

本文档遵循知识共享许可协议(Creative Commons license 4.0),以下简称CC 4.0, 使用者需遵循以下授权内容:

<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.zh>

您必须遵守下列条件:

1.署名: 使用本系列文档及文档内容, 您必须注明源作者信息(马哥教育张士杰老师, Email:2973707860@qq.com), 并提供本许可证的链接(<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.zh>),

同时需要声明您的内容是否经过修改, 你可以采用适当的方式将相关修改信息表述出来, 但是您不能在信息中以任何形式暗示授权人已正式认可了您以及您的使用行为。

2.非商业性使用: 您不能将本系列文档用于其他任何商业用途。

3.相同方式授权: 如您对本系列文档进行修改或使用本文档中内容, 那么您必须基于本协议分发您的作品(署名源作则马哥教育张士杰老师并使用本许可协议)。

您的权利:

当遵循本协议后, 您将有以下权限:

共享 – 允许以非商业性质复制本作品。

改编 – 在原基础上修改、复制或以本作品为基础进行重新编辑并用于个人非商业使用。

只要您遵守本许可协议条款, 许可人(马哥教育张士杰老师)将授权您使用本系列文档, 如果您违背本许可协议条款, 许可人(马哥教育张士杰老师)将有权收回您的授权并保留进一步追究其法律责任的权利。

<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.en>

Under the following terms:

Attribution – You must give appropriate credit(www.magedu.com, Zhang ShiJie, Email:2973707860@qq.com), provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

NonCommercial – You may not use the material for commercial purposes.

ShareAlike – If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

Ubuntu

讲师：张士杰（杰哥）

<http://www.magedu.com>

一：Debian与Ubuntu简介：

1.1：Debian简介：

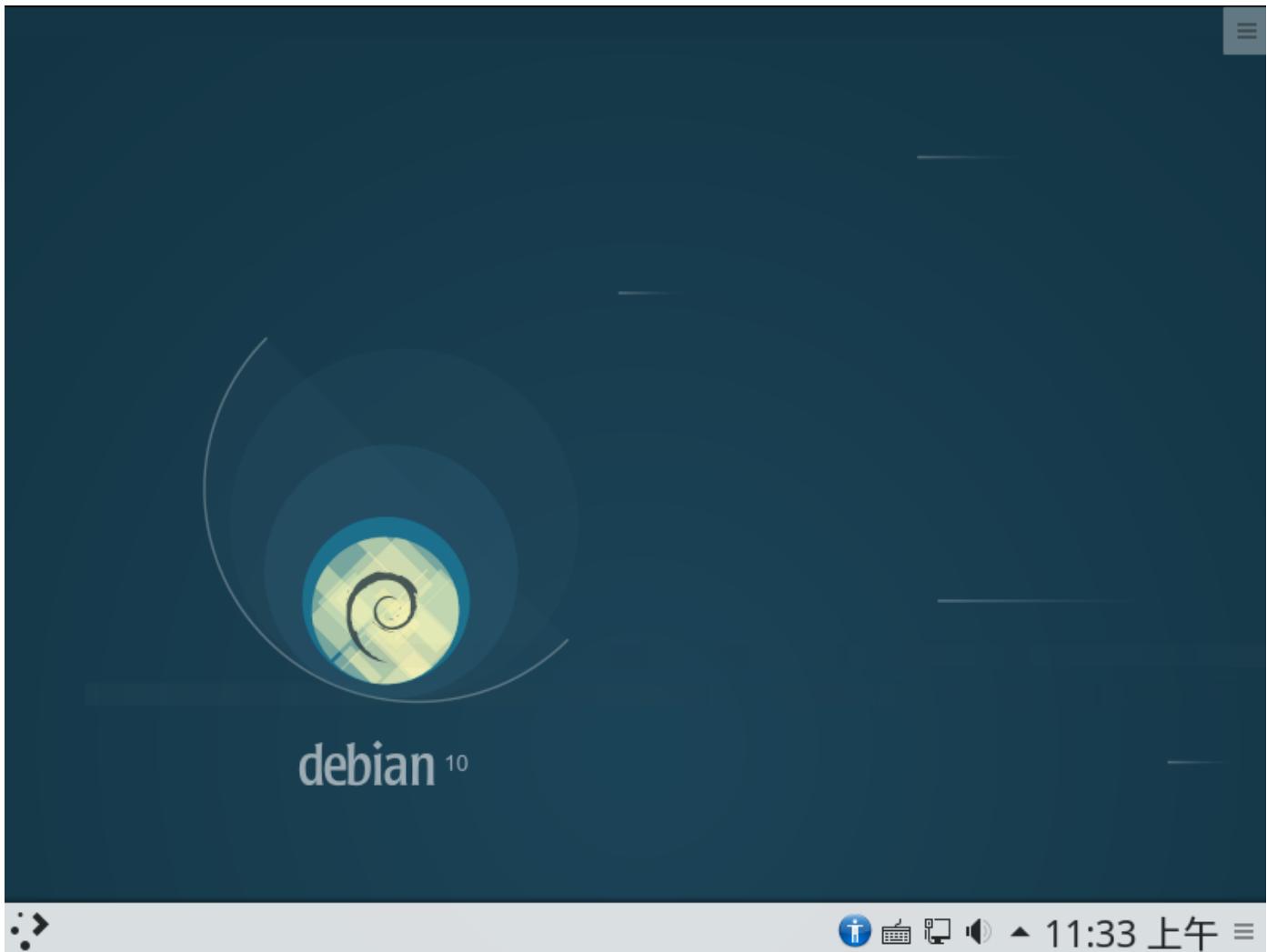
是从 1993 年由 Ian Murdock(伊恩·默多克) 发起的，受到当时 Linux 与 GNU 的鼓舞，目标是成为一个公开的发行版，经过二十几年的迭代更新Debian 从一个小型紧密的自由软件骇客（hacker）小组，逐渐成长成今日庞大且运作良好的开发者与用户社群，Debian 的名字是由 Debian 的创始人 Ian Murdock (伊恩·默多克) 和他的爱妻 Debra(黛布拉) 两人的名字组合而成的。

Debian 是由大约一千个分布在世界各地的开发者无偿地利用他们的业余时间开发的，而这些开发者实际上大部分都没见过面，彼此之间的通讯大多是通过电子邮件（lists.debian.org 上的邮件列表）和 IRC（irc.debian.org 上的 #debian 频道）来完成的，目前Debian 提供59000多个软件包的维护与更新。

Debian官网：<https://www.debian.org/>

官方镜像地址：<https://www.debian.org/mirror/list>

清华大学下载地址：<https://mirrors.tuna.tsinghua.edu.cn/debian-cd/10.1.0-live/amd64/iso-hybrid/>



1.2: Ubuntu简介：

Ubuntu（友帮拓、优般图、乌班图）早期是一个开源的GNU/Linux操作系统，Ubuntu是基于Debian GNU/Linux，支持x86、amd64（即x64）和ppc架构，由全球化的专业开发团队（Canonical Ltd）打造的，其名称来自非洲南部祖鲁语或豪萨语的“ubuntu”一词，类似儒家“仁爱”的思想，意思是“人性”、“我的存在是因为大家的存在”，是非洲传统的一种价值观，Ubuntu基于Debian发行版和GNOME桌面环境，而从11.04版起，Ubuntu发行版放弃了Gnome桌面环境，改为Unity，与Debian的不同在于它每6个月会发布一个新版本，Ubuntu的目标在于为一般用户提供一个最新的、同时又相当稳定的主要由自由软件构建而成的操作系统，Ubuntu具有庞大的社区力量，用户可以方便地从社区获得帮助，Ubuntu对GNU/Linux的普及特别是桌面普及作出了巨大贡献，由此使更多人共享开源的成果与精彩。

Canonical公司是一个私营公司，由南非的企业家马克·沙特尔沃思（Mark Shuttleworth）创建，主要为了促进开源软件项目，Canonical在马恩岛登记注册，其雇员分布在世界各地，其主要办事处在伦敦，在蒙特利尔也有办事处，Canonical公司支持并创建了几个项目，主要都是自由/开源软件（FOSS）或是一些旨在改善自由软件开发者和贡献者的工具。

<http://www.techweb.com.cn/business/2013-08-20/1318187.shtml> #ubuntu公司目前并未实现盈利

Ubuntu是各种公有云上最受欢迎的操作系统，这是因为Ubuntu与各大云运营商密切合作，针对云服务商的基础架构优化内核和系统，加快安全更新速度，并最大限度降低用户默认需要的网络或存储成本。

```
https://cn.ubuntu.com/ #中文官网  
https://ubuntu.com/download #英文官网
```

1.3: Ubuntu历史发行版本:

Ubuntu 没有像其它软件一样有 1.0 版本，是因为其第一个版本是发布于 2004 年，所以Ubuntu的生日是2004年的10月20日。

ubuntu的版本号有个固定的特征，其使用一个形容词和一个动物名称组成，并且形容词和名词的首字母都是一致的，Ubuntu版本的命名规则是根据正式版发布的年月命名，Ubuntu 16.04 也就意味着 2016年4月发布的Ubuntu，研发人员与用户可从版本号码就知道正式发布的时间。

每两年的 4 月份，都会推出一个长期支持版本 (LTS) ，其支持期长达五年，而非 LTS 版本的支持期通常只有半年。

```
https://baike.baidu.com/item/ubuntu #Ubuntu历史版本一览表
```

版本	别名	发布日期
4.10	Warty Warthog(长疣的疣猪)	2004年10月20日
5.04	Hoary Hedgehog(灰白的刺猬)	2005年4月8日
5.10	Breezy Badger(活泼的獾)	2005年10月13日
6.06(LTS)	Dapper Drake(整洁的公鸭)	2006年6月1日
6.10	Edgy Eft(急躁的水蜥)	2006年10月6日
7.04	Feisty Fawn(坏脾气的小鹿)	2007年4月19日
7.10	Gutsy Gibbon(勇敢的长臂猿)	2007年10月18日
8.04(LTS)	Hardy Heron(耐寒的苍鹭)	2008年4月24日
8.10	Intrepid Ibex (勇敢的野山羊)	2008年10月30日
9.04	Jaunty Jackalope(得意洋洋的怀俄明野兔)	2009年4月23日
9.10	Karmic Koala(幸运的考拉)	2009年10月29日
10.04(LTS)	Lucid Lynx(清醒的猞猁)	2010年4月29日
11.10	Oneiric Ocelot(梦幻的豹猫)	2010年10月13日
11.04	Natty Narwhal(敏捷的独角鲸)	2011年4月28日
12.04(LTS)	Precise Pangolin(精准的穿山甲)	2012年的4月26日
12.10	Quantal Quetzal(量子的绿咬鹃)	2012年的10月20日
13.04	Raring Ringtail(铆足了劲的猫熊)	2013年4月25日

13.10	Saucy Salamander(活泼的蝾螈)	2013年10月17日
14.04(LTS)	Trusty Tahr (可靠的塔尔羊)	2014年4月18日
	Utopic Unicorn(乌托邦独角兽)	2014年10月23日
15.04	Vivid Vervet (活泼的小猴)	2015年4月
15.10	Wily Werewolf (狡猾的狼人)	2015年10月
16.04(LTS)	Xenial Xerus (好客的地松鼠)	2016-04-21
16.10	Yakkety Yak(喋喋不休的牦牛)	2016-10-13
17.04	Zesty Zapus(热情的美洲林跳鼠)	2017-04-13
17.10	Artful Aardvark(巧妙的土豚)	2017-10-19
18.04(LTS)	Bionic Beaver(仿生的海狸)	2018-04-26
18.10	Cosmic Cuttlefish(宇宙的墨鱼)	2018-10-18
19.04	Disco Dingo(迪斯可的澳洲野犬)	2019-04-18
19.10	Eoan Ermine(短尾的黄鼠狼)	2019-10-17

1.4: Ubuntu使用场景：

Ubuntu系统在物联网、桌面系统、云环境、服务器系统与容器有广泛的使用场景，如下：

1.4.1: Ubuntu 桌面系统：

Ubuntu作为个人电脑既快速又安全、开源、免费，有数以千计的应用程序可供选择。从工作学习、到娱乐游戏，无所不能。

1.4.1.1: Ubuntu桌面系统简介：



1.4.1.2: 基于Ubuntu的优麒麟:

Canonical, 工业和信息化部软件与集成电路促进中心 (CSIP) 以及国防科学技术大学 (NUDT) 携手打造, 目前, 开发团队正在与开源社区合作打造Ubuntu的核心功能, 以更好地适应中国用户, 除了Ubuntu软件中心推出的本土化版本以外, 优麒麟还带最常用的中文程序。

优麒麟是由

Canonical, 工业和信息化部软件与集成电路促进中心 (CSIP) 以及国防科学技术大学 (NUDT) 携手打造。目前, 开发团队正在与开源社区合作打造Ubuntu的核心功能, 以更好地适应中国用户。除了Ubuntu软件中心推出的本土化版本以外, 优麒麟还带最常用的中文程序。

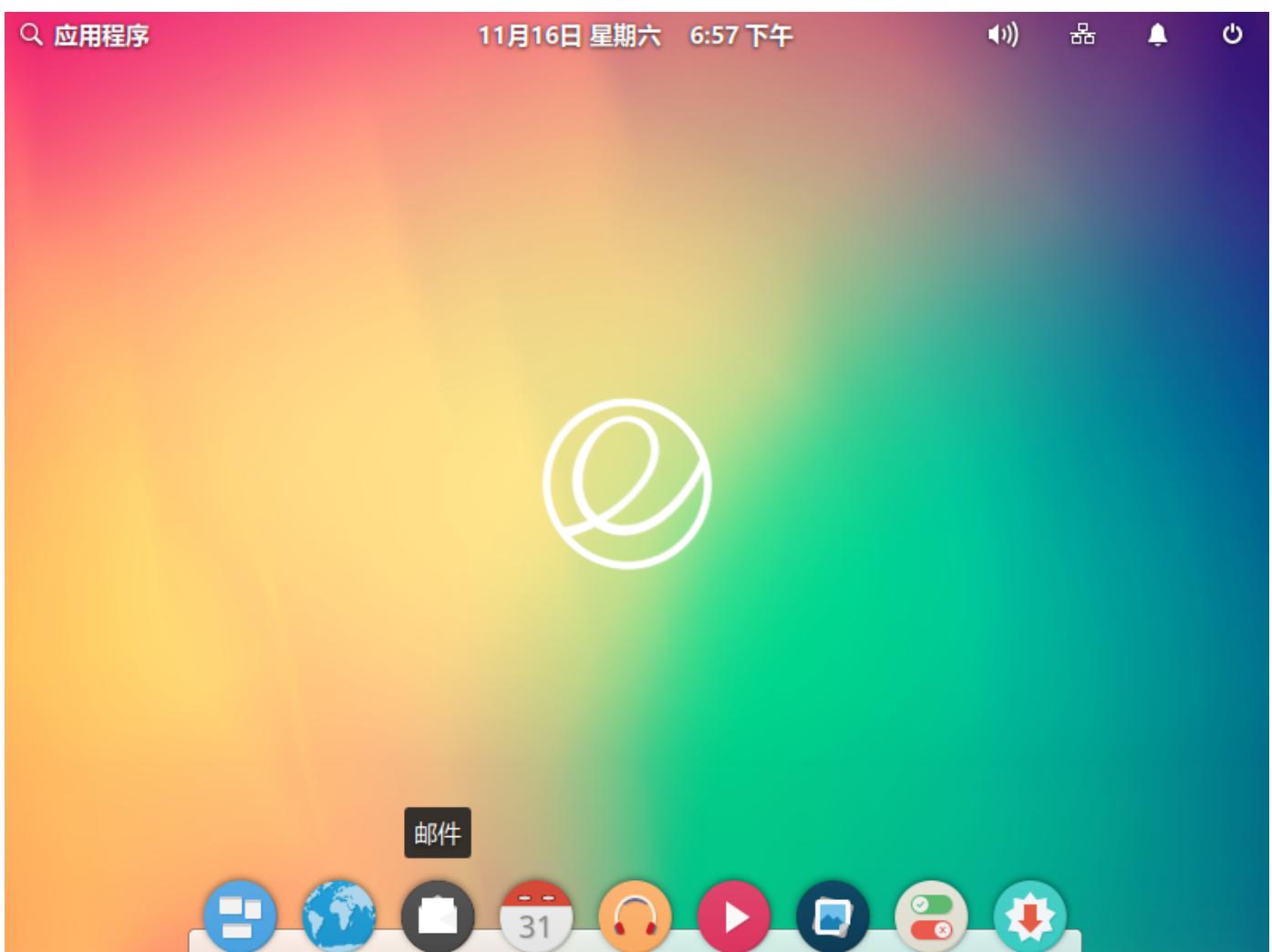
CANONICAL

CSIP

国防科学技术大学
NATIONAL UNIVERSITY OF DEFENSE TECHNOLOGY

1.4.1.3: 基于Ubuntu的Elementary OS:

Elementary OS系统是一款基于Ubuntu精心打磨美化的桌面Linux发行版, 号称“最漂亮的Linux系统”, 它最早是Ubuntu的一个美化主题项目, 后来太受欢迎演变成了独立的发行版, Elementary OS由设计师主导开发, 风格与macOS相似, 它借鉴了ChromeOS的菜单设计, 系统核心依然是Ubuntu并完全兼容所有Ubuntu软件包, elementary os从界面外观到系统设计都很精美, 视觉效果很好!



1.4.1.4：基于Ubuntu的其他分支：

Ubuntu版本之外，Ubuntu除了标准版之外还有几大主要分支，分别是Edubuntu、Kubuntu、Lubuntu、Mythbuntu、Ubuntu MATE、Ubuntu GNOME、Ubuntu Kylin、Ubuntu Studio和Xubuntu

Edubuntu是Ubuntu的教育发行版，专注于学校(教育)的需求，是由Ubuntu社区和K12-LTSP社区合作开发的，适合儿童、学生、教师使用的基础发行版，其内置了大量适合教学的应用软件和游戏。

Kubuntu是使用KDE桌面管理器取代GNOME桌面管理器作为其默认的桌面管理器的版本。Kubuntu的推出，为喜爱KDE桌面环境的使用者的安装和使用带来了很大的便利。

Lubuntu是一个后起之秀，以轻量级桌面环境LXDE替代Ubuntu默认的Unity。由于LXDE是一个轻量级桌面环境，所以Lubuntu所需的计算机资源很少，十分适合追求简洁或速度，以及还在使用老旧硬件的朋友选用。

Mythbuntu是一个用来实现媒体中心的Ubuntu发行版本，其核心组件是MythTV，所以Mythbuntu可以视为Ubuntu和MythTV的结合体。

Ubuntu GNOME是采用GNOME3作为Ubuntu默认桌面管理器的发行版本。由于Ubuntu的默认桌面环境是Unity，为了满足Linux用户的不同需求和使用习惯，Ubuntu GNOME项目应运而生。

Ubuntu Kylin(优麒麟)是一个专门为中文用户定制的Ubuntu版本，预置了大量中国用户熟悉的应用，是开箱即用的Ubuntu官方中国定制版本，适合中国用户使用。

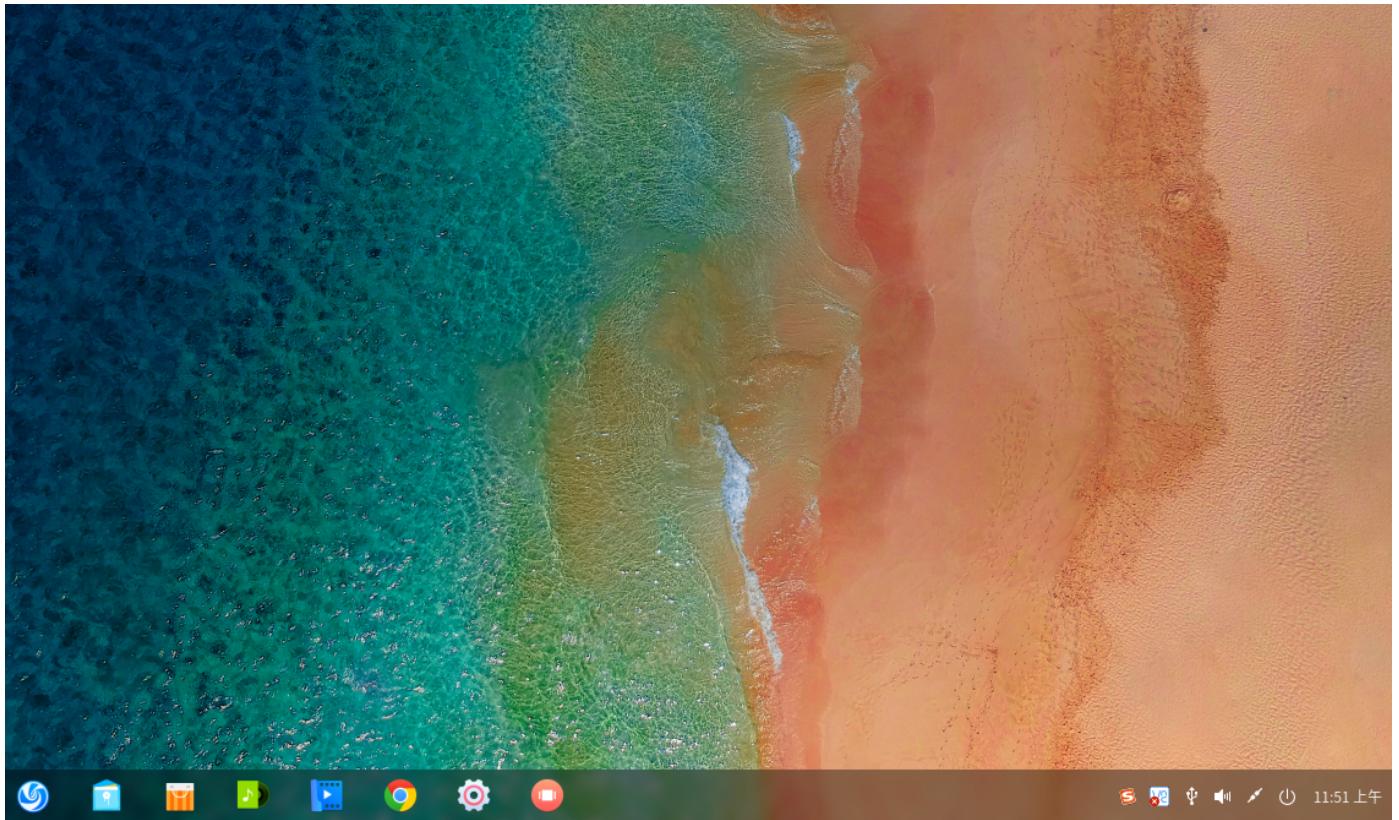
Ubuntu Studio则是一个为专业多媒体制作而打造的Ubuntu版本，可以编辑和处理音频、视频和图形图像等多媒体文件，对于多媒体专业人士而言，是一个鱼和熊掌兼得的好选择。

Xubuntu采用了小巧和高效的xfce作为桌面环境，界面简约，类似于GNOME2，功能全面，系统资源消耗较小，是追求速度和低配置计算机用户的福音，同时也为老旧计算机提供了发挥余热的机会

1.4.1.5：基于debain的deepin：

deepin：基于debain的发行版，Deepin原名Linux Deepin、deepin os、深度系统、深度操作系统，在2014年4月改名Deepin，由武汉深之度科技有限公司开发，被誉为“最美国产操作系统”，其以桌面应用为主的开源GNU/Linux操作系统，支持笔记本、台式机和一体机。

深度操作系统（deepin）是中国第一个具备国际影响力的Linux发行版本，截止至2019年7月25日，深度操作系统支持33种语言，用户遍布除了南极洲的其它六大洲，深度桌面环境（deepinDDE）和大量的应用软件被移植到了包括Fedora、Ubuntu、Arch等十余个国际Linux发行版和社区，在开源操作系统统计网站DistroWatch上，deepin长期位于世界前十。



1.4.1.6：其他Linux发行版：

<http://www.distrowatch.org/index.php?dataspan=2018>

Manjaro是一款基于Arch Linux对用户友好的Linux发行版，使用pacman管理安装包。

MX Linux是基于Debian稳定分支的面向桌面的Linux发行版本。

elementary OS是一个基于ubuntu的衍生版本，早期是一个ubuntu主题，后来衍生为一个专门的发行版。



1.4.2: Ubuntu与云环境:

<https://cn.ubuntu.com/cloud>

云计算的标准操作系统

70%的公有云工作负载和54%的OpenStack云*

[安装OpenStack](#)

如您已明确企业的需求，可直接[联系我们](#)获得技术支持以及认证服务。

1.4.2.1: Ubuntu与私有云:

ubuntu在私有云环境大量在openstack、容器与kubernetes环境使用。

openstack解决方案:

聊聊OpenStack

Ubuntu是目前生产环境使用的大多数OpenStack云的首选平台，并且有Canonical为全球众多领先公司提供咨询、培训、支持和完全托管的云服务。

[进一步了解Canonical在Ubuntu上提供的OpenStack服务](#)

完全托管的云服务

利用我们的BootStack服务对您的云进行全面的远程操作控制，打造生产用OpenStack私有云的捷径。

[进一步了解BootStack](#)

咨询服务方案

我们的Foundation Cloud Build服务，可在3个星期内为您构建一个固定价格的内部云，并提供经过验证的参考架构。

[进一步了解我们的咨询服务方案](#)

立即安装

利用conjure-up在单个笔记本电脑或多个机架服务器上运行OpenStack或Kubernetes。

[下载](#)

自动化物理服务器

利用MAAS裸机部署功能将数据中心变成物理云。内部部署、开源且可支持。

[进一步了解MAAS](#)

在认证的公有云上使用Ubuntu

Ubuntu是领先的云客户机操作系统，当今公有云的大部分工作负载均由它运行。

[开始使用Ubuntu服务器](#)

使用Juju进行设计和部署

数百种服务和预配置的应用程序，只需一个命令即可部署到任何云。

[开始使用Juju](#)

openstack客户：

我们的部分OPENSTACK客户



ubuntu与kubernetes:



Canonical的Kubernetes®发行版® *

这是在AWS、Google Cloud、Azure、Oracle或您的私有VMware、OpenStack和裸机设备上运行Kubernetes最简单的方法。

Canonical提供上游项目Kubernetes的一个发行版，它在每个基础架构上都能同样正常运行。您可以通过简单的自动升级来跟踪上游项目的更改，并且只需一个命令便可集成喜欢的监控系统。

[进一步了解在Ubuntu上使用Kubernetes](#)

1.4.2.2: Ubuntu与公有云:

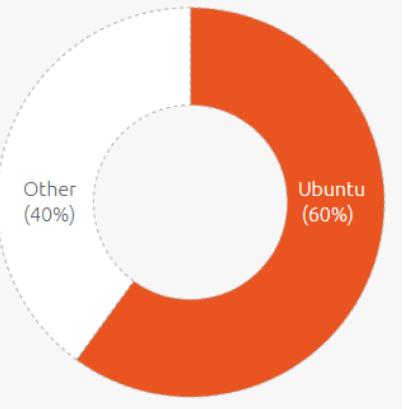
是各种公有云上最受欢迎的操作系统，这是因为我们与各大云运营商密切合作，针对他们的基础架构优化内核和系统，加快安全更新速度，并最大限度降低用户默认需要的网络或存储成本。

在公有云上使用Ubuntu

Ubuntu是各种公有云上最受欢迎的操作系统，这是因为我们与各大云运营商密切合作，针对他们的基础架构优化内核和系统，加快安全更新速度，并最大限度降低用户默认需要的网络或存储成本。

- ✓ Canonical保证Ubuntu在各种公有云上的一致性、安全性和性能
- ✓ 提供企业级商业支持
- ✓ 在Amazon云上的受欢迎程度是其他所有操作系统总和的两倍*

[在公有云上使用Ubuntu](#)



1.4.3: Ubuntu server系统:

Ubuntu Server 可让您的公共或私有数据中心在经济和技术上都具有出色的可扩展性。无论是部署 OpenStack 云、Hadoop 集群还是 50,000 个节点的大型渲染场，Ubuntu Server 都能提供性价比最佳的横向扩展能力。

1.4.3.1: Ubuntu Server 18.04 LTS 新增功能:

<https://wiki.ubuntu.com/BionicBeaver/ReleaseNotes/>

由 Canonical 提供到 2023 年的长期支持

可在所有主流架构上运行 - x86、x86-64、ARM v7、ARM64、POWER8 和 IBM s390x (LinuxONE)
占用空间更小、启动速度更快的全新 `ubuntu-minimal` 映像

通过 `chrony` 实现快速准确的时间同步 新的默认服务器安装程序 ISO 带来全新界面和更快的安装速度
支持 ZFS (非常适合服务器和容器的新一代卷管理/文件系统)

LXD 3.0 - 提供包括集群、QoS 和资源控制 (CPU、内存、块 I/O/ 图形处理器和存储配额) 等在内的多种 Linux 容器

更新了 LXD (v3.0)、DPDK (v17.11.1)、Postgresql (v10.3)、Libvirt (v4.0)、NGINX (v1.13)、
Qemu (v2.11.1)、Docker (v17.03)、Puppet (v4.10)、MySQL (v5.7)、PHP (v7.2) 等

安装 snap 来支持简单的应用程序安装和发布管理

Linux 4.15 内核

获得 AWS、Microsoft Azure、Joyent、IBM、Google Cloud Platform 和 Rackspace 客户机认证
通过 netplan.io 轻松配置网络连接

1.4.3.2: Ubuntu Server 19.10 新增功能:

https://wiki.ubuntu.com/EoanErmine/ReleaseNotes?_ga=2.36184999.620886555.1573809180-329374192.1565688951

由Canonical提供9个月的支持 Linux 5.3内核
更新为qemu (v4.0) , libvirt (v5.4) , mysql (v8.0) , postgresql (v11) 等
新的修复程序集和对Ubuntu Server安装程序的刷新
新的Ubuntu优势体验

1.4.4: Ubuntu与容器环境:

Ubuntu是容器的首选平台，从LXD到Kubernetes再到Docker， Canonical与行业领先的公司合作， 提供全方位的技术和服务， 帮助用户在公有云、私有云和混合云或裸机云上大规模运行容器。

[#docker的历史版本更新记录](https://github.com/moby/moby/blob/master/CHANGELOG.md)

1.4.4.1: 企业级Docker:

与Docker Inc.密切合作，在Ubuntu上提供商业支持的集成式Docker引擎。
Docker引擎的映像以snap包的形式发布，并会自动进行事务性更新。
Canonical为CS Docker引擎提供1级和2级技术支持，Docker Inc.则负责3级支持。

1.4.4.2: Canonical的Kubernetes发行版:

内置弹性扩展能力：

Canonical的Kubernetes简单易用，而且本身具有弹性。

规模化成本效益：

Canonical的Kubernetes使用Juju进行建模，并拥有管理持久性集群所需的所有操作脚本和工具，包括升级和弹性扩展。

随时可用的分析功能：

可轻松与Prometheus集成以进行监控，与Ceph集成以进行存储，以及与Elastic Stack(包括Kibana)集成以进行分析和可视化。

与行业领先公司合作



1.5: Ubuntu镜像下载:

Ubuntu这么好，哪里可以下载的到呢？

```
http://cdimage.ubuntu.com/releases/ #ubuntu server(服务器版)
http://releases.ubuntu.com/ #ubuntu desktop(桌面版)
http://cdimage.ubuntu.com/daily-live/current/ #20.04每天镜像
```

1.5.1：系统镜像版本区别：

<https://packages.ubuntu.com/search?lang=zh-cn&arch=any&keywords=libfuse-dev>

```
ubuntu-18.04.3-live-server-amd64.iso #带live, ISO镜像提供不安装就可以试用系统的功能
ubuntu-18.04.3-server-amd64.iso #不带live, 不可用试用, 但是可以直接进行系统安装
```

1.5.2：不同CPU指令集的ISO镜像：

CPU架构	安装包标识	备注	镜像名称示例
x86----->i386		#32位, server版已不再支持32位	ubuntu-16.04.6-desktop-i386.iso
x86-64----->amd64		#64位	ubuntu-18.04.3-server-amd64.iso
ARM v7----->ARM64		#arm平台	ubuntu-18.04.3-server-arm64.iso
IBM s390x----->s390x	#IBM System z		ubuntu-18.04.3-server-s390x.iso
POWER----->PPC64el	#PowerPC		ubuntu-18.04.3-server-ppc64el.iso
preinstalled-server-arm64+raspi3	#预安装的树莓派系列 arm64+raspi3.img.xz:		ubuntu-18.04.3-preinstalled-server-

二：Ubuntu Server版安装及使用：

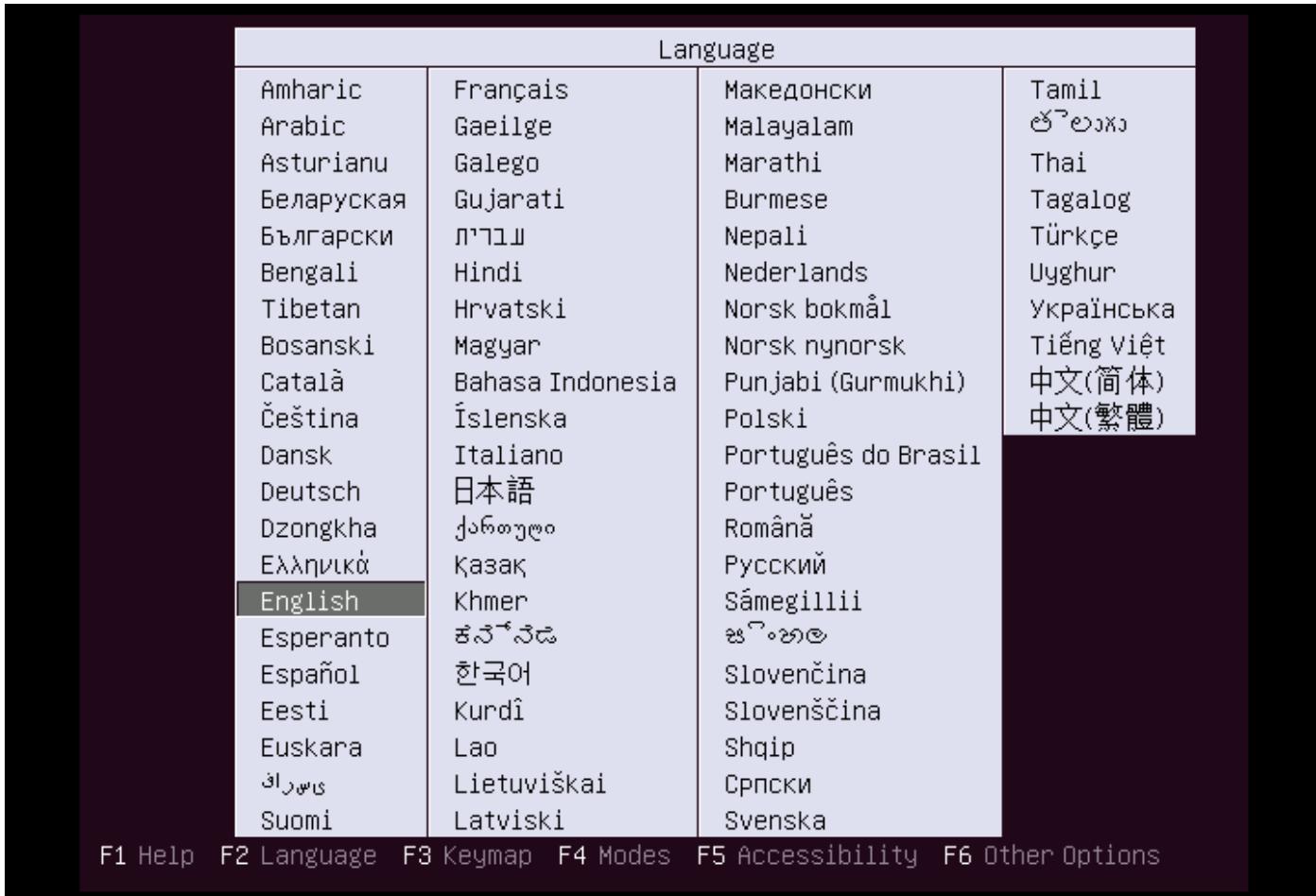
ubuntu 18.04服务器版本和桌面版的安装介绍

<https://help.ubuntu.com/> #官方文档

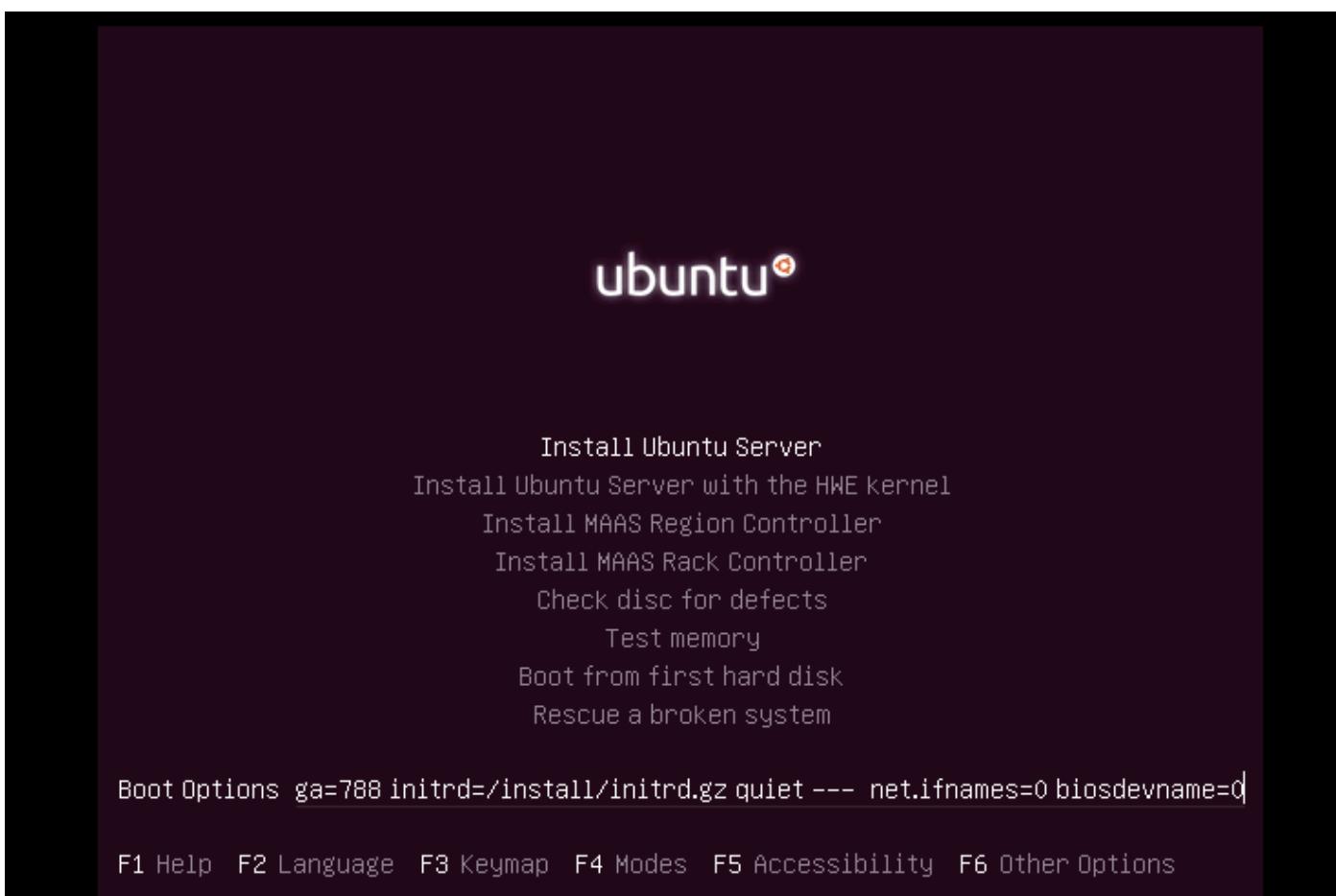
2.1：Ubuntu Server版本安装：

ubuntu 18.04服务器版本安装

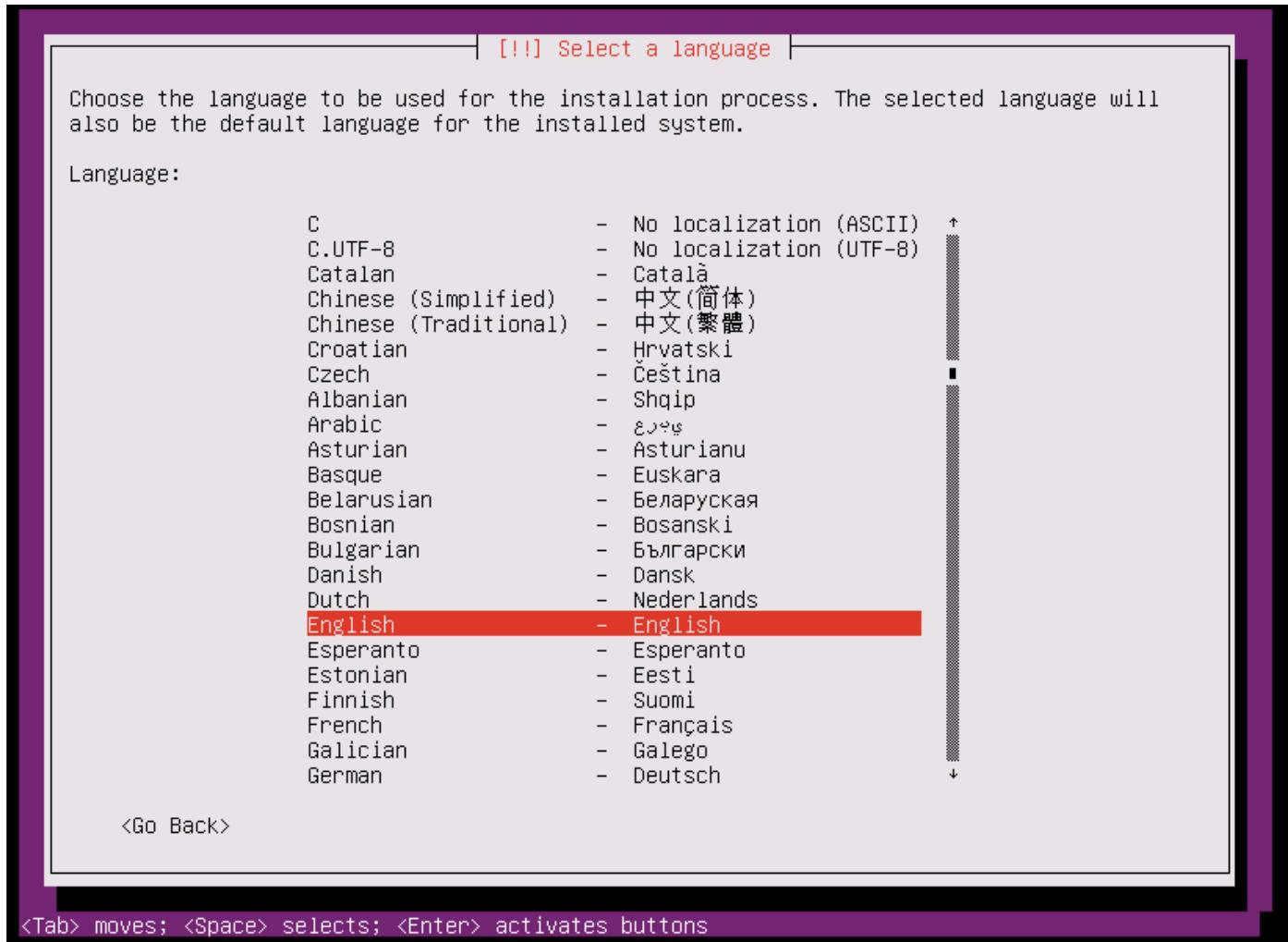
2.1.1：选择安装语言：



2.1.1：安装界面传递内核参数：



2.1.3: 选择安装过程语言:



2.1.4: 地区设置:

[!] Select your location

The selected location will be used to set your time zone and also for example to help select the system locale. Normally this should be the country where you live.

This is a shortlist of locations based on the language you selected. Choose "other" if your location is not listed.

Country, territory or area:

- Antigua and Barbuda
- Australia
- Botswana
- Canada
- Hong Kong
- India
- Ireland
- Israel
- New Zealand
- Nigeria
- Philippines
- Seychelles
- Singapore
- South Africa
- United Kingdom
- United States
- Zambia
- Zimbabwe
- other

<Go Back>

<Tab> moves; <Space> selects; <Enter> activates buttons

2.1.5: 键盘选择:

不进行探测，手动设置

[!] Configure the keyboard

You can try to have your keyboard layout detected by pressing a series of keys. If you do not want to do this, you will be able to select your keyboard layout from a list.

Detect keyboard layout?

<Go Back>

<Yes>

<No>

2.1.6: 选择键盘:

选美式键盘

[!] Configure the keyboard

The layout of keyboards varies per country, with some countries having multiple common layouts. Please select the country of origin for the keyboard of this computer.

Country of origin for the keyboard:

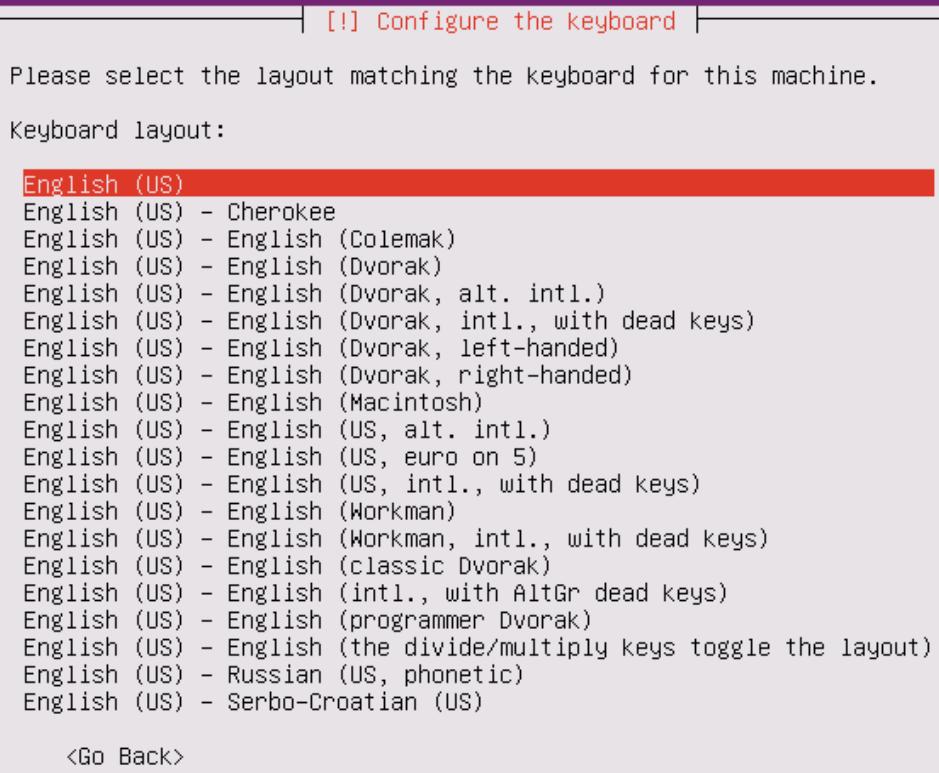
- Bambara
- Bangla
- Belarusian
- Belgian
- Berber (Algeria, Latin)
- Bosnian
- Braille
- Bulgarian
- Burmese
- Chinese
- Croatian
- Czech
- Danish
- Dhivehi
- Dutch
- Dzongkha
- English (Australian)
- English (Cameroon)
- English (Ghana)
- English (Nigeria)
- English (South Africa)
- English (UK)
- English (US)

<Go Back>



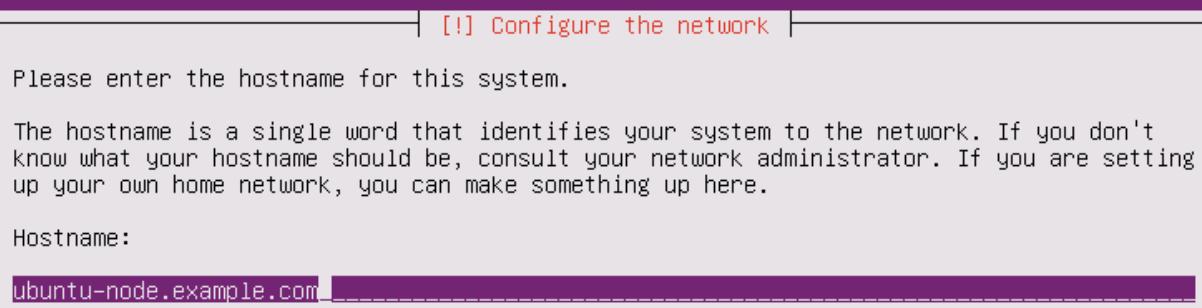
<Tab> moves; <Space> selects; <Enter> activates buttons

2.1.7: 键盘确认:



<Tab> moves; <Space> selects; <Enter> activates buttons

2.1.8：配置主机名：



2.1.9：创建新的普通系统用户：

[!!] Set up users and passwords

A user account will be created for you to use instead of the root account for non-administrative activities.

Please enter the real name of this user. This information will be used for instance as default origin for emails sent by this user as well as any program which displays or uses the user's real name. Your full name is a reasonable choice.

Full name for the new user:

magedu

<Go Back> <Continue>

2.1.10: 确认新账户名称:

[!!] Set up users and passwords

Select a username for the new account. Your first name is a reasonable choice. The username should start with a lower-case letter, which can be followed by any combination of numbers and more lower-case letters.

Username for your account:

magedu

<Go Back> <Continue>

2.1.11: 设置用户密码:

[!!] Set up users and passwords

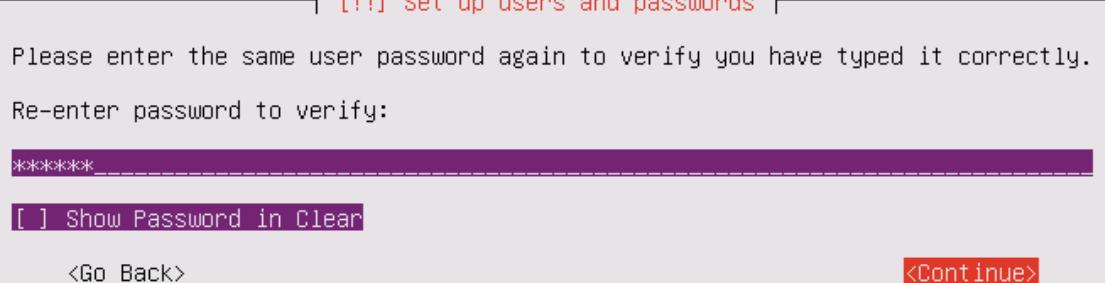
A good password will contain a mixture of letters, numbers and punctuation and should be changed at regular intervals.

Choose a password for the new user:

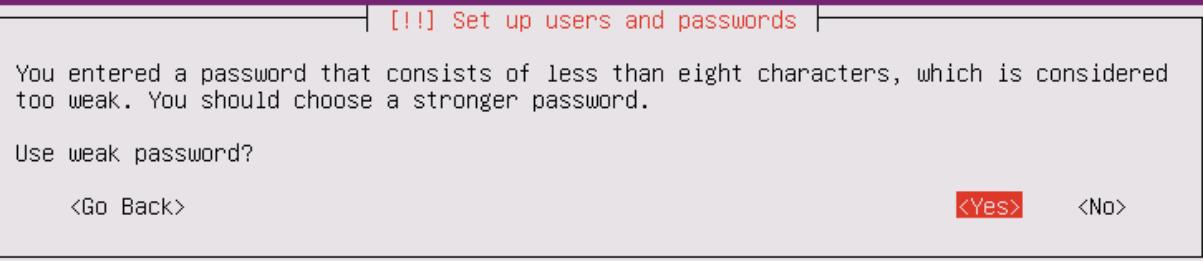
[] Show Password in Clear

<Go Back> <Continue>

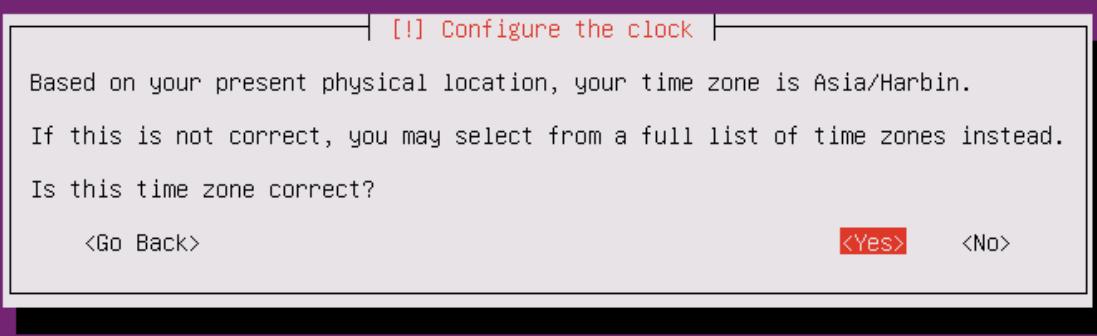
2.1.12: 确认账户密码:



2.1.13: 确认使用密码:



2.1.14: 确认时区:



2.1.15: 手动分区:

[!!] Partition disks

The installer can guide you through partitioning a disk (using different standard schemes) or, if you prefer, you can do it manually. With guided partitioning you will still have a chance later to review and customise the results.

If you choose guided partitioning for an entire disk, you will next be asked which disk should be used.

Partitioning method:

- Guided – use entire disk
- Guided – use entire disk and set up LVM
- Guided – use entire disk and set up encrypted LVM
- Manual

<Go Back>

2.1.16: 选择磁盘：

[!!] Partition disks

This is an overview of your currently configured partitions and mount points. Select a partition to modify its settings (file system, mount point, etc.), a free space to create partitions, or a device to initialize its partition table.

Guided partitioning
Configure iSCSI volumes

SCSI33 (0,0,0) (sda) - 107.4 GB VMware, VMware Virtual S

Undo changes to partitions
Finish partitioning and write changes to disk

<Go Back>

2.1.17: 确认使用磁盘：

[!!] Partition disks

You have selected an entire device to partition. If you proceed with creating a new partition table on the device, then all current partitions will be removed.

Note that you will be able to undo this operation later if you wish.

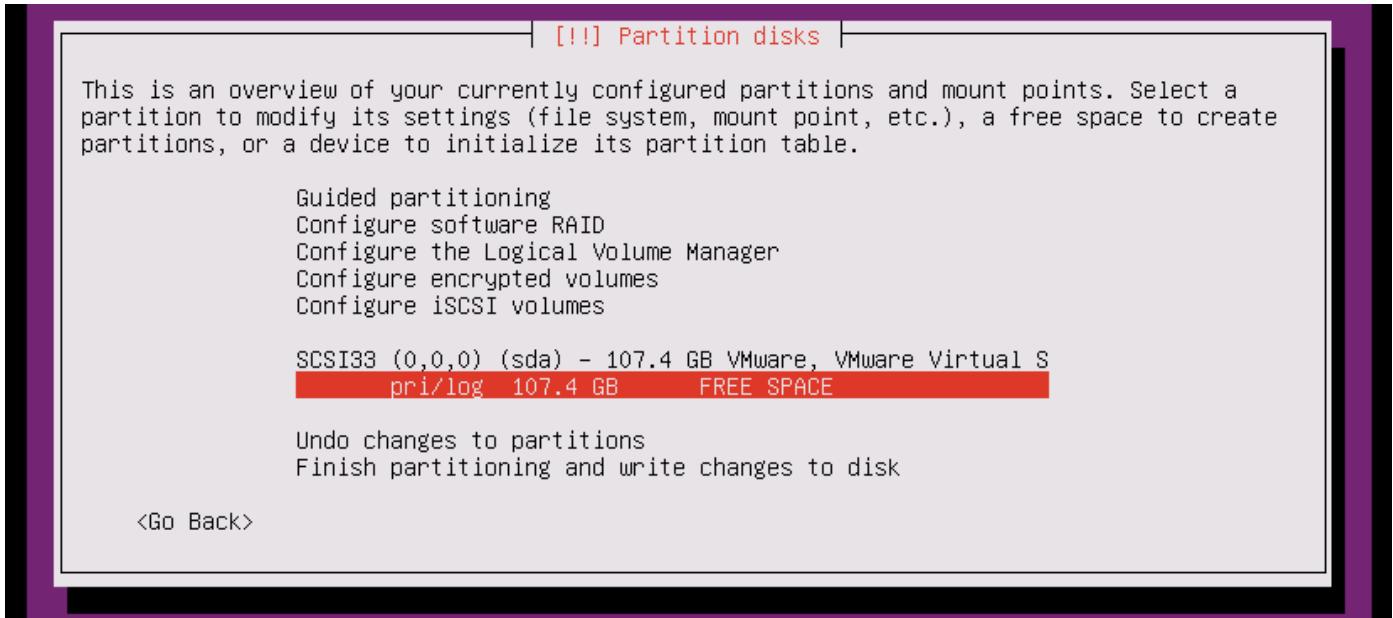
Create new empty partition table on this device?

<Go Back>

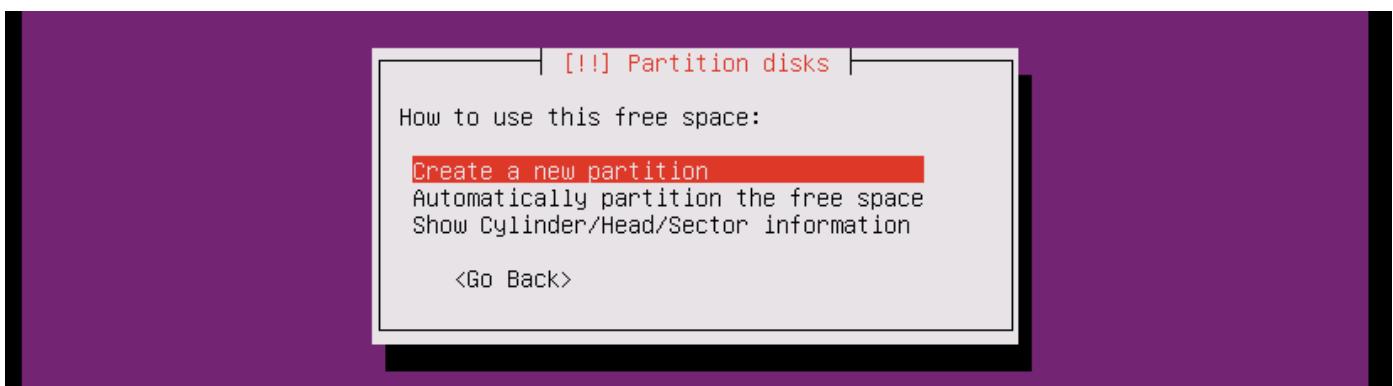
<Yes>

<No>

2.1.18: 对磁盘分区：



2.1.19: 创建新分区：



2.1.20: 指定分区大小：



2.1.21: 选择分区类型：

分区类型为主分区



2.1.22: 分区完成:

文件系统类型可以选择ext4/xfs/等

```
You are editing partition #1 of SCSI33 (0,0,0) (sda). No existing file system was detected in this partition.
```

Partition settings:

Use as:	Ext4 journaling file system
Mount point:	/
Mount options:	defaults
Label:	none
Reserved blocks:	5%
Typical usage:	standard
Bootable flag:	off

Delete the partition
Done setting up the partition

<Go Back>

2.1.23: 完成分区并写入数据:

```
This is an overview of your currently configured partitions and mount points. Select a partition to modify its settings (file system, mount point, etc.), a free space to create partitions, or a device to initialize its partition table.
```

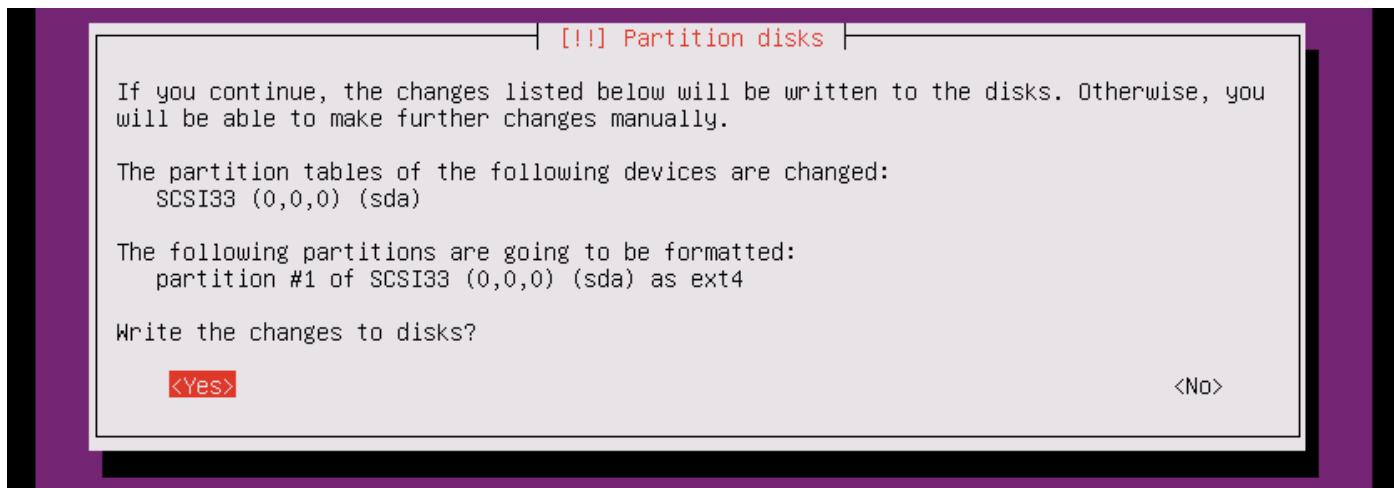
Guided partitioning
Configure software RAID
Configure the Logical Volume Manager
Configure encrypted volumes
Configure iSCSI volumes

```
SCSI33 (0,0,0) (sda) - 107.4 GB VMware, VMware Virtual S
#1 primary 107.4 GB f ext4 /
```

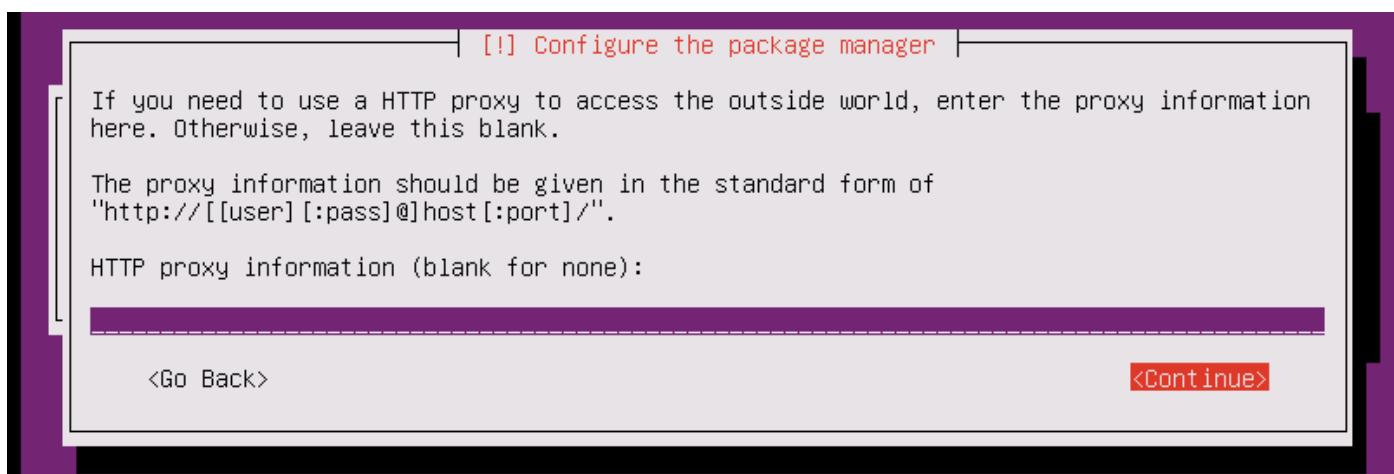
Undo changes to partitions
Finish partitioning and write changes to disk

<Go Back>

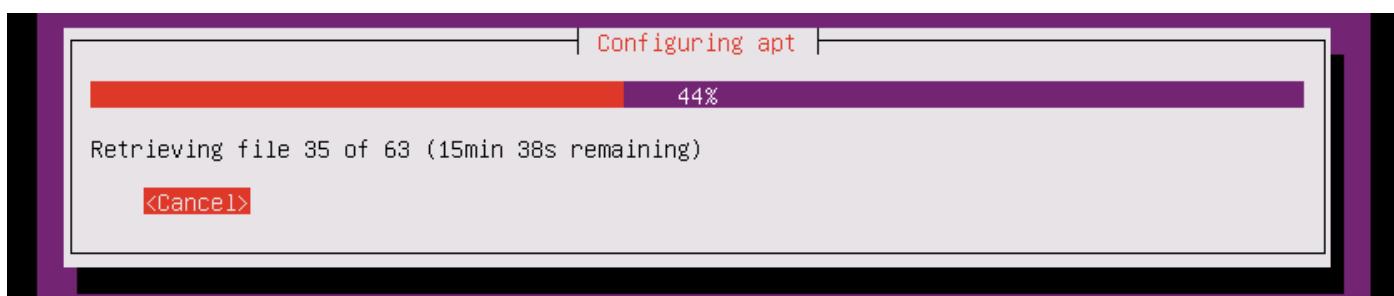
2.1.24: 确认写入磁盘:



2.1.15: 是否使用代理:



2.1.26: 系统安装过程中:



2.1.27: 是否自动更新:

```
[!] Configuring tasksel

Applying updates on a frequent basis is an important part of keeping your system secure.

By default, updates need to be applied manually using package management tools.
Alternatively, you can choose to have this system automatically download and install
security updates, or you can choose to manage this system over the web as part of a group
of systems using Canonical's Landscape service.

How do you want to manage upgrades on this system?

No automatic updates
Install security updates automatically
Manage system with Landscape
```

2.1.28: 选择安装组件:

```
[!] Software selection

At the moment, only the core of the system is installed. To tune the system to your
needs, you can choose to install one or more of the following predefined collections of
software.

Choose software to install:

[ ] DNS server
[ ] LAMP server
[ ] Mail server
[ ] PostgreSQL database
[ ] Print server
[ ] Samba file server
[*] OpenSSH server

<Continue>
```

2.1.29: 写入grub引导:

```
[!] Install the GRUB boot loader on a hard disk

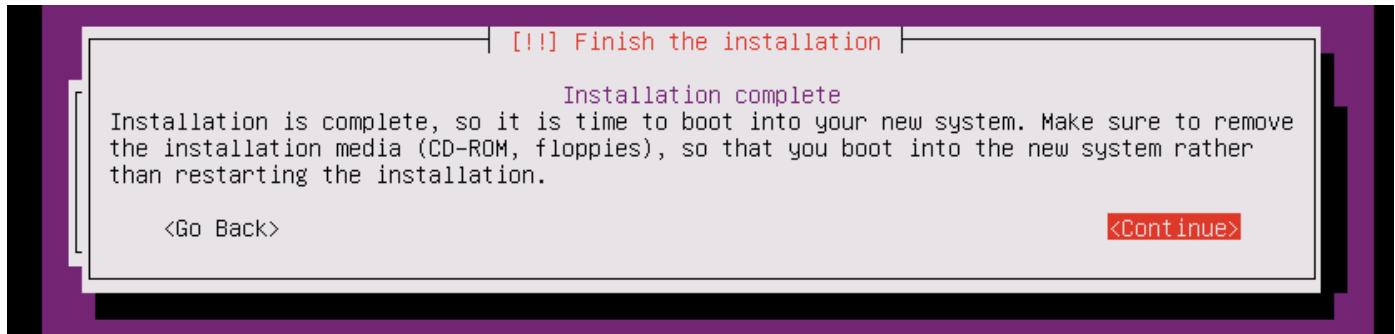
It seems that this new installation is the only operating system on this computer. If so,
it should be safe to install the GRUB boot loader to the master boot record of your first
hard drive.

Warning: If the installer failed to detect another operating system that is present on
your computer, modifying the master boot record will make that operating system
temporarily unbootable, though GRUB can be manually configured later to boot it.

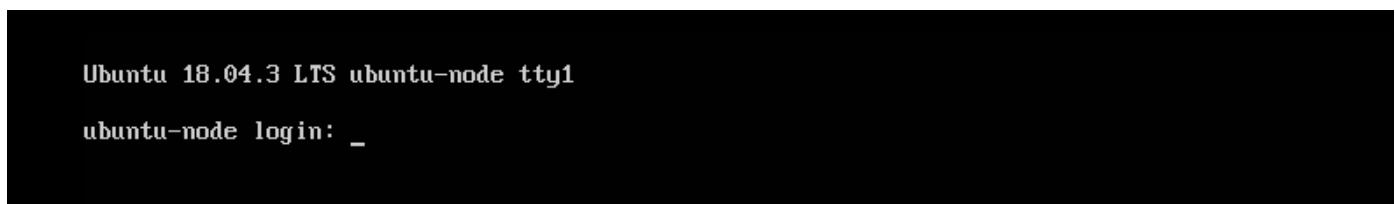
Install the GRUB boot loader to the master boot record?

<Go Back> <Yes> <No>
```

2.1.30：确认重启服务器：



2.1.31：系统登录界面：



2.2：Ubuntu Server系统基础配置：

官方使用文档：

<https://help.ubuntu.com>

2.2.1：更改主机名：

```
~$ cat /etc/hostname  
ubuntu-node.example.com
```

2.2.2：更改网卡名称为eth*：

如果没有在安装系统之前传递内核参数将网卡名称更改为eth*，则可以在安装系统之后使用以下方式修改：

```
~$ sudo vim /etc/default/grub  
GRUB_DEFAULT=0  
GRUB_TIMEOUT_STYLE=hidden  
GRUB_TIMEOUT=2  
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`  
GRUB_CMDLINE_LINUX_DEFAULT=""  
GRUB_CMDLINE_LINUX="net.ifnames=0 biosdevname=0"  
  
~$ sudo update-grub  
Sourcing file `/etc/default/grub'  
Generating grub configuration file ...  
Found linux image: /boot/vmlinuz-4.15.0-55-generic  
Found initrd image: /boot/initrd.img-4.15.0-55-generic  
done
```

```
~$ sudo reboot
```

2.2.3：配置root远程登录：

默认情况下，ubuntu不允许root用户远程ssh，如果有实际场景需要允许root用户远程ssh，则需要设置root密码，并且编辑/etc/ssh/sshd_config文件修改如下：

```
~$ sudo vim /etc/ssh/sshd_config
32 #PermitRootLogin prohibit-password #默认为禁止登录
33 PermitRootLogin yes #改为允许登录

57 #PasswordAuthentication yes
58 PasswordAuthentication yes #打开密码认证，其实默认就是允许通过密码认证登录

~$ sudo su - root #切换到root用户环境
~# passwd #设置密码
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully

~# systemctl restart sshd #重启ssh服务并测试root用户远程ssh连接
```

2.2.4：Ubuntu 18.04网络配置：

<https://netplan.io/> #官方文档

Ubuntu 从 17.10 开始，已放弃在 /etc/network/interfaces 里固定IP的配置，而是改成 netplan 方式，配置文件是：/etc/netplan/01-netcfg.yaml

ubuntu 17.04及之前的静态IP配置方式：

```
~# cat /etc/network/interfaces
root@magedu:~# cat /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

auto eth0 #网卡自启动，写自己要配置IP的实际网卡名称
iface eth0 inet static #配置静态IP，写自己要配置IP的实际网卡名称
address 172.18.3.12 #IP地址
netmask 255.255.0.0 #掩码
gateway 172.18.0.1 #网关
dns-nameservers 223.6.6.6 #DNS
dns-nameservers 223.5.5.5

#重启网络服务
~# /etc/init.d/networking restart
~# systemctl restart networking.service
```

2.2.4.1：单网卡静态IP地址：

ubuntu 18.04配置静态IP地址

```
~#sudo vim /etc/netplan/01-netcfg.yaml
# This file describes the network interfaces available on your system
# For more information, see netplan(5).

network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: no
      dhcp6: no
      addresses: [172.18.3.18/16]
      gateway4: 172.18.0.1
      nameservers:
        addresses: [223.6.6.6]

# sudo netplan apply

# vim ~/.vimrc #取消vim 自动缩进功能
set paste
```

2.2.4.2：配置多网卡静态IP：

ubuntu 18.04配置多网卡静态IP以及静态路由

```
~# sudo vim /etc/netplan/01-netcfg.yaml
# This file describes the network interfaces available on your system
# For more information, see netplan(5).

network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: no
      dhcp6: no
      addresses: [172.18.3.18/16]
      gateway4: 172.18.0.1
      nameservers:
        addresses: [223.6.6.6]
    eth1:
      dhcp4: no
      dhcp6: no
      addresses: [10.20.3.18/16]
      routes:
        - to: 172.20.0.0/16
```

```
    via: 10.20.0.1
  - to: 10.20.0.0/16
    via: 10.20.0.1
  - to: 10.2.0.0/16
    via: 10.20.0.1
  - to: 10.8.0.0/16
    via: 10.20.0.1

~# sudo netplan apply
```

2.2.4.3: 单网卡桥接:

ubuntu 18.04的单网卡桥接配置

```
root@s1:~# cat /etc/netplan/01-netcfg.yaml
```

```
~# sudo cat /etc/netplan/01-netcfg.yaml
# This file describes the network interfaces available on your system
# For more information, see netplan(5).

network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: no
      dhcp6: no

  bridges:
    br0:
      dhcp4: no
      dhcp6: no
      addresses: [172.18.3.18/16]
      gateway4: 172.18.0.1
      nameservers:
        addresses: [223.6.6.6]
      interfaces:
        - eth0

~# sudo netplan apply
```

2.2.4.4: 多网卡桥接:

ubuntu 18.04多网卡的桥接配置，将br0和br1分别桥接到eth0和eth1。

```
~# sudo cat /etc/netplan/01-netcfg.yaml
# This file describes the network interfaces available on your system
# For more information, see netplan(5).

network:
  version: 2
```

```

renderer: networkd
ethernets:
    eth0:
        dhcp4: no
        dhcp6: no
    eth1:
        dhcp4: no
        dhcp6: no

bridges:
    br0:
        dhcp4: no
        dhcp6: no
        addresses: [172.18.3.18/16]
        gateway4: 172.18.0.1
        nameservers:
            addresses: [223.6.6.6]
    interfaces:
        - eth0

    br1:
        dhcp4: no
        dhcp6: no
        addresses: [10.20.3.18/16]
        routes:
            - to: 172.20.0.0/16
              via: 10.20.0.1
            - to: 10.20.0.0/16
              via: 10.20.0.1
            - to: 10.2.0.0/16
              via: 10.20.0.1
            - to: 10.8.0.0/16
              via: 10.20.0.1
        interfaces:
            - eth1

```

2.2.4.4: 双网卡绑定:

七种bond模式说明:

第一种模式: `mod=0`, 即: `(balance-rr)` Round-robin policy (平衡轮循策略)

特点: 传输数据包顺序是依次传输 (即: 第1个包走eth0, 下一个包就走eth1....一直循环下去, 直到最后一个传输完毕), 此模式提供负载平衡和容错能力。

第二种模式: `mod=1`, 即: `(active-backup)` Active-backup policy (主-备份策略)

特点: 只有一个设备处于活动状态, 当一个宕掉另一个马上由备份转换为主设备。mac地址是外部可见得, 从外面看来, bond的MAC地址是唯一的, 以避免switch(交换机)发生混乱。此模式只提供了容错能力; 由此可见此算法的优点是可以提供高网络连接的可用性, 但是它的资源利用率较低, 只有一个接口处于工作状态, 在有 N 个网络接口的情况下, 资源利用率为 $1/N$ 。

第三种模式：mod=2，即：(balance-xor) XOR policy (平衡策略)

特点：基于指定的传输HASH策略传输数据包。缺省的策略是：(源MAC地址 XOR 目标MAC地址) % slave数量。其他的传输策略可以通过xmit_hash_policy选项指定，此模式提供负载平衡和容错能力。

第四种模式：mod=3，即：broadcast (广播策略)

特点：在每个slave接口上传输每个数据包，此模式提供了容错能力。

第五种模式：mod=4，即：(802.3ad) IEEE 802.3adDynamic link aggregation (IEEE 802.3ad 动态链接聚合)

特点：创建一个聚合组，它们共享同样的速率和双工设定。根据802.3ad规范将多个slave工作在同一个激活的聚合体下。

必要条件：

条件1：ethtool支持获取每个slave的速率和双工设定。

条件2：switch(交换机)支持IEEE 802.3ad Dynamic link aggregation。

条件3：大多数switch(交换机)需要经过特定配置才能支持802.3ad模式。

第六种模式：mod=5，即：(balance-tlb) Adaptive transmit load balancing (适配器传输负载均衡)

特点：不需要任何特别的switch(交换机)支持的通道bonding。在每个slave上根据当前的负载（根据速度计算）分配外出流量。如果正在接受数据的slave出故障了，另一个slave接管失败的slave的MAC地址。

该模式的必要条件：

ethtool支持获取每个slave的速率

第七种模式：mod=6，即：(balance-alb) Adaptive load balancing (适配器适应性负载均衡)

特点：该模式包含了balance-tlb模式，同时加上针对IPV4流量的接收负载均衡(receive load balance, rlb)，而且不需要任何switch(交换机)的支持。

ubuntu 18.04的双网卡绑定配置。

需要提前安装好bridge命令

```
# sudo vim /etc/netplan/01-netcfg.yaml
# This file describes the network interfaces available on your system
# For more information, see netplan(5).

network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: no
      dhcp6: no
    eth1:
      dhcp4: no
      dhcp6: no

  bonds:
    bond0:
      interfaces:
        - eth0
        - eth1
      addresses: [172.18.3.18/16]
```

```

gateway4: 172.18.0.1
nameservers:
  addresses: [223.6.6.6,223.5.5.5]
parameters:
  mode: active-backup
  mii-monitor-interval: 100

~# sudo netplan apply

```

2.2.4.5：双网卡绑定+桥接：

网卡绑定用于提供网卡接口冗余以及高可用和端口聚合功能，桥接网卡再给需要桥接设备的服务使用。

```

~# sudo cat /etc/netplan/01-netcfg.yaml
# This file describes the network interfaces available on your system
# For more information, see netplan(5).

network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: no
      dhcp6: no
    eth1:
      dhcp4: no
      dhcp6: no

  bonds:
    bond0:
      interfaces:
        - eth0
        - eth1
      #addresses: [172.18.3.18/16]
      #gateway4: 172.18.0.1
      #nameservers:
      #  addresses: [223.6.6.6,223.5.5.5]
      parameters:
        mode: active-backup
        mii-monitor-interval: 100

  bridges:
    br0:
      dhcp4: no
      dhcp6: no
      addresses: [172.18.3.18/16]
      gateway4: 172.18.0.1
      nameservers:
        addresses: [223.6.6.6,223.5.5.5]
      interfaces:

```

```
- bond0
```

```
~#reboot
```

2.2.4.6：内外多网卡绑定：

多网络情况下实现网卡绑定。

```
~# sudo cat /etc/netplan/01-netcfg.yaml
# This file describes the network interfaces available on your system
# For more information, see netplan(5).

network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: no
      dhcp6: no
    eth1:
      dhcp4: no
      dhcp6: no
    eth2:
      dhcp4: no
      dhcp6: no
    eth3:
      dhcp4: no
      dhcp6: no

  bonds:
    bond0:
      interfaces:
        - eth0
        - eth1
      addresses: [172.18.3.18/16]
      gateway4: 172.18.0.1
      nameservers:
        addresses: [223.6.6.6,223.5.5.5]
      parameters:
        mode: active-backup
        mii-monitor-interval: 100

    bond1:
      interfaces:
        - eth2
        - eth3
      addresses: [10.20.3.18/16]
      parameters:
        mode: active-backup
        mii-monitor-interval: 100
```

```
routes:
  - to: 172.20.0.0/16
    via: 10.20.0.1
  - to: 10.20.0.0/16
    via: 10.20.0.1
  - to: 10.2.0.0/16
    via: 10.20.0.1
  - to: 10.8.0.0/16
    via: 10.20.0.1
```

2.2.4.7：内网多网卡绑定+桥接：

```
~# sudo cat /etc/netplan/01-netcfg.yaml
# This file describes the network interfaces available on your system
# For more information, see netplan(5).

network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: no
      dhcp6: no
    eth1:
      dhcp4: no
      dhcp6: no
    eth2:
      dhcp4: no
      dhcp6: no
    eth3:
      dhcp4: no
      dhcp6: no

  bonds:
    bond0:
      interfaces:
        - eth0
        - eth1
      parameters:
        mode: active-backup
        miimon-interval: 100

    bond1:
      interfaces:
        - eth2
        - eth3
      parameters:
        mode: active-backup
        miimon-interval: 100
```

```
bridges:
  br0:
    dhcp4: no
    dhcp6: no
    addresses: [172.18.3.18/16]
    gateway4: 172.18.0.1
    nameservers:
      addresses: [223.6.6.6,223.5.5.5]
  interfaces:
    - bond0

br1:
  dhcp4: no
  dhcp6: no
  interfaces:
    - bond1
  addresses: [10.20.3.18/16]
  routes:
    - to: 172.20.0.0/16
      via: 10.20.0.1
    - to: 10.20.0.0/16
      via: 10.20.0.1
    - to: 10.2.0.0/16
      via: 10.20.0.1
    - to: 10.8.0.0/16
      via: 10.20.0.1
```

2.3：Ubuntu软件包管理：

在ubuntu安装、升级、卸载软件包等常规操作。

2.3.1：修改软件仓库：

阿里云仓库地址：<https://opsx.alibaba.com/mirror>

中科大：<http://mirrors.ustc.edu.cn/help/ubuntu.html>

清华大学：<https://mirror.tuna.tsinghua.edu.cn/help/ubuntu/>

华为：<https://mirrors.huaweicloud.com/>

2.3.2：apt/apt-get：

```
# apt list #apt列出仓库软件包, 等于yum list
# apt search NAME #搜索安装包
# apt show apache2 #查看某个安装包的详细信息
# apt install apache2 #在线安装软件包
# apt remove apache2 #卸载单个软件包但是保留配置文件
```

```
# apt autoremove apache2 #删除安装包并解决依赖关系
# apt update #更新本地软件包列表索引，修改了apt仓库后必须执行
# apt purge apache2 #卸载单个软件包删除配置文件
# apt upgrade #升级所有已安装且可升级到新版本的软件包
# apt full-upgrade #升级整个系统，必要时可以移除旧软件包。
# apt edit-sources #编辑source源文件

# apt-cache madison nginx #查看仓库中软件包有哪些版本可以安装
# apt install nginx=1.14.0-0ubuntu1.6 #安装软件包的时候指定安装具体的版本
```

2.3.3：设置oracle JDK环境：

```
# pwd
/usr/local/src
```

解压二进制文件并设置软连接：

```
# tar xf jdk-8u212-linux-x64.tar.gz
# ln -sv /usr/local/src/jdk1.8.0_212 /usr/local/jdk
```

配置环境变量：

```
# vim /etc/profile
export JAVA_HOME=/usr/local/jdk
export PATH=$JAVA_HOME/bin:$JAVA_HOME/jre/bin:$PATH
export CLASSPATH=.:$CLASSPATH:$JAVA_HOME/lib:$JAVA_HOME/jre/lib:$JAVA_HOME/lib/tools.jar
```

重新导入环境变量并验证：

```
# source /etc/profile
# java -version
java version "1.8.0_212"
Java(TM) SE Runtime Environment (build 1.8.0_212-b10)
Java HotSpot(TM) 64-Bit Server VM (build 25.212-b10, mixed mode)
```

2.3.4：安装OpenJDK：

```
# apt install openjdk-8-jdk
```

2.3.5：安装常用系统命令：

```
# apt purge ufw lxd lxd-client lxcfs lxc-common
# apt install iproute2 ntpdate tcpdump telnet traceroute nfs-kernel-server nfs-common lrzsz tree openssl libssl-dev libpcap-dev libpcap3-dev zlib1g-dev ntpdate tcpdump telnet traceroute gcc openssh-server lrzsz tree openssl libssl-dev libpcap3-dev zlib1g-dev ntpdate tcpdump telnet traceroute iotop unzip zip
```

2.3.6：系统资源限制优化：

```
#cat /etc/security/limits.conf
#root账户的资源软限制和硬限制
root          soft  core      unlimited
root          hard  core      unlimited
root          soft  nproc     1000000
root          hard  nproc     1000000
root          soft  nofile   1000000
root          hard  nofile   1000000
root          soft  memlock   32000
root          hard  memlock   32000
root          soft  msgqueue  8192000
root          hard  msgqueue  8192000

#其他账户的资源软限制和硬限制
*          soft  core      unlimited
*          hard  core      unlimited
*          soft  nproc     1000000
*          hard  nproc     1000000
*          soft  nofile   1000000
*          hard  nofile   1000000
*          soft  memlock   32000
*          hard  memlock   32000
*          soft  msgqueue  8192000
*          hard  msgqueue  8192000
```

2.3.7：内核参数优化：

```
# Controls source route verification
net.ipv4.conf.default.rp_filter = 1
net.ipv4.ip_nonlocal_bind = 1
net.ipv4.ip_forward = 1

# Do not accept source routing
net.ipv4.conf.default.accept_source_route = 0

# Controls the System Request debugging functionality of the kernel
kernel.sysrq = 0

# Controls whether core dumps will append the PID to the core filename.
# Useful for debugging multi-threaded applications.
kernel.core_uses_pid = 1

# Controls the use of TCP syncookies
net.ipv4.tcp_syncookies = 1

# Disable netfilter on bridges.
```

```
net.bridge.bridge-nf-call-ip6tables = 0
net.bridge.bridge-nf-call-iptables = 0
net.bridge.bridge-nf-call-arptables = 0

# Controls the default maximum size of a message queue
kernel.msgmnb = 65536

# # Controls the maximum size of a message, in bytes
kernel.msgmax = 65536

# Controls the maximum shared segment size, in bytes
kernel.shmmax = 68719476736

# # Controls the maximum number of shared memory segments, in pages
kernel.shmall = 4294967296

# TCP kernel parameter
net.ipv4.tcp_mem = 786432 1048576 1572864
net.ipv4.tcp_rmem = 4096      87380    4194304
net.ipv4.tcp_wmem = 4096      16384    4194304
net.ipv4.tcp_window_scaling = 1
net.ipv4.tcp_sack = 1

# socket buffer
net.core.wmem_default = 8388608
net.core.rmem_default = 8388608
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
net.core.netdev_max_backlog = 262144
net.core.somaxconn = 20480
net.core.optmem_max = 81920

# TCP conn
net.ipv4.tcp_max_syn_backlog = 262144
net.ipv4.tcp_syn_retries = 3
net.ipv4.tcp_retries1 = 3
net.ipv4.tcp_retries2 = 15

# tcp conn reuse
net.ipv4.tcp_timestamps = 0
net.ipv4.tcp_tw_reuse = 0
net.ipv4.tcp_tw_recycle = 0
net.ipv4.tcp_fin_timeout = 1

net.ipv4.tcp_max_tw_buckets = 20000
net.ipv4.tcp_max_orphans = 3276800
net.ipv4.tcp_synack_retries = 1
net.ipv4.tcp_syncookies = 1
```

```
# keepalive conn
net.ipv4.tcp_keepalive_time = 300
net.ipv4.tcp_keepalive_intvl = 30
net.ipv4.tcp_keepalive_probes = 3
net.ipv4.ip_local_port_range = 10001      65000

# swap
vm.overcommit_memory = 0
vm.swappiness = 10

#net.ipv4.conf.eth1.rp_filter = 0
#net.ipv4.conf.lo.arp_ignore = 1
#net.ipv4.conf.lo.arp_announce = 2
#net.ipv4.conf.all.arp_ignore = 1
#net.ipv4.conf.all.arp_announce = 2
```

2.3.8: dpkg安装包管理:

rpm: RPM(Red Hat Package Manager), 是基于Red hat的Linux Distribution的包管理系统, 同时也指rpm包本身, RPM用于rpm包的管理 (诸如安装、卸载、升级等)

"dpkg "是"Debian Packager "的简写,为 "Debian"专门开发的套件管理系统, 方便软件的安装、更新及移除。所有源自"Debian"的"Linux "发行版都使用 "dpkg", 例如 "Ubuntu"、"Knoppix "等。

```
# dpkg -i gitlab-ce_11.9.8-ce.0_amd64.deb #安装某个软件包
# dpkg -r gitlab-ce #删除某个软件包保留配置文件
# dpkg -r -P gitlab-ce #删除某个软件包不留配置文件
# dpkg -I gitlab-ce_11.9.8-ce.0_amd64.deb #查看软件包信息
# dpkg -c gitlab-ce_11.9.8-ce.0_amd64.deb #查看软件包内的文件及目录内容
# dpkg -l #列出本机已经安装的所有软件
```

2.3.9: apt安装Nginx:

```
~$ sudo apt install nginx=1.14.0-0ubuntu1.6
~$ systemctl start nginx
```

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

三：Ubuntu Desktop版本安装及使用：

3.1：安装过程：

参考server版安装过程

3.2：环境配置：

1、设置软件源及安装常用命令：

```
https://opsx.alibaba.com/mirror
```

```
sudo apt-get install build-essential cmake pkg-config qt4-qmake libqt4-dev desktop-file-utils \
libavformat-dev libavcodec-dev libavutil-dev libswscale-dev libasound2-dev libpulse-dev
libjack-jackd2-dev \
libgl1-mesa-dev libglu1-mesa-dev libx11-dev libxfixes-dev libxext-dev libxi-dev
libxinerama-dev
```

2、系统更新及配置中文语言环境：

3、安装搜狗拼音输入法

```
https://pinyin.sogou.com/linux/?r=pinyin
```

4、安装转码器ffmpeg：

多媒体视频处理工具FFmpeg有非常强大的功能包括视频采集功能、视频格式转换、视频抓图、给视频加水印等。

```
sudo apt-get install ffmpeg
```

5、安装视频播放器： smplayer <https://www.jianshu.com/p/f24252c632d0>

```
sudo apt-get install smplayer
```

6、办公软件WPS：

```
https://www.wps.cn/product/wpslinux
```

7、单机VNC工具：

```
x11vnc server
```

8、RealVNC v6.6：

支持多个用户同时连接

<https://www.realvnc.com/en/connect/download/vnc/linux/>

9、文本编辑器：

visual studio code

10、markdown工具：

<https://www.typora.io/#linux>

`sudo apt-get install typora=0.9.60-1`

11、Ubuntu 桌面3D特效：

`sudo apt-get install compiz-plugins compizconfig-settings-manager`

12、左侧菜单在底栏显示：

`~$ gsettings set com.canonical.Unity.Launcher launcher-position Bottom`

13、VMware workstion桥接网卡设置

https://kb.vmware.com/s/article/287?lang=zh_CN

`chmod a+rwx /dev/vmnet0 #解决网卡桥接不通问题`

14、远程工具：

SecureCRT

15、python开发：

pycharm

16、java开发：

Eclipse

IntelliJ IDEA

17、vmware workstation：

#网卡开启混杂模式 https://kb.vmware.com/s/article/287?lang=zh_CN

```
$ sudo vim /etc/init.d/vmware
126 # Start the virtual ethernet kernel service
127 vmwareStartVmnet() {
128     vmwareLoadModule $vnet
129     "$BINDIR"/vmware-networks --start >> $VNETLIB_LOG 2>&1
130     chgrp magedu /dev/vmnet*
131     chown magedu /dev/vmnet*
132     chmod a+rwx /dev/vmnet*
133 }
```



Thank You!

讲师：张士杰（杰哥）

<http://www.magedu.com>