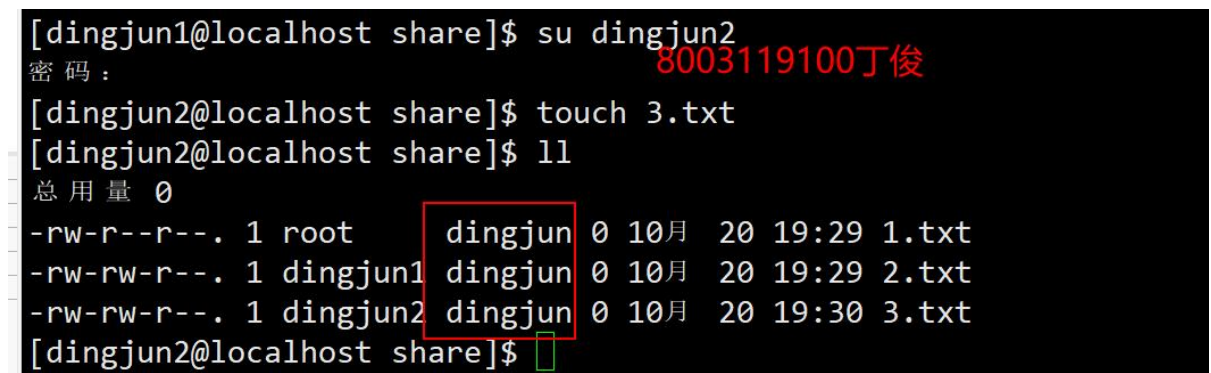
```
[dingjun2@localhost Dj]$ touch 1.txt
```

```
[dingjun2@localhost Dj]$ touch 2.txt
```

步骤四: dingjun3 用户在共享文件夹 Dj 中新建文件

```
[root@localhost Dj]# su dingjun3
```

```
[dingjun3@localhost Dj]$ touch 3.txt
```



```
[dingjun1@localhost share]$ su dingjun2
密码:
[dingjun2@localhost share]$ touch 3.txt
[dingjun2@localhost share]$ ll
总用量 0
-rw-r--r--. 1 root dingjun 0 10月 20 19:29 1.txt
-rw-rw-r--. 1 dingjun1 dingjun 0 10月 20 19:29 2.txt
-rw-rw-r--. 1 dingjun2 dingjun 0 10月 20 19:30 3.txt
[dingjun2@localhost share]$
```

步骤五: 查看文件所属组

```
[dingjun3@localhost Dj]$ ll
```

```
总用量 0
```



```
-rw-rw-r--. 1 dingjun2 dingjun1 0 12 月  8 00:33 1.txt
```

```
-rw-rw-r--. 1 dingjun2 dingjun1 0 12 月  8 00:34 2.txt
```

```
-rw-rw-r--. 1 dingjun3 dingjun1 0 12 月  8 00:35 3.txt
```

可以看到新建的文件用户组都是 dingjun 组，这样该部门都能读取共享目录的内容。

七、总结反思

grub 是多数 Linux 发行版中使用的 boot loader 系统引导程序，其主要作用是计算机从 boot sequence 中选取指定设备之后，由 grub 找到系统内核 kernel 文件并加载内核文件的作用。grub 作用过程中主要分为三个阶段的作用，stage1: 启动 grub 程序，位于 MBR 的前 446Byte 中；stage1.5: 识别内核所在分区的文件系统类型，位于内核文件所在分区的 boot sector 中；stage2: 根据 grub 的配置文件 grub.conf 中的启动菜单，读取菜单中指定的内核 vmlinuz 文件及读取参数和 initramfs 文件，位于内核 vmlinuz 文件所在的 boot 目录中。

SGID 即 Set GID 的缩写，它出现在文件所属组权限的执行位上面，它对普通二进制文件和目录都有效。当它作用于普通文件时，和 SUID 类似，在执行该文件时，用户将获得该文件所属组的权限。当 SGID 作用于目录时，意义就非常重大了。当用户对某一目录有写和执行权限时，该用户就可以在该目录下建立文件，如果该目录用 SGID 修饰，则该用户在这个目录下建立的文件都是属于这个目录所属的组。

八、参考资料

[1] 文东戈,赵艳芹.Linux 操作系统实用教程（第 2 版）[M].北京：清华大学出版社.2019,9.