

Linux的OTA系统核心实现理论

文件版本号管理

[参考链接](#)

在 Linux 中，版本号通常遵循一定的命名规则，如主版本号.次版本号.修订版本号。每个数字代表不同的含义：

1. **主版本号 (Major version)**：通常表示软件或系统的重大更新或功能改进。当主版本号增加时，表明软件或系统的核心功能发生了较大的变化，可能包括向后不兼容的改动。
2. **次版本号 (Minor version)**：通常表示一些较小的功能更新、改进或 bug 修复。当次版本号增加时，意味着软件或系统添加了新的功能，但仍然保持向后兼容。
3. **修订版本号 (Patch version)**：通常表示一些小的改动、修复或优化，如修复了一些错误或漏洞。修订版本号的增加表明对软件或系统进行了一些细微的改动，以提高稳定性和性能。

此外，有时版本号可能会包含额外的信息，如预发布版本、构建号等，以便更详细地标识软件或系统的状态和构建信息。

总的来说，主版本号、次版本号和修订版本号的不同位数代表了软件或系统的不同级别的变化，帮助用户了解软件或系统的更新内容和重要性

三种文件系统的更新概述

注意:以下场景适用于ubuntu系统和银河麒麟系统 (已测试)

三种文件系统分别是initrd,kernel,rootfs。

- initrd:系统启动的时候引导真正的根文件系统rootfs挂载
- kernel:linux系统的真正内核文件
- rootfs:根文件系统,储存 各种系统配置/应用文件等

所谓更新就是在当前的linux系统下进行对这三种文件系统的更替。首先需要得知的前置知识

前置知识1:Grub引导启动

在linux系统启动的时候流程就是首先启动grub启动配置 grub的作用就是负责整个系统的启动,通过grub配置文件 他首先先启动内核,然后内核启动成功后 启动initrd文件系统 然后initrd文件系统开始运行挂载真正的根文件系统rootfs到当前系统运行后 整个系统启动的结束,所以我们要完成对三种文件系统的更新就是要在grub配置上下手 , 因为在Linux系统里面有一个经典概论: 一切皆文件 ,所有的东

西都是文件 包括进程,配置,文件系统,所以我们可以更改grub配置来达到对三种文件系统的更改,也就是说如果grub配置了什么文件系统在哪个地方 我们就可以修改这个地方的路径到新的地方,也就完成了文件系统的更新

前置知识2:磁盘分区与挂载

众所周知,linux系统的一切都是文件,那文件应该是有一个储存的地方和地址,也就是linux系统有很多文件,这些文件是什么 + 这些文件放在哪 就构成了一个linux系统。那么通过lsblk 指令 ,你可以得知当前这整个linux系统的空间大小和分配,也可以看到哪些文件系统挂载到哪里

```
foryous@foryous-pc:~/桌面$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   70G  0 disk
├─sda1       8:1    0    2G  0 part /boot
├─sda2       8:2    0    1K  0 part
├─sda4       8:4    0   20G  0 part
├─sda5       8:5    0  26.1G  0 part /
├─sda6       8:6    0  17.4G  0 part /data
└─sda7       8:7    0   4.6G  0 part [SWAP]
sr0         11:0    1 1024M  0 rom
```

在这个目录结构下这一整个所有的分支加起来构成了一个Linux系统,也可以说linux系统把所有文件分成了许多个sda分区 ,每个分区在物理地址上是连续的,每个分区的作用都不一样,在右边一栏 显示了他的作用,比如sda1是存放boot目录(内核镜像文件 和grub配置) sda5的右边是个 / 意思是根文件目录,那么我们得知根文件系统在物理上真实在sda5下 而引导配置文件在sda1下,他们的真实路径 对应 在/dev/sda5和/dev/sda1。上面说到每个分区的地址是连续的 不同分区的地址是不同的 形象来说 存放boot配置的地方是在四川省,存放根文件的地方是在湖北省,他们不是在一个地方储存的,但是当你打开这个系统的根目录 ,就会发现在根文件目录下有一个boot目录,其实这个目录就是sda1分区下的boot配置 , 那这是怎么做到的,就是通过"挂载" , 通过一种链接的方式类似于快捷键 在根文件目录下创建了一个boot目录的链接 连接到boot 分区, 也就是说boot文件实际上并不储存在根文件目录下 只是通过挂载的方式。

前置知识3: A/B分区更新策略 (也称为无缝更新)

在使用grub更新的时候我们应该要考虑更新的流程是怎么样的,究竟是在系统运行的期间直接把新的文件系统覆盖到对应的分区吗? 那这样系统正在运行的其他的应用不就出问题了吗。于是有一个策略通过A/B分区的方式来进行更新。还记得上个提到的磁盘分区和挂载吗?他的原理就是在这个系统上除了当前正在运行的根文件系统,再开一个空白的根文件系统,比如当前我的根文件系统运行在sda5(A)下 , 我创建一个新的sda6(B) 给这个分区空间配置大小跟sda5差不多,然后得到新的根文件系统的时候把这个新的文件系统装在这个新的sda6分区里面,这样更新的时候 就不会影响正在运行的根文件系统了,当成功更新了新的文件系统到这个新的盘,我只需要修改一下grub配置,配置下一次系统重新启动的时候要挂载的根文

件系统是sda6而不是sda5了，同时在使用sda6的分区的时候,如果系统出现运行问题导致各种不兼容的问题的话还可以轻松回滚到运行sda5盘。另外如果这个新的sda6下运行如果成功 ,下次要更新就可以把下一个新的文件系统拉到sda5下启动了。

kernel和initrd的更新

众所周知,Linux系统有一个官方的内核网站kernel.org,在这个网站上可以现在历史的linux内核源码,但是他是没有经过编译的,编译在本地上进行还非常占空间和资源,如果要为系统升级新的内核版本 可以直接在<https://kernel.ubuntu.com/mainline/>网站上进行选择内核版本进行下载 这里网站与kernel.org官方内核网站不同的是 这里下载的内核是预编译后的 也就是不需要重新编译 并且更加贴合ubuntu系统 (银河麒麟系统是基于Ubuntu开发的 所以也同样适用)从这个ubuntu官网下载指定版本的内核后 选择那个版本文件夹里面自己的架构 如amd64然后下载里面四个以deb结尾的文件 到ubuntu创建的一个文件夹里面 接下来在这个文件夹下打开终端 输入sudo dpkg --install*.deb进行安装内核等待安装完成,这个指令会对四个文件进行安装,把新的内核和initrd文件安装到boot目录等配置。在grub配置里面选择启动的时候启动这个版本的内核然后sudo update-grub就可以完成重启系统后自动运行新内核 uname -r 查看运行的内核版本

注意:在使用A/B分区策略的时候 因为使用sudo dpkg --install.deb指令安装内核的时候默认会装在当前正在运行的根文件系统和Boot 如果要指定分区 需要先备用的根文件目录挂载到一个文件夹下,然后把备用的boot目录挂载到这个备用的根文件目录下,指令
sudo dpkg --root=/mnt/backup_root --install .deb

然后安装新的内核后 你可以在boot目录下看到新版本的内核文件和initrd文件 这些是镜像文件,另外除了在boot目录下的文件 内核模块文件(这些才是内核真正的文件核心代码)在/lib文件夹下存放,然后如果你要进行系统运行新的内核和新的initrd文件的话,需要修改grub配置

修改grub配置让系统运行新的内核和initrd文件系统

在目录/etc/default/grub 这个配置文件里面你可以看到下图

```
6 GRUB_DEFAULT=0
7 GRUB_TIMEOUT=1
8 GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
9 GRUB_DISTRIBUTOR_RELEASE=`lsb_release -d -s | awk -F" " '{print $2 " " $3}' 2> /dev/null || echo ""`
10 GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"
11 GRUB_CMDLINE_LINUX=""
12 GRUB_CMDLINE_LINUX_SECURITY="security=kysec"
```

第一行GRUB_DEFAULT是指grub菜单的启动默认选择的菜单项 0代表第一个菜单项 1是第二个
第二行GRUB_TIMEOUT是指grub启动菜单显示的超时时间,以秒为单位
第三行GRUB_DISTRIBUTOR是指定义grub菜单中显示发行版的名称
第四行GRUB_DISTRIBUTOR_RELEASE定义了 GRUB 菜单中显示的发行版的版本信息。

第五行GRUB_CMDLINE_LINUX_DEFAULT设置定义了启动 Linux 内核时的默认命令行参数

第六行GRUB_CMDLINE_LINUX设置定义了启动 Linux 内核时的额外命令行参数。

第七行GRUB_CMDLINE_LINUX_SECURITY设置定义了与安全相关的内核参数。

一般我们只关注第一行的信息选择启动哪一项(因为不同项启动的内核文件和initrd文件也不同), 那如何查看当前系统的grub菜单呢 在终端cat /boot/grub/grub.cfg

查找到 menuentry 从第一个menuentry开始,就代表是第一个菜单项也就是0 第二个菜单项就是2, 但是你会发现可能在一个菜单下面还有一个菜单这时候怎么选择呢 可以设置

GRUB_DEFAULT="1>0" 也就是选择第二个菜单下的第一个子菜单

修改完成后输入sudo update-grub 保存grub配置 接着进行重启, 那么下一次系统运行的时候就会按照修改后的默认启动菜单启动某个版本的内核

rootfs根文件系统的更新

如果说上一个内核和Initrd的更新还可以很简单地完成 只需要网上拉取然后进行安装 改配置这么简单,那么rootfs根文件系统的更新稍微就有点学问了,因为你当前系统运行就是运行在根文件系统下的,当前任何的进程信息包括内存信息也都是在这个根文件系统下的,任何的修改和更新都有可能当前系统出现异常崩溃,所以如果采用传统的方式直接把一整个文件系统直接覆盖 是显然很困难的,根据上述的前置知识我们可以用A/B分区策略 秒了, 具体操作是

1:首先如果你要更新 那么你总要有一个新的文件系统 ,那么问题来了既然他是根文件目录 根据经典的 linux一切皆是文件的理念 有些内存也当然运行在根文件目录下 并且也有一些其他的分区比如Boot分区也挂载在rootfs目录下的,那么如何选取更新新的根文件系统应该包括哪些文件呢

- |—— bin //必须要:存放基本用户命令
- |—— boot//根据情况排除:如果要更新 rootfs 而不涉及内核或引导加载程序, 可能不需要更新 /boot 。但在某些情况下(如内核更新), 需要包含在更新中。
- |—— dev //必须排除:包含设备文件, 这些文件由系统动态创建, 不需要备份或更新
- |—— etc//必须要:系统配置文件
- |—— home//根据情况排除:包含用户数据,
- |—— lib//必须要:包含系统库文件
- |—— lib64//必须要:包含系统库文件
- |—— media//必须排除:用于临时挂载点和可移动媒体挂载点, 不需要备份或更新
- |—— mnt//必须排除:用于临时挂载点和可移动媒体挂载点, 不需要备份或更新
- |—— opt//必须要:用于可选的应用程序包。
- |—— proc// 必须排除:虚拟文件系统, 提供进程和系统信息, 不需要备份或更新。
- |—— root//必须要:超级用户 (root) 的主目录
- |—— run//必须排除:包含运行时数据, 例如 PID 文件和临时数据, 不需要备份或更新
- |—— sbin//必须要:包含系统管理命令。
- |—— srv//必须要:存放由该系统提供的服务的数据。
- |—— sys//必须排除:虚拟文件系统, 提供系统和硬件信息, 不需要备份或更新
- |—— tmp//必须排除:存放临时文件, 系统重启后内容会丢失, 不需要备份或更新

- └─ usr//必须要:包含用户程序和库文件。
- └─ var//必须要 但不包括run lock log cache

2: 创建一个新的/dev/sda下的分区

指令:sudo fdisk /dev/sda

接着会进入启动一个交互模式 输入m查看帮助菜单了解所有可用指令

接着输入n创建新分区

选择分区类型 (主分区或者扩展分区) 通常选择p创建主分区

选择新区编号也就是他是sda几 如果你要创建/dev/sda6就输入6 也可以默认回车

然后指定大小,也可以按默认值回车创建一个使用剩余的磁盘空间的新分区

输入w写入更改并退出

这样一个新的分区就创建完成了

然后你需要格式化新分区

sudo mkfs.ext4 /dev/sda6 这里假如你的分区编号是6

然后你就可以把这个分区挂载到一个文件夹下供你安装新的rootfs文件系统了

然后为了使用这个新的rootfs文件系统 你可以修改/boot/grub/grub.cfg文件 这个文件下是grub图形化菜单的配置目录 你可以找到menuentry并添加一个新的menuentry作为你的新的根文件系统的启动配置 如下

```
menuentry 'Foryous' {  
  set root='(hd0,1)'  
  linux /vmlinuz-6.1.0-060100-generic root=/dev/sda5 ro quiet  
  initrd /initrd.img-6.1.0-060100-generic  
}
```

这里我添加了一个叫Foryous的启动项, 通过启动这个项目 他的引导设置是第一个磁盘(hd0)的第一个分区 也就是/boot目录下的启动程序 linux右边的是内核文件(也有版本信息), root右边是根文件系统的真实路径 你可以修改/dev/sda5作为你的新根文件系统的路径 下面initrd 指定了initrd 镜像文件

添加完成后 保存 从此你的系统启动后又多了一个可选项,然后你可以把这个启动项放到一个位置 比如第一个还是第几个 然后再在/etc/default/grub配置文件里面选择启动第几个菜单项

这样就完成了对三个根文件系统的更新

升级包的制作和挂载流程

升级包需要有的文件目录包括一个rootfs和boot目录

rootfs目录下需要有

/etc /bin /lib /lib64 /opt /root /sbin /srv /usr /var

boot目录下需要有

正常的Boot目录下的所有内容

```
sudo rsync -aAXv --one-file-system --exclude={"/proc/", "/dev/", "/sys/", "/tmp/", "/run/", "/mnt/", "/media/*", "/lost+found"} / /data/rootfs
```

rootfs + boot备份分区策略(A/B)

1: 在data分区里面新建两个文件夹一个是rootfs_b一个是boot_b 然后把备用的根分区和备用的boot分区分别挂载到这两个文件夹。

2: 从后端拉取升级包后 (是一个压缩包的形式) 这个压缩包里面包含一个rootfs的文件夹和一个boot文件夹

把这两个目录 分别解压到备用对应的rootfs_b和boot_b文件夹下

3: 为了保持用户数据一致性 需要将新rootfs目录下的home删除
还有- /etc/passwd , /etc/shadow : 用户信息和密码文件。

- /etc/group : 用户组信息。
- /etc/fstab : 文件系统挂载信息。
- /etc/hostname : 系统主机名。
- /etc/network/interfaces 或 /etc/netplan/ : 网络配置。
- /etc/apt/ : 包管理器的配置 (如 /etc/apt/sources.list) 。

还有

- /usr/bin/ , /usr/sbin/ , /usr/local/bin/ : 二进制可执行文件。
- /usr/lib/ , /usr/local/lib/ : 库文件。
- /usr/share/ : 应用程序共享数据, 如图标、文档、样本配置文件。
- /opt/ : 可选的第三方应用程序及其文件。

旧rootfs这些文件覆盖到新rootfs对应位置

4: 修改当前根文件系统的启动配置 /etc/grub.d/40_custom

在这个文件中新增菜单

```
menuentry 'Foryous' {  
  set root='(hd0,1)'  
  linux /vmlinuz-6.1.0-060100-generic root=/dev/sda3 ro quiet  
  initrd /initrd.img-6.1.0-060100-generic  
}
```

Foryous是新的备用菜单名 set root是新的boot位置 如果是/dev/sda3 则是(hd0,3)

linux右边是内核的镜像文件名字 root是新的rootfs分区

initrd是镜像文件名字

然后当前根文件系统的修改/etc/default/grub

中的GRUB_DEFAULT="Foryous"

更新grub配置 `sudo update-grub`

//待定:然后确保新的boot分区里面的gurb配置是指向新的rootfs和新的boot(因为不同设备的分区号可能会不同) 步骤跟上方一样

5:取消挂载新的rootfs文件系统boot文件系统

6:升级步骤完成 等待更新

检测系统升级后异常

通过检查系统的 日志 特定时间段 筛选特定的关键字和错误消息来检测异常

使用 `journalctl` 监控日志

`journalctl` 是一个强大的日志查看工具, 适用于所有使用 `systemd` 的 Linux 系统。

```
# 实时查看系统日志 sudo journalctl -f # 查看特定时间段的日志 sudo journalctl --since "2023-05-21 00:00:00" --until "2023-05-21 23:59:59"
```

差分升级

bsdiff 在stm32里面的差分升级

<https://blog.csdn.net/cplaf2012/article/details/117769624>

https://blog.csdn.net/gitblog_00088/article/details/136703885

第三题6题

国密算法 rsa非对称加密

麒麟系统的debug

新建分区

当我扩展系统的大小后 新建一个磁盘分区 他不会默认从未分区的物理始末位置建立 极为不便

强制自动升级

ssh协议