

# Linux服务器构建与运维管理

## 从基础到实战 (基于 openEuler)

### 第1章：初识Linux

阮晓龙

13938213680 / ruanxiaolong@hactcm.edu.cn

<https://internet.hactcm.edu.cn>

<http://www.51xueweb.cn>

河南中医药大学信息技术学院互联网技术教学团队  
河南中医药大学医疗健康信息工程技术研究所

2024.9

### 提纲

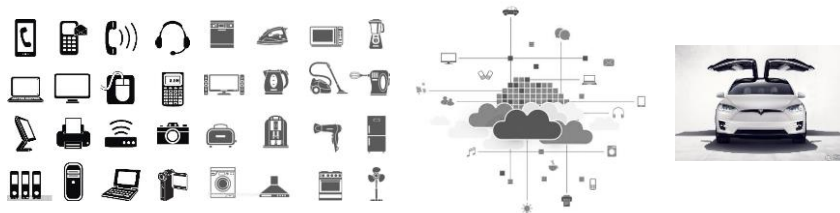
- 了解Linux
- 认识openEuler
- 基于VirtualBox安装openEuler
  - VirtualBox的网络配置
  - 创建虚拟机并安装openEuler
- 使用YUM/DNF管理openEuler
  - YUM与DNF
  - openSSH
- 通过SSH远程管理openEuler
  - 远程管理 SSH、VNC、Telnet
- 电源管理



## 1. 了解Linux

### 1.1 Linux简介

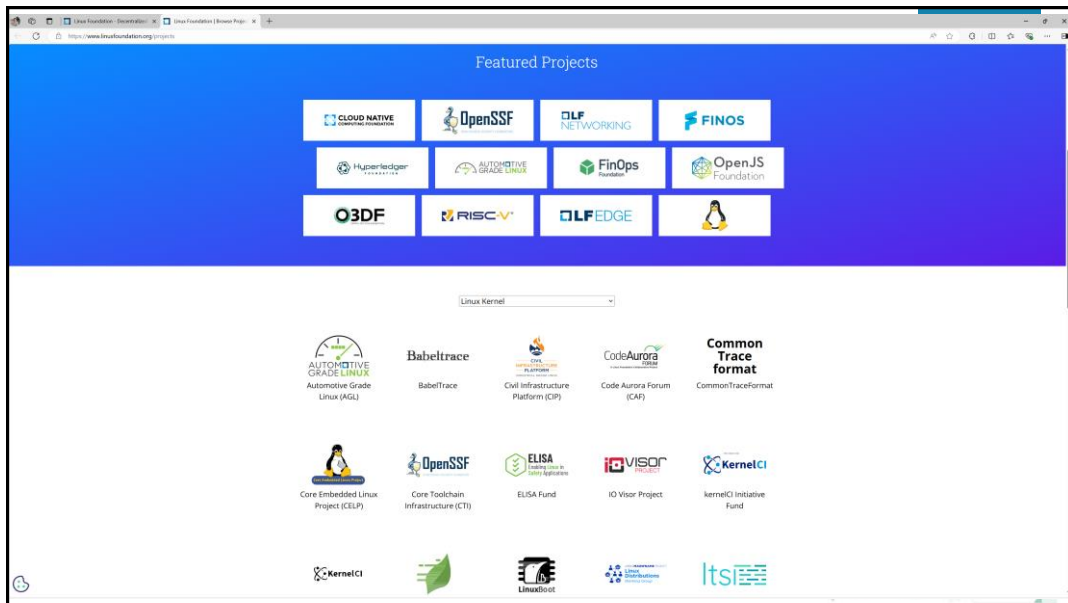
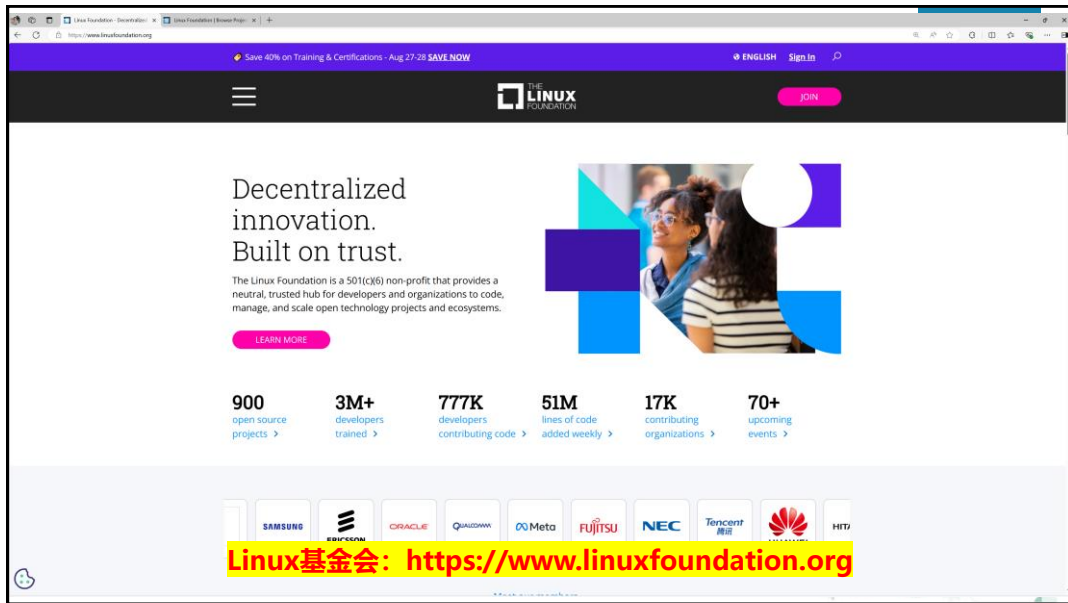
- 通常所说的Linux指**Linux操作系统**。
  - Linux全称GNU/Linux，是一套免费使用和自由传播的类UNIX操作系统，是多用户、多任务、支持多线程和多CPU的操作系统。
  - Linux操作系统主要包含 **Linux Kernel**、**GNU** 和**应用程序**三部分。



## 1. 了解Linux

### 1.2 Linux Kernel

- 原生Linux指**Linux Kernel**。
  - Linux Kernel 指的是一个提供设备驱动、文件系统、进程管理、网络通信等功能的系统软件，俗称Linux内核。
  - Linux Kernel 不是一套完整的操作系统，只是操作系统的核心。
  - Linux Kernel 是开源项目，主要由**Linux基金会**负责维护。
- 关于Linux基金会的更多信息，可以访问其官方网站 (<http://www.linuxfoundation.org>) 详细了解。



# 1. 了解Linux

## 1.2 Linux Kernel

- Linux内核源代码是公开的，任何人都可以对内核加以修改并发布给其他人使用，内核版本通过项目管理工具进行着规范化管理。
- Linux内核版本号有两种体系，一个是内核版本号，一个是发行版本号。
  - Linux内核版本号由r、x、y三组数字组成，目前Linux内核版本有稳定版和开发版。
    - 第一个组数字r：当前发布的内核主版本号。
    - 第二个组数字x：偶数表示稳定版，奇数表示开发版。
    - 第三个组数字y：错误修补的次数。
  - 例如：6.10.6 (r.x.y) :
    - 主版本号r=6；
    - 次版本号x=10，表示稳定版；
    - 错误修补的次数y=6。
    - 前两个数字组合描述内核系列，如稳定版的6.10.6是6.10版内核系列。

网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn>

棒棒堂智能运维 <http://www.51xueweb.cn>



**The Linux Kernel Archives**

Latest Release: **6.10.6**

| mainline   | 6.10.rc5      | 2024-08-25 | [patch] | [inc. patch] | [new diff] | [browse] |
|------------|---------------|------------|---------|--------------|------------|----------|
| stable     | 6.10.6        | 2024-08-19 | [patch] | [inc. patch] | [new diff] | [browse] |
| longterm   | 6.6.47        | 2024-08-19 | [patch] | [inc. patch] | [new diff] | [browse] |
| longterm   | 6.10.6        | 2024-08-19 | [patch] | [inc. patch] | [new diff] | [browse] |
| longterm   | 5.15.165      | 2024-08-19 | [patch] | [inc. patch] | [new diff] | [browse] |
| longterm   | 5.10.224      | 2024-08-19 | [patch] | [inc. patch] | [new diff] | [browse] |
| longterm   | 5.4.262       | 2024-08-19 | [patch] | [inc. patch] | [new diff] | [browse] |
| longterm   | 4.19.320      | 2024-08-19 | [patch] | [inc. patch] | [new diff] | [browse] |
| linux-next | next-20240828 | 2024-08-28 |         |              |            | [browse] |

**Other resources**

Get Traces: Documentation, Kernel Mailing Lists, Patchwork, Wikis, Bugzilla, Release Atom Feed, Mimers, Linux.com, Kernel Planet

**Social**

Site Atom Feed, Release Atom Feed, Kernel Planet

This site is operated by the Linux Kernel Organization, Inc., a 501(c)(3) nonprofit corporation, with support from the following sponsors:

SPONSORS: EQUINIX, METAL, fastly, CONSTELLIX, RedHat, PURE STORAGE, Google, LINUX FOUNDATION

**index : kernel/git/stable/linux.git**

Linux kernel stable tree

about summary refs log tree commit diff stats

Diffstat

| File                           | Changes |
|--------------------------------|---------|
| arch/arm64/include/asm/cpuid.h | 2/1     |
| arch/arm64/include/asm/cpuid.h | 1/1     |
| arch/arm64/include/asm/cpuid.h | 10/000  |
| arch/arm64/include/asm/cpuid.h | 14/000  |
| arch/arm64/include/asm/cpuid.h | 148/000 |
| arch/arm64/include/asm/cpuid.h | 9/000   |
| arch/arm64/include/asm/cpuid.h | 8/000   |
| arch/arm64/include/asm/cpuid.h | 2/1     |
| arch/arm64/include/asm/cpuid.h | 31/000  |
| arch/arm64/include/asm/cpuid.h | 1/1     |
| arch/arm64/include/asm/cpuid.h | 31/000  |
| arch/arm64/include/asm/cpuid.h | 148/000 |
| arch/arm64/include/asm/cpuid.h | 9/000   |
| arch/arm64/include/asm/cpuid.h | 60/000  |
| arch/arm64/include/asm/cpuid.h | 4/000   |
| arch/arm64/include/asm/cpuid.h | 8/000   |
| arch/arm64/include/asm/cpuid.h | 40/000  |
| arch/arm64/include/asm/cpuid.h | 2/1     |
| arch/arm64/include/asm/cpuid.h | 10/000  |
| arch/arm64/include/asm/cpuid.h | 2/1     |
| arch/arm64/include/asm/cpuid.h | 2/1     |
| arch/arm64/include/asm/cpuid.h | 71/000  |
| arch/arm64/include/asm/cpuid.h | 4/000   |
| arch/arm64/include/asm/cpuid.h | 2/1     |
| arch/arm64/include/asm/cpuid.h | 1/1     |
| arch/arm64/include/asm/cpuid.h | 7/000   |

29 files changed, 488 insertions, 315 deletions

**Linux Kernel: <https://www.kernel.org>**

## 1. 了解Linux

### 1.2 Linux Kernel

- 内核是操作系统最基本的部分，它有多种不同的实现模式，主要有单内核和微内核，以及主要在科研领域使用的外内核。
- 单内核
  - Linux大部分都是单内核的，单内核有一个较大的进程管理系统，属于集中式操作系统，它的内部又被分为用户程序、系统调用以及硬件控制三个层次和若干个模块。
  - 运行时每一个模块都是一个独立的二进制映像或者进程，并且可通过直接调用其他模块中的函数来实现通信。
  - 单内核的所有模块都在同一块寻址空间内运行，倘若某个模块有错误，运行时就会损及整个操作系统运行。
  - 如果单内核架构的操作系统在开发设计时相当完善，并经测试验证后具有高度可靠性，则操作系统内的各软件组件因具有高度紧密性，系统运行效率也会大幅提高。



## 1. 了解Linux

### 1.2 Linux Kernel

- 内核是操作系统最基本的部分，它有多种不同的实现模式，主要有单内核和微内核，以及主要在科研领域使用的外内核。
- 微内核
  - 微内核出现于单内核之后，它仅将操作系统中最核心的功能写入内核，如线程管理、内存管理、地址空间、进程间通信等。
  - 微内核模式是面向对象理论在操作系统设计中的产物，其通过对系统逻辑功能的划分，将操作系统结构中的处理管理、文件管理、存储管理和设备管理等高级功能服务模块从内核中分离出来，成为在一定的特权方式下运行并且具有独立运行空间的独立进程，同时各个进程之间以消息机制来完成信息交换。
  - 微内核以模块化的方式完成操作系统中比较高级的操作，这样的设计使内核中最核心部分的设计更简单，增加了内核的灵活性，使之易于维护和移植。
  - 在微内核操作系统中，一个服务组件失效并不会导致整个系统崩溃，内核需要做的，仅仅是重新启动这个组件，而不影响其它的部分。



## 1. 了解Linux

### 1.3 Linux发行版与衍生发行版

- 许多个人、组织和企业使用Linux Kernel开发了遵循GNU/Linux协议的完整操作系统，叫做Linux发行版。
- 通常所说的Linux操作系统就是**基于Linux Kernel的发行版**。
- Linux衍生发行版是基于Linux发行版再次改造所衍生出的Linux操作系统，其目的通常是为了进一步简化Linux发行版的安装、使用以及提供应用软件等。
  - 知名Linux发行版有Debian、SlackWaer、RedHat、Gentoo、ArchLinux、红旗Linux等。
  - 知名Linux衍生发行版有Ubuntu、SUSE、openSUSE、CentOS、Fedora等。

网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn> 棒棒堂智能运维 <http://www.51sucweb.cn>

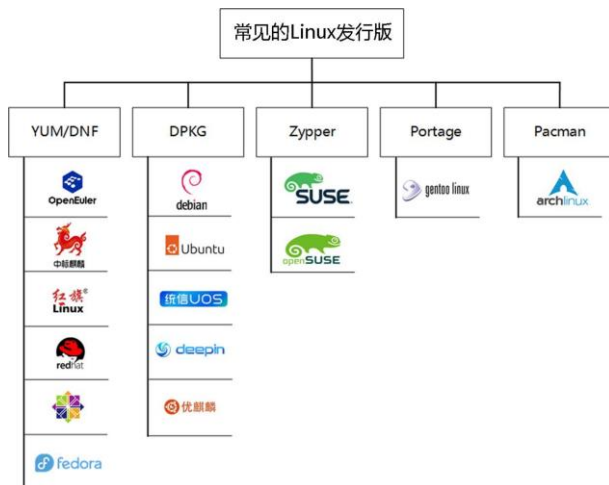


主要的Linux发行版与衍生发行版

| 序号 | 名称          | 发行商          | 网站地址                                                                  |
|----|-------------|--------------|-----------------------------------------------------------------------|
| 1  | RedHat      | 红帽公司         | <a href="https://www.redhat.com">https://www.redhat.com</a>           |
| 2  | Fedora      | Fedora 项目社区  | <a href="https://fedoraproject.org">https://fedoraproject.org</a>     |
| 3  | CentOS      | 红帽公司         | <a href="https://www.centos.org">https://www.centos.org</a>           |
| 4  | Debian      | Debian 项目社区  | <a href="https://www.debian.org">https://www.debian.org</a>           |
| 5  | Ubuntu      | Canonical    | <a href="https://cn.ubuntu.com">https://cn.ubuntu.com</a>             |
| 6  | SUSE        | Novell       | <a href="https://www.suse.com">https://www.suse.com</a>               |
| 7  | openSUSE    | Novell       | <a href="https://www.opensuse.org">https://www.opensuse.org</a>       |
| 8  | Gentoo      | Gentoo 项目社区  | <a href="https://www.gentoo.org">https://www.gentoo.org</a>           |
| 9  | Arch        | Arch 项目社区    | <a href="https://archlinux.org">https://archlinux.org</a>             |
| 10 | openEuler   | openEuler 社区 | <a href="https://www.openeuler.org">https://www.openeuler.org</a>     |
| 11 | 统信 UOS      | 统信软件技术有限公司   | <a href="https://www.chinauos.com">https://www.chinauos.com</a>       |
| 12 | NeoKylin    | 麒麟软件有限公司     | <a href="https://www.kylinos.cn">https://www.kylinos.cn</a>           |
| 13 | FusionOS    | 超聚变数字技术有限公司  | <a href="https://www.xfusion.com">https://www.xfusion.com</a>         |
| 14 | UbuntuKylin | 麒麟软件有限公司     | <a href="https://www.ubuntukylin.com">https://www.ubuntukylin.com</a> |
| 15 | 红旗 Linux    | 中科红旗信息科技产业集团 | <a href="https://www.chinaredflag.cn">https://www.chinaredflag.cn</a> |



Linux 操作系统中使用软件包管理器进行软件安装、卸载和管理。  
按照包管理方式划分常见的 Linux 发行版



## 1. 了解Linux

### 1.4 开源 open source

□ 开源（Open Source）即开放源代码。

- 开源系统同样有版权，受到法律保护。
- 开放源代码的定义由 Bruce Perens（Debian 的创始人之一）创立，关键内容：
  - ① 自由再散布（Free Distribution）：获得源代码的人可自由再将此源代码散布。
  - ② 源代码（Source Code）：程序的可执行版本在散布时，必须随附完整源代码或是可让人方便地取得源代码。
  - ③ 衍生著作（Derived Works）：任何人依此源代码修改后，依照同一授权条款再散布。
  - ④ 原创作者程序源代码的完整性（Integrity of The Author's Source Code）：修改后的版本，需以不同的版本号与原始的程序源代码进行区分，保障原始代码的完整性。
  - ⑤ 不得对任何人或团体有差别待遇（No Discrimination Against Persons or Groups）：开放源代码软件不得因性别、团体、国家、族群等设定限制，但若是因为法律规定的情形则为例外。
  - ⑥ 对程序在任何领域内的利用不得有差别待遇（No Discrimination Against Fields of Endeavor）：不得限制商业使用。

# 1. 了解Linux

## 1.4 开源 open source

- 开源 (Open Source) 即开放源代码。
  - 开源系统同样有版权, 受到法律保护。
  - 开放源代码的定义由 Bruce Perens (Debian 的创始人之一) 创立, 关键内容:
    - ⑧ 散布授权条款 (Distribution of License): 若软件再散布, 必须以同一条款散布。
    - ⑨ 授权条款不得专属于特定产品 (License Must Not Be Specific to a Product): 若多个程序组合成一套软件, 则当某一开放源代码的程序单独散布时, 也需要符合开放源代码的条件。
    - ⑩ 授权条款不得限制其他软件 (License Must Not Restrict Other Software): 当某一开放源代码软件与其他非开放源代码软件一起发布时, 不得限制其他软件的授权条件, 也要遵照开放源代码的授权。
    - ⑪ 授权条款必须技术中立 (License Must Be Technology-Neutral): 授权条款不得限制为电子格式才有效, 纸质授权条款也应视为有效。

网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn>

棒棒堂智能运维 <http://www.51xueweb.cn>



**DEFINING OPEN SOURCE AI**

We're driving a multi-stakeholder process to define an "Open Source AI", with online discussions and in-person workshops.

[Join us!](#)

**About the Open Source Initiative**

The OSI is the authority that defines Open Source, recognized globally by individuals, companies, and by public institutions.

[Learn more](#)

**The Open Source Definition**

Page created on July 7, 2006 | Last modified on February 16, 2024

**Introduction**

Open source doesn't just mean access to the source code. The distribution terms of open source software must comply with the following criteria:

**1. Free Redistribution**

The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

**2. Source Code**

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost, preferably downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

**open source: <https://opensource.org>**

## 1. 了解Linux

### 1.4 开源 open source

#### □ 开源协议

- 为了维护作者和贡献者的合法权利，保证开源软件不被商业机构或个人窃取，影响软件发展，开源社区开发出了多种开源许可协议。
- 常见的许可协议有 GPL、LGPL、MPL、Apache、MIT、BSD、QPL、QNCL、Jabber、IBM 等。
  - GPL 许可协议 (GNU General Public License) 保证了所有开发者的权利，同时为使用者提供了足够的复制、分发、修改的权利，是开源界最常用的许可模式。
  - LGPL 许可协议 (GNU Lesser General Public License) 是 GPL 的一个主要为类库设计的开源协议。
  - MPL (Mozilla Public License) 许可协议主要平衡开发者对源代码的需求和他们利用源代码获得的利益。



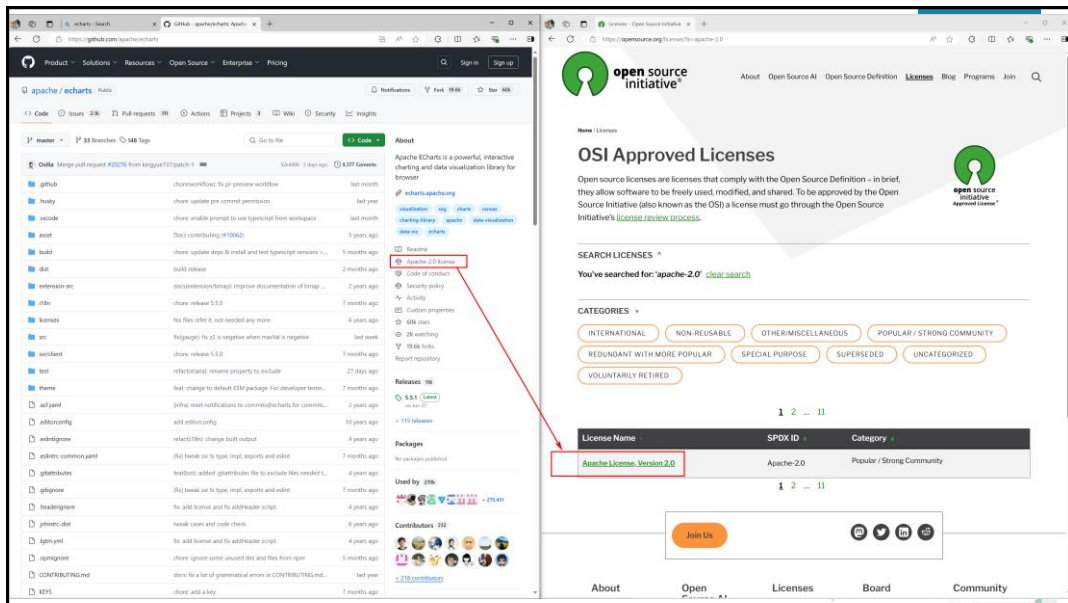
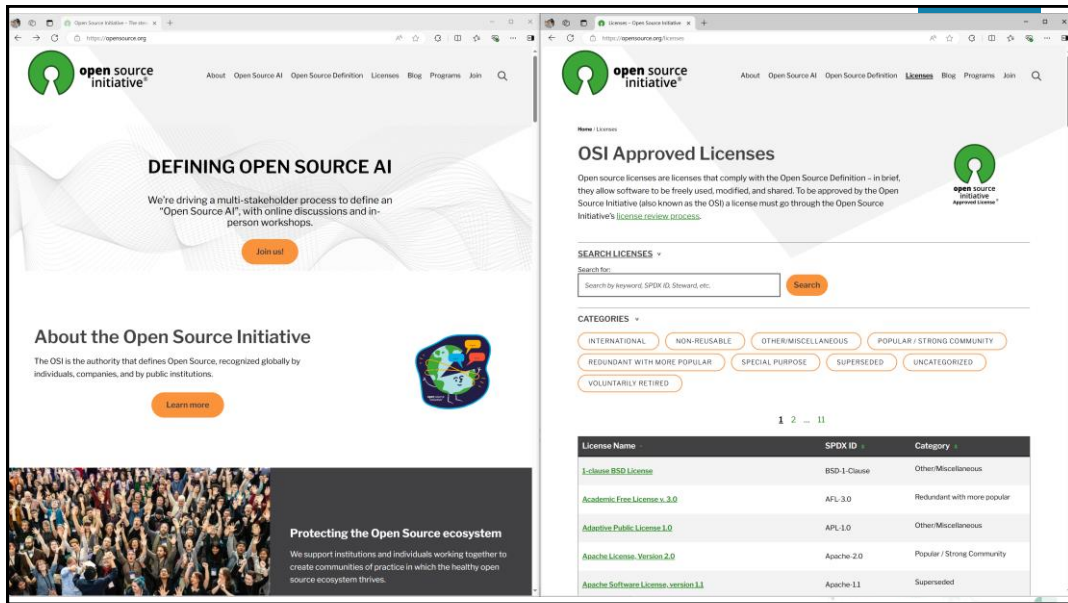
## 1. 了解Linux

### 1.4 开源 open source

#### □ 开源协议

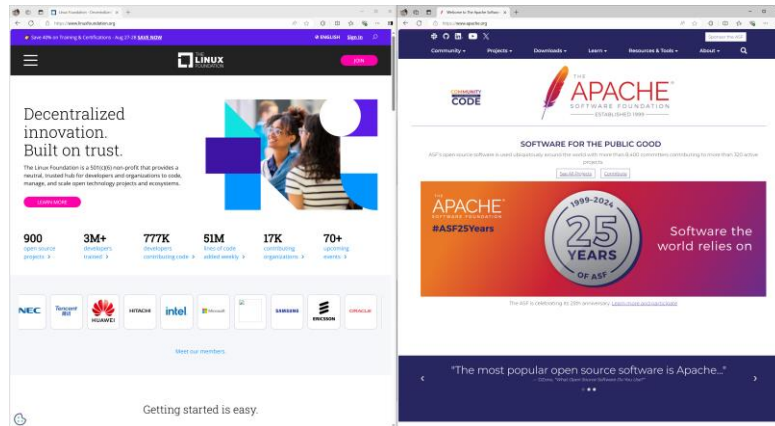
- 为了维护作者和贡献者的合法权利，保证开源软件不被商业机构或个人窃取，影响软件发展，开源社区开发出了多种开源许可协议。
- 常见的许可协议有 GPL、LGPL、MPL、Apache、MIT、BSD、QPL、QNCL、Jabber、IBM 等。
  - Apache 许可协议 (Apache License) 是著名的非营利开源组织 Apache 采用的协议，主要特点有永久权利、全球范围权利、授权免费且无版权、授权无排他性、授权不可撤销等。
  - MIT 许可协议 (Massachusetts Institute of Technology) 是广泛使用的开源协议中最宽松的，其软件及相关文档对所有人免费，允许使用者修改、复制、合并、发表、授权甚至销售等，唯一限制是软件中必须包含上述版权和许可声明。
  - BSD 许可协议 (Berkeley Software Distribution License) 不仅需要附上许可证的原文，它还要求开发者上传自己的版权资料，所以 BSD 许可证发行的软件版权资料许可证的所占空间可能比程序还大。





# 1. 了解Linux

## 1.4 开源 open source



世界知名的开源组织：Linux Foundation 和 Apache Software Foundation

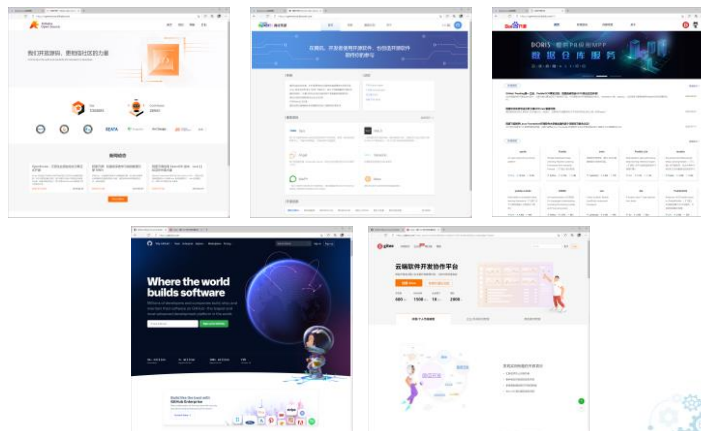
网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn>

棒棒堂智能运维 <http://www.51xueweb.cn>

# 1. 了解Linux

## 1.4 开源 open source

国内知名的IT厂商都建立了开源社区



网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn>

棒棒堂智能运维 <http://www.51xueweb.cn>

## 2. 认识openEuler

### 2.1 初识openEuler

#### □ openEuler操作系统

- 是全球开源贡献者构建的高效、稳定、安全的开源操作系统。
- 是最初面向服务器场景的操作系统，目前已从服务器操作系统正式升级为面向数字基础设施的操作系统。
- 是由openEuler开源社区负责维护。
  - openEuler开源社区是一个面向全球的操作系统开源社区，通过社区合作，打造创新平台，构建支持多处理器架构、统一和开放的操作系统，推动软、硬件应用生态繁荣发展。



## 2. 认识openEuler

### 2.1 初识openEuler

#### □ openEuler的发展历程

- 2019年12月31日，面向多样性计算的操作系统开源社区openEuler正式成立。
- 2020年3月27日，openEuler 20.03 LTS (Long Term Support, 长生命周期支持) 版本正式发布，为Linux世界带来一个全新的、具备独立技术演进能力的Linux发行版。
- 2020年9月30日，首个openEuler 20.09创新版发布，该版本是openEuler社区中的多个企业、团队、独立开发者协同开发的成果，在openEuler社区的发展进程中具有里程碑式的意义，也是中国开源历史上的标志性事件。
- 2021年3月31日，发布openEuler 21.03内核创新版，该版本将内核升级到5.10，还在内核方向实现内核热升级、内存分级扩展等多个创新特性，加速提升多核性能，构筑千核运算能力。
- 2021年9月30日，全新openEuler 21.09创新版如期而至，这是openEuler全新发布后的第一个社区创新版本，实现了全场景支持。增强服务器和云计算的特性，发布面向云原生的业务混部CPU调度算法、容器化操作系统KubeOS等关键技术，同时发布边缘和嵌入式版本。
- 2022年3月30日，基于统一的5.10内核，发布面向服务器、云计算、边缘计算、嵌入式的全场景openEuler 22.03 LTS版本，聚焦算力释放，持续提升资源利用率，打造全场景协同的数字基础设施操作系统。
- 2022年9月30日，发布openEuler 22.09创新版本，持续补齐全场景的支持。



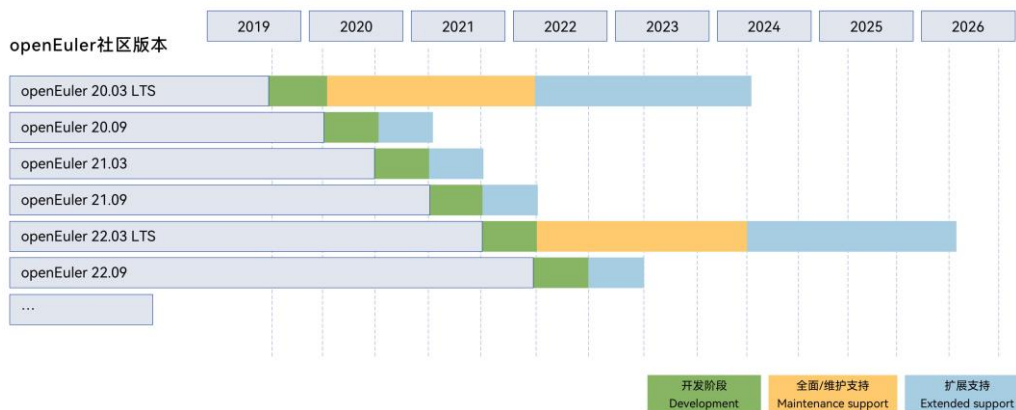
## 2. 认识openEuler

### 2.2 openEuler生命周期管理规范

- openEuler生命周期管理是指对openEuler版本和软件包的整个生命周期进行管理，包括版本号命名、版本发布、版本更新、社区支持以及版本生命周期的终止等方面。
  - 可通过访问地址“<https://www.openeuler.org/zh/other/lifecycle/>”查看社区版本生命周期管理。
- **社区版本生命周期管理规范（总体）**
  - 社区版本按照交付年份和月份进行版本号命名。例如：openEuler 20.09于2020年9月发布。
  - 社区版本分为长期支持版本和创新版本。
    - 长期支持版本：发布间隔周期定为2年，提供4年社区支持。
    - 社区创新版本：openEuler每隔6个月会发布一个社区创新版本，提供6个月社区支持。



## openEuler社区版本生命周期管理规范（总体）

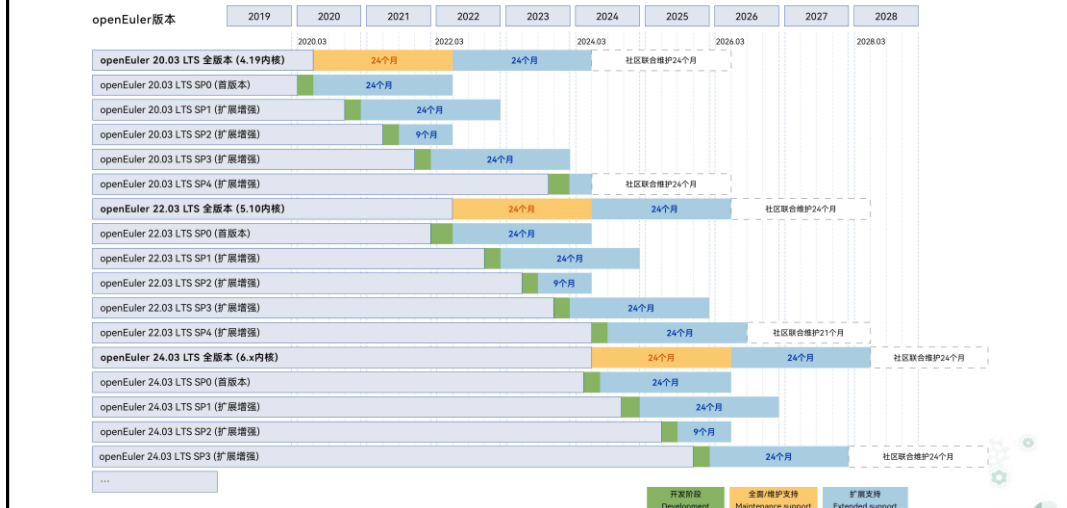


## 2. 认识openEuler

### 2.2 openEuler生命周期管理规范

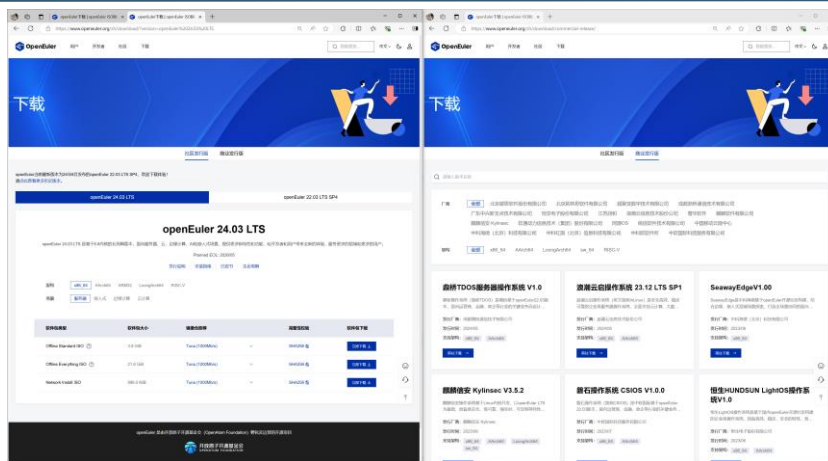
- openEuler生命周期管理是指对openEuler版本和软件包的整个生命周期进行管理，包括版本号命名、版本发布、版本更新、社区支持以及版本生命周期的终止等方面。
  - 可通过访问地址“<https://www.openeuler.org/zh/other/lifecycle/>”查看社区版本生命周期管理。
- 社区版本生命周期管理规范（LTS+SP）
  - 目前LTS版本全版本生命周期4年（2+2），到生命周期结束前半年至1年由相关团队组建联合维护团队，申请延长至6年。
  - LTS版本的SP版本生命周期原则上按照：
    - 小SP（6月Release，可选）9个月执行，
    - 大SP（12月Release）24个月执行；
    - 大规模使用建议选择大SP。

## openEuler社区版本生命周期管理规范 (LTS+SP)



## 2. 认识openEuler

### 2.3 openEuler的发行版与衍生发行版



注：按公司名拼音首字母顺序排序

|                                                                                   |                                                                                   |                                                                                   |                                                                                    |
|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|------------------------------------------------------------------------------------|
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn> 棒棒堂智能运维 <http://www.51xueweb.cn>



- ❑ 木兰协议是我国首个开源协议，这一开源协议共有五个主要方面，涉及授予版权许可、授予专利许可、无商标许可、分发限制和免责申明与责任限制。
- ❑ 在版权许可方面，木兰协议允许“每个‘贡献者’根据”本许可证“授予您永久性的、全球性的、免费的、非独占的、不可撤销的版权许可，您可以复制、使用、修改、分发其‘贡献’，不论修改与否。”
- ❑ 木兰协议比Apache License更好一些，Apache License要求列出每个修改文件，其实很多项目做不到这一点，所以MulanPSL直接取消了这个要求。



The image shows a screenshot of the OpenEuler website on the left and a document titled '木兰宽松许可证 第2版' (Mulan Relax License Version 2.0) on the right. The website displays various logos of partner companies, including PowerLeader, αFUSION, 星光长空, HUND, Intewell, kaihong, 联众信安, HPSUN, CEIC, K-Security, 麒麟信安, 浪潮软件, SUSE, turbolinux, 浪潮软件, H3C, ISCAS, iSOFTSTONE, 国盾量子, ThunderSoft, 中科方德, 中科红旗, and SEAWAY. Below the logos is a section titled '友情链接' (Friendly Links) with links to various communities and organizations. The document on the right is titled '木兰宽松许可证 第2版' and contains text in Chinese, including a section for '适用范围' (Scope of Application) and '许可证条款' (License Terms). A red arrow points from the '木兰宽松许可证 第2版' title in the document to the '友链' (Friends Link) section on the website.

### 3. 基于VirtualBox安装openEuler

#### 3.1 安装前准备

##### 了解操作系统所需的硬件需求

- Linux操作系统的硬件需求取决于所使用的发行版。
  - Linux操作系统兼容绝大多数计算机硬件，通常不需要考虑硬件兼容性问题。
  - 由于硬件的发展速度很快，很难保证Linux与所有的硬件都能达到100%的兼容，因此在安装之前需要了解操作系统所需的硬件配置，相关发行版的硬件兼容性信息可通过官方网站获取。
- 例如：
  - CentOS官方硬件支持列表：
    - <https://wiki.centos.org/AdditionalResources/HardwareList>
  - Ubuntu官方硬件支持列表：
    - <https://wiki.ubuntu.com/HardwareSupport>
  - openEuler的兼容性列表：
    - <https://www.openeuler.org/zh/compatibility>

网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn>

棒棒堂智能运维 <http://www.51xueweb.cn>



The image displays two side-by-side screenshots of the openEuler compatibility list website. Both screenshots show the '兼容性列表' (Compatibility List) page. The left screenshot shows the '操作系統' (Operating System) filter set to '全部' (All) and the 'CPU' filter set to '全部' (All). The right screenshot shows the '操作系統' (Operating System) filter set to '全部' (All) and the 'CPU' filter set to '全部' (All). The website header includes the openEuler logo and navigation links. The main content area lists various hardware configurations and their compatibility with different openEuler versions. The table columns include '硬件名称' (Hardware Name), '操作系統' (Operating System), '版本' (Version), '软件类型' (Software Type), '软件描述' (Software Description), '下载地址' (Download Address), and '开源协议' (Open Source License).

| 硬件名称   | 操作系統                    | 版本          | 软件类型                       | 软件描述        | 下载地址 | 开源协议           |
|--------|-------------------------|-------------|----------------------------|-------------|------|----------------|
| x86_64 | openEuler 20.03 LTS SP1 | 18.11.3-1   | ApplicationSystem          | Open Source | link | BSD and LGPLv2 |
| x86_64 | openEuler 20.03 LTS SP1 | 0.98.22.4.5 | SystemEnvironmentDatabases | Open Source | link | GPLv2+         |
| x86_64 | openEuler 20.03 LTS SP1 | 0.2.41-12   | SystemEnvironmentDatabases | Open Source | link | LGPLv2+        |

## 3. 基于VirtualBox安装openEuler

### 3.1 安装前准备

- 了解操作系统的磁盘分区
  - 在安装Linux操作系统之前，规划操作系统的磁盘分区很有必要。
    - 磁盘的分区主要有主分区（Primary partion）、扩展分区（Extension partion）和逻辑分区（Logical partion）。
    - Linux分区是通过将分区挂载到目录上实现对分区的访问，其分区方式根据系统的应用环境不同，各个分区的规划各不相同，好的分区规划可以有效提升系统的可用性。
    - 对于初学者，建议使用默认分区方案以快速开始学习。

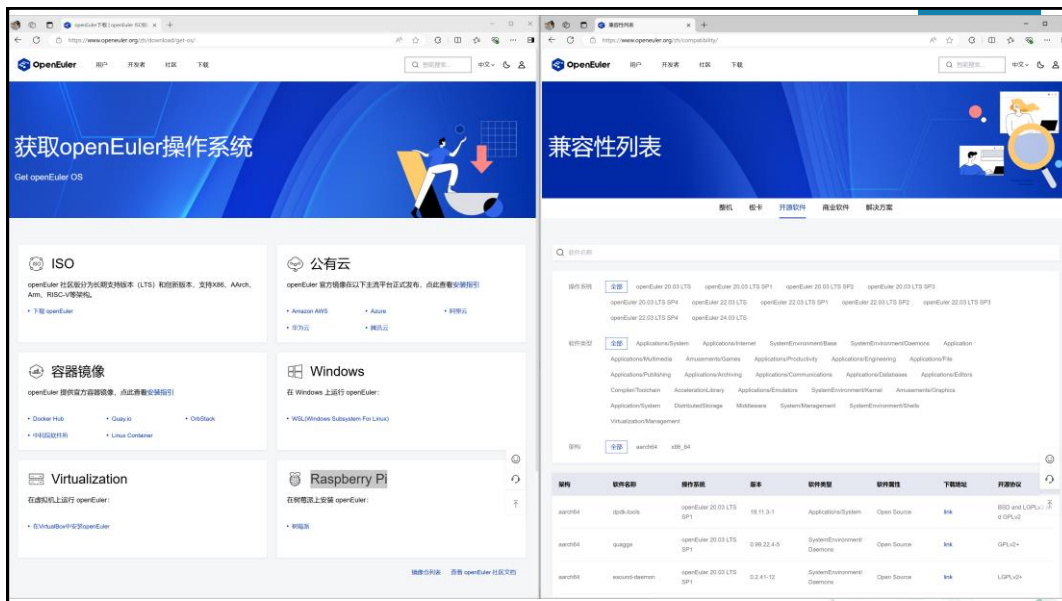


## 3. 基于VirtualBox安装openEuler

### 3.1 安装前准备

- 获取openEuler操作系统
  - 获取渠道：
    - 官网：<http://https://www.openeuler.org/zh/download/get-os/>
    - 镜像服务：<http://mirrors.aliyun.com> <http://mirrors.163.com>  
<http://mirrors.sohu.com> <http://mirrors.ustc.edu.cn>
  - 选择合适的版本：
    - ISO: X86\_64 AArch64 ARM32 LoongArch64 RISC-V
    - 公有云版本: Amazon AWS Azure 阿里云 华为云 腾讯云
    - 容器镜像: Docker Hut Quay.io OrbStack 中科院软件所 Linux Container
    - Windows上使用
    - Virtualization上运行: VirtualBox
    - Raspberry Pi上安装
  - 最简单：
    - 从课程学习平台 (<https://internet.hactcm.edu.cn>) 获取





### 3. 基于VirtualBox安装openEuler

40  
3.2 安装方式

- openEuler操作系统的安装方式灵活多样，可根据环境的不同进行选择。
- 安装方式当前仅支持：
  - 光盘
  - USB盘安装
  - 网络安装
    - 使用PXE通过网络安装。
  - qcow2镜像安装
  - 私有镜像安装
    - 仅华为公有云的x86虚拟化平台支持私有镜像安装。

### 3. 基于VirtualBox安装openEuler

#### 3.3 安装流程



### 3. 基于VirtualBox安装openEuler

#### 3.4 为什么使用Linux操作系统

- 为什么选用Linux作为服务器操作系统？
  - 当一种产品对用户的价值随着采用相同的产品、或可兼容产品的用户增加而增大时，就出现了网络外部性。
  - 在具有网络效应的产业，“先下手为强”（first-mover advantage）和“赢家通吃”（winner - take - all）是市场竞争的重要特征。
  - 以如今 Linux 服务器的用户规模，任何技术上的优势都是次要的。
  - 决定 Linux 作为服务器操作系统的主要因素是【网络效应】。

知乎

发现更大的世界

<https://www.zhihu.com/question/19738282>

<https://wiki.mbalib.com/wiki/网络效应>



### 3. 基于VirtualBox安装openEuler

#### 3.4 为什么使用Linux操作系统

##### □ 为什么选用openEuler?

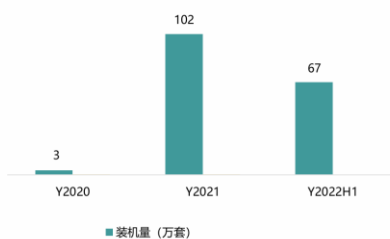


网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn> 棒棒堂智能运维 <http://www.51xueweb.cn>

#### openEuler生态现状——openEuler系产品市场高速增长

- 现阶段，随着生态伙伴对于社区版本的掌握能力增强，openEuler版本的服务器操作系统逐渐成为众多国内操作系统厂商的主力产品。同时伴随着产品在市场中的稳定表现，openEuler系产品市场装机量在2022年继续突破，上半年openEuler系产品整体装机量达67万套，其中商业发行版装机量达到33.6万套，几乎与2021年全年商业发行版装机量持平。

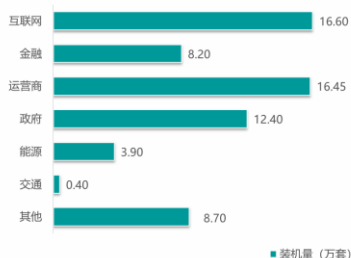
2020-2022年上半年中国openEuler系物理机装机量



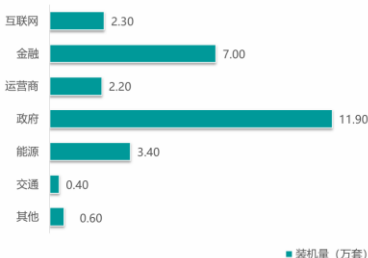
## openEuler生态现状——行业渗透较快，但集中度很高

- 2022年openEuler产品主要的行业市场中，运营商占比较高，党政机关、互联网和金融行业中渗透率可观。同时在CentOS 8停服的背景下，openEuler又在2022年3月推出新LTS版本，覆盖多个行业场景，未来潜力巨大。
- 从具体行业来看，政府、金融和能源市场以商业发行版为主，而运营商、互联网市场则以企业自用版及社区发行版为主。

2022年上半年openEuler系产品行业装机量分布



2022年上半年openEuler系商业版产品行业装机量分布



## openEuler生态现状——初具规模走向繁荣

- OpenEuler全面支持数字基础设施多样性设备，服务数字多元场景。通过开源社区的建设，openEuler将芯片厂商、整机厂商、操作系统厂商、应用软件厂商等厂商集合到社区中，成员实现共建、共享、共治，全球资源逐渐向社区汇集，从而推动了整个产业链条的发展。截至目前，欧拉社区伙伴已发展到360余家，全球下载量超过57万，PR合入超过6.5万。

openEuler生态伙伴版本汇总

| 生态伙伴  | 系统名称                                    | 自用版/商业发行版 |
|-------|-----------------------------------------|-----------|
| 百度智能云 | 百度Linux智能云操作系统                          | 自用        |
| 中国电信  | CTyunOS                                 | 自用        |
| 中国移动  | BC-Linux for Euler 20.12                | 自用        |
| 中国联通  | CULinux                                 | 自用        |
| 麒麟软件  | 银河麒麟高级服务器操作系统技术预览版                      | 商业发行      |
| 统信软件  | 统信服务器操作系统V20                            | 商业发行      |
| 麒麟信安  | 麒麟信安操作系统（服务器版）V3.5<br>麒麟信安操作系统（工控版）V3.5 | 商业发行      |
| SUSE  | SUSE Euler 2.0                          | 商业发行      |
| 超聚变   | Fusion OS 22超聚变服务器操作系统                  | 商业发行      |

| 生态伙伴   | 系统名称                         | 自用版/商业发行版 |
|--------|------------------------------|-----------|
| 普华基础软件 | 普华服务器操作系统V5.1                | 商业发行      |
| 中科院软件所 | 中科微末服务器操作系统V2.0              | 商业发行      |
| 润和软件   | HopeEdge V1.0/HopeStage V1.0 | 商业发行      |
| 新华三    | H3Linux V2.0                 | 商业发行      |
| 北京托林思  | 拓林思企业级服务器操作系统TLES16          | 商业发行      |
| 中科方德   | 方德高可信服务器操作系统V4.0（欧拉版）        | 商业发行      |
| 凝思     | 凝思安全操作系统V6.0.99              | 商业发行      |
| 红旗     | 红旗Asianux服务器操作系统（欧拉版）V8.1    | 商业发行      |
| 同源软件   | 同源OS8.1 发行版                  | 商业发行      |

- 从兼容性来看，openEuler已完成了与多个厂商操作系统的兼容，通过了CPU架构（x86/ARM）、跨代CPU、主板（USB/拓扑）、外设等方面的兼容性测试。

2022年第二季度新增openEuler整机兼容性列表

| 架构      | 硬件厂家 | 硬件型号                    | CPU         | 操作系统                    | 架构      | 硬件厂家 | 硬件型号                    | CPU         | 操作系统                    |
|---------|------|-------------------------|-------------|-------------------------|---------|------|-------------------------|-------------|-------------------------|
| aarch64 | 长城超云 | R2211 F11               | FT-2000+/B4 | openEuler 20.03 LTS SP3 | x86_64  | 超聚变  | 2288H V6                | icelake     | openEuler 20.03 LTS SP3 |
| x86_64  | 中科可控 | R6230HA                 | Hygon 2     | openEuler 20.03 LTS SP3 | x86_64  | 超聚变  | 2488H V6                | cooperlake  | openEuler 20.03 LTS SP3 |
| x86_64  | 长城超云 | R5210 G11               | cascade     | openEuler 20.03 LTS SP3 | x86_64  | 超聚变  | FusionPoD 600 DH121C V6 | icelake     | openEuler 20.03 LTS SP1 |
| x86_64  | 中兴   | R5300 G4X               | icelake     | openEuler 20.03 LTS SP3 | x86_64  | 超聚变  | 1288H V6                | icelake     | openEuler 20.03 LTS SP1 |
| aarch64 | 百信   | Hengshan TS02F-F30      | Kunpeng 920 | openEuler 22.03 LTS     | x86_64  | 超聚变  | 5288 V6                 | icelake     | openEuler 20.03 LTS SP1 |
| x86_64  | 超聚变  | FusionPoD 600 DH121C V6 | icelake     | openEuler 22.03 LTS     | x86_64  | 超聚变  | 2288H V6                | icelake     | openEuler 20.03 LTS SP1 |
| x86_64  | 超聚变  | 1288H V6                | icelake     | openEuler 22.03 LTS     | x86_64  | 超聚变  | 2488H V6                | cooperlake  | openEuler 20.03 LTS SP1 |
| x86_64  | 超聚变  | 5288 V6                 | icelake     | openEuler 22.03 LTS     | x86_64  | 超聚变  | FusionPoD 600 DH121C V6 | icelake     | openEuler 20.03 LTS     |
| x86_64  | 超聚变  | 2288H V6                | icelake     | openEuler 22.03 LTS     | x86_64  | 超聚变  | 1288H V6                | icelake     | openEuler 20.03 LTS     |
| x86_64  | 超聚变  | 2488H V6                | cooperlake  | openEuler 22.03 LTS     | x86_64  | 超聚变  | 5288 V6                 | icelake     | openEuler 20.03 LTS     |
| x86_64  | 超聚变  | FusionPoD 600 DH121C V6 | icelake     | openEuler 20.03 LTS SP3 | x86_64  | 超聚变  | 2288H V6                | icelake     | openEuler 20.03 LTS     |
| x86_64  | 超聚变  | 1288H V6                | icelake     | openEuler 20.03 LTS SP3 | aarch64 | 华为   | TaiShan 2280            | Kunpeng 916 | openEuler 22.03 LTS     |
| x86_64  | 超聚变  | 5288 V6                 | icelake     | openEuler 20.03 LTS SP3 | aarch64 | 华为   | TaiShan 2280            | Kunpeng 916 | openEuler 20.03 LTS SP3 |
| x86_64  | 超聚变  | 2288H V6                | icelake     | openEuler 20.03 LTS SP3 | x86_64  | 中科可控 | R6240H0                 | Hygon 2     | openEuler 20.03 LTS SP1 |

资料来源：openEuler社区（详见<https://www.openeuler.org/zh/compatibility/>）

### 3. 基于VirtualBox安装openEuler



#### 基于VirtualBox安装openEuler

##### 任务目标：

- 掌握Oracle VM VirtualBox的安装。
- 掌握Oracle VM VirtualBox的基本操作。
- 掌握在VirtualBox上创建虚拟机的方法。
- 掌握openEuler的安装方法和基本应用。

##### 操作演示：



##### 操作步骤：

- 获取VirtualBox安装程序
- 安装VirtualBox
- 获取openEuler
- 创建虚拟机
- 配置虚拟机
- 安装openEuler
- 初次使用openEuler





- ✓ 现场演示:
  - ✓ 基于VirtualBox安装openEuler



## 4. 使用YUM/DNF管理openEuler

### 4.1 YUM 与 DNF

- YUM (Yellow dog Updater, Modified) 是在Fedora和RedHat以及SUSE中的Shell前端软件包管理器。
  - YUM能够从指定的服务器自动下载RPM包并且安装，可以自动处理依赖性关系，并且一次安装所有依赖的软件包，无须繁琐地一次次下载、安装。
  - YUM提供了查找、安装、删除某一个、一组甚至全部软件包的命令。
- YUM语法
  - yum [options] [command] [package ...]
    - options: 可选
      - 选项包括-h (帮助)，-y (当安装过程提示选择全部为 "yes")，-q (不显示安装的过程) 等。
    - command: 要进行的操作
    - package: 安装的包名



## 4. 使用YUM/DNF管理openEuler

### 4.1 YUM 与 DNF

- YUM是C/S结构，由YUM服务器和YUM客户端两部分组成。
  - YUM服务器：
    - 所有要发行的rpm包都放在YUM服务器上以提供别人来下载，rpm包根据kernel版本号、cpu版本号分别编译发布。
    - YUM服务器通过http或ftp协议提供下载。
    - YUM服务器管理每个rpm包的基本信息，包括rpm包的版本号、conf文件、binary信息及依赖信息。
    - YUM服务器提供了createrepo工具，用于把rpm包的基本概要信息做成一张“清单”，这张“清单”就是描述每个rpm包的spec文件中信息。
  - YUM客户端：
    - YUM客户端每次调用yum install或者search的时候，都会解析/etc/yum.repos.d下面所有以.repo结尾的配置文件，这些配置文件指定了YUM服务器的地址。
    - YUM客户端定期“更新”YUM服务器上的rpm包“清单”，然后把“清单”下载保存到cache。cache的位置由/etc/yum.conf配置确定，默认是/var/cache/yum。
    - YUM调用yum install时会查询cache目录下的“清单”，根据“清单”里的rpm包描述来确定安装包的名字、版本号、所需要的依赖包等，然后再去yum服务器下载对应的rpm包安装。



## 4. 使用YUM/DNF管理openEuler

### 4.1 YUM 与 DNF

- DNF (Dandified YUM) 是在YUM基础上改进的软件包管理器，同时继承了YUM的特性。
  - 虽然DNF在许多方面优于YUM，但YUM在一些旧的或者不完全支持DNF的环境中更为适用。
  - DNF与YUM的主要区别：
    - 依赖解决算法
      - DNF使用libsolv库进行依赖解析，是一个独立的依赖解析器，可以有效地解决复杂的依赖关系，YUM基于PCL算法进行依赖解析。
    - 性能
      - DNF支持并行操作，可以同时下载多个软件包；
      - YUM是单线程的，每次只能下载一个软件包。
    - 配置文件
      - DNF使用以.repo为后缀的配置文件；
      - YUM使用以.repo为后缀的配置文件和/etc/yum.conf的主配置文件。
    - 用户界面
      - DNF输出信息更加详细；YUM输出信息相对简单。



## 4. 使用YUM/DNF管理openEuler

### 4.2 任务



### 使用YUM管理软件包

#### 任务目标：

- 掌握YUM的使用。
- 掌握使用YUM管理软件包。
- 掌握使用YUM管理软件包组。

#### 操作演示：

Code

#### 操作步骤：

- 管理软件包
  - 搜索软件包
  - 输出软件包列表
  - 查看RPM包信息
  - 安装、下载、卸载软件包
- 管理软件包组
  - 输出软件包组列表
  - 显示软件包组信息
  - 安装、卸载软件包组

网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn>

棒棒堂智能运维 <http://www.51xueweb.cn>



## 4. 使用YUM/DNF管理openEuler

### 4.2 任务



### 使用YUM管理软件包

#### 操作命令与脚本程序：

- yum search keywords: 搜索软件包
- yum list all:
  - 输出所有软件包的信息
- yum list glob\_expression
  - 输出系统中特定的软件包信息
- yum info package\_name
  - 查看RPM包信息
- yum install package\_name: 安装软件包
- yum download package\_name: 下载软件包
- yum remove packages\_name: 卸载软件包
- yum group summary
  - 输出软件包组列表
- yum group list
  - 输出系统中所有软件包组及其组ID
- yum group info glob\_expression
  - 显示软件包组信息
- yum group install group\_name
  - 安装软件包组
- yum group remove group\_name
  - 卸载软件包组

网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn>

棒棒堂智能运维 <http://www.51xueweb.cn>



## 4. 使用YUM/DNF管理openEuler

### 4.3 任务



### 使用YUM进行系统升级更新

#### 任务目标:

- 掌握YUM的使用。
- 掌握使用YUM进行系统升级。

#### 操作步骤:

- 检查更新
- 升级软件
- 更新所有软件及依赖

#### 操作演示:

Code



## 4. 使用YUM/DNF管理openEuler

### 4.3 任务



### 使用YUM进行系统升级更新

#### 操作命令与脚本程序:

- yum update package\_name
  - 升级单个软件包
- yum group update group\_name
  - 升级软件包组
- yum update
  - 更新所有软件包及其依赖的软件包
- yum check-update
  - 列出所有可更新的软件清单命令
- 清除缓存
  - yum clean packages:
    - 清除缓存目录下的软件包
  - yum clean headers:
    - 清除缓存目录下的 headers
  - yum clean oldheaders:
    - 清除缓存目录下旧的 headers
  - yum clean, yum clean all
    - = yum clean packages + oldheaders
    - 清除缓存目录下的软件包及旧的 headers



## 5. 远程管理openEuler

### 5.1 什么是远程管理?

- 远程管理是指在网络上由一台计算机（主控端）远距离控制另一台计算机（被控端）的技术。
  - 远程不是字面意思的远距离，一般指通过网络控制远端计算机。
  - 当操作者使用主控端计算机控制被控端计算机时，就如同坐在被控端计算机的屏幕前一样，可启动被控端计算机的应用程序，可使用被控端计算机的文件数据，甚至可以利用被控端计算机的外部打印设备和通信设备来进行打印和访问互联网。
  - 如果是个人计算机，就没有远程管理的概念，想用的时候直接开机，而对于服务器来说，远程管理就变的十分重要，服务器一般放置在数据中心的，对服务器进行管理的主要途径就是通过远程方式。



## 5. 远程管理openEuler

### 5.1 什么是远程管理?

- Linux常用的远程管理协议有Telnet、SSH、VNC。
  - Telnet:
    - 所有数据在网络上都是明文传输，比如用户在登录服务器时输入的用户名和密码，因为在网络中以明文方式传输，存在一定的安全隐患。
  - SSH:
    - 全称是Secure Shell（安全Shell），使用SSH可将客户机与远程服务器之间的通信数据进行加密，从而提高远程登录的安全性。
    - 目前UNIX/Linux操作系统最常用的远程管理方式。
  - VNC:
    - 可在本机计算机以图形的方式显示远程服务器的图形界面。
    - 进行VNC登录时，既可使用专门的客户端软件，也可使用浏览器进行登录。



## 5. 远程管理openEuler

- 通过SSH协议进行远程管理是C/S结构，需要：
  - 受控端（例如安装Linux操作系统的服务器）安装支持SSH协议的服务器端软件
  - 主控端（例如用于远程管理的Windows计算机）安装支持SSH协议的客户端软件
- OpenSSH是Linux操作系统广泛使用的支持SSH协议的服务器端软件。
  - OpenSSH是使用SSH协议进行远程登录的首要连接工具。
  - OpenSSH服务端由 `sshd`、`sftp-server` 和 `ssh-agent` 三个软件组成。
  - 远程操作通过 `ssh`、`scp` 和 `sftp` 三种协议进行。
  - OpenSSH支持SSH协议的1.3、1.5和2版本。



网络与信息系统智能运维课程体系 <https://internet.hactcm.edu.cn>

棒棒堂智能运维 <http://www.51xueweb.cn>



OpenSSH 9.8 released July 1, 2024

OpenSSH is the premier connectivity tool for remote login with the SSH protocol. It encrypts all traffic to eliminate eavesdropping, connection hijacking, and other attacks. In addition, OpenSSH provides a large suite of secure tunneling capabilities, several authentication methods, and sophisticated configuration options.

The OpenSSH suite consists of the following tools:

- Remote operations are done using `ssh`, `sftp`, and `scp`.
- Key management with `ssh-add`, `ssh-keygen`, `ssh-keyscan`, and `ssh-keysign`.
- The service side consists of `sshd`, `ssh-agent`, and `ssh-keyexch`.

OpenSSH is developed by a few developers of the [OpenBSD Project](#) and made available under a BSD-style license.

OpenSSH is incorporated into many commercial products, but very few of those companies assist OpenSSH with funding.

Contributions towards OpenSSH can be sent to the [OpenBSD Foundation](#).

### OpenSSH Manual Pages

Web manual pages are available from OpenBSD for the following commands. These manual pages reflect the latest development release of OpenSSH.

- `ssh(1)` — The basic rlogin-like client program
- `ssh-add(1)` — The daemon that permits you to log in
- `ssh_config(5)` — The client configuration file
- `sshd_config(5)` — The daemon configuration file
- `ssh-agent(1)` — An authentication agent that can store private keys
- `ssh-add(1)` — Tool which adds keys to the above agent
- `ssh(1)` — FTP-like program that works over SSH and SFTP protocol
- `scp(1)` — File copy program that acts like `cp`
- `ssh-keygen(1)` — Key generation tool
- `ssh-keyscan(1)` — SFTP server subsystem (started automatically by `sshd`)
- `ssh-keysign(8)` — Utility for gathering public host keys from a number of hosts
- `ssh-keyexch(8)` — Helper program for host-based authentication

The SFTP protocol implemented in OpenSSH is standardized by the IETF [sftp](#) working group and is specified in several RFCs and drafts. The overall structure of SFTP is described in the [architecture](#) RFC. It is composed of three layered components:

- The [transport layer](#) provides algorithm negotiation and a key exchange. The key exchange includes server authentication and results in a cryptographically secured connection. It provides integrity, confidentiality and optional compression.
- The user [authentication layer](#) uses the established connection and relies on the services provided by the transport layer. It provides several mechanisms for user authentication. These include traditional password authentication as well as public key or host-based authentication mechanisms.
- The [connection layer](#) multiplexes many different concurrent channels over the authenticated connection and allows tunneling of login sessions and TCP forwarding. It provides a flow control service for these channels. Additionally, various channel-specific options can be negotiated.

Additional documents specify:

- The [interactive authentication](#) RFC provides support for new authentication schemes like S/Key or TIS authentication.
- The SFTP file transfer protocol is specified in the [draft](#) draft. OpenSSH implements a SFTP `client` and `server`.
- A file format for public keys is specified in the [publickey](#) draft. The command `ssh-keygen(1)` can be used to convert an OpenSSH public key to this file format.
- The [Diffie-Hellman Group Exchange](#) allows clients to request more secure groups for the Diffie-Hellman key exchange.
- OpenSSH implemented a compression method `"zlib@openssh.com"` that delays starting compression until after user authentication, to eliminate the risk of pre-authentication attacks against the compression code. It is described in [draft-miller.ssh.compression-delayed-00.txt](#).
- OpenSSH implements an additional MAC (Message Authentication Code) `"umac-64@openssh.com"`, which has superior performance to the ones specified in RFC 4253. It is described in [draft-miller.ssh.umac-01.txt](#).
- The authentication agent protocol used by `ssh-agent` is documented in the [PROTOCOL.agent](#) file.
- OpenSSH makes various other minor extensions to and divergences from the standard SSH protocols. These are documented in the [PROTOCOL](#) file.

## 5. 远程管理openEuler

- 支持SSH协议的客户端软件有：
  - Windows平台：
    - Putty、Termius、XShell、Bitvise SSH、MobaXterm
    - DameWare SSH、SmarTTY、Cygwin
  - Mac平台：
    - OpenSSH Client、Shuttle、Secure Shell、Termius
  - Android平台：
    - JuiceSSH、Termius、Android Terminal Emulator
  - iOS平台：
    - Prompt、Termius、iSSH



## 5. 远程管理openEuler

- MobaXterm (Moba + X + Terminal)
  - MobaXterm 是一款功能强大的终端软件，集成了X服务器和许多网络工具，实现便捷的远程连接和管理功能。
  - MobaXterm 具有以下特点：
    - 丰富的连接方式：
      - 支持 SSH、Telnet、RDP 等多种协议，可以连接到不同类型的远程服务器。
    - 内置 X 服务器：
      - 允许在远程服务器上运行图形化应用程序，并将其显示在本地。
    - 多标签界面：
      - 方便用户同时管理多个远程连接，提高工作效率。
    - 文件传输功能：
      - 可以通过 SFTP、FTP 等协议进行文件的上传和下载。
    - 终端功能强大：
      - 提供了命令行历史记录、自动补全、语法高亮等功能，方便用户在终端中进行操作。



## 5. 远程管理openEuler

### □ MobaXterm (Moba + X + Terminal)

- MobaXterm 是一款功能强大的终端软件，集成了X服务器和许多网络工具，实现便捷的远程连接和管理功能。
- MobaXterm 具有以下特点：
  - 丰富的连接方式：
    - 支持 SSH、Telnet、RDP 等多种协议，可以连接到不同类型的远程服务器。
  - 内置 X 服务器：
    - 允许在远程服务器上运行图形化应用程序，并将其显示在本地。
  - 多标签界面：
    - 方便用户同时管理多个远程连接，提高工作效率。
  - 文件传输功能：
    - 可以通过 SFTP、FTP 等协议进行文件的上传和下载。
  - 终端功能强大：
    - 提供了命令行历史记录、自动补全、语法高亮等功能，方便用户在终端中进行操作。

**X server:**  
X 服务器是一种在图形用户界面 (GUI) 环境中负责显示图形和处理用户输入的软件组件。  
通常在类 Unix 操作系统中使用，允许远程客户端连接并在本地显示图形界面。



## 5. 远程管理openEuler



### 通过SSH远程管理openEuler

#### 任务目标：

- 掌握sshd服务的安装。
- 掌握MobaXterm远程管理openEuler。

#### 操作演示：



#### 操作步骤：

- 在openEuler上安装sshd服务
- sshd服务准备
  - 服务启动
  - 服务运行正常
  - 防火墙允许远程访问
- 安装MobaXterm
- 使用MobaXterm实现远程连接
  - 服务器地址和端口
  - 远程管理的账户权限





#### ✓ 现场演示:

- ✓ 基于VirtualBox安装openEuler
- ✓ 配置openEuler的sshd服务及FW
- ✓ 安装并使用MobaXterm



## 6. 电源管理

- openEuler中通过systemctl命令进行系统电源管理，实现系统重启、休眠、关机操作。
  - 当前仍兼容部分Linux中的管理命令。
  - 建议用户使用systemctl 命令来进行管理。

| 序号 | Linux 管理命令 | systemctl 命令       | 命令描述     |
|----|------------|--------------------|----------|
| 1  | reboot     | systemctl reboot   | 重启系统     |
| 2  | halt       | systemctl halt     | 关闭系统但不下电 |
| 3  | poweroff   | systemctl poweroff | 关闭系统并下电  |

推荐使用



## 网络与信息系统智能运维 课程体系学习平台

\*\*\*

本课程体系由  
河南中医药大学信息技术学院建设

课程体系学习平台由河南中医药大学医疗健康信息  
工程技术研究所开发与技术保障

网络与信息系统智能运维课程体系学习平台  
<https://internet.hactcm.edu.cn>

互联网运维管理工程应用丛书  
<http://www.51xueweb.cn>

扫码学习  
并获取课程资源

