

第4部分

软件与网络管理

2024年6月30日

软件与网络管理

- 软件管理
- 网络管理

软件包的封装类型

■ 常见的软件包封装类型

软件管理

- 使用YUM
- RPM包管理
- 源代码安装

YUM

- Yum (Yellow dog Updater, Modified) 是一个在Linux系统中**基于Shell模式的软件包管理器**。
- 能够从**指定的服务器**自动下载**RPM包**并安装，可以处理依赖性关系，并且一次安装所有依赖的软件包。
- 如果同一软件存在多个可用的更新版本，Yum能够自动选择最新的软件版本。

YUM

■ yum命令的语法格式如下：

yum [选项] [命令] packages

■ 基本 yum 命令

- 1.yum help 显示使用信息
- 2.yum list 列出软件包
- 3.yum search keyword 搜索关键字
- 4.yum info packagename 列出软件包详细信息
- 5.yum install packagename 安装软件包
- 6.yum remove packagename 删除软件包
- 7.yum update packagename 升级软件包

使用yum安装软件

```
[root@localhost ~]# yum install php
```

```
.....
依赖关系解决
```

Package	架构	版本	仓库	大小
正在安装：				
php	i686	5.4.1-1.fc17	fedora	2.6 M
为依赖而安装：				
apr	i686	1.4.6-1.fc17	fedora	105 k
apr-util	i686	1.4.1-2.fc17	fedora	80 k
apr-util-ldap	i686	1.4.1-2.fc17	fedora	17 k
httpd	i686	2.2.22-4.fc17	fedora	825 k
httpd-tools	i686	2.2.22-4.fc17	fedora	75 k
libzip	i686	0.10.1-1.fc17	fedora	46 k
php-cli	i686	5.4.1-1.fc17	fedora	2.5 M
php-common	i686	5.4.1-1.fc17	fedora	805 k

事务概要

安装 1 软件包 (+8 Dependent packages)

总下载量：7.0 M

安装大小：25 M

确定吗？[y/N]：

- 有些软件需要多个软件配合才能使用，涉及**一系列软件包**，而这些软件包又互相依赖，用户在安装时就会产生困扰。
- yum提供了命令来一次性完成整个软件组的安装。

使用yum安装软件组

■ 基本命令

- 1.yum grouplist 列出所有可用组
- 2.yum groupinfo **group1**提供特定组的信息
- 3.yum groupinstall **group1**安装软件包组
- 4.yum groupupdate **group1**更新软件包组
- 5.yum groupremove **group1** 删除软件包组

使用yum安装软件组

```
[root@localhost ~]# yum groupinstall 'Web 服务器'
```

```
.....
```

依赖关系解决

Package	架构	版本	仓库	大小
正在安装：				
crypto-utils	i686	2.4.1-34.fc17	fedora	72 k
distcache	i686	1.4.5-23	fedora	114 k
httpd	i686	2.2.22-4.fc17	fedora	825 k
httpd-manual	noarch	2.2.22-4.fc17	fedora	878 k
mod_python	i686	3.3.1-17.fc17	fedora	282 k
mod_ssl	i686	1:2.2.22-4.fc17	fedora	88 k
php	i686	5.4.1-1.fc17	fedora	2.6 M
php-ldap	i686	5.4.1-1.fc17	fedora	46 k
php-mysql	i686	5.4.1-1.fc17	fedora	116 k
webalizer	i686	2.23_05-3	fedora	126 k
为依赖而安装：				
apr	i686	1.4.6-1.fc17	fedora	105 k
apr-util	i686	1.4.1-2.fc17	fedora	80 k
apr-util-ldap	i686	1.4.1-2.fc17	fedora	17 k
httpd-tools	i686	2.2.22-4.fc17	fedora	75 k
libzip	i686	0.10.1-1.fc17	fedora	46 k
perl-Newt	i686	1.08-31.fc17	fedora	64 k
php-cli	i686	5.4.1-1.fc17	fedora	2.5 M
php-common	i686	5.4.1-1.fc17	fedora	805 k
php-pdo	i686	5.4.1-1.fc17	fedora	112 k

```
.....
```



安装本地软件包

■ 使用yum安装本地软件包文件

- yum install packagefile.rpm

```
[root@localhost ~]# yum install wps-office-8.1.0.3724-0.1.blp2.i686.rpm
已加载插件：downloadonly, fastestmirror, langpacks, presto, refresh-packagekit
诊断 wps-office-8.1.0.3724-0.1.blp2.i686.rpm: wps-office-8.1.0.3724-0.1.blp2.i686
wps-office-8.1.0.3724-0.1.blp2.i686.rpm 将被安装
解决依赖关系
--> 执行事务检查
---> 软件包 wps-office.i686.0.8.1.0.3724-0.1.blp2 将被 安装
--> 处理依赖关系 libpng12.so.0, 它被软件包 wps-office-8.1.0.3724-0.1.blp2.i686 需要
Loading mirror speeds from cached hostfile
* fedora: ftp-stud.hs-esslingen.de
--> 执行事务检查
---> 软件包 libpng-compat.i686.2.1.5.10-1.fc17 将被 安装
--> 完成依赖关系计算
```

依赖关系解决

Package	架构	版本	仓库	大小
正在安装:				
wps-office	i686	8.1.0.3724-0.1.blp2	/wps-office-8.1.0.3724-0.1.blp2.i686	348 M
为依赖而安装:				
libpng-compat	i686	2:1.5.10-1.fc17	fedora	94 k

事务概要

安装 1 软件包 (+1 Dependent package)

总文件大小: 348 M

总下载量: 94 k

安装大小: 348 M

确定吗? [Y/N]:



软件包更新及删除

■ 软件包更新命令

- yum update packagefile
- yum groupupdate 'packagefile'
- 如果没有指定软件包名，则会更新整个系统

■ 软件包删除命令

- yum remove packagefile
- 在删除软件包时，yum会检索依赖于该软件包的其他软件包。如果存在，yum将会列出需要删除的软件包名
- 用户要谨慎执行删除操作，以防止由于依赖关系而误删其它软件。

RPM软件包管理

■ 命令：rpm

■ 主要功能

- 1. 查询RPM软件、包文件的相关信息
- 2. 安装、升级、卸载RPM软件包
- 3. 维护RPM数据库信息

查询RPM软件信息

■ 查询已安装的RPM软件信息

- 格式：rpm -q[子选项] [软件名]
- 用法：结合不同的子选项 完成不同查询
 - qa: 查看系统中已安装的所有RPM软件包列表
 - qi: 查看指定软件的详细信息
 - ql: 查询指定软件包所安装的目录、文件列表
 - qc: 仅显示指定软件包安装的配置文件
 - qd: 仅显示指定软件包安装的文档文件

■ 查询文件/目录属于哪个RPM软件

- 格式：rpm -qf 文件或目录名

查询RPM软件信息

```
[root@localhost ~]# rpm -q bash
```

```
bash-3.1-16.1
```

```
[root@localhost ~]# rpm -qa | grep bash
```

```
bash-3.1-16.1
```

查询是否已安装有
bash软件包

```
[root@localhost ~]# rpm -qf /usr/bin/vim
```

```
vim-enhanced-7.0.109-3
```

查询vim是由哪个
软件包安装的

```
[root@localhost ~]# rpm -ql dhcp
```

```
/etc/dhcpd.conf
```

```
/etc/rc.d/init.d/dhcpd
```

```
.....
```

查询dhcp软件包
安装的文件列表

查询RPM软件信息

■ 查询未安装的RPM包文件

- 格式：rpm -qp[子选项] RPM包文件
- 用法：结合不同的子选项 完成不同查询
 - qpi: 通过.rpm包文件查看该软件的详细信息
 - qpl: 查看.rpm安装包内所包含的目录、文件列表
 - qpc: 查看.rpm安装包内包含的配置文件列表
 - qpd: 查看.rpm安装包内包含的文档文件列表

查询RPM软件信息

```
[root@localhost ~]# rpm -qpi lynx-2.8.5-28.1.i386.rpm
```

```
warning: lynx-2.8.5-28.1.i386.rpm: Header V3 DSA signature: NOKEY,  
key ID 37017186
```

```
Name       : lynx       Relocations: (not relocatable)
```

```
Version    : 2.8.5      Vendor: Red Hat, Inc.
```

```
Release    : 28.1       Build Date: 2006年07月13日 星期四 14时10分20秒
```

```
.....
```

查询lynx软件安装包的摘要信息

```
[root@localhost Server]# rpm -qpl lynx-2.8.5-28.1.i386.rpm
```

```
/etc/lynx-site.cfg
```

```
/etc/lynx.cfg
```

```
.....
```

查询lynx安装包中的文件列表

安装、升级、卸载RPM软件包

■ 安装或升级RPM软件

- 格式：rpm [选项] RPM包文件...
- 用法：不同选项适用于不同情况
 - **-i**：安装一个新的rpm软件包
 - **-U**：升级某个rpm软件，若原本未装，则进行安装
 - **-F**：更新某个rpm软件，若原本未装，则放弃安装

■ 卸载指定的RPM软件

- 格式：rpm -e RPM软件名

安装、升级、卸载RPM软件包

■ 辅助选项

- --force: 强制安装所指定的rpm软件包
- --nodeps: 安装、升级或卸载软件时, 忽略依赖关系
- -h: 以“#”号显示安装的进度
- -v: 显示安装过程中的详细信息

安装、升级、卸载RPM软件包

```
[root@localhost ~]# rpm -ivh lynx-2.8.5-28.1.i386.rpm
```

```
warning: lynx-2.8.5-28.1.i386.rpm: Header V3 DSA signature: NOKEY,  
key ID 37017186
```

```
Preparing... #####  
[100%]
```

```
1:lynx #####  
[100%]
```

安装lynx软件包同时显示安装进度

```
[root@localhost ~]# rpm -q lynx
```

```
lynx-2.8.5-28.1
```

```
[root@localhost ~]# rpm -e lynx
```

```
[root@localhost ~]# rpm -q lynx
```

```
package lynx is not installed
```

卸载lynx软件包

安装、升级、卸载RPM软件包

```
[root@localhost ~]# rpm -i wget-1.10.2-3.2.1.i386.rpm
```

```
[root@localhost ~]# rpm -q wget
```

```
wget-1.10.2-3.2.1
```

```
[root@mail ~]# rpm -Uvh /media/cdrom/wget-1.10.2-7.el5.i386.rpm
```

```
warning: /media/cdrom/wget-1.10.2-7.el5.i386.rpm: Header V3 DSA  
signature: NOKEY, key ID 37017186
```

```
Preparing... ##### [100%]
```

```
1:wget ##### [100%]
```

```
[root@localhost ~]# rpm -q wget
```

```
wget-1.10.2-7.el5
```

升级wget软件包

源代码安装

- 与RPM安装方式相比，使用**源代码**进行软件安装会复杂一些，但是用源代码安装软件是Linux下进行软件安装的重要手段，也是运行Linux的最主要的优势之一。

源代码安装

■ 使用源代码安装软件的优点

- 获得最新的软件版本，及时修复bug
- 根据用户需要，灵活定制软件功能

■ 应用场合

- 安装较新版本的应用程序时
 - 自由软件的最新版本大都以源码的形式最先发布
- 当前安装的程序无法满足需要时，需要为应用程序添加新的功能时
 - 编译安装可由用户自行修改、定制功能或加入新的功能

源代码安装

■ Tarball 封包:

- .tar.gz 和 .tar.bz2 格式居多
- 软件素材参考: <http://sourceforge.net>

■ 完整性校验

- md5sum 校验工具
 - 计算MD5校验和, 并与官方提供的值相比较, 判断是否一致

```
[root@localhost ~]# md5sum axel-1.0a.tar.gz  
2d94c0b36b374834567f1fcec5f89119 axel-1.0a.tar.gz
```

MD5校验和

源代码安装

■ 确认源代码编译环境

- 需安装支持 C/C++ 程序语言的编译器
 - gcc-4.1.1-52.el5、gcc-c++-4.1.1-52.el5
 - **make**-3.81-1.1.i386

```
[root@localhost ~]# gcc --version
```

```
gcc (GCC) 4.1.1 20070105 (Red Hat 4.1.1-52)
```

```
Copyright (C) 2006 Free Software Foundation, Inc.
```

本程序是自由软件；请参看源代码的版权声明。本软件没有任何担保；
包括没有适销性和某一专用目的下的适用性担保。

```
[root@localhost ~]# g++ --version
```

```
g++ (GCC) 4.1.1 20070105 (Red Hat 4.1.1-52)
```

```
.....
```

源码编译安装的基本过程

- **解包 —— tar**
 - 解包、释放出源代码文件
- **配置 —— ./configure**
 - 针对当前系统、软件环境，配置好安装参数
- **编译 —— make**
 - 将源代码文件变为二进制的可执行程序
- **安装 —— make install**
 - 将编译好的程序文件复制到系统中

源码编译安装的基本过程

■ 1. 解包

- 习惯上将软件包释放到 `/usr/src/` 目录
- 解包后的源代码文件位置：
 - `/usr/src/软件名-版本号/`

```
[root@localhost ~]# tar zxvf axel-1.0a.tar.gz -C /usr/src/
```

```
[root@localhost ~]# ls /usr/src/axel-1.0a/
```

```
API    axelrc.example  configure  CREDITS  gui    nl.po    tcp.c
axel.1  CHANGES      http.c    de.po    http.c  README  tcp.h
axel.c  http.h        search.c  ftp.c    http.h  search.c
axel.h  conf.h        COPYING  ftp.h    Makefile  se
```

configure 执行脚本
用于下一步的配置

README 或 INSTALL
文件可提供安装方法

源码编译安装的基本过程

■ 2. 配置

- 需将工作目录切换到软件源码所在目录
- 使用源码目录中的 `configure` 脚本
- 执行“`./configure --help`” 可以查看帮助
- 典型的配置选项：
 - `--prefix=软件安装目录` // 配置了安装路径

```
[root@localhost ~]# cd /usr/src/axel-1.0a
```

```
[root@localhost axel-1.0a]# ./configure --prefix=/usr/local/axel
```

若不指定任何配置选项，将采用默认值

源码编译安装的基本过程

■ 3. 编译

- 执行 **make** 命令

■ 4. 安装

- 执行 **make install** 命令

```
[root@localhost axel-1.0a]# make
```

```
[root@localhost axel-1.0a]# make install
```

这两个步骤可以合写为一行：
make && make install

源码编译安装的基本过程

■ 验证已编译安装完成的程序

- 查看安装后的软件目录、了解软件程序的使用方法

```
[root@localhost axel-1.0a]# ls /usr/local/axel/ /usr/local/axel/bin/
```

```
/usr/local/axel/:
```

```
bin etc share
```

```
/usr/local/axel/bin/:
```

```
axel
```

```
[root@localhost axel-1.0a]# /usr/local/axel/bin/axel -help
```

```
Usage: axel [options] url1 [url2] [url...]
```

```
--max-speed=x    -s x    Specify maximum speed (bytes per second)
```

```
.....
```

网络管理

- 网络配置基础
- 网络的基本配置
- 系统网络配置文件
- 网络设置工具

网络信息查看

- Linux系统中的普通用户可以使用命令查看系统的网络属性信息
 - 查看网络接口信息
 - 测试网络连接状态
 - 查看网关地址和路由信息
 - 查看主机名称信息

查看网络接口信息

■ **ifconfig**命令用于查看网络接口信息

- 查看所有**活动**网络接口的信息

```
$ /sbin/ifconfig
```

- 查看所有网络接口（包括非活动网络接口）的信息

```
$ /sbin/ifconfig -a
```

- 查看指定网络接口信息

```
$ /sbin/ifconfig eth0
```

命令名，普通用户执行需输入命令的全路径名

命令选项，要显示的网络接口名称

查看主机路由信息

■ route命令可查看主机的路由信息

\$ /sbin/route

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
192.168.152.0	*	255.255.255.0	U	0	0	0	eth0
169.254.0.0	*	255.255.0.0	U	0	0	0	eth0
default	<u>192.168.152.2</u>	0.0.0.0	UG	0	0	0	eth0

缺省网关地址

使用ping命令测试网络连接

使用“Ctrl + c”组合键
结束ping命令执行

- ping命令用于测试当前主机与其他

```
$ ping 192.168.152.131
```

```
PING 192.168.152.131 (192.168.152.131) 56(84) bytes of data.
```

```
64 bytes from 192.168.152.131: icmp_seq=1 ttl=64 time=0.057 ms
```

```
64 bytes from 192.168.152.131: icmp_seq=2 ttl=64 time=0.584 ms
```

命令名

测试连接的目的主机地址

```
--- 192.168.152.131 ping statistics ---
```

```
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
```

```
rtt min/avg/max/mdev = 0.057/0.584/1.112/0.528 ms, pipe 2
```

ping命令结束执行，显示统计信息

指定ping命令的测试数据包

■ ping命令可以指定发送的测试数据包的数量

```
$ ping -c 4 192.168.152.2
```

```
PING 192.168.152.2 (192.168.152.2) 56(84) bytes of data.
```

```
64 bytes from 192.168.152.2: icmp_seq=0 ttl=64 time=0.361 ms
```

命令名

命令选项

测试连接的目的
主机地址

```
64 bytes from 192.168.152.2: icmp_seq=1 ttl=64 time=0.798 ms
```

```
64 bytes from 192.168.152.2: icmp_seq=2 ttl=128 time=0.361 ms
```

```
64 bytes from 192.168.152.2: icmp_seq=3 ttl=128 time=1.30 ms
```

-c表示指定测试数
据包的数量

4是为-c选项设置
的值，即发送4各
测试数据包

```
--- 192.168.152.2 ping statistics ---
```

```
4 packets transmitted, 4 received, 0% packet loss, time 3000ms
```

```
rtt min/avg/max/mdev = 0.361/0.798/1.306/0.438 ms, pipe 2
```

ping命令发送指定的数据包后自动
结束执行，显示统计信息

测试与其他主机的网络连接路径

- traceroute命令测试当前主机到目的主机之间经过的所有网络节点

```
$ traceroute 192.168.1.1
```

```
traceroute to 192.168.1.1 (192.168.1.1), 30 hops max, 38 byte packets
```

```
1 192.168.152.2 93 ms 1.300
```

```
2 192.168.1.1 (192.168.1.1) 4.821 ms 3.518 ms
```

命令名

测试的目标主机地址
作为命令参数

当前主机到目的主机经过的网络中间节点

查看当前主机名称

- **hostname**命令可显示当前主机的名称

```
$ hostname
```

```
localhost.localdomain
```

设置主机名称

■ hostname命令可以对当前主机的名称进行设置

- hostname命令设置的主机名称是即时生效的
- 系统重新启动后将会恢复为系统原有的主机名称

hostname Kylinserver

命令名

设置的主机名称作为命令参数

使用命令进行网络设置

- 使用命令配置网络属性可以即时生效，但主机重新启动后将丢失配置
 - 网络设置的两种方法
 - ip地址配置命令
 - 路由配置命令
 - 主机名称配置命令
 - DNS域名解析

网络设置的两种方法

■ DHCP网络配置

- Linux主机通过网络中已有的DHCP服务器获取必要的设置信息进行网络配置
- 网络客户端的配置简单，需要在DHCP服务器端进行正确的配置

■ 手工网络配置

- 在每台主机中独立配置网络属性
- 主要应用于网络服务器主机

IP地址配置命令

- ifconfig命令可以设置网络接口的ip地址

```
# ifconfig eth0 192.168.152.133 \  
> netmask 255.255.255.0
```

命令名

设置网络掩码的
命令选项

被设置的网络
接口名

网络掩码的设置值

网络接口
ip地址

命令续行
符号

路由配置命令

■ route命令可以维护系统路由表信息

- 删除缺省网关

```
# route del default gw 192.168.152.2
```

- 添加缺省网关

```
# rou 命令名 add表示 default gw 缺省网关的地址 192.168.152.2
```

add表示
路由表

default gw 缺省网关的地址

del表示删除路
由表记录

修改配置文件进行网络设置

- 通过**配置文件**进行网络配置可以使配置持久有效
 - 使用工具进行网络配置
 - 网络服务启动脚本
 - 网络接口配置文件
 - 主机名称配置文件
 - 本地主机名称解析文件
 - 域名服务器配置文件

网络服务启动脚本

- “network”服务程序用于对网络属性进行配置生效

- 重新启动network服务

```
# service network restart
```

网络接口配置文件

■ 网络接口的文件位于目录“/etc/sysconfig/network-scripts/”中

- 网络接口eth0的配置文件是ifcfg-eth0

```
# cat /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
DEVICE=eth0
```

```
ONBOOT=yes
```

```
BOOTPROTO=static
```

```
IPADDR=192.168.1.1
```

```
NETMASK=255.255.255.0
```

```
GATEWAY=192.168.1.254
```

```
DNS1=192.168.0.254
```

BOOTPROTO用于设置网络接口是配置为静态地址（static）或是配置为使用DHCP（dhcp）获得地址

IPADDR用于设置网络接口的ip地址

NETMASK用于设置网络接口的子网掩码

GATEWAY用于设置网络接口的缺省网关地址

网络接口配置文件

■ 动态 ifcfg-eth0 文件内容

- DEVICE=eth0
- ONBOOT=yes
- BOOTPROTO=dhcp

■ 静态 ifcfg-eth0 文件内容

- DEVICE=eth0
- ONBOOT=yes
- BOOTPROTO=static
- IPADDR=192.168.0.131
- PREFIX=24
- GATEWAY=192.168.0.254
- DNS1 =192.168.0.254

停止和启动指定的网络接口

■ 网络接口配置文件修改后并不会自动生效，需要执行命令使配置生效

- 停用网络接口

```
# ifdown eth0
```

- 启用网络接口

```
# ifup eth0
```

- 网络接口停用再启用时将按照配置文件的内容设置
命令名 网络接口名作为命令参数

网络设置工具

图形模式

