



21世纪高等学校计算机
专业实用规划教材

Linux 操作系统 应用与开发教程

◎ 邱铁 编著

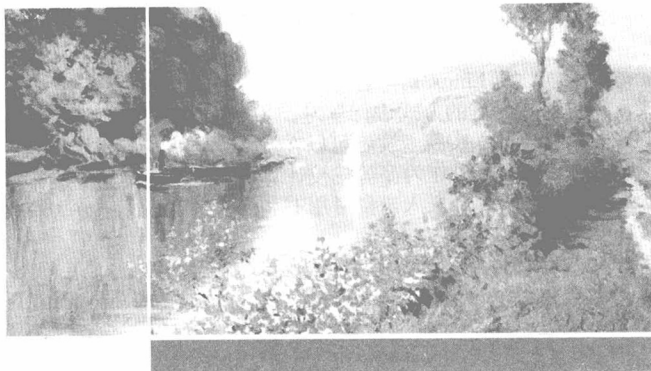


清华大学出版社





21世纪高等学校计算机
专业实用规划教材



Linux 操作系统 应用与开发教程

◎ 邱铁 编著

清华大学出版社
北京

内 容 简 介

本书针对学习者如何快速掌握 Linux 应用、开发、内核编程与高级编程,以最新的 Linux 内核版本 3.0.x~3.19.x 为依据,设计典型实例,并对开发场景进行详细讲解。在内容设计上,本书采取了循序渐进的原则,对 Linux 系统知识进行精心策划,使 Linux 初学者能够以“Linux 应用基础→编程开发→内核源码与场景分析→高级编程”为主线,以阶梯式前进的方式进行学习。

本书立足于基础,实例经典,深入实践。读者可以在较低起点下进行高效的理论与实践学习,为 Linux 系统应用与开发打下坚实的基础。本书可作为高等院校计算机、信息类大学生、研究生进行操作系统学习和开发的教材或参考书,也可作为 Linux 开发人员参考用书和广大的 Linux 爱好者自学教程。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Linux 操作系统应用与开发教程/邱铁编著. --北京:清华大学出版社,2016

21 世纪高等学校计算机专业实用规划教材

ISBN 978-7-302-44032-1

I. ①L… II. ①邱… III. ①Linux 操作系统-高等学校-教材 IV. ①TP316.89

中国版本图书馆 CIP 数据核字(2016)第 127827 号

责任编辑:魏江江 赵晓宁

封面设计:刘 键

责任校对:胡伟民

责任印制:何 芊

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:三河市春园印刷有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:24.75

字 数:615 千字

版 次:2016 年 8 月第 1 版

印 次:2016 年 8 月第 1 次印刷

印 数:1~2000

定 价:49.50 元

产品编号:064844-01

出版说明

随着我国改革开放的进一步深化，高等教育也得到了快速发展，各地高校紧密结合地方经济建设发展需要，科学运用市场调节机制，加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度，通过教育改革合理调整和配置了教育资源，优化了传统学科专业，积极为地方经济建设输送人才，为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是，高等教育质量还需要进一步提高以适应经济社会发展的需要，不少高校的专业设置和结构不尽合理，教师队伍整体素质亟待提高，人才培养模式、教学内容和教学方法需要进一步转变，学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007年1月，教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》，计划实施“高等学校本科教学质量与教学改革工程（简称‘质量工程’）”，通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容，进一步深化高等学校教学改革，提高人才培养的能力和水平，更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中，各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势，对其特色专业及特色课程（群）加以规划、整理和总结，更新教学内容、改革课程体系，建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上，经教育部相关教学指导委员会专家的指导和建设，清华大学出版社在多个领域精选各高校的特色课程，分别规划出版系列教材，以配合“质量工程”的实施，满足各高校教学质量和教学改革的需要。

本系列教材立足于计算机专业课程领域，以专业基础课为主、专业课为辅，横向满足高校多层次教学的需要。在规划过程中体现了如下一些基本原则和特点。

(1) 反映计算机学科的最新发展，总结近年来计算机专业教学的最新成果。内容先进，充分吸收国外先进成果和理念。

(2) 反映教学需要，促进教学发展。教材要适应多样化的教学需要，正确把握教学内容和课程体系的改革方向，融合先进的教学思想、方法和手段，体现科学性、先进性和系统性，强调对学生实践能力的培养，为学生知识、能力、素质协调发展创造条件。

(3) 实施精品战略，突出重点，保证质量。规划教材把重点放在公共基础课和专业基础课的教材建设上；特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版，逐步形成精品教材；提倡并鼓励编写体现教学质量和教学改革成果的教材。

(4) 主张一纲多本，合理配套。专业基础课和专业课教材配套，同一门课程有针对不同层次、面向不同应用的多本具有各自内容特点的教材。处理好教材统一性与多样化，基本教材与辅助教材、教学参考书，文字教材与软件教材的关系，实现教材系列资源配套。

(5) 依靠专家，择优选用。在制定教材规划时要依靠各课程专家在调查研究本课程教

材建设现状的基础上提出规划选题。在落实主编人选时，要引入竞争机制，通过申报、评审确定主题。书稿完成后要认真实行审稿程序，确保出书质量。

繁荣教材出版事业，提高教材质量的关键是教师。建立一支高水平教材编写梯队才能保证教材的编写质量和建设力度，希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机专业实用规划教材

联系人：魏江江 weijj@tup.tsinghua.edu.cn

前 言

在 IT 文化繁荣的今天，Linux 操作系统作为源码开放的自由软件，是迄今为止，由网络黑客参与开发的规模、性能完善的操作系统。从它的产生、发展和不断完善的历程中，凝聚了众多 IT 人对最优方案的不懈努力。时至今日，Linux 内核版本发展到了 3.x 版本以后，其版本更新速度相对以前逐渐趋于缓慢。这也正是 Linux 内核走向成熟化的标志之一。

Linux 的设计参照了流行的网络操作系统 UNIX，具有很强的兼容性和稳定性。Linux 还是自由软件项目 GNU 的重要组成部分。它目前广泛地应用于计算机科学研究、软件技术研究、网络服务后台系统等需要高可靠性、高复杂度的环境之中。因此，学习使用 Linux 也成为计算机专业人员所必备的技能之一。

面对庞大而复杂的 Linux 操作系统，令很多学习者无从下手。本书正是在这种背景下编写的，紧紧围绕着本书的写作主线“Linux 应用→编程开发→内核源码与场景分析→图形界面高级编程”，以当前最新的 Linux 内核源代码为依据，以软件开发人员学习的实际需要为基本，采用“理论讲解+实例解析”的方式对 Linux 进行了全面细致的讲解。主要分为以下 4 大部分：

- 基于流行的 Ubuntu 发行的 Linux 操作系统，对 Linux 常用的命令配合实例进行详细讲解；介绍了利用命令脚本进行 Shell 编程；介绍了 Linux 下文本编辑器 vi 的使用方法；介绍了 GCC 源代码安装过程以及 GNU 开发工具链的使用。
- 对常用的 Linux 中 C 函数库进行了讲解；介绍了 Linux 多进程处理与进程之间的通信；介绍了 Linux 内存资源管理函数；介绍了 Linux 中用户管理和对用户信息进行编程操作；介绍了文件和目录的处理、文件内容的处理等编程方法。
- 以当前最新的 Linux 内核源代码为依据，分析内核各功能模块原理，包括 Linux 内核裁剪与编译、模块机制与操作方式、中断上半部和下半部处理机制、系统调用机制的实现方式、内存管理、定时器管理以及向内核添加驱动程序的方法，并给出大量的场景分析和实例验证。
- 在高级编程里介绍了 QT 图形界面的开发方法、MySQL 数据库程序的开发方法，还介绍了以 CS 模式 Socket 模型为例的基于 TCP 协议的网络应用开发。

感谢所有参与本书构思、解决方案、编辑和出版工作的同事、同行和为本书编写提供灵感的同学们。于玉龙参加了代码调试和部分文档整理，陈宝超、陈宁、王云成、王逸非、王鑫等参加了代码的版本升级工作，在此向他们表示感谢。

Linux 在众多的网络黑客的参与下，其内核版本和代码结构不断更新。书中错误和不妥之处，恳请读者批评指正。

邱 铁

2015 年 10 月于大连

目 录

第 1 部分 Linux 系统应用篇

第 1 章 Linux 的安装与配置	2
1.1 旅程开始	2
1.2 本书使用 Linux 环境介绍	2
1.3 Linux 的安装	4
1.3.1 获取 Ubuntu 14.04	4
1.3.2 选择安装平台	4
1.3.3 进入 Ubuntu 14.04 的安装程序	9
1.3.4 安装 Ubuntu 14.04	10
1.4 Linux 的配置	19
1.4.1 认识 Gnome 桌面环境	19
1.4.2 Ubuntu 的配置	22
1.4.3 软件安装与升级	24
第 2 章 Linux 常用命令训练	26
2.1 关于 Shell	26
2.2 文件操作命令	27
2.2.1 调用终端控制台	27
2.2.2 文件浏览	27
2.2.3 文件复制	29
2.2.4 文件移动	29
2.2.5 文件链接	30
2.2.6 文件删除	30
2.2.7 文件压缩和备份	31
2.2.8 修改文件属性	32
2.2.9 文件搜索	34
2.3 目录操作	36
2.3.1 创建目录	37
2.3.2 删除目录	37
2.3.3 修改当前目录	37

2.3.4	查看当前目录	37
2.4	用户与系统操作	38
2.4.1	用户切换	38
2.4.2	用户信息修改	39
2.4.3	关闭系统	39
2.5	获得帮助	40
2.5.1	获取简要帮助	40
2.5.2	获得详细帮助	40
2.6	变量、流、管道操作	41
2.6.1	变量赋值	41
2.6.2	变量的使用	41
2.6.3	流输出	42
2.6.4	流的重定向	44
2.6.5	管道	45
2.7	进程操作	45
2.7.1	进程查看	46
2.7.2	发送信号	46
2.7.3	进程切换	47
2.8	网络操作	48
2.8.1	网络配置	48
2.8.2	ping	48
2.8.3	ARP	49
2.8.4	FTP	50
2.9	其他命令	50
2.9.1	日历	50
2.9.2	命令历史记录	51
2.9.3	后台操作	51
2.10	思考与练习	52
第3章 vi/vim 编辑器的使用		53
3.1	vi 的介绍	53
3.2	vi 操作模式	53
3.3	vi 的命令	54
3.3.1	状态切换命令	54
3.3.2	文件保存与退出	54
3.3.3	光标移动	54
3.3.4	编辑操作	55
3.3.5	字符串搜索替换	55
3.3.6	撤销与重做	55

3.4	启动 vi 编辑器	56
3.5	使用 vi 进行文字录入	56
3.6	使用 vi 修改文本	57
3.7	思考与练习	58
第 4 章	Shell 程序设计	59
4.1	Shell 编程简介	59
4.2	系统变量	59
4.3	条件测试	60
4.3.1	文件状态测试	60
4.3.2	逻辑操作	60
4.3.3	字符串测试	61
4.3.4	数值测试	61
4.4	Shell 流程控制语句	61
4.4.1	if 语句	61
4.4.2	case 语句	62
4.4.3	while 语句	62
4.4.4	for 语句	63
4.5	Shell 编程中的常用命令与符号	63
4.5.1	read 命令	63
4.5.2	select 命令	63
4.5.3	大括号	64
4.5.4	引号	64
4.5.5	注释	65
4.6	函数	65
4.7	应用实例训练	65
4.8	思考与练习	67
第 5 章	GCC 的安装	68
5.1	GCC 简介	68
5.2	解压缩工具 tar	68
5.3	在 Linux 下使用源代码安装软件的基本步骤	69
5.4	获得 GCC 软件包	70
5.5	解压缩软件包	72
5.6	对源文件进行配置	72
5.7	编译 GCC	74
5.8	安装 GCC	75
5.9	测试 GCC 安装结果	76
5.10	思考与练习	77

第 6 章 GNU 开发工具链的使用	78
6.1 gcc 命令的使用	78
6.2 调试工具 gdb	79
6.2.1 gdb 简介	79
6.2.2 gdb 的使用方法	79
6.3 代码管理 make	81
6.3.1 make 简介	81
6.3.2 Makefile 文件的格式	82
6.3.3 Makefile 文件的一些特性	83
6.3.4 make 命令的使用	85
6.4 实例训练	85
6.4.1 编写程序	87
6.4.2 调试程序	95
6.4.3 编写 Makefile	98
6.5 思考与练习	99

第 2 部分 编程开发篇

第 7 章 Linux 常用 C 函数	102
7.1 使用函数库	102
7.2 字符操作	103
7.3 字符串操作	107
7.3.1 数据类型转换	108
7.3.2 字符串数据处理	111
7.4 数学计算操作	116
7.5 数据结构与算法操作	118
7.6 日期时间操作	123
7.7 实例训练	125
7.7.1 任务分析	125
7.7.2 编写程序	127
7.7.3 编译、运行	134
7.8 思考与练习	135
第 8 章 进程操作	136
8.1 Linux 进程工作原理	136
8.2 进程操作函数	137
8.3 信号量	144
8.4 信号量操作的函数	145

8.5	应用实例训练	146
8.5.1	问题分析	147
8.5.2	代码编写	147
8.5.3	编译与运行	152
8.6	思考与练习	153
第 9 章	信号与定时器	154
9.1	进程间通信与信号	154
9.2	Linux 系统中的信号	154
9.3	信号操作相关数据结构	155
9.4	信号操作相关函数	156
9.5	定时器操作相关函数	160
9.6	应用实例训练	161
9.6.1	程序分析	162
9.6.2	程序编写	163
9.6.3	编译与运行	166
9.7	思考与练习	167
第 10 章	内存管理与用户操作	168
10.1	Linux 内存管理	168
10.2	内存操作相关函数	169
10.3	Linux 系统中的用户操作	175
10.4	用户管理相关数据结构	176
10.5	用户管理相关函数	177
10.6	用户组管理相关函数	182
10.7	应用实例训练	184
10.7.1	编写代码	185
10.7.2	编译与运行	193
10.8	思考与练习	194
第 11 章	文件操作	195
11.1	Linux 的文件系统	195
11.2	文件操作相关函数	197
11.2.1	文件控制	197
11.2.2	目录操作	200
11.2.3	文件流读写控制	204
11.2.4	文件读写操作	211
11.3	应用实例训练	217
11.3.1	程序分析	217

11.3.2	程序编写	218
11.3.3	编译与运行	225
11.4	思考与练习	225

第3部分 Linux 内核篇

第12章	Linux 内核裁剪与编译	228
12.1	内核编译选项	228
12.1.1	常规设置	228
12.1.2	可加载模块支持	229
12.1.3	处理器类型及特性	229
12.1.4	可执行文件格式	229
12.1.5	网络支持	229
12.1.6	设备驱动程序选项	230
12.1.7	文件系统	231
12.1.8	对于其他配置选项的说明	231
12.2	内核编译与定制	231
12.2.1	获得 Linux 内核与补丁	231
12.2.2	准备编译需要的工具	233
12.2.3	解压内核	234
12.2.4	给内核打补丁	234
12.2.5	设定编译选项	235
12.2.6	编译与安装内核	237
12.3	安装引导配置	238
12.3.1	创建 initramfs	238
12.3.2	设置 grub	239
12.3.3	启动选项	239
12.4	思考与练习	240
第13章	模块机制与操作	241
13.1	关于内核编程	241
13.2	Linux 的模块机制	241
13.2.1	Linux 内核结构	241
13.2.2	模块的实现	242
13.2.3	Linux 模块导出符号表	244
13.2.4	模块参数	244
13.2.5	模块使用计数	245
13.3	内核调试函数 <code>printk()</code>	245
13.4	应用实例训练	247

13.4.1	编写模块源程序	247
13.4.2	Linux kernel 2.6.26 之前版本模块编译、安装及退出	251
13.4.3	Linux kernel 2.6.26 以后版本模块编译、安装及退出	253
13.5	思考与练习	256
第 14 章	Linux 中断管理	257
14.1	Linux 中断原理	257
14.1.1	中断控制器	257
14.1.2	中断处理	258
14.1.3	中断处理的下半部机制	261
14.2	Tasklet 实例解析	262
14.2.1	编写测试函数	262
14.2.2	编写 Makefile	263
14.2.3	实验结果分析	263
14.3	在嵌入式 Linux 下开中断实例解析	264
14.3.1	硬件电路组成	264
14.3.2	编写中断服务模块	265
14.3.3	结果分析	267
14.4	思考与练习	268
第 15 章	系统调用	269
15.1	系统调用原理	269
15.2	系统调用函数分析	270
15.2.1	系统调用入口函数	270
15.2.2	系统调用表	273
15.3	添加系统调用实例训练	275
15.4	思考与练习	277
第 16 章	内存管理	278
16.1	关于 Linux 的内存管理	278
16.1.1	动态存储管理	279
16.1.2	页面管理	279
16.1.3	slab 分配模式	280
16.2	Linux 的内存管理函数	281
16.3	实例训练与分析	282
16.3.1	在用户空间用 valloc/malloc 分配内存	282
16.3.2	在内核空间用 kmalloc/vmalloc 分配内存	282
16.4	思考与练习	290

第 17 章 时钟定时管理	291
17.1 内核定时器分类	291
17.1.1 实时时钟 RTC	291
17.1.2 时间戳计数器 TSC	291
17.1.3 可编程间隔定时器 PIT	291
17.1.4 SMP 系统上的本地 APIC 定时器	292
17.1.5 高精度计时器	293
17.2 内核时钟管理分析	293
17.2.1 时钟源及其初始化	293
17.2.2 软定时器	295
17.3 应用实例训练	296
17.3.1 编写测试实例	296
17.3.2 编写 Makefile	298
17.3.3 编译及运行结果	298
17.4 思考与练习	299
第 18 章 设备驱动程序的编写	300
18.1 Linux 驱动程序	300
18.1.1 驱动程序分类	300
18.1.2 驱动程序开发的注意事项	301
18.1.3 设备目录	301
18.2 Linux 驱动数据结构分析	302
18.2.1 Linux 驱动核心结构体	302
18.2.2 设备的内核操作函数	304
18.3 驱动程序实例训练	306
18.3.1 以模块的方式加载驱动程序	306
18.3.2 测试驱动程序	311
18.4 编译时向内核添加新设备	312
18.5 思考与练习	315

第 4 部分 高级编程篇

第 19 章 Qt 图形界面设计	318
19.1 X-Windows 概述	318
19.2 Qt 编程	318
19.2.1 概述	318
19.2.2 Qt Creator	319

19.2.3	Qt 信号与 Slot 机制	319
19.3	Qt 安装方法	321
19.4	应用实例训练	324
19.4.1	创建工程目录打开 Qt Creator	324
19.4.2	新建工程	324
19.4.3	绘制窗体	325
19.4.4	编写代码	327
19.4.5	编译运行	330
19.5	思考与练习	330
第 20 章	MySQL 数据库设计与编程	331
20.1	MySQL 的特性	331
20.2	数据库编程概述	333
20.3	Qt 中的数据库编程	333
20.3.1	QSqlDriver	333
20.3.2	QSqlDatabase	334
20.3.3	QSqlQuery	336
20.4	应用实例训练	338
20.4.1	数据库的建立	338
20.4.2	应用程序的建立	344
20.4.3	运行结果	351
20.5	思考与练习	353
第 21 章	网络通信高级编程	354
21.1	网络编程概述	354
21.2	Socket 编程模型	355
21.3	Qt 网络编程中用到的类和方法	355
21.3.1	QtcpSocket	355
21.3.2	QTcpServer	357
21.3.3	QThread	359
21.4	应用实例训练	359
21.4.1	建立工程	360
21.4.2	数据结构设计	362
21.4.3	界面设计	367
21.4.4	动作设计	368
21.4.5	编译与运行	372
21.5	思考与练习	374
	后记	375

第 1 部分 Linux 系统应用篇

欢迎来到多彩的 Linux 世界。Linux 是一款专门为计算机软件开发人员设计的操作系统，不仅包含了丰富的开发工具，并且提供了许多令人惊叹的功能。为了更好地利用 Linux，必须要深入了解 Linux。首先，要学习如何使用 Linux，如何利用 Linux 中的各种工具。

作为全书的第一部分，这里从最基础的系统安装和基本的命令操作入手，一步一步地进入 Linux 的软件开发世界。相信在阅读这一部分之后，读者会对 Linux 有一个感性的认识，并为之后的阅读打下一个坚实的基础。

本部分主要内容提要：

- 第 1 章介绍基于流行的 Ubuntu 14.04 对 Linux 操作系统的安装，并对 Linux 在安装后的设置进行了说明。
- 第 2 章介绍 Linux 常用的命令，学习 Linux 命令是操作 Linux 的基础，按照功能分类介绍了大量常用的 Linux 命令。
- 第 3 章介绍 Linux 下最常用的基于命令行的全屏幕文字编辑器 vi，它通常作为程序开发的代码输入工具，是软件开发人员必须掌握的工具之一。
- 第 4 章介绍 Shell 编程，常常用于系统配置，详细介绍了 Linux 下 Shell 编程及其特性。
- 第 5 章介绍 Linux 下最常用的开源代码编译工具 GCC，并以 GCC 为例介绍 Linux 下使用源代码安装软件的过程。
- 第 6 章初步介绍了 GNU 开发工具链的使用，包括编译工具 gcc、调试工具 gdb、工程管理工具 make 等。通过实例，全面展示了各种工具的具体使用方法。

学习本章要达到的目标:

- (1) 对 GNU 有一定的了解。
- (2) 初步建立对 Linux 操作系统的感性认识。
- (3) 为后面章节的学习进行一系列准备。

1.1 旅程开始

这里开始一段难忘的“Linux 之旅”，相信在认真阅读过本书并完成相应的实例之后，读者会对 Linux 有一个全新的了解和认识。本书由浅入深，一步一步地把读者引进一个奇妙而又充满乐趣的 Linux 世界。

首先对 Linux 的历史做一个了解。说到 Linux 就不得不提到一个词——GNU (如图 1.1 所示)。GNU 项目 (the GNU Project) 是一个自由软件计划，旨在建设一个完全自由开放通用的 UNIX 软件平台。它的创始人 Richard M. Stallman 于 1983 年提出了 GNU 操作系统 (GNU Operating System) 的方案，并于 1984 年启动了 GNU 项目。GNU 的含义是 GNU's Not UNIX，是一个非常有趣的递归定义。自由软件项目吸引了一大批开发人员和电脑黑客，此后越来越多的人加入了自由软件开发的行列。这些自由软件的开发为计算机软件的发展做出了非常重要的贡献。

但是，到目前为止，庞大的 GNU 项目仍没有完成其开发。到 1990 年，GNU 项目完成了大量的外围的具有高度可移植性的软件的开发，但是却没有完成 GNU 自己的操作系统内核的开发。1991 年，一名叫做 Linus Torvalds 的大学生，在一些网络黑客的协助下，开发了一款类 UNIX 操作系统内核——Linux (如图 1.2 所示)。1992 年，Linux 成为了自由软件的一部分，将 GNU 项目的所有软件和 Linux 内核结合起来构成了一个完整的操作系统——GNU/Linux。通过各种不同的发行版本，如 Ubuntu、Debian、Red Hat 等，全世界估计有数以千万的用户在使用 GNU/Linux 系统。



图 1.1 GNU 标志



图 1.2 Linux 图标是一个小企鹅

1.2 本书使用 Linux 环境介绍

当前官方提供最新的 Ubuntu 版本是 14.10，但 14.10 只支持更新 9 个月，而 14.04 支