

Linux软件管理

1

作者：牟建波

2

时间：2025-01-07

3

描述：日常学习笔记

1.包管理器分类

在 Linux 中，软件管理是系统运维的重要部分，主要涉及 **包管理器**（Package Manager）、**软件包操作**、**依赖管理**等。不同的 Linux 发行版使用不同的包管理工具：

发行版	包管理器	包格式
Debian、Ubuntu	apt、dpkg	.deb
CentOS、RHEL、Fedora	dnf、yum、rpm	.rpm
Arch Linux	pacman	.tar.zst
openSUSE	zypper	.rpm

2.Ubuntu包管理器

2.1 rpm包管理器

```
1  # 更新和升级
2  sudo apt update           # 更新软件源信息
3  sudo apt upgrade         # 更新已安装软件
4  sudo apt full-upgrade    # 更新软件，同时处理依赖变更
5
6  # 查找软件
7  sudo apt search package_name # 搜索软件
8  sudo apt show package_name  # 显示软件详细信息
9
10 # 安装软件
11 sudo apt install package_name # 安装软件包
12
```

```
13  # 卸载软件
14  sudo apt remove package_name      # 删除软件，并保留配置文件
15  sudo apt purge package_name       # 删除软件，并清理配置文件
16
17  # 清除无用的软件包
18  sudo apt autoremove               # 移除不需要的依赖包
19  sudo apt clean                    # 清理本地缓存的软件包
```

2.2 dpkg包管理器(基本不用)

```
1  # 安装.deb软件包
2  sudo dpkg -i package_name
3
4  # 卸载.deb软件包
5  sudo dpkg -r package_name      # 只删除软件
6  sudo dpkg -P package_name     # 删除软件及其配置文件
7
8  # 查询已安装的软件
9  sudo dpkg -l | grep package_name
```

3.CentOS/Rocky包管理器

3.1 dnf包管理器(新版)

```
1  # 安装、卸载、更新
2  sudo dnf update                # 更新
3  sudo dnf install package_name # 安装
4  sudo dnf remove package_name  # 卸载
5
6  # 搜索和查询
7  sudo dnf search package_name  # 搜索软件
8  sudo dnf info package_name    # 查询软件详细信息
9
10 # 清理缓存
11 sudo dnf clean all
```

3.2 yum包管理器(旧版)

```
1  # 说明: dnf代替了yum, 比yum更快更稳定
2
3  # 安装、卸载、更新
4  sudo yum update           # 更新
5  sudo yum install package_name # 安装
6  sudo yum remove  package_name # 卸载
7
8  # 搜索和查询
9  sudo yum search package_name # 搜索软件
10 sudo yum info  package_name  # 查询软件详细信息
11
12 # 清理缓存
13 sudo yum clean all
```

3.3 rpm包管理器(基本不用)

```
1  # 安装.rpm包
2  sudo rpm -ivh package.rpm
3
4  # 卸载.rpm包
5  sudo rpm -e package_name
6
7  # 查询已安装的软件
8  sudo rpm -qa | grep package_name
```

4.程序包的编译过程

☁ 步骤一：安装必要软件，例如： `gcc make cmake autoconf pkg-config` 等

- Ubuntu: `sudo apt install build-essential cmake pkg-config`
- CentOS: `sudo dnf install cmake pkg-config`

步骤二：下载源码

步骤三：配置编译环境

- `./configure --prefix=/usr/local`
- 若缺少依赖: `sudo apt install libssl-dev`

步骤四：编译安装

- `sudo make` , 编译
- `sudo make install` , 安装

```
1  # 示例: 编译安装 Nginx
2
3  # 1. 安装编译工具
4  sudo apt install build-essential libpcre3-dev zlib1g-dev libssl-dev
5
6  # 2. 下载 Nginx 源码
7  git clone http://nginx.org/download/nginx-1.24.0.tar.gz
8  tar -xvf nginx-1.24.0.tar.gz
9  cd nginx-1.24.0
10
11 # 3. 配置编译环境
12 ./configure --prefix=/usr/local/nginx --with-http_ssl_module
13
14 # 4. 编译
15 make
16
17 # 5. 安装
18 sudo make install
19
20 # 6. 验证安装
21 /usr/local/nginx/sbin/nginx -v
```