

# tar高级&vi文本编辑器<sup>2020219lrf</sup>

## 一、tar命令实现增量备份

- 1.完整备份
- 2.差异+ 增量备份
- 3.还原备份资料:清空测试资料
- 4.打包某个目录下的所有文件时忽略某个文件
- 5.tar命令打包压缩忽略目录

## 二、文本编辑器 vim

- 1.文本编辑器的作用
- 2.Linux中最常用的文本编辑器
- 3.三种工作模式
- 4.不同模式之间的切换
- 5.常规操作的补充（笔试题）

## 三、软件包安装及管理

- 1.rpm格式应用程序与系统命令的关系
- 2.常见的软件包封装类型
- 3.查询已安装的RPM软件信息
- 4.挂载光盘镜像文件
- 5.查询未安装的RPM包文件
- 6.安装或升级RPM软件
- 7.卸载指定的RPM软件
- 8.升级安装
- 9.重建rpm数据库

10.配置本地yum仓库

11.源码包 (Tarball--> .tar.gz|.tar.bz2|.tar.xz) 方式安装

12.备份配置文件后, 修改配置文件

## 一、tar命令实现增量备份

### 1.完整备份

- 建立测试路径与档案

```
[root@localhost ~]# mkdir test
```

```
[root@localhost ~]# touch test/{a,b,c}
```

- 执行完整备份

```
[root@localhost ~]# tar -g snapshot -zcf full.tar.gz test/
```

- 查看tarball内容

```
[root@localhost ~]# tar tf full.tar.gz
```

```
test/
```

```
test/a
```

```
test/b
```

```
test/c
```

### 2.差异+增量备份

- 新增一个档案, 并修改一个档案内容

```
[root@localhost ~]# touch test/e
```

```
[root@localhost ~]# echo 123 > test/a
```

- 执行第二次的增量备份 (注意tarball档名)

```
[root@localhost ~]# tar -g snapshot -zcf full_2.tar.gz test
```

- 查看tarball内容

```
[root@localhost ~]# tar tf full_2.tar.gz
```

```
test/
```

```
test/a
```

```
test/e
```

### 3.还原备份资料:清空测试资料

```
[root@localhost ~]# rm -rf test/
```

- 开始进行资料还原

```
[root@localhost ~]# tar xf full.tar.gz
```

```
[root@localhost ~]# tar xf full_2.tar.gz
```

- 查看测试资料

```
[root@localhost ~]# ls test/
a      b      c      e
```

## 4.打包某个目录下的所有文件时忽略某个文件

方法一：

```
[root@localhost ~]# mkdir /tardir
[root@localhost ~]# touch /tardir/{11,22,33,44,55}
[root@localhost ~]# ls -l /tardir/
总用量 0
-rw-r--r-- 1 root root 0 12月 14 16:59 11
-rw-r--r-- 1 root root 0 12月 14 16:59 22
-rw-r--r-- 1 root root 0 12月 14 16:59 33
-rw-r--r-- 1 root root 0 12月 14 16:59 44
-rw-r--r-- 1 root root 0 12月 14 16:59 55
[root@localhost ~]# tar zcf num.tar.gz --exclude=/tardir/11 --exclude=/tardir/22 /tardir
tar: 从成员名中删除开头的“/”
[root@localhost ~]# tar tf num.tar.gz
```

方法二：

```
[root@localhost ~]# vim /tardir/excludefile
11
22
[root@localhost ~]# tar zcf num.tar.gz --exclude-from /tardir/excludefile /tardir
tar: 从成员名中删除开头的“/”
[root@localhost ~]# tar tf num.tar.gz
```

## 5.tar命令打包压缩忽略目录

```
[root@localhost ~]# tar zcf file.tar.gz -C /etc/ passwd
[root@localhost ~]# tar tf file.tar.gz
passwd
```

# 二、文本编辑器 vim

## 1.文本编辑器的作用

- 创建或修改文本文件
- 维护Linux系统中的各种配置文件

## 2.Linux中最常用的文本编辑器

- vi:类UNIX操作系统的默认文本编辑器
- vim: vim是vi文本编辑器（一般简称为vi编辑器）的增强版本

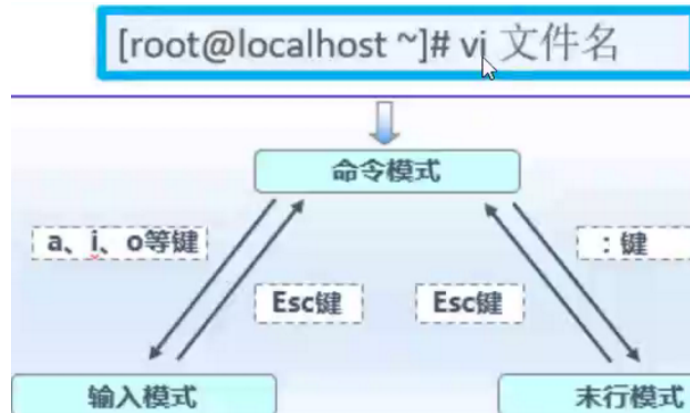
## 3.三种工作模式

- 命令模式：主要做复制，粘贴，剪切，删除

- 输入模式：主要输入文本信息，编辑等
- 末行模式：主要保存退出等

ps:并不完全是这些功能，还有其他功能

## 4.不同模式之间的切换



### 快捷键

- a :在光标后插入内容
- A : 在光标所在行末尾插入内容
- i: 从当前光标前插入内容
- I: 在光标所在行行首插入内容
- o: 在当前光标下插入空行
- O: 在当前光标上插入空行
- 方向移动:上下左右
- 翻页: pg up,pg dn
- 行内快速跳转:home/^/0 end/\$
- 行间快速跳转:1G/gg 首行 G 末行 #G 第几行
- 行号显示: set nu 显示 行号 set nonu 不显示行号
- 删除: x/del 删单个字符 dd 删除所在行 #dd 删第几行 d^ 删光标到行首 d\$ 删光标到行尾
- 复制: yy 复制整行 #yy 复制从光标所在行开始的#行内容
- 粘贴: p 将缓冲区中的内容粘贴到光标位置处之后一行 P粘贴到光标位置处之前

文件内容查找

/word : 从上而下查找

?word: 从下而上查找

n: 定位下一个

N: 定位上一个

撤销编辑及保存退出

u: 取消最近的一次操作

--u是对整个文件的修改

U: 取消对当前行所做的所有编辑—大写U针对行

ZZ: 保存并退出vi编辑器

保存文件及退出vi编辑器

w:保存文件 :w /root/newfile 另存为其他文件

退出vi: q : 未修改退出 q!:放弃对文件内容的修改

wq:保存文件退出vi

打开新文件或读入其他文件内容

:e ~/install.log 打开新的文件进行编辑

:r /etc/filesystems

ps:记忆一个就好。e

### 查找替换

:s /old/new

:s /old/new/g

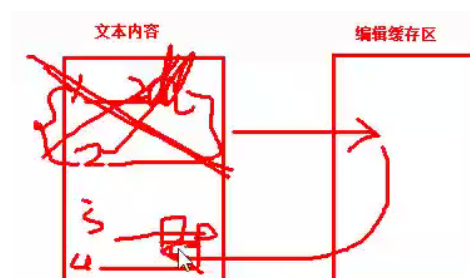
:#, # s/old/new/g

:% s/old/new/g

:s /old/new/c

ps:

1. esc返回命令模式
2. 注意下方提示, 在英文状态写入命令
3. 命令模式输入快捷键
4. 编辑缓存区



- 用系统数据做实验时, 不要用它本身, 把它copy出来, 不影响该系统运行

```
[root@localhost ~]# rm -rf test
[root@localhost ~]# cat /etc/passwd > test.txt
[root@localhost ~]# cat /etc/passwd >> test.txt
[root@localhost ~]# vim test.txt
```

```
1、复制/etc/passwd文件到/root/pass
2、vim编辑/root/pass文件，将root替换为admin。
3、按u撤销
4、vim编辑/root/pass文件，将/sbin/nologin替换为/1111/2222。
方法一：转义
: % s/\sbin\/nologin\/\1111\/2222/g
```

方法二：

```
: % s#/sbin/nologin#/1111/2222#g
```

1. cp /etc/passwd ~/pass

2. :s /oroot/admin

4. 方法三： % s/sbin\/nologin/1111\/2222/g

注：符号不要冲突，用三个###或三个iii也可以

## 5.常规操作的补充（笔试题）

- x: 保存退出，但如果未修改，时间戳不改变
- w file: 将当前文件另存为file (vim)
- X: 加密 (vim)
- 7: 跳转到第7行
- 1,\$ co \$: 从第一行到最后一行进行复制，并粘贴到最后一行后面
- J: 合并当前行和下一行
- ctrl+r: 对使用u命令撤销操作进行恢复

vim可视化操作: visual

vim学习工具: vimtutor

1、删除了第1行到第3行

2、u 1~3恢复

3、ctrl+r把恢复去掉

问题:

```
E325: 注意
发现交换文件 ".pass.swp"
 所有者: root    日期: Sat Feb 15 20:32:53 2020
 文件名: ~root/pass
 修改过: 否
 用户名: root    主机名: lx
 进程 ID: 9776
正在打开文件 "pass"
 日期: Tue Feb 18 10:45:18 2020
比交换文件新!

(1) Another program may be editing the same file.  If this is the case,
    be careful not to end up with two different instances of the same
    file when making changes.  Quit, or continue with caution.
(2) An edit session for this file crashed.
    如果是这样，请用 ":recover" 或 "vim -r pass"
    恢复修改的内容 (请见 ":help recovery")。
    如果你已经进行了恢复，请删除交换文件 ".pass.swp"
-- 更多 --
```

当打开vim pass 时弹出的界面

解决办法: rm -rf .pass.swp

```
[root@localhost ~]# vim 123
[root@localhost ~]# vim 123
[root@localhost ~]# rm -r .123.swp
rm: 是否删除普通文件 ".123.swp"? y
[root@localhost ~]# vim 123
[root@localhost ~]#
```

表达式敲错了

```
18 color: x:32:32:Rpcbind Daemon:/var/lib/rpck
19 rpc:x:32:32:Rpcbind Daemon:/var/lib/rpck
E486: Pattern not found: sbin\nologin
  仅将文本发送到当前选项卡
```

## 三、软件包安装及管理

### 1.rpm格式应用程序与系统命令的关系

- 文件位置
  1. 系统命令：一般在/bin和/sbin目录中，或为Shell内部指令
  2. 应用程序：通常在/usr/bin和、usr/sbin目录中
- 主要用途
  1. 系统命令：完成对系统的基本管理工作，例如IP配置工具
  2. 应用程序：完成相对独立的其他辅助任务，例如网页浏览器
- 适用环境
  1. 系统命令：一般只在字符操作界面中运行
  2. 应用程序：根据实际需要，有些程序可在图形界面运行
- 运行格式
  1. 系统命令：一般包括命令字、命令选项和命令参数
  2. 应用程序：通常没有固定的执行格式

注：windows里面软件包格式：exe是第三方格式，msi官方  
简称rhel，企业环境（红帽系列）

文件类型	保存目录
普通执行程序文件	/usr/bin
服务器执行程序文件和管理程序文件	/usr/sbin
应用程序配置文件	/etc
日志文件	/var/log
应用程序文档文件	/usr/share/doc
应用程序手册页文件	/usr/share/man

注：

- 1. windows装软件：
  - 在指定安装位置生成各种文件
  - 修改注册表
- 2. linux装软件：
  - RPM
  - 就是生成了各种文件
- 3. 源码包

2.常见的软件包封装类型

常见的软件包封装类型	
文件类型	保存目录
rpm软件包	扩展名为 “.rpm”
deb软件包	扩展名为 “.deb”
源代码软件包	一般为 “.tar.gz” 、 “.tar.bz2” 等格式的压缩包，包含程序的原始代码
绿色免安装的软件包	在压缩包内提供已编译好的执行程序文件，解开压缩包后的文件即可直接使用

注：

源代码软件包： tar.xz

由源码包转换成.rpm和.deb，要获得最新最全的防bug的就找源码包

主要学rpm 和源码包

RPM Package Manager

由Red Hat公司提出，被众多Linux发行版所采用

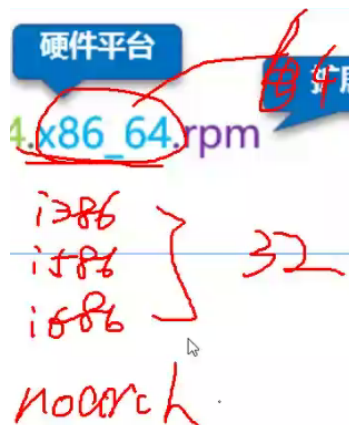
rpm命令建立统一的数据库文件、详细记录软件包安装、卸载等变化信息、自动分析软件包依赖关系

一般命名格式：



注：RPM 只能管rpm；我们用到的软件包镜像里面都有；网站基本用不到





ps: 软件名称 软件包名, 要严格区分

### 3. 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

- rpm -q 软件名 查询指定软件是否安装 (“-q” 选项时实际上调用了 “/usr/bin/rpmquery” 程序完成查询工作)
- rpm -qa 查询系统已经安装所有的软件信息 (软件由哪个软件包生成)
- rpm -qa | grep 查询当前系统安装了哪些与软件名称相关的包
- rpm -qi 软件名 查询已经安装软件的详细信息
- rpm -ql 软件名 查询已安装软件安装到什么地方去了
- rpm -qf 文件的绝对路径 查询该文件由哪个软件产生
- rpm -qc 软件名 查询软件生成的配置文件
- rpm -qd 软件名 查询软件生成的文档文件

注: 查询的都是已安装的, 最后两个不用背4.

### 4. 挂载光盘镜像文件

(将CentOS操作系统IOS镜像文件放入虚拟机光驱, 注意: 设备状态的“对勾”必须打上)

```
[root@localhost ~]# mount | grep "sr0" 查询光盘是否挂载
[root@localhost ~]# umount /dev/sr0 卸载光盘 /dev/sr0 = /dev/cdrom
[root@localhost ~]# mkdir /media/cdrom 准备空目录作为挂载点
[root@localhost ~]# mount /dev/sr0 /media/cdrom 挂载光盘
[root@localhost ~]# ls -l /media/cdrom/Packages 查看rpm软件包列表
```

### 5. 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

- rpm -qpi 完整软件包名称 通过.rpm包文件查看该软件的详细信息

- rpm -qpl 完整软件包名称 通过.rpmrpm安装包内所包含的目录、文件列表
- rpm -qpc 完整软件包名称 通过.rpm安装包内包含的配置文件列表
- rpm -qpd 完整软件包名称 通过.rpm安装包内包含的文档文件列表

## 6.安装或升级RPM软件

格式: rpm [选项] RPM包文件

rpm -ivh 完整软件包名称

- -i 安装一个新的rpm软件包
- -h 以“#”号显示安装的进度
- -v 显示安装过程中的详细信息
- --force 强制安装（主要用在安装旧的软件代替新的软件）
- --nodeps 安装、升级或卸载软件时，忽略依赖关系
- --test 测试安装

安装软件可以用-U、-i

安装的时候忽略依赖关系，卸载的时候更要忽略依赖关系，笔试题

```
[root@localhost ~]# rpm -ivh /media/cdrom/Packages/dhcp-4.2.5-68.el7.centos.x86_64.rpm
```

```
root@localhost ~]# rpm -q vsftpd
vsftpd-3.0.2-22.el7.x86_64
root@localhost ~]# rpm -ivh /media/cdrom/Packages/vsftpd-3.0.2-22.el7.x86_64.rpm
准备中... ##### [100%]
软件包 vsftpd-3.0.2-22.el7.x86_64 已经安装
root@localhost ~]#
root@localhost ~]#
root@localhost ~]# rpm -ivh /media/cdrom/Packages/vsftpd-3.0.2-22.el7.x86_64.rpm --force
准备中... ##### [100%]
正在升级/安装...
1:vsftpd-3.0.2-22.el7 ##### [100%]
root@localhost ~]#
```

```
[root@localhost ~]# rpm -ivh /media/cdrom/Packages/httpd-2.4.6-80.el7.centos.x86_64.rpm
错误: 依赖检测失败:
/etc/mime.types 被 httpd-2.4.6-80.el7.centos.x86_64 需要
httpd-tools = 2.4.6-80.el7.centos 被 httpd-2.4.6-80.el7.centos.x86_64 需要
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# rpm -ivh /media/cdrom/Packages/httpd-tools-2.4.6-80.el7.centos.x86_64.rpm
准备中... ##### [100%]
正在升级/安装...
1:httpd-tools-2.4.6-80.el7.centos ##### [100%]
[root@localhost ~]# rpm -ivh /media/cdrom/Packages/httpd-2.4.6-80.el7.centos.x86_64.rpm
错误: 依赖检测失败:
/etc/mime.types 被 httpd-2.4.6-80.el7.centos.x86_64 需要
[root@localhost ~]#
```

```
root@localhost ~]# rpm -ivh /media/cdrom/Packages/httpd-2.4.6-80.el7.centos.x86_64.rpm --nodeps
准备中... ##### [100%]
正在升级/安装...
1:httpd-2.4.6-80.el7.centos ##### [100%]
root@localhost ~]#
```

## 7.卸载指定的RPM软件

- rpm -e 软件

```
root@localhost ~]# rpm -q dhcp
hcp-4.2.5-68.el7.centos.x86_64
root@localhost ~]# rpm -e dhcp
root@localhost ~]# rpm -q dhcp
卡安装软件包 dhcp
root@localhost ~]#
```

## 8.升级安装

- rpm -Uvh 完整软件包名称 #无论旧版本软件是否安装，都安装新版本
- rpm -Fvh 完整软件包名称 #若旧版本软件没有安装，则放弃安装新版本

注：警告类似于安装非手机应用商店的应用

```
root@localhost ~]# rpm -ivh /media/cdrom/Packages/lrzs-0.12.20-36.el7.x86_64.rpm
```

lrzs软件的作用      rz:上传文件      sz: 下载文件

```
root@localhost ~]# lrzs   rz上传文件   sz下载文件   ^C
root@localhost ~]#
root@localhost ~]#
root@localhost ~]#
root@localhost ~]# ls
23            ansible-2.5.0-2.el6.noarch.rpm    full.tar.gz
bc            epel-release-latest-7.noarch.rpm   initial-setup-ks.cfg
naconda-ks.cfg   full_2.tar.gz            num.tar.gz
root@localhost ~]#
root@localhost ~]#
root@localhost ~]# sz epel-release-latest-7.noarch.rpm
root@localhost ~]#
```

```
root@localhost ~]#
root@localhost ~]# rz
```

I

```
root@localhost ~]# ls
23            ansible-2.5.0-2.el6.noarch.rpm    full.tar.gz            password.lst    模板    文档    桌面
bc            epel-release-latest-7.noarch.rpm   initial-setup-ks.cfg   snapshot        视频    下载
naconda-ks.cfg   full_2.tar.gz            num.tar.gz        公共        图片    音乐
root@localhost ~]#
```

## 9.重建rpm数据库

- rpm --rebuilddb
- rpm --initdb

```
root@localhost ~]# rpm -q dhcp
hcp-4.2.5-68.el7.centos.x86_64
root@localhost ~]# rpm -e dhcp
root@localhost ~]# rpm -q dhcp
卡安装软件包 dhcp
root@localhost ~]#
```

警告类似于安装非手机应用商店的应用

```

root@localhost ~]# rpm -ivh ansible-2.5.0-2.el6.noarch.rpm --nodeps
警告: ansible-2.5.0-2.el6.noarch.rpm: 头V3 RSA/SHA256 Signature, 密钥 ID 0608b895: NOKEY
准备中... ##### [100%]
正在升级/安装...
1:ansible-2.5.0-2.el6 ##### [100%]
root@localhost ~]#

root@localhost ~]# ls /media/cdrom/
CentOS_BuildTag  EULA  images  LiveOS  repodata  RPM-GPG-KEY-CentOS-Testing-7
FI  GPL  isolinux  Packages  RPM-GPG-KEY-CentOS-7  TRANS.TBL
root@localhost ~]# rpm --import /media/cdrom/RPM-GPG-KEY-CentOS-*
root@localhost ~]#

```

## 10.配置本地yum仓库

(务必把光盘镜像文件挂载/media/cdrom下)

yum命令：便于自动解决软件包的依赖关系

挂载光盘镜像文件：

```

[root@localhost ~]# mount | grep "sr0" 查询光盘是否挂载
[root@localhost ~]# umount /dev/sr0 卸载光盘 /dev/sr0 = /dev/cdrom

[root@localhost ~]# mkdir /media/cdrom 准备空目录作为挂载点
[root@localhost ~]# mount /dev/sr0 /media/cdrom 挂载光盘
[root@localhost ~]# ls -l /media/cdrom/Packages 查看rpm软件包列表

```

修改yum仓库配置文件

```

[root@localhost ~]# cd /etc/yum.repos.d/
[root@localhost yum.repos.d]# mkdir backup
[root@localhost yum.repos.d]# mv *.repo backup/

```

方法一：

```

[root@localhost yum.repos.d]# mv backup/CentOS-Media.repo ./
[root@localhost yum.repos.d]# vim CentOS-Media.repo
enabled=0 改为 enabled=1

```

方法二：

```

[root@localhost yum.repos.d]# vim local.repo
[c7-media]
name=CentOS-Media
baseurl=file:///media/cdrom
gpgcheck=1
enabled=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7

```

```
[root@localhost ~]# yum clean all && yum makecache
```

```
[root@localhost ~]# yum clean all #清除yum仓库缓存
```

```
[root@localhost ~]# yum makecache #重建yum仓库缓存
```

```
[root@localhost ~]# yum list #显示yum仓库软件列表
```

```
[root@localhost ~]# yum -y install software #安装software软件
```

```
[root@localhost ~]# yum -y update software #升级software软件（需要配置互联网yum仓库）
```

```
[root@localhost ~]# yum -y remove software #卸载software软件
```

remove=erase（生产环境中慎用）一般用 -e

-y: 命令执行过程不与管理员交互

## 11.源码包 (Tarball--> .tar.gz|.tar.bz2|.tar.xz) 方式安装

- 源代码编译概述

获得最新的软件版本，及时修复bug

根据用户需要，灵活制定软件功能

- 应用场合举例

安装较新版本的应用程序时

当前安装的rpm程序无法满足需要时

需要为应用程序添加新的功能时

软件包对内使用--> rpm

软件包对外使用--> tarball

- Tarball封包

.tar.gz和.tar.bz2及.tar.xz格式居多 (tar -Jcf file.tar.xz file1 file2)

软件素材参考: <http://sourceforge.net>

软件包官方网站

md5sum完整性校验

计算MD5校验和，并与官方提供的值相比较，判断是否一致，可能源码包破损或被人植入程序，需谨慎

```
[root@localhost ~]# md5sum httpd-2.4.37.tar.gz
```

```
9ca56d8636944a4989de5af608714086 httpd-2.4.37.tar.gz
```

想通过源码包安装软件系统中需要安装支持C/C++程序语言的编译器 (gcc)

查询编译器是否完善



```
[root@localhost ~]# gcc --version
[root@localhost ~]# g++ --version
[root@localhost ~]# rpm -q gcc gcc-c++ make
gcc-4.8.5-28.el7.x86_64
gcc-c++-4.8.5-28.el7.x86_64
make-3.82-23.el7.x86_64
```

### 步骤1: tar解包

用途: 解压并释放源代码包到指定的目录 (习惯将软件包放到/usr/src/目录)

```
[root@localhost ~]# tar xzf httpd-2.4.37.tar.gz -C /usr/src/
```

### 步骤2: ./configure 配置

用途: 设置安装目录、安装功能模块等选项、生成Makefile

```
[root@localhost ~]# cd /usr/src/httpd-2.4.37/
```

```
[root@localhost httpd-2.4.37]# ./configure --prefix=/usr/local/httpd
```

#--prefix指定安装路径, 部分人安装到/opt或者用户宿主目录下

### 步骤3: make编译

用途: 基于configure步骤生成的Makefile, 编译生成可执行的二进制文件或普通文件

```
[root@localhost httpd-2.4.37]# make -j 2
```

-j 通过多个CPU内核共同编译

**敲make 就行, 敲make -j 2 容易造成当机**

### 步骤4: make install 安装

用途: 复制二进制文件或普通文件到系统, 配置应用环境

```
[root@localhost httpd-2.4.37]# make install
```

逻辑与: &&

```
[root@localhost httpd-2.4.37]# ./configure --prefix=/usr/local/httpd && make && make install
```

### 步骤5: 验证安装效果

前期准备

```
[root@localhost ~]# systemctl stop httpd |
```

```
[root@localhost ~]# rpm -e httpd --nodeps
```

## 12.备份配置文件后, 修改配置文件

```
[root@localhost httpd-2.4.37]# cd /usr/local/apache/conf/
```

```
[root@localhost conf]# cp httpd.conf httpd.conf.bak
```

```
[root@localhost httpd-2.4.37]# vim /usr/local/apache/conf/httpd.conf 进入文件后搜索
```

ServerName将#注释去掉

```
#ServerName www.example.com:80
```

```
ServerName www.example.com:80
```

```

[root@localhost ~]# rpm -q httpd
httpd-2.4.6-80.el7.centos.x86_64
[root@localhost ~]#
[root@localhost ~]# systemctl stop httpd
[root@localhost ~]# rpm -e httpd

[root@localhost ~]# cd /usr/local/httpd/conf/
[root@localhost conf]# ls
extra httpd.conf magic mime.types original
[root@localhost conf]# cp httpd.conf httpd.conf.bak
[root@localhost conf]# vim httpd.conf
[root@localhost conf]# /usr/local/httpd/

[root@localhost conf]# /usr/local/httpd/bin/apachectl start
[root@localhost conf]#

```

## 关闭防火墙

iptables -F

systemctl stop firewalld

## 关闭selinux防护机制

setenforce 0

注：再改任何文件之前，养成备份的习惯

卸载软件包，只能用rm，删除路径，不是rpm格式

清理tarball

```

[root@localhost ~]# /usr/local/httpd/bin/apachectl stop
[root@localhost ~]# rm -rf /usr/local/httpd/
[root@localhost ~]# cd /usr/src/httpd-2.4.37
[root@localhost httpd-2.4.37]# make clean
[root@localhost httpd-2.4.37]# cd
[root@localhost ~]# rm -rf /usr/src/httpd-2.4.37

```

```

[root@localhost httpd-2.4.41]# cd
[root@localhost ~]#
[root@localhost ~]# rm -rf /usr/src/httpd-2.4.41/
[root@localhost ~]#

```

