

Linux GRUB2 配置简介

内容

学习 GRUB 引导加载程序是如何预备你的系统并启动操作系统内核的。

自从上个月为我的文章《[Linux 引导和启动过程简介](#)》做研究开始，我对更深入了解 GRUB2 产生了兴趣。这篇文章提供了配置 GRUB2 的简要介绍。为了简便起见，我大多数情况下会使用 GRUB 指代 GRUB2。

GRUB

GRUB 来自 GRand Unified Bootloader 的缩写。它的功能是在启动时从 BIOS 接管掌控、加载自身、加载 Linux 内核到内存，然后再把执行权交给内核。一旦内核开始掌控，GRUB 就完成了它的任务，也就不再需要了。

GRUB 支持多种 Linux 内核，并允许用户在启动时通过菜单在其中选择。我发现这是一种非常有用的工具，因为我有很多次遇到一个应用程序或者系统服务在特定内核版本下失败的问题。有好几次，引导到一个较旧的内核时就可以避免类似的问题。默认情况下，使用 yum 或 dnf 进行更新时会保存三个内核 - 最新的以及两个比较旧的。在被包管理器删除之前所保留的内核数目可以在 /etc/dnf/dnf.conf 或 /etc/yum.conf 文件中配置。我通常把 installonly_limit 的值修改为 9 以便保留 9 个内核。当我不得不恢复到几个版本的内核时这非常有用。

GRUB 菜单

GRUB 菜单的功能是当默认的内核不是想要的时，允许用户从已经安装的内核中选择一个进行引导。通过上下箭头键允许你选中想要的内核，敲击回车键会使用选中的内核继续引导进程。

GRUB 菜单也提供了超时机制，因此如果用户没有做任何选择，GRUB 就会在没有用户干预的情况下使用默认内核继续引导。敲击键盘上除了回车键之外的任何键会停止终端上显示的倒计时器。立即敲击回车键会使用默认内核或者选中的内核继续引导进程。

GRUB 菜单提供了一个“救援”^{rescue}内核，用于故障排除或者由于某些原因导致的常规内核不能完成启动过程。不幸的是，这个救援内核不会引导到救援模式。文章后面会更详细介绍这方面的东西。

grub.cfg 文件

grub.cfg 文件是 GRUB 配置文件。它由 grub2-mkconfig 程序根据用户的配置使用一组主配置文件以及 grub 默认文件而生成。/boot/grub2/grub.cfg 文件在 Linux 安装时会初次生成，安装新内核时又会重新生成。

grub.cfg 文件包括了类似 Bash 脚本的代码以及一个按照安装顺序排序的已安装内核列表。例如，如果你有 4 个已安装内核，最新的内核索引是 0，前一个内核索引是 1，最旧的内核索引是 3。如果你能访问 grub.cfg 文件，你应该去看看感受一下它看起来是什么样。grub.cfg 太大也就没有包含在这篇文章中。

GRUB 配置文件

grub.cfg 的主要配置文件都在 /etc/grub.d 目录。该目录中的每个文件都包含了最终会整合到 grub.cfg 文件中的 GRUB 代码。这些配置文件的命名模式以排序方式设计，这使得最终的 grub.cfg 文件可以按正确的顺序整合而成。每个文件都有注释表明该部分的开始和结束，这些注释也是最终的 grub.cfg 文件的一部分，从而可以看出每个部分是由哪个文件生成。分隔注释看起来像这样：

```
### BEGIN /etc/grub.d/10_linux ###
```

```
### END /etc/grub.d/10_linux ###
```

不要修改这些文件，除非你是一个 GRUB 专家并明白更改会发生什么。无论如何，修改 grub.cfg 文件时你也总应该保留一个原始文件的备份。40_custom 和 41_custom 这两个特别的文件用于生成用户对 GRUB 配置的修改。你仍然要注意对这些文件的更改的后果，并保存一份原始 grub.cfg 文件的备份。

你也可以把你自己的文件添加到 /etc/grub.d 目录。这样做的一个可能的原因是为非 Linux 操作系统添加菜单行。要注意遵循命名规则，确保配置文件中额外的菜单选项刚好在 10_linux 条目之前或之后。

GRUB 默认文件

老版本 GRUB 的配置非常简单而明了，我只需要修改 /boot/grub/grub.conf 就可以了。对于新版本的 GRUB2，我虽然还可以通过更改 /boot/grub2/grub.cfg 来修改，但和老版本的 GRUB 相比，新版本相对更加复杂。另外，安装一个新内核时 grub.cfg 可能会被重写，因此任何修改都可能消失。当然，GNU.org 的 GRUB 手册确实有过直接创建和修改 /boot/grub2/grub.cfg 的讨论。

一旦你明白了如何做，更改 GRUB2 配置就会变得非常简单。我为之前的文章研究 GRUB2 的时候才明白这个。秘方就在 /etc/default 目录里面，一个自

然而然称为 grub 的文件，它可以通过简单的终端命令操作。/etc/default 目录包括了一些类似 Google Chrome、 useradd、 和 grub 程序的配置文件。

/etc/default/grub 文件非常简单。这个 grub 默认文件已经列出了一些有效的键值对。你可以简单地更改现有键值或者添加其它文件中还没有的键。下面的列表 1 显示了一个没有更改过的 /etc/default/grub 文件。

```
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR="$(sed 's, release .*$,g'
/etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console"
GRUB_CMDLINE_LINUX="rd.lvm.lv=fedora_fedora25vm/root
rd.lvm.lv=fedora_fedora25vm/swap
rd.lvm.lv=fedora_fedora25vm/usr rhgb quiet"
GRUB_DISABLE_RECOVERY="true"
```

列表 1: Fedora 25 一个原始 grub 默认文件。

[GRUB 手册 5.1 章节](#)包括了所有可以添加到该 grub 文件的键的信息。我只需要修改 grub 默认文件已经有的一些键值就够了。让我们看看这些键值以及一些在 grub 默认文件中没有出现的每个键的意义。

- GRUB_TIMEOUT 这个键的值决定了显示 GRUB 选择菜单的时间长度。GRUB 提供了同时保存多个安装内核并在启动时使用 GRUB 菜单在其中选择的功能。这个键的默认值是 5 秒，但我通常修改为 10 秒使得有更多时间查看选项并作出选择。
- GRUB_DISTRIBUTOR 这个键定义了一个从 /etc/system-release 文件中提取发行版本的 sed 表达式。这个信息用于生成出现在 GRUB 菜单中的每个内核发布版的文本名称，例如“Fedora”等。由于不同发行版之间 system-release 文件结构的差异，在你的系统中这个 sed 表达式可能有些不同。
- GRUB_DEFAULT 决定默认引导哪个内核。如果是 saved，这代表最新内核。这里的其它选项如果是数字则代表了 grub.cfg 中列表的索引。使用索引号 3，就会总是加载列表中的第四个内核，即使安装了一个新内核之后也是。因此使用索引数字的话，在安装一个新内核后会加载不同的内核。要确保引导特定内核版本的唯一方法是设置 GRUB_DEFAULT 的值为想要内核的名称，例如 4.8.13-300.fc25.x86_64。
- GRUB_SAVEDEFAULT 通常，grub 默认文件中不会指定这个选项。当选择不同内核进行引导时，正常操作下该内核只会启动一次。默认内核不会改变。当其设置为 true 并和 GRUB_DEFAULT=saved 一起使用时，这个选项会保存一个不同内核作为默认值。当选择不同内核进行引导时会发生这种情况。
- GRUB_DISABLE_SUBMENU 一些人可能会希望为 GRUB 菜单创建一个内核的层级菜单结构。这个键和 grub.cfg 中一些额外内核配置允许创建这样的层级结构。例如，主菜单中可能有 production 和 test 子菜单，每个子菜单中包括了一些合适的内核。设置它为 false 可以启用子菜单。
- GRUB_TERMINAL_OUTPUT 一些环境下可能需要或者必要将输出重定向到一个不同的

显示控制台或者终端。默认情况下是把输出发送到默认终端，通常 `console` 等价于 Intel 系列个人电脑的标准输出。另一个有用的选择是在使用串行终端或者 Integrated Lights Out (ILO) 终端连接的数据中心或者实验室环境中指定 `serial`。

- `GRUB_TERMINAL_INPUT` 和 `GRUB_TERMINAL_OUTPUT` 类似，可能需要或者必要重定向输入为串行终端或者 ILO 设备、而不是标准键盘输入。
- `GRUB_CMDLINE_LINUX` 这个键包括了在启动时会传递给内核的命令行参数。注意这些参数会被添加到 `grub.cfg` 所有已安装内核的内核行。这意味着所有已安装的内核在启动时都会有相同的参数。我通常删除 `rhgb` 和 `quiet` 参数以便我可以看到引导和启动时内核和 `systemd` 输出的所有内核信息消息。
- `GRUB_DISABLE_RECOVERY` 当这个键的值被设置为 `false`，GRUB 菜单中就会为每个已安装的内核创建一个恢复条目。当设置为 `true` 时就不会创建任何恢复条目。但不管这个设置怎样，最后的内核条目总是一个 `rescue` 选项。不过在 `rescue` 选项中我遇到了一个问题，下面我会详细介绍。

还有一些你可能觉得有用但我没有在这里介绍的键。它们的描述可以在 [GRUB 手册 2](#) 的 5.1 章节找到。

生成 `grub.cfg`

完成所需的配置之后，就需要生成 `/boot/grub2/grub.cfg` 文件。这通过下面的命令完成。

```
grub2-mkconfig > /boot/grub2/grub.cfg
```

这个命令按照顺序使用位于 `/etc/grub.d` 的配置文件构建 `grub.cfg` 文件，然后使用 `grub` 默认文件的内容修改输出以便获得最终所需的配置。`grub2-mkconfig` 命令会尝试定位所有已安装的内核并在 `grub.cfg` 文件的 `10_Linux` 部分新建条目。它还创建一个 `rescue` 条目提供一个用于从 Linux 不能启动的严重问题中恢复的方法。

强烈建议你不要手动编辑 `grub.cfg` 文件，因为任何对该文件的直接修改都会在下一次安装新内核或者手动运行 `grub2-mkconfig` 时被重写。

问题

我遇到一个如果没有意识到就可能导致严重后果的 GRUB2 问题。这个救援内核没有启动，反而启动了另外一个内核。我发现那是列表中索引为 1 的内核，也就是列表中的第二个内核。额外的测试发现不管使用原始的还是我生成的 `grub.cfg` 配置文件都会发生这个问题。我在虚拟机和真实硬件上都尝试过而且都发生了这个问题。我只测试了 Fedora 25，因此其它 Fedora 发行版本可能没有这个问题。

注意，从救援内核生成的“`recovery`”内核条目不能引导到维护模式。

我推荐将 grub 默认文件中 GRUB_DISABLE_RECOVERY 的值更改为 “false”，然后生成你自己的 grub.cfg。这会在 GRUB 菜单中为每个已安装的内核生成可用的恢复条目。这些恢复配置能像期望那样工作，从而从那些需要输入密码登录的内核条目中引导到运行级别 1，也就是进入（不需要密码的）单用户维护模式。你也可以按 Ctrl-D 继续正常的引导进入默认运行级别。

总结

GRUB 是引导 Linux 计算机到可用状态过程的一系列事件中，发生在 BIOS 之后的第一步。理解如何配置 GRUB 对于恢复或者处理多种类型的问题非常重要。

这么多年来我多次不得不引导到恢复或者救援模式以便解决多种类型的问题。其中的一些问题确实是类似 /etc/fstab 或其它配置文件中不恰当条目导致的引导问题，也有一些是由于应用程序或者系统软件和最新的内核不兼容的问题。硬件兼容性问题也可能妨碍特定的内核启动。

我希望这些信息能对你开启 GRUB 配置之旅有所帮助。

（题图：Internet Archive [Book Images](#). Opensource.com 修改。CC BY-SA 4.0）

作者简介：

David Both - David Both 是一个居住在 Raleigh，北卡罗来纳州的 Linux 和开源倡导者。他在 IT 界已经有超过 40 年，并在他工作的 IBM 执教 OS/2 20 多年。在 IBM 的时候，他在 1981 年开设了第一个最初 IBM 个人电脑的培训课程。他在红帽教授过 RHCE 课程并在 MCI Worldcom、Cisco、和北卡罗来纳州工作过。他已经在 Linux 和开源软件方面工作将近 20 年。