

GRUB 2 中文指南

本文档为 [Grub2](#) 指南的简体中文版本，繁体由 elmaz 翻译，简体由[鱼漂(ipaddr)]在繁体的基础上修订。

本文档可免费转载及传播，但必须注明作者和出处，且不能用于商业用途。

本文档最新版本可在此找到：<http://www.eit.name/blog/read.php?442>

目前只有 Ubuntu9.10 在全新安装时会选择 Grub2，其它 Linux 发行版本还未采用 Grub2。网上有部分 Grub2 配置文档描述有误。

关键字：[Grub2](#) [中文指南](#) [中文手册](#) [中文参考](#)

GRUB 2 中文指南	1
简介	4
功能增强	4
GRUB vs GRUB 2	5
安装	5
默认安装	6
升级至 GRUB 2	6
配置文件结构	9
boot/grub/grub.cfg	9
etc/default/grub	10
etc/grub.d/（目录）	10
配置 GRUB 2	11
etc/default/grub	11
启动显示过程	14
初始化默认值	14
超时显示	15
隐藏	15
保存选择	15
自定义菜单项	15
建立自定义菜单	16
命令行&救援模式	18
启动时编辑菜单	18
命令行模式	19
使用命令行启动	21
救援模式	23
GRUB 错误信息	24
「Unrecognized Device String」 (Error 11)	24
「File Not Found 」 (Error 15)	25
引导画面和主题	26
GRUB 2 引导画面	26
图片建立和配置	30

主题	30
重新安装 GRUB 2.....	31
更改或移动 GRUB 2.....	31
从启动光盘重新安装	32
卸载 GRUB 2.....	34
回复至传统 GRUB.....	34

简介

GRUB 2 是 Ubuntu 9.10 (Karmic Koala) 的默认引导程序。在计算机启动时，GRUB 2 将显示菜单等待用户选择或者自动将控制权转交给操作系统内核。GRUB 2 继承自 GRUB (GRand Unified Bootloader)。它经过了完全的重写，以提供使用者更大的灵活性和性能改进。GRUB 2 是开放源代码软件。

在这份指南中，GRUB 2 版本为 1.96 或更新版本。传统 GRUB (0.97 版) 将以 GRUB 称之。

功能增强

GRUB 2 与原本的 GRUB 相比之下主要增加的功能包含：

- 支持脚本(Scripting)，包括条件语法和函数
- 动态加载模块
- 救援模式
- 可定制菜单
- 主题
- 支持图形化启动菜单以及增强的引导画面功能
- 可直接由硬盘中的启动光盘映像文件来启动
- 新的配置文件结构
- 支持非 x86 平台（例如 PowerPC）
- 全面支持 UUID（不只在 Ubuntu）
- openSUSE 和 Fedora 尚未采用 GRUB 2

GRUB vs GRUB 2

GRUB 2 的默认菜单对 GRUB 使用者来说看起来很熟悉，但是在相似的表面下有着非常巨大的差异：

- 在没有安装其他操作系统的全新 Ubuntu 9.10 安装下，GRUB2 将会直接启动到登录提示符号或是桌面。将不会显示菜单。
- 启动时，按住 SHIFT 键强制显示菜单(类似 Grub 中的 ESC)
- 没有 /boot/grub/menu.lst。已被 /boot/grub/grub.cfg 取代。
- 在 grub 提示符下没有「/find boot/grub/stage1」。Stage 1.5 已被淘汰。
- 主菜单文件，/boot/grub/grub.cfg，不应再被手动编辑，即使是「root」用户。
- grub.cfg 会在任何有更新、内核被加入/移除或是使用者执行 update-grub 的时候被覆盖。
- 使用者可以编辑一个定制化文件，/etc/grub.d/40_custom，在其中用户可放入自己的选项。此文件将不会被覆盖。
- 主要用来改变显示设置的配置文件是 /etc/default/grub。
- 有好几个文件用来配置菜单 - 上面提到的 /etc/default/grub，还有所有在 /etc/grub.d/ 目录下的文件。
- 磁盘分区编号变更。现在第一个磁盘分区是 1 而不是 0。然而第一个设备仍然是 0（没有改变）。
- 无论何时执行 update-grub 命令时，自动搜索其他操作系统，例如 Windows。
- 在 update-grub 命令执行前，所有配置文件的修改都不会生效。

安装

GRUB 2 会在全新安装的 Ubuntu 9.10 (Karmic Koala) 中默认安装。升级至 9.10，例如从 Ubuntu 9.04 Jaunty Jackalope 升级，将保留原本的 GRUB，除非使用者选择升级至 GRUB 2。若使用者需要的话，之前版本的 Ubuntu 可以升级至 GRUB 2。

您可以通过 Synaptic 或是从命令行来检查目前安装的版本。打开终端（应用程序 > 附件 > 终端）并输入以下命令：

```
grub-install -v
```

GRUB 2 应该会显示版本编号 1.96 或是更新。传统 GRUB 版本为 0.97。

默认安装

GRUB 2 将会是 Ubuntu 9.10 Karmic Koala 默认安装的启动加载程序。

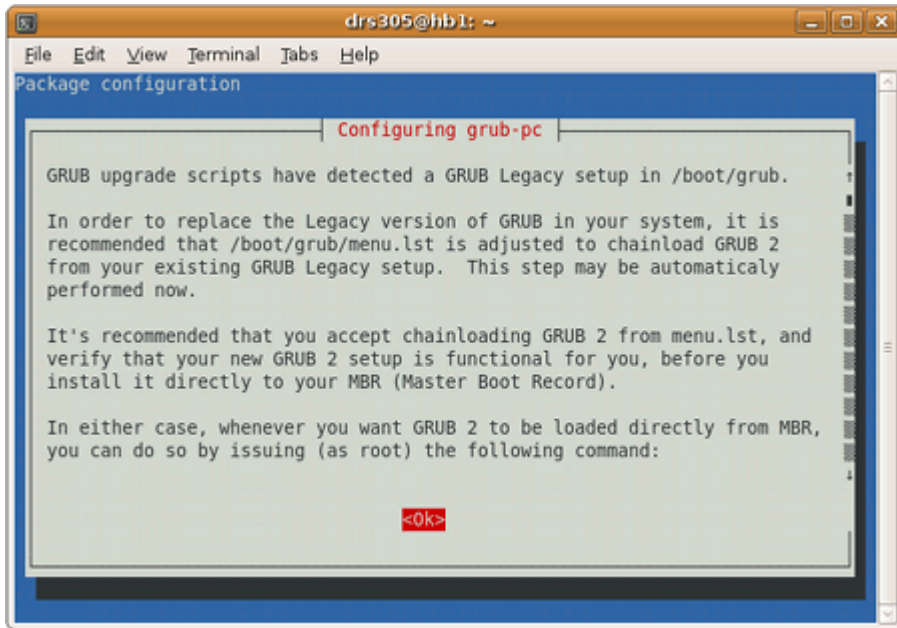
- GRUB 2 组件为 grub-pc。grub-common 组件也将会一并安装。
- grub2 是过渡期虚拟组件。他将会被安全移除，并不会在 Synaptic 中显示为已安装。

升级至 GRUB 2

在 Ubuntu 9.10 或更早期的版本将 GRUB（传统）升级至 GRUB 2 是相当容易的。在使用者确认完全的转换前，升级过程将提供一个程序来确认 GRUB 2 能在您的机器上工作。然而，默认情况下并不会自动升级至 GRUB 2。升级至 Ubuntu 9.10 的使用者仍然选择可以继续沿用 GRUB。

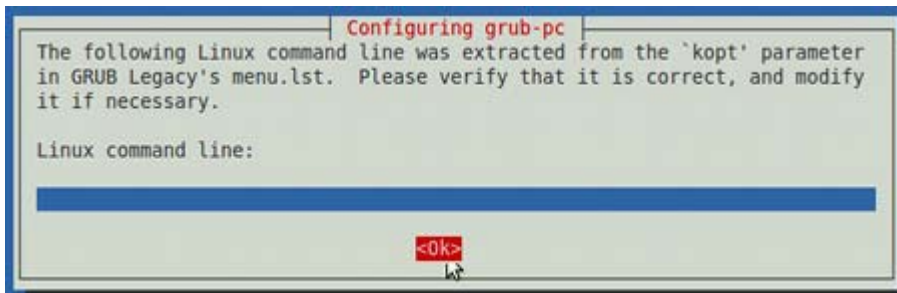
- 在安装 GRUB 2 之前，更新组件库索引以确保系统有最新的软件组件列表。一般在终端中通过 `apt-get update` 命令或是通过图形用户界面组件管理器，如 Synaptic，来更新列表。旧的 GRUB 组件可能未包含早期版本的一些错误修正更新。
- 欲升级至 GRUB 2，请使用 `sudo apt-get install grub-pc` 或 Synaptic 安装「grub-pc」组件。一旦开始升级，将有一个简单的脚本引导用户操作：

1. 在第一个提示中，阅读这些信息，选中「OK」并按下 ENTER。



2. 在选择「OK」后，下一个选项允许用户在正常的 GRUB 菜单中通过新增一个菜单项来测试 GRUB 2。选择「Yes」将在 GRUB 菜单中增加一个 Chainload 项。当下次使用 GRUB 启动时，使用者可以选择正常的 GRUB 菜单项或是将控制权通过 Chainload 菜单项转到 GRUB 2。

3. 下一步将会把来自 GRUB 的 menu.lst 中的内容导入到「Linux command line」（如下图），或者导入到「Linux default command line」。这些项目允许用户输入任意的特殊命令，例如 quiet, splash, noapic 或是其他同样性质的命令。GRUB 2 将会自动导入这些配置。通常用户不需要修改并选择「OK」。若有需要，选项也可以在以后再加入。更多细节请参考配置 GRUB 2 小节。



4. 当系统已安装完成 GRUB 2，使用者可重新启动并看到已更新的 GRUB 传统菜单欢迎画面。如图所示，使用者将会看到一个「Chainload into GRUB 2」菜单项用来测试 GRUB 2 安装。选择 Chainload 选项将会把控制权转到 GRUB 2 并显示 GRUB 2 菜单。

```
Ubuntu 8.10, kernel 2.6.27-9-generic
Ubuntu 8.10, kernel 2.6.27-9-generic (recovery mode)
Ubuntu 8.10, kernel 2.6.27-2-generic
Ubuntu 8.10, kernel 2.6.27-2-generic (recovery mode)
Chainload into GRUB 2
Ubuntu 8.10, memtest86+

Use the ↑ and ↓ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

5. Jaunty 用户可能会碰到一个「Unrecognized device string」Error 11 的错误。如果发生此情况，按下「e」直到您看到一行开头为「grub edit>root」的文字。把「root」改为「uuid」，按下 ENTER，然后按「b」来启动。更多关于 Error 11 的细节可以在命令行和救援模式小节中找到。

6. 测试新选项并注意到 Grub 1.9x 版本编号（新增高亮度部分）。

```
GNU GRUB version 1.96

Ubuntu, linux 2.6.27-9-generic
Ubuntu, linux 2.6.27-9-generic (single-user mode)
Ubuntu, linux 2.6.27-2-generic
Ubuntu, linux 2.6.27-2-generic (single-user mode)
Memory test (memtest86+)
```

7. 若 GRUB 2 菜单功能一切正常，用户可在任何时间执行以下命令以完全升级至 GRUB 2:

```
sudo upgrade-from-grub-legacy
```



在成功升级到 GRUB 2 之后，会有一个提示告知用户有些 GRUB 传统配置文件仍然存在系统中，并可被删除。此说明包含以下的命令，必须以「root」身份执行：`rm -f /boot/grub/menu.lst`

*以「root」执行 `rm` 命令时请特别慎重。不正确输入此命令可能导致系统文件永久性的被删除。

配置文件结构

GRUB 2 包含一个完全修改过的配置目录及文件。传统的 GRUB 下的 `menu.lst` 将不再继续使用。

用户可以执行以下命令找到 GRUB 2 安装位置：

```
sudo grub-probe -t device /boot/grub
```

可找到安装的设备

```
sudo grub-probe -t fs_uuid /boot/grub
```

可找到 UUID。

GRUB 2 使用的主要配置目录和文件包括：

`/boot/grub/grub.cfg`

这是类似 GRUB 下 `/boot/grub/menu.lst` 的文件。此文件中包含了 GRUB 2 菜单信息，但不像 GRUB 的 `menu.lst`，`grub.cfg` 文件不应该手动编辑。

- 每个段落明确地由「(### BEGIN)」界定并参考在 `/etc/grub.d` 目录中的文件，由该目录中的文件来产生信息。

- `grub.cfg` 由 `root` 身份执行 `update-grub` 来更新。`update-grub` 命令其实是 `sudo grub-mkconfig -o /boot/grub/grub.cfg` 的简化。为了标准化，请使用 `update-grub` 命令。(需要加上 `sudo` 以 `root` 权限执行)
- 此文件将会在 Linux 内核安装/更新时或是其他会影响到它的输出时自动更新。
- 默认情况下，无论何时，当 `update-grub` 命令执行后，此文件会被配置为「只读」。这是刻意所为，以提醒用户不要手动编辑此文件。

用户也会在 `/boot/grub` 目录下看到许多 `*.mod` 文件，这些文件表明 GRUB 2 的模块化本质；GRUB 2 启动时按需要加载不同的模块。

`/etc/default/grub`

- 此配置文件包含以前在 GRUB 传统的 `menu.lst` 前半部配置内容；以及每行内核配置后面的追加配置。在此文件中的配置可以由具有 `root` 权限的使用者编辑，并会在更新时合并至 `grub.cfg` 中。具体的配置选项可以在此小节稍后的配置 GRUB 2 部分中找到详细信息。

`/etc/grub.d/`（目录）

- 在此目录中的文件将会在执行 `update-grub` 命令的时候读取，并将其中的命令合并至 `/boot/grub/grub.cfg` 中。
- 在 `grub.cfg` 菜单中的菜单项顺序是依照在此目录中文件的顺序来决定。具有较小数字前缀文件名的文件将会先执行。`10_linux` 将会在 `20_memtest` 之前执行，而 `40_custom` 将在最后执行。若有文件名全为英文字母名称，那么将会在数字名称的文件名后执行。
- 自定义选项可以加入在 `40_custom` 文件中，或新建的文件中。自定义文件开头为 `06_` 将会出现在菜单的最上方，因为他的字母排序在 `10_` 和 `40_` 之前。
- 在 `update-grub` 执行过程中只有可执行文件会输出至 `grub.cfg`。
- 在此目录中的默认文件为：
 - ◆ `00_header` 配置初始显示项目，例如图形模式、默认选项、时间限制等等。这些配置信息一般由 `/etc/default/grub` 中导入。使用者一般不需要对此文件做任何更动。
 - ◆ `05_debian_theme` 此文件中的配置用来配置引导画面、文字颜色、高亮度选项及主题。再没有引导画面的情况下，此文件会配置一个单色的主题来显示初始菜单画面。如何配置字体显示及引导画面将会在引导画面和主题小节中讨论。
 - ◆ `10_hurd` 定位 Hurd 内核位置。目前未使用。
 - ◆ `10_linux` 定位当前操作系统使用中的 `root` 设备内核位置。此信息将会用来建立这些内核显示在菜单中的名称。
 - ◆ `20_memtest86+` 搜索 `/boot/memtest86+.bin` 并将之包含在 GRUB 2 启动菜单选项中。目前没有选项能将此条目从启动菜单中移除。但可以通过移除 `memtest86+` 的执行权限来禁用：`sudo chmod -x /etc/grub.d/20_memtest86+`。

- ◆ `30_os-prober` 此文件和 `os-prober`，用来搜索 Linux 和其他操作系统。结果将会依据此文件中的内容而放至 `grub.cfg` 中。此文件被划分成四个段落，代表由该脚本所处理的四种操作系统类型：Windows, Linux, OSX 和 Hurd。此脚本中的变量用来指定在 `/boot/grub/grub.cfg` 和 GRUB 2 菜单中的名称显示格式。熟悉基本脚本的用户可以修改这些变量来改变菜单选项的显示格式。此外，用户可以修改 `/etc/default/grub` 的一个选项来关闭这个脚本。（参考下方）
- ◆ `40_custom` 用来加入自定义菜单项的样板，将会在执行 `update-grub` 命令时嵌入至 `grub.cfg` 中。此文件中的内容，在「`exec tail -n +3 $0`」这一行和默认的注释后，会毫无改变地直接导入至 `/etc/default/grub.cfg` 中。
- 在这些目录中的文件必须要设为可执行才能被 `update-grub` 使用。没有设为可执行的文件将会被忽略。欲将文件设为可执行，执行

```
sudo chmod +x /etc/grub.d/folder/filename
```

配置 GRUB 2

配置 GRUB 2 通常需要修改 `/etc/default/grub` 和 `/etc/grub.d` 目录中的自定义文件。`/boot/grub/grub.cfg` 文件不应该直接编辑；此文件由 `update-grub` 来更新。



某些最常见的修改，例如默认的操作系统/内核和菜单超时，可以使用一个图形用户界面应用程序「`StartUp-Manager`」来修改。如何安装及使用此应用程序的相关信息请参考 Ubuntu 社区文档：[StartUpManager](#)。

`/etc/default/grub`

- 可修改的主要默认配置文件。在安装时，以下内容可由使用者修改：
- `GRUB_DEFAULT`
 - ◆ 配置默认菜单选项。选项可以是数字或是「`saved`」。

- ◆ GRUB_DEFAULT=0 以菜单位置来配置默认菜单选项。如同 GRUB，在 grub.cfg 中的第一个「菜单选项」为 0，第二个为 1，以此类推。
- ◆ GRUB_DEFAULT=saved 配置默认选项为上一次选择的项目。若启动过程中显示菜单，则上一次选择的选项将会以高亮度表示。若没有进行任何动作，此选项将会在超时时间到达或是菜单设为隐藏时用来启动。
 - 当此参数设置为 saved 时，可以使用 grub-set-default 命令快速地改变默认启动的操作系统/内核。
 - ✧ 命令格式为 sudo grub-set-default X，此处的 X 代表菜单选项的位置（第一个由 0 开始）或是确切的选项字符串。例如：sudo grub-set-default 3 将使用菜单选项位置，或是 sudo grub-set-default "Ubuntu, Linux 2.6.31-14-generic" 表示菜单选项字符串。
 - ✧ 欲获取现有的菜单选项编号（由 0 开始）或是菜单选项名称，执行 grep menuentry /boot/grub/grub.cfg
 - GRUB_DEFAULT="xxxx" 一个确切的菜单选项，包含引号也可使用。在此例中，菜单中的位置并不重要。例如：GRUB_DEFAULT="Ubuntu, Linux 2.6.31-14-generic"
 - 如何启用「saved」选项及自定义菜单的范例，请参考「自定义用户选项」小节。

◆ GRUB_HIDDEN_TIMEOUT=0

- ✧ 此配置将影响菜单显示。若设置此选项，将在此时间内隐藏菜单而显示引导画面。



- ✧ 菜单将会被隐藏，除非在此行开头加上一个 # 符号。（# GRUB_HIDDEN_TIMEOUT=0）。
- ✧ GRUB 2 第一次执行时将会寻找其他操作系统。若没有其他操作系统被检测到，菜单将会配置为隐藏。若辨认出其他操作系统，菜单将会显示。
- ✧ 若是大于 0 的整数，系统将会依此配置的秒数暂停，但不会显示菜单。
- ✧ 0 则菜单不会显示，也不会有延迟。
 - 使用者可以在启动时按住 SHIFT 键不放以强制显示菜单。
 - 启动过程中，系统将会检查 SHIFT 键状态。若无法辨识按键状态，会有一个短时间的延迟让使用者可通过按下 ESC 键来显示菜单。

◆ GRUB_HIDDEN_TIMEOUT_QUIET=true

- ✧ true 不显示倒计时。屏幕将会是空白的。
- ✧ false 在 GRUB_HIDDEN_TIMEOUT 中配置的时间，空白屏幕上会有一个倒数计时器。

◆ GRUB_TIMEOUT=10

- ✧ 此命令将顺从 GRUB_HIDDEN_TIMEOUT 配置，除非 GRUB_HIDDEN_TIMEOUT 被注释掉（#）。若 GRUB_HIDDEN_TIMEOUT 启用，则当菜单显示时，GRUB_TIMEOUT 将会只执行一次。

✧ 配置此值为 -1 将会导致菜单一直显示，直到用户选择。

✧ GRUB 2 菜单默认为隐藏，除非其他操作系统被系统检测到。若没有其他操作系统，此行将会被注释掉，除非使用者修改它。为了在每次启动时显示菜单，去掉此行的注释并使用 1 或更大的值。

◆ GRUB_DISTRIBUTOR=lsb_release -i -s 2> /dev/null || echo Debian
取得菜单选项中的描述名称。（Ubuntu, Xubuntu, Debian 等等。）

◆ GRUB_CMDLINE_LINUX

✧ 若存在，无论在一般或是救援模式，此行将追加到所有的 'linux' 命令行后面（传统 GRUB 的「kernel」选项）。类似于 menu.lst 中的「altoptions」选项。

◆ GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"

✧ 此行将追加在 'linux' 命令行后面（传统 GRUB 的「kernel」选项）。此选项只会追加在一般模式的最后方。类似于 menu.lst 中的「defoptions」选项。如果想显示黑色屏幕以及启动进程文字，请移除「quiet splash」。若想看到 grub 引导画面及简短的文字输出，使用「splash」。若有需要的话，也可以在此行输入选项「acpi=off」。

◆ #GRUB_TERMINAL=console

如果想启用 console，请删除此行前的注释启用此选项。（只限 grub-pc）

◆ #GRUB_DISABLE_LINUX_UUID=true

如果您不想让 GRUB 传递「root=UUID=xxx」参数给 Linux，请删除此行前的注释启用此选项。

◆ #GRUB_GFXMODE=640x480

■ 您可移除注释启用此选项。此选项会配置图形化菜单的分辨率（菜单文字大小）。它将提供用户的显示器所支持的分辨率（例如 640x480, 800x600, 1280x1024 等等）。此配置只适用于启动菜单显示，并不适用于完成启动后的操作系统分辨率。

◆ 小技巧：在 GRUB 2 中配置与操作系统相同的分辨率将会稍微减少启动所需时间。

- 使用者也可以在后面加上颜色位数。比如 1280x1024x24 或 640x480x32。
- 用户可以加入多个分辨率。若 GRUB 2 无法使用第一个选项，它会试着使用下一个配置。配置由逗号分开。例如：1280x1024x16,800x600x24,640x480。
- 若有使用引导画面，请确定分辨率配置和引导画面大小兼容。
- 若在执行 update-grub 时有「无法找到 (not found)」信息，试着加上或修改颜色深度配置。
- 能够在 GRUB 2 中使用的分辨率可通过在 GRUB 2 命令行模式下键入 vbeinfo 得知。此命令行模式可在 GRUB 2 菜单画面显示时按下「c」键进入。
- 若此行被注释掉（#）或是分辨率无法使用，GRUB 2 将使用由 /etc/grub.d/00_header 中的默认配置。
- 欲参考关于使用引导画面时改变分辨率的指南请参考引导画面和主题小节。

◆ GRUB_DISABLE_LINUX_RECOVERY=true

加上或移除此行注释来禁止「救援」模式内核选项显示在菜单中。若您只想要在其中一个内核有「救援」选项，请在 /etc/grub/40_custom 中制作一个特殊选项。

◆ 额外可包含的选项：

✧ GRUB_DISABLE_OS_PROBER=true

- 启用/禁止 os-prober 对其他磁盘分区操作系统的检查，包括 Windows, Linux, OSX 和 Hurd。

启动显示过程

GRUB 2 会在任何操作系统前加载。它的模块化组件根据需求被加载。菜单显示规则由 /etc/default/grub 来生成。配置选项和格式化指引请参考「配置 GRUB 2」小节。

初始化默认值

- 若没有检测到其它操作系统，GRUB 2 将会直接启动默认的操作系统。不会显示菜单。若检测到其他操作系统，GRUB 2 将会显示菜单。

超时显示

- 菜单将会持续在预定的秒数中显示。默认选项会以高亮度表示，使用者可以在超时时间到期前选择其他选项。默认的超时时间为 3 秒。计时器会在任何按键被按下前或是按下 **ENTER** 选择高亮度标明的选项前持续计时。若使用者未进行输入则 **GRUB 2** 将会在超时时间到期时以默认选项启动。
- 当按下任何按键时，超时将会失效，且使用者必须手动进行选择。
- 默认选项由 `/etc/default/grub` 中的 `DEFAULT=` 来配置；第一个「菜单选项」的值为「0」。

隐藏

- 不显示菜单选项。若有配置引导画面，将会显示引导画面。
- 使用者可以通过按住 **SHIFT** 键，以此来中断启动程序并显示菜单。

GRUB 2 会在启动过程中搜索 **SHIFT** 键按下的信号。若此键按下或是 **GRUB 2** 无法确认此键的状态，菜单将会显示。
- 屏幕保持空白但允许显示菜单的时间是由 `/etc/default/grub` 中的配置来决定。
- **GRUB 2** 可以显示一个倒数计时器以提供剩余时间的视觉回馈，直到默认的选项被选定。此超时配置在 `/etc/default/grub` 中。
- 在超时时间结束时，在 `/etc/default/grub` 中所配置的默认选项将会被选定。

保存选择

- 若默认选项配置为「**saved**」，使用者未选择菜单的话，则将启动上一次成功启动的内核/系统。

自定义菜单项

GRUB 2 允许用户建立自定义菜单选项，在执行 `sudo update-grub` 时此选项将会自动加入主菜单中。在 `/etc/grub.d/` 中有一个空的 `40_custom` 文件可供使用，或是以其当作建立其他自定义菜单的模板。这里有一些自定义菜单需要注意的地方：

- 文件的名称将会决定选项在菜单中的排序。数字文件名会在前执行，跟着是文本文件名。
 - ◆ 10_linux 选项会出现在 30_os-prober 选项之前，然后跟着是 40_custom 选项，最后是 my.custom.entries。
 - ◆ 用户欲将自定义选项排在主菜单第一位，可以将文件命名为 06_xxx 如此一来它将会是菜单中的第一个选项。
- 文件必须有可执行权限：`sudo chmod +x /etc/grub.d/filename`。
- 在 `update-grub` 命令未执行前，`grub.cfg` 的内容不会被更新。
- 和 `grub.cfg` 不同，自定义文件可以在任何时间编辑，不是只读权限，而且可以修改文件属主。
- 自定义文件是静态的。当新的内核加入或移除时其内容将不会被修改（除非内容受使用者加入的脚本影响）。



若「自定义文件」将某个菜单项放在 `grub.cfg` 的最前面，使用者需要特别注意 `/etc/default/grub` 的 `DEFAULT=` 配置，确保默认值是指向你所要的菜单项。

建立自定义菜单

使用者可选择默认的 `/etc/grub.d/40_custom` 文件来修改或是建立一个新文件。建立一个新自定义菜单最简单的方法为从 `/boot/grub/grub.cfg` 复制一份使用中的选项。复制之后，在 `40_custom` 中的内容可剪裁为用户想要的样式。

- ◆ 根据默认的范例自定义文件（`/etc/grub.d/40_custom`）在 `/etc/grub.d` 中任何自定义文件的头两行应该是：

```
#!/bin/sh
exec tail -n +3 $0
```

- ◆ 使用者可从 `/boot/grub/grub.cfg` 或是从传统 GRUB 文件中复制已存在的菜单选项。若使用者想从传统 GRUB 的 `menu.lst` 中复制，需要注意：
 - 若使用者从 GRUB 升级至 GRUB 2 或许可以在 `/boot/grub` 目录下找到一份备份文件。
 - 以下来自 GRUB 菜单中的项目必须修改才能作用：
 - `title` 更改为 `menuentry`。此行必须由 `{` 作为结束
 - `root` 更改为 `set root=`
 - `kernel` 必须改为 `linux`

- 任何分区编号（sda4, sda6）必须修改，因为 GRUB 和 GRUB 2 的磁盘分区计数方式不同。GRUB 2 中的第一个磁盘分区为 1，不是 0。设备仍然由 0 开始计数。
- 仔细检查任何作用中的 GRUB 2 menulist 条目格式以确保使用正确的格式。

◆ 一般 menuentry 建构规则：

- 第一行一定由 menuentry 开头，并以 { 结束
- 大括号之间的区域就是会出现在 GRUB 2 的内容。依照需求编辑。
- menuentry 最后一行一定是 }
- 每行最后面不要留下空白
- set root= 行应该指到 GRUB 2 的 /boot 地址（sdXY）
- 在 linux 行中的 root 应指向系统磁盘分区。
- 若 GRUB 2 无法找到指定的内核，试试看以设备名称来取代 UUID（例如：/dev/sda6）。

◆ 一份从 grub.cfg 复制来由用户修改过的选项范例也许看起来像这样：

```
menuentry "My Default Karmic" {  
set root=(hd0,1)  
search --no-floppy --fs-uuid --set cb201140-52f8-4449-9a95-749b27b58ce8  
linux /boot/vmlinuz-2.6.31-11-generic root=UUID=cb201140-52f8-4449-9a95-749b27b58ce8 ro quiet splash  
initrd /boot/initrd.img-2.6.31-11-generic  
}
```

一份由网站上的命令将系统还原光盘（SystemRescueCD）安装为硬盘版（在 sda8 上）的选项范例：

```
menuentry "System Rescue CD" {  
set root=(hd0,8)  
linux /sysrcd/rescuecd subdir=sysrcd setkmap=us  
initrd /sysrcd/initram.igz  
}
```

一份连锁加载另一个 GRUB 启动加载程序的选项范例：

```
menuentry "Grub 1 Bootloader" {  
  set root=(hd0.8)  
  chainloader +1  
}
```

命令行&救援模式

使用者可以有数种方法来尝试从失败的启动中恢复。如果菜单仍然可以使用，菜单配置可以由使用者来检查和修改。此外，若 **GRUB 2** 检测到启动时因为 `grub.cfg` 的错漏而无法将控制权转移到所选择内核的错误，它将会回复到救援模式。两种方式都允许使用者由受限的命令行来恢复为可用的系统。如果只有「Grub」字样出现在屏幕顶端而没有提示符号（命令行）或菜单，参考从启动光盘重新安装小节。

启动时编辑菜单

- 若启动过程中菜单无法正常显示，按住 **SHIFT** 键让计算机试着显示 **GRUB 2** 菜单。
- 当菜单显示后，按下任何键（**ENTER** 除外）来停止超时计时器并以上/下键选择想要的项目。
- 按下「e」键进入所选选项的配置。

GNU GRUB version 1.97~beta3

```
recordfail=1
save_env recordfail
set quiet=1
insmod ext2
set root=(hd0,1)
search --no-floppy --fs-uuid --set 904bf39-9234
linux /boot/vmlinuz-2.6.31-9 root=UUID=904bf39-9234 ro quiet splash
initrd /boot/initrd.img-2.6.31-9-generic
```

Minimum Emacs-like screen editing is supported. TAB lists completions. Press **Ctrl-x** to boot, Ctrl-c for a command-line or ESC to return menu.

- 使用键盘来定光标位置。在此例中，光标已经被移动，因此用户可以修改或删除数字 9。。对任一行或每一行做单一或多种修改。不要使用 ENTER 来做行列间的移动。
- 可使用 Tab 键自动完成功能，这在输入内核和 initrd 项目的时候特别好用。
- 当完成时，决定下一个步骤：
 - ◆ CTRL-x - 以修改过后的配置启动（以高亮度强调）
 - ◆ CTRL-c - 进入命令行来进行诊断、加载模块、修改配置等等。
 - ◆ ESC - 放弃所有修改并回到主菜单
 - ◆ 这些选择也会列在屏幕底部作为提示。

命令行模式

从 GRUB 2 主菜单画面使用者可选择按下「c」进入命令行模式。



这里有一些好用的诀窍和特性来使用 GRUB 2 命令行：

- 关闭引导画面也许会让终端更容易检查。按下「c」进入命令行然后输入： `set color_normal=white/blue` 或是任何你想使用的颜色组合。第二个选项使用「black」会保留菜单的透明度，若使用者想使用实色当作背景色的话应该避免此选择。
- 为了预防画面卷动到屏幕最上方以外，首先执行 `set pager=1`，此配置类似 Linux 下的「| more」命令。将允许一个全画面的数据，并在按下「ENTER」键时才会继续往下卷动。
- 输入 `help` 可列出所有命令的列表。输入 `help x` 来检查所有以「x」字母开头的命令。Tab 键自动完成以及使用向上键来重复上一个命令与一般终端动作相同。
- TAB 自动完成功能非常的好用。在任何时间，若组件是独一的，按下 TAB 键可以自动完成一个项目，否则会显示所有可用的选项。输入一些额外的字符后再按下 TAB 可自动完成剩余的部分。

一些好用的命令包括：

命令	结果
background_image	加载一个现存的背景图片。使用 <code>tab</code> 自动完成来协助输入文件名！ <code>background_image=/usr/share/images/grub/</code> 然后用 <code>tab</code> 来显示所有图片的列表。当初测试引导画面时 很有用的命令。
cat	用来显示目前文件内容。 <code>cat /boot/grub/grub.cfg</code> , <code>cat /etc/fstab</code>
help	列出所有可用命令。输入 <code>help x</code> 来取得一份以字母 <code>x</code> 开头的命令列表。 <code>help xxx</code> 取得关于 <code>xxx</code> 命令的信息。此命令应该接在「 <code>set pager=1</code> 」命令之后，会让屏幕显示类 似终端「 more」命令的作用，以预防卷动超过屏幕顶端。
ls	单独使用，列出所有设备 (<code>hd0</code>) (<code>hd0,1</code>) 等等。使用 <code>ls</code> 来检查可用的磁盘分区。使用 <code>ls /boot</code> , <code>ls /boot/grub</code> 来检查 <code>/boot</code> 和 <code>/grub</code> 目录内容。
initrd	载入 <i>initrd</i> 。
linux	载入 <i>linux</i> 内核。
loopback	把文件挂载成设备。 <code>loopback loop (hd0,2)/iso/my.iso</code>
lsmod	显示已加载模块
reboot	重新启动
rescue	进入救援模式
set	显示适当的环境变量。与其他结合，配置变量如 <code>set gfxmode=640x480</code> , <code>set color_normal=cyan/black</code> 在命令行模式中使用后面的命令来测试各种菜单字体颜色。
vbeinfo	显示可用的屏幕分辨率。

使用命令行启动

若使用者有启动上的麻烦但菜单可用，最简单的启动方法就是编辑现有的菜单。参考启动时编辑菜单。若 GRUB 2 寻找可用 `grub.cfg` 文件失败，它会恢复到 `grub-rescue` 模式。命令行提示符号显示 `grub-rescue>` 且将不会显示菜单。用户可由此命令行来尝试手动输入命令来启动至可用的系统。

若命令行提示符号没有启动，按下「c」来进入命令行模式。您将会看到 GRUB 2 提示符号：`grub>` 或 `grub rescue>`



若您任何时间想要取消并重来，**ESC** 将会移除修改部分并让使用者返回前一个菜单。使用以下两个命令来决定您想要启动的系统所在设备（磁盘驱动器）和磁盘分区。

set	当输入 set 且没有附加项目时，此命令会显示目前的 GRUB 2 配置。
ls	执行 ls 来查看 GRUB 2 认出的设备。例如：(hd0) = sda; (hd0,1)=sda1; (hd0,3)=sdb3

以最近的内核快速启动

输入以下所有命令：（每个命令后按 ENTER）

1. `set root=(hdX,Y)` 输入来自 ls 命令正确的 X,Y 值并按 ENTER。记住 GRUB 2 将第一个设备算为 0，第一个磁盘分区为 1。例如：若 Ubuntu 系统安装在 sda5，输入：`set root=(hd0,5)`
2. `linux /vmlinuz root=/dev/sdXY ro` （范例：`linux /vmlinuz root=/dev/sda3`）
3. `initrd /initrd.img` 选择最新的 initrd 影像。
4. `boot` 以选定的磁盘分区最新的内核来启动。

手动启动到指定的内核

若在启动程序期间 GRUB 2 菜单无法编辑，使用者可在 GRUB 2 命令行中输入所有需要用来启动到可用内核的信息。此小节将提供一个逐步的指引来说明如何输入此信息。此行完成后会看起来类似以下图型：

只有当完成每一小节（「A」，「B」，「C」和「D」）后才按下 ENTER。

A. 配置根磁盘分区（步骤 1）

1. **set root=(hdX,Y)**

输入来自 `ls` 命令的正确 `X,Y` 值并按下 `ENTER`。记住 GRUB 2 将第一个设备算为 0，第一个磁盘分区为 1。例如：若 Ubuntu 系统安装在 `sda5`，输入：`set root=(hd0,5)`

B. 输入「linux」行信息（步骤 2-5）

linux kernel root ro

2. **linux**

输入并继续。

3. **/boot/vml**

输入，然后使用 `TAB` 来显示可用的内核。输入正确的内核后继续。

4. **root=/dev/sdXY**

输入（使用正确的设备如「`sda1`」、「`sdb5`」等等）然后继续。

5. **ro**

加入任何选项，例如 **ro** 等等。

当此行的所有信息都完成后，它看起来将会像是以下的范例。按下 `ENTER`。检查并确认。若成功输入，GRUB 2 将在 `[]` 符号中显示储存的信息。（以下的第一个高亮度段落）。

linux /boot/vmlinuz-2.6.31-9-generic root=/dev/sda1 ro

```
sh:grub> ls
(hd0) (hd0,5) (hd0,1) (fd0)
sh:grub> linux /boot/vmlinuz-2.6.31-9-generic root=/dev/sda1 ro
[Linux-bzImage, setup=0x3c00, size=0x3b7f00]
sh:grub> initrd /boot/initrd.img-2.6.31-9-generic
[Initrd, addr=0x2f932000, size=0x7018d5]
sh:grub>
```

C. 输入「initrd」行信息（步骤 6-8）

initrd /boot/initrd.img-2.6.31.9-generic

6. **linux**

输入并继续。

7. **initrd**

输入，然后使用 `TAB` 来显示可用的内核。输入正确的内核后继续。

8. **/boot/init**

输入并使用 `TAB` 来显示可用的内核，输入正确的 `initrd` 影像。参考上图高亮度类似确认行的那一行。

D. 启动（步骤 9）

9.boot 以输入的内核/操作系统启动。CTRL-X 也可行。

救援模式


救援模式是 GRUB 2 的主要强化部分。若 GRUB 2 寻找可用的 grub.cfg 失败并无法转移控制权至内核，他将会进入 grub-rescue> 提示符号。在此提示符号下用户可检查问题、进行修改，以及重新尝试启动。

救援模式所提供的命令比起一般 GRUB 提示符号少，但也提供额外的命令：

命令	结果
dump	清除内存
exit	退出 GRUB 2
normal	若可能的话回到标准的「grub>」模式。

以下命令可以在 grub 救援模式中使用：

boot	cat	chainloader	dump	exit	freebsd
freebsd_loadenv	freebsd_module	help	initrd	insmod	linux
lsmod	multiboot	normal	rmmod	set	unset

 若需要额外的命令，用户可以试着用 insmod normal 来加载一般的 GRUB 2 模块。若成功， help 和其他额外的命令就可以使用。

在 grub rescue> 提示符号下，进行以下步骤来尝试启动至最新的内核。

1. ls

2. set root=/dev/sdXY

3. ls /boot

4.insmod /boot/grub/_linux.mod
- 此命令将显示已知的设备和磁盘分区。由此信息，用户必须决定系统安装在哪一个设备和磁盘分区。

X 为设备/磁盘，由 0 开始。Y 为磁盘分区，由 1 开始。（范例：(hd0,1) 是 sda1。(hd3.5) 是 sdc5。）

检验内容。用户应该能看到多个内核、initrd 影像和 grub 目录。若没有，使用 ls 命令来检查设备并试着找寻文件和目录。若有必要，配置另一个设备 为 root。

加载 linux 模块。此模块未加载，若没有加载此模块，用户将会收到一个「Unknown command linux」

5. **linux /vmlinuz root=/dev/sdXY ro**

6. **initrd /initrd.img**

7. **boot**

信息。

载入 *linux* 内核，以正确的名称替换掉「X」和「Y」（例如：sda1）。用户将会看到内核已被加载的信息。（参看上图）

加载 *initrd* 影像。当按下 **enter** 时，使用者可能会或不会看到终端中的信息。（参看上图的高亮度部分）

启动。

成功启动进入系统后，用户必须重新安装 GRUB 2（`sudo grub-install /dev/sdX`）或搜索及编辑 GRUB 2 配置文件来做必要的修改。

GRUB 错误信息

GRUB 2 并不会回报错误码。如果出现错误信息并带着一个错误码，这是由 GRUB 过渡至 GRUB 2 时发生的错误。

「Unrecognized Device String」 (Error 11)

此错误信息通常是因为菜单选项中的异常内容。最常见的原因因为使用者试图从使用连锁加载（Chainload）功能的 Jaunty 升级至 GRUB 2 导致。

1. 当启动 GRUB 2 时显示以下信息：

```
Error 11: Unrecognized device string
Press any key to continue..._
```

2. 记住 Jaunty 的连锁加载功能是有问题的，使用者按下任意键来回到主菜单，然后按「e」直到看见「grub edit>」。编辑高亮度的「Chainload into GRUB 2」选项或任何用户试图用来启动的选项。


```
[ 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.  ESC at any time
exits. ]
```

```
grub edit> root  b42c61b0-2520-4838-9deb-c9168f946a02
```

3. 有敏锐记忆力的使用者应该还记得「root」需要修改为「uuid」。使用箭头键来移动光标，用「uuid」来取代「root」。当结束编辑时，使用者按下 ENTER 键，然后再按「b」来使用修改过的选项来启动。注意这些命令与 GRUB 2 中使用的不同。

```
grub edit> uuid  b42c61b0-2520-4838-9deb-c9168f946a02
```

若使用者打算保留双重 GRUB 选择，以 root 身份开启 /boot/grub/menu.lst 并将「Chainload into Grub 2」菜单选项第二行「root」修改为「uuid」以修正错误。

```
title          Chainload into GRUB 2
uuid          65106cde-757e-49ff-a44c-8918100289a0
kernel         /boot/grub/core.img
```

「File Not Found 」(Error 15)

此错误为 GRUB 2 安装至 /boot 但主启动扇区（MBR）仍然为传统 Grub。此状况发生在当您执行 `sudo update-from-grub-legacy` 时未选择您的磁盘。执行此命令后用户马上会被要求选择一个设备（sda、sdb 等）。当显示此画面时选择磁盘并按下空格键。没有选择将会导致 Error 15 的产生。

欲修复此错误，必须要重新安装 GRUB 2。详细命令请参考从[启动光盘重新安装](#)小节。

引导画面和主题

GRUB 2 引导画面

GRUB 2 较之前版本大大的加强了引导画面功能。配置引导画面需要进行两个步骤：选择图片以及选择显示的文字颜色。

安装默认的 GRUB 2 图片组件，请安装「grub-splashimages」组件：

```
sudo apt-get install grub2-splashimages
```

不要把它与 *grub-splashimages* 组件搞错，否则会与 GRUB 2 不兼容。

这些下载图片的默认路径在 */usr/share/images/grub*。这个路径或文件所放置的其他路径必须加进 *05_debian_theme* 中。（参见下方）

用户可以建立并使用他们自己的引导画面。



配置引导画面

GRUB 2 的引导画面由 */etc/grub.d/05_debian_theme* 所控制。以 root 身份打开此文件进行编辑。

找到以下行并编辑粗体部分，以你想要使用的 GRUB 2 引导画面路径和名称来取代：

```
for i in { /boot/grub,/usr/share/images/desktop-base }/moreblue-orbit-grub.{png,tga} ; do
```

在此项目中有些地方需要注意：

- GRUB 2 默认会在 */boot/grub* 和 */usr/share/images/desktop-base* 中寻找引导画面
- *grub2-splashimages* 组件并非解压缩至以上的两个目录中，而是在 */usr/share/images/grub*
- 使用者必须在以下两动作中择一：
 - ◆ 修改上面一行的目录位置并指向正确的位置，或
 - ◆ 将引导画面移动到上面所提到的两个目录之一。

如果想增加搜索的目录，按如下方式修改此文件：

```
for i in { /boot/grub,/usr/share/images/desktop-base,/usr/share/images/grub }/moreblue-orbit-grub.{png,tga} ; do
```

- 文件名以句点结束 (.) (/moreblue-orbit-grub.)
- 引导画面选择：
 - ◆ 若相同文件名文件存在于超过一个目录中，最早指定的目录将会被使用。
 - ◆ 若文件名相同而扩展名不同（多种扩展名），则最早指定的扩展名文件将会被选择。

使用者可以对上述配置行做更多修改以增加引导画面的弹性：

- 加入用户拥有的目录，让 GRUB 2 可以搜索到更多可用的引导画面。
- 加入使用 JPEG 格式启动图像文件的能力。此时只有 8-bit JPEG 格式图片能被正确显示。

以下展示了新增加的用户拥有的引导画面目录以及增加 JPEG 支持：

```
for i in { /boot/grub,/usr/share/images/grub,/home/drs/mysplash }/moreblue-orbit-grub.{png,tga,jpg,jpeg} ; do
```

一旦您修改了 *05_debian_theme* 中，您可执行 `sudo update-grub` 来加新增信息加入 *grub.cfg*。若引导画面加入成功，用户将会在终端执行命令时看到「*Found Debian image...*」。若没有看到此信息，则背景图片未配置成功。

* 鱼漂特别提醒：通常你还需要修改这里的图片文件名，也就是 **moreblue-orbit-grub**，比如改成 **grub**，那么它将搜索 **grub.png,grub.tga** 文件。

配置菜单字体&高亮度色彩


此行包含在 `/etc/grub.d/05_debian_theme` 中。这些命令不会在靠近文件上方的主题配置设为「mono」的状态下生效。

- 未选取的菜单选项色彩如以下范例行来配置。第一个色彩是文字色彩；第二个是背景色彩。

```
set color_normal=black/black
```

- 以下的代码配置被选取选项的色彩。第一个色彩是高亮度文字色彩；第二个是被选定行的背景色彩。若 `/black` 为第二个选项，则高亮度行将会是透明的，且只有文字色彩会改变。

```
set color_highlight=magenta/black
```

 当作为第二个色彩时 `/black` 是 GRUB 2 的默认透明色。若使用引导画面时，只要改变第一个色彩（例如 `xxxxxx/black`）。若第二个色彩更改为黑色以外的颜色，引导画面将会被藏在实色背景之后。


- ◆ 在下图中，其 `color_normal` 配置为 `white/black`。高亮度色彩配置为 `cyan/blue`

```
Example: set color_normal=red/black Note /black is transparent
Example: set color_highlight=white/blue
```

- ◆ 在 GRUB 2 中可使用这些色彩。第二选项设为「black」则会被认定为「透明」。

black	blue	brown	cyan	dark-gray	green	light-cyan	light-blue
light-green	light-gray	light-magenta	light-red	magenta	red	white	yellow

测试字体和引导画面

 与其重新启动来测试色彩组合，用户也可通过在启动过程使用 GRUB 2 命令行来测试。

- 依照上方说明来选择引导画面和初始字体色彩，以 `sudo update-grub` 来更新 GRUB 2 的文件后重新启动。
- 当 GRUB 2 引导画面/菜单出现时，按下任何键以停止倒计时。
- 按 `c` 进入 GRUB 2 命令行。
 - ✧ `set color_normal=first-color/black` 或 `set color_highlight=first-color/second-color`
 - ✧ 在命令输入后立即可以看到文字的色彩。您可以使用「向上」键来呼叫最后执行的命令然后编辑就能快速测试不同的色彩组合。

- ✧ 一旦您想结束，按下 **ESC** 键回到菜单。
- ✧ 随意重复上述步骤。
- ✧ 记下想使用的色彩组合，正常启动然后编辑 `/etc/grub.d/05_debian_theme` 并让修改永久储存。
- ✧ 保存，更新 GRUB 2: `sudo update-grub`

修改分辨率及引导画面

若用户想在使用引导画面时修改 GRUB 2 屏幕分辨率，请按以下步骤：

1. 在 `/etc/default/grub` 中配置欲使用的分辨率
 - 修改 **GRUB_GFXMODE=** 之值（例如：`GRUB_GFXMODE=800x600`）
 - 若无法确认 GRUB 2 可以使用何分辨率，可以在 GRUB 2 命令行下输入 `vbeinfo` 来显示。命令行可以在当 GRUB 2 主菜单显示的时候按下「c」。
2. 选择一个同样大小的图片并在 `/etc/grub.d/05_debian_theme` 中修改。
 - 图片名称位置在以「`for i in {/boot}`」为开头的那一行里
 - 若没有使用正确大小的图片，菜单的位置将不会是正确的。
 - 使用您选择的图片编辑程序来建立/修改正确尺寸的图片。
 - 用户也可以通过文件浏览器来检查图片的尺寸（在 Nautilus 中检查属性）。
3. 以 root 权限执行 `update-grub` 将新增的配置加入 `/boot/grub/grub.cfg`

建立用户引导画面

GRUB 2 的引导画面管理让各式各样的引导画面变得容易。

分辨率配置

在 `grub2-splashimages` 组件中的引导画面主要为 640x480 的图片。

GRUB 2 会在 `/etc/default/grub` 中寻找分辨率配置。若取消注释，则分辨率由此行决定：

```
GRUB_GFXMODE=640x480
```

- 若在 `/etc/default/grub` 中没有找到分辨率配置，GRUB 2 会使用配置在 `/etc/grub.d/00_header` 中 `640x480` 这个配置值。

图片建立和配置



在 Ubuntu 中，参考系统的偏好配置中的显示以取得可用的屏幕分辨率，或是在 GRUB 2 命令行下执行 `vbeinfo` 来显示可用的分辨率。

1. 决定您想在 GRUB 2 引导画面中使用的分辨率。
2. 建立或修改任何图片至合适的尺寸
 - GRUB 2 目前支持 `.png`, `.tga`, 以及 8-bit `.jpg/.jpeg` 图片。
 - ✧ 若使用 GIMP，将文件以适合的扩展名命名可存盘为该格式。
3. 在 GRUB 2 配置文件中作合适的修改：
 - `/etc/default/grub`: 取消 `GRUB_GFXMODE=640x480` 行的注释，若分辨率不是 `640x480` 则修正其值。
 - `/etc/grub.d/05_debian_theme`: 在以下行中输入正确的路径及文件名。更多如何修改此行的信息请参考前一小节。

```
for i in {/boot/grub,/usr/share/images/desktop-base}/moreblue-orbit-grub.{png,tga} ; do
```

4. 执行 `sudo update-grub` 将修改合并至 GRUB 2

主题

GRUB 2 主题还在开发中，将与 `gfxmenu` 整合。主题元素包含了色彩、字体、进度条、菜单和标签。这些功能尚未在 Ubuntu 9.10 发行时准备好。



重新安装 GRUB 2

有时使用者可能需要移动或是重新安装 GRUB 2。前者也许发生在当用户想要将系统移动到另一台设备上或是重新配置安装命令。若用户完全无法启动进入操作系统，甚至无法使用[救援模式]，那么也许必须要重新安装 GRUB 2。

更改或移动 GRUB 2

更改 GRUB 2 安装设备或启动文件的命令为以 root 身份执行 `grub-install`。此命令允许用户配置根（ROOT）目录、加载模块、执行指定的配置文件或其他方式来修改安装。当执行时，`grub-install` 可能会执行一或多个其他命令，例如 `grub-probe`、`grub-mkimage` 和 `grub-setup`。以下是当执行 `grub-install` 时一些需要考虑的：

- 在一般情况下应该使用 `grub-install` 命令而不是 `grub-setup`。当需要的时候 `grub-install` 会调用 `grub-setup`。
- 此命令应指定一设备，当执行时将会安装 GRUB 需要的文件到指定的设备上。（例如：`sudo grub-install /dev/sda`）
- 若用户试图在执行此命令时指定一个磁盘分区，（例如：`sudo grub-install /dev/sda6`）会发出一个警告信息。不建议将 GRUB 2 安装在分区上。然而还是可以选择安装在分区上。

- `grub-install` 可用选项列表可在终端中执行 `grub-install --help` 来显示。
- `grub-install` 的帮助文件（man page）目前尚不会显示所有可用的选项。

从启动光盘重新安装

若您无法从 GRUB 2 启动，参考启动疑难和救援模式小节。若需要重新安装，跟着以下命令：

1. 使用 9.10 Karmic Desktop 启动光盘启动。
2. 开启终端 - 应用程序（Applications），附件（Accessories），终端（Terminal）。
3. 判断您的标准系统磁盘分区 - （参数是一个小写的「L」）

```
sudo fdisk -l
```

若您无法确定，执行 `df -Th`。找出正确的磁盘容量和 `ext3` 或 `ext4` 格式。挂载您的标准系统磁盘分区：

4. 挂载您的标准系统磁盘分区：

- ✧ 以正确的磁盘分区代号替换：`sda1`、`sda5` 等等。
- ✧ GRUB 2 计算第一台磁盘驱动器为「0」，而第一个磁盘分区为「1」。

5. `sudo mount /dev/sdXX /mnt` # 范例：`sudo mount /dev/sda1 /mnt` 注意：若用户有任何其他独立的系统磁盘分区，例如 `/usr`，这些也应用同样的方式挂载。

6. 只在您有独立的 `/boot` 磁盘分区才需要执行：

- ✧ `sdYY` 为独立 `/boot` 磁盘分区代号（例如 `sdb3`）
- ✧ `sudo mount /dev/sdYY /mnt/boot`

7. 挂载设备：

```
sudo mount --bind /dev/ /mnt/dev
```

8. Chroot 至您的标准系统设备：

```
sudo chroot /mnt
```

9. 重新安装 GRUB 2：

以正确设备替换 - **sda**、**sdb** 等等。不要指定磁盘分区数字。

10. `sudo grub-install /dev/sdX` 验证安装（使用正确设备，例如 **sda**。不要指定磁盘分区数字）：

```
sudo grub-install --recheck /dev/sdX
```

11. 退出 chroot：按下键盘 CTRL-D

12. 卸除设备：

```
sudo umount /mnt/dev
```

✧ 若您挂载独立的 **/boot** 磁盘分区：

■ `sudo umount /mnt/boot`

13. 最后卸除设备：

```
sudo umount /mnt
```

14. 重新启动。

```
reboot
```

一旦用户可以启动至正常运作的系统，试着判断系统先前为何无法启动。以下命令证实对于找到和修复此问题非常有效。

- 更新 `/boot/grub/grub.cfg` 中的可用设备及配置
 - ◆ `sudo update-grub`
- 寻找启动加载程序位置。
 - ◆ `grub-probe -t device /boot/grub`
- 安装 GRUB 2 至 `sdX` 磁盘分区的主要启动扇区（MBR）（`sda`、`sdb` 等等）
 - ◆ `sudo grub-install /dev/sdX`
- 再度检查安装（`sda`、`sdb` 等等）
 - ◆ `sudo grub-install --recheck /dev/sdX`

卸载 GRUB 2

回复至传统 GRUB

若使用者选择回复至传统 GRUB（0.97），以下步骤将卸载 GRUB 2 并安装 GRUB。

命令行模式提供较清楚干净的卸载和重新安装。然而 `Synaptic` 也可以做到新增和卸载组件，某些步骤仍然要进入终端来完成。

1. 开启终端：应用程序（Applications），附件（Accessories），终端（Terminal）。

2. 备份 GRUB 2 目录和文件。（可选）

- ◆ `sudo cp /etc/default/grub /etc/default/grub.old`
- ◆ `sudo cp -R /etc/grub.d /etc/grub.d.old`
- ◆ `sudo cp -R /boot/grub /boot/grub.old`

3. 卸载 GRUB 2

- ◆ `sudo apt-get purge grub2 grub-pc`



此时在其他启动加载程序安装前，系统将无法启动。

- ◆ 当组件卸载后，许多文件仍然会留在「/boot/grub」下

4. 安装 GRUB 0.97

- ◆ `sudo apt-get install grub`

5. 当 grub 安装后，使用者仍然需要执行以下两个命令来建立 menu.lst 和 stage1/stage2 文件。

- ◆ `sudo update-grub`
 - ✧ 产生 menu.lst
 - ✧ 提示时选择「Yes」。
- ◆ `sudo grub-install /dev/sdX`
 - ✧ 选择正确的设备（sda、sdb 等），一般而言就是 Ubuntu 安装的那个设备
 - ✧ 在 /boot/grub 中建立 stage1/stage2 文件并写入至主启动扇区（MBR）。

6. 重启动。



若使用者在重新启动时收到一个「Unrecognized device string Error 11」错误信息，参考解决「Unrecognized Device String」 (Error 11) 小节，有如何编辑菜单和使系统可启动的命令。