

# Hadoop 集群（第 3 期）

## ——VSFTP 安装配置

### 1、VSFTP简介

**VSFTP** 是一个基于 GPL 发布的类 Unix 系统上使用的 **FTP 服务器** 软件,它的全称是 Very Secure FTP 从此名称可以看出来,编制者的初衷是代码的安全。

安全性是编写 VSFTP 的初衷,除了这与生俱来的安全特性以外,高速与高稳定性也是 VSFTP 的两个重要特点。

在速度方面,使用 ASCII 代码的模式下载数据时,VSFTP 的速度是 Wu-FTP 的两倍,如果 Linux 主机使用 2.4.\*的内核,在千兆以太网上的下载速度可达 86MB/S。

在稳定方面,VSFTP 就更加的出色,VSFTP 在单机(非集群)上支持 4000 个以上的并发用户同时连接,根据 Red Hat 的 Ftp 服务器(ftp.redhat.com)的数据,VSFTP 服务器可以支持 15000 个并发用户。

### 2、VSFTP安装及配置

安装该软件需要使用最高用户(**root**)进行安装,否则不能进行。

```
Last login: Fri Feb 24 05:12:50 2012 from 192.168.1.102
[hadoop@Master ~]$ su -
密码:
[root@Master ~]#

[root@Master ~]# █
```

#### 2.1 安装

首先用命令检查 VSFTP 是否已经安装。

```
chkconfig --list | grep vsftpd
```

显示结果如下, **没有任何反应**, 说明没有安装 VSFTP。

```
[root@Master ~]# chkconfig --list | grep vsftpd
[root@Master ~]# █
```

接着使用 **yum** 命令直接安装

```
yum -y install vsftpd
```

```
[root@Master ~]# yum -y install vsftpd
Loaded plugins: fastestmirror
http://mirrors.grandcloud.cn/centos/6.2/os/i386/repodata/repomd.xml: [
Errno 14] PYCURL ERROR 6 - ""
Trying other mirror.
base                                     | 3.7 kB      00:00
base/primary_db                         85% [=====] 47 kB/s | 3.0 MB      00:10 ETA
```

上图中表示正在下载，需要耐心等待一下，如果网络不畅通，也可能需要下载失败，不过不要担心，只要在重新提交一次命令就行。

```
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing      : vsftpd-2.2.2-6.el6_2.1.i686                      1/1

Installed:
  vsftpd.i686 0:2.2.2-6.el6_2.1

Complete!
[root@Master ~]#
```

从上面的结果中看出，已经成功安装。然后为它创建日志文件：

```
touch /var/log/vsftpd.log
```

```
[root@Master ~]# touch /var/log/vsftpd.log
[root@Master ~]#
```

这样简单的两个命令就完成了 **vsftp** 的安装，但是如果你现在想这样 `ftp://your_ip` 来访问的话，那还不行，还需要配置权限！

## 2.2 启动与配置自启动

再次使用 “**chkconfig -list | grep vsfpd**” 来查看 **vsftpd** 服务启动项情况；  
如果看到的是如下显示的结果：

```
[root@Master ~]# chkconfig --list | grep vsftpd
vsftpd      0:关闭  1:关闭  2:关闭  3:关闭  4:关闭  5:关闭  6:关闭
[root@Master ~]#
```

服务全部都是关闭（**off**）的，注意这里的关闭（**off**）表示的是服务器启动的时候是否会自启动服务，我们使用如下命令来配置其自启动：

```
chkconfig vsftpd on
```

或者

```
chkconfig --level 2345 vsftpd on
```

执行结果如下：

```
[root@Master ~]# chkconfig --list | grep vsftpd
vsftpd      0:关闭  1:关闭  2:关闭  3:关闭  4:关闭  5:关闭  6:关闭
[root@Master ~]# chkconfig vsftpd on
[root@Master ~]# chkconfig --list | grep vsftpd
vsftpd      0:关闭  1:关闭  2:启用  3:启用  4:启用  5:启用  6:关闭
[root@Master ~]#
```

查看与管理 ftp 服务：

```
启动 ftp 服务： service vsftpd start
查看 ftp 服务状态： service vsftpd status
重启 ftp 服务： service vsftpd restart
关闭 ftp 服务： service vsftpd stop
```

## 2.3 配置vsftpd服务

编辑/etc/vsftpd/vsftpd.conf 文件，配置 vsftp 服务：

```
vim /etc/vsftpd/vsftpd.conf
```

```
[root@Master ~]# ls /etc/vsftpd
ftppusers  user_list  vsftpd.conf  vsftpd_conf_migrate.sh
[root@Master ~]# vim /etc/vsftpd/vsftpd.conf
```

上图中先显示出“**/etc/vsftpd**”下面有哪些文件，然后在用“**vim**”编辑器进行修改。

先按键盘上的“a”就可以进行编辑了，按照下面进行操作。

```
anonymous_enable=YES --> anonymous_enable=NO //不允许匿名用户访问，默认是允许

xferlog_file=/var/log/vsftpd.log #设定 vsftpd 的服务日志保存路径。注意，该文件默认不存在。
必须要手动 touch 出来

#idle_session_timeout=600 --> idle_session_timeout=600 //会话超时，客户端连接到 ftp 但
未操作，默认被注释掉，可根据个人情况修改

#async_abor_enable=YES --> async_abor_enable=YES //支持异步传输功能，默认是
注释掉的，去掉注释

#ascii_upload_enable=YES --> ascii_upload_enable=YES //支持 ASCII 模式的下载功
能，默认是注释掉的，去掉注释

#ascii_download_enable=YES --> ascii_download_enable=YES //支持 ASCII 模式的上
传功能，默认是注释掉的，去掉注释

#ftpd_banner=Welcome to blah FTP service //FTP 的登录欢迎语，本身是被注释掉的，去不
去都行

#chroot_local_user=YES --> chroot_local_user=YES
//禁止本地用户登出自己的 FTP 主目录，本身被注释掉，去掉注释
```

下面几个都是已经去掉的，只是写出来，明白其用意。

```
local_enable=YES //允许本地用户访问，默认就是 YES，不用改
write_enable=YES //允许写入，默认是 YES，不用改
local_umask=022 //上传后文件的权限掩码，不用改
dirmessage_enable=YES //开启目录标语，默认是 YES，开不开无所谓，我是默认就行
xferlog_enable=YES //开启日志，默认是 YES，不用改
connect_from_port_20=YES //设定连接端口 20
xferlog_std_format=YES //设定 vsftpd 的服务日志保存路径，不用改

pam_service_name=vsftpd //设定 pam 服务下 vsftpd 的验证配置文件名，不用改
userlist_enable=YES //拒绝登录用户名单，不用改
TCP_wrappers=YES //限制主机对 VSFTP 服务器的访问，不用改（通过/etc/hosts.deny 和
/etc/hosts.allow 这两个文件来配置）
```

按照上面修改完之后，按键盘“**Esc**”退出编辑，再按“**:**”，并在后面输入“**wq**”，进行保存并退出。

## 2.4 配置iptables防火墙

按照以上步骤还不能运行，用下面命令可以查看一下防火墙允许的端口号，我们知道ftp的端口号是“20、21”，从结果中可以看出，并没有。

```
service iptables status
```

```
[root@Master ~]# service iptables status
表格: filter
Chain INPUT (policy ACCEPT)
num  target    prot opt source                destination            state
1    ACCEPT    all  --  0.0.0.0/0            0.0.0.0/0              state RELATED, ESTABLISHED
2    ACCEPT    icmp --  0.0.0.0/0            0.0.0.0/0
3    ACCEPT    all  --  0.0.0.0/0            0.0.0.0/0
4    ACCEPT    tcp  --  0.0.0.0/0            0.0.0.0/0              state NEW tcp dpt:22
5    REJECT    all  --  0.0.0.0/0            0.0.0.0/0              reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
num  target    prot opt source                destination            reject-with
1    REJECT    all  --  0.0.0.0/0            0.0.0.0/0              icmp-host-prohibited

Chain OUTPUT (policy ACCEPT)
num  target    prot opt source                destination
```

这时需要对“/etc/sysconfig/iptables”进行配置：

```
vim /etc/sysconfig/iptables
```

```
[root@Master ~]# ls /etc/sysconfig | grep iptables
iptables
iptables-config
iptables.old
[root@Master ~]# vim /etc/sysconfig/iptables

## Firewall configuration written by system-config-firewall
# Manual customization of this file is not recommended.
*filter
```

上图中先列出以“iptables”模糊匹配的文件，然后在用上面的命名对“iptables”文件进行添加下面的内容。

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 21 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 20 -j ACCEPT
```

具体的添加操作命令和前面对“/etc/vsftpd/vsftpd.conf”的操作一样。

```
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 21 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 20 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
"/etc/sysconfig/iptables" 15L, 606C 已写入
[root@Master ~]#
```

## 2.5 设置selinux问题

**SELinux**(Security-Enhanced Linux) 是美国国家安全局 (NSA) 对于强制访问控制的实现, 是 Linux 史上最杰出的新安全子系统。SELinux 是 2.6 版本的 Linux 内核中提供的强制访问控制 (MAC) 系统。对于目前可用的 Linux 安全模块来说, SELinux 是功能最全面, 而且测试最充分的, 它是在 20 年的 MAC 研究基础上建立的。SELinux 在类型强制服务器中合并了多级安全性或一种可选的多类策略, 并采用了基于角色的访问控制概念 SELinux 提供了比传统的 UNIX 权限更好的访问控制。

正式如此, 所以此时还不能访问 FTP 服务器。用下面命令查询:

```
getsebool -a | grep ftp
```

```
[root@Master ~]# getsebool -a | grep ftp
allow_ftpd_anon_write --> off
allow_ftpd_full_access --> off
allow_ftpd_use_cifs --> off
allow_ftpd_use_nfs --> off
ftp_home_dir --> off
ftpd_connect_db --> off
httpd_enable_ftp_server --> off
sftpd_anon_write --> off
sftpd_enable_homedirs --> off
sftpd_full_access --> off
sftpd_write_ssh_home --> off
tftp_anon_write --> off
[root@Master ~]#
```

可以通过下面命令把上图中红色圈住的两个的值设置为“on”。

```
setsebool -P ftp_home_dir 1
setsebool -P allow_ftpd_full_access 1
```

```
[root@Master ~]# setsebool -P ftp_home_dir 1
[root@Master ~]# setsebool -P allow_ftpd_full_access 1
[root@Master ~]# getsebool -a | grep ftp
allow_ftpd_anon_write --> off
allow_ftpd_full_access --> on
allow_ftpd_use_cifs --> off
allow_ftpd use nfs --> off
ftp_home_dir --> on
ftpd_connect_db --> off
httpd_enable_ftp_server --> off
sftpd_anon_write --> off
sftpd_enable_homedirs --> off
sftpd_full_access --> off
sftpd_write_ssh_home --> off
tftp_anon_write --> off
[root@Master ~]#
```

## 2.6 重启相关服务

最后就是把“vsftpd”和“iptables”两个服务重新启动一下。

```
service vsftpd restart
```

```
[root@Master ~]# service vsftpd restart
关闭 vsftpd: [确定]
为 vsftpd 启动 vsftpd: [确定]
```

如果出现下图，说明刚才没有把 vsftpd 服务启动起来。

```
service vsftpd start
```

```
[root@Master ~]# service vsftpd restart
关闭 vsftpd: [失败]
为 vsftpd 启动 vsftpd: [确定]
```

最后把防火墙重新启动一下。

```
service iptables restart
```

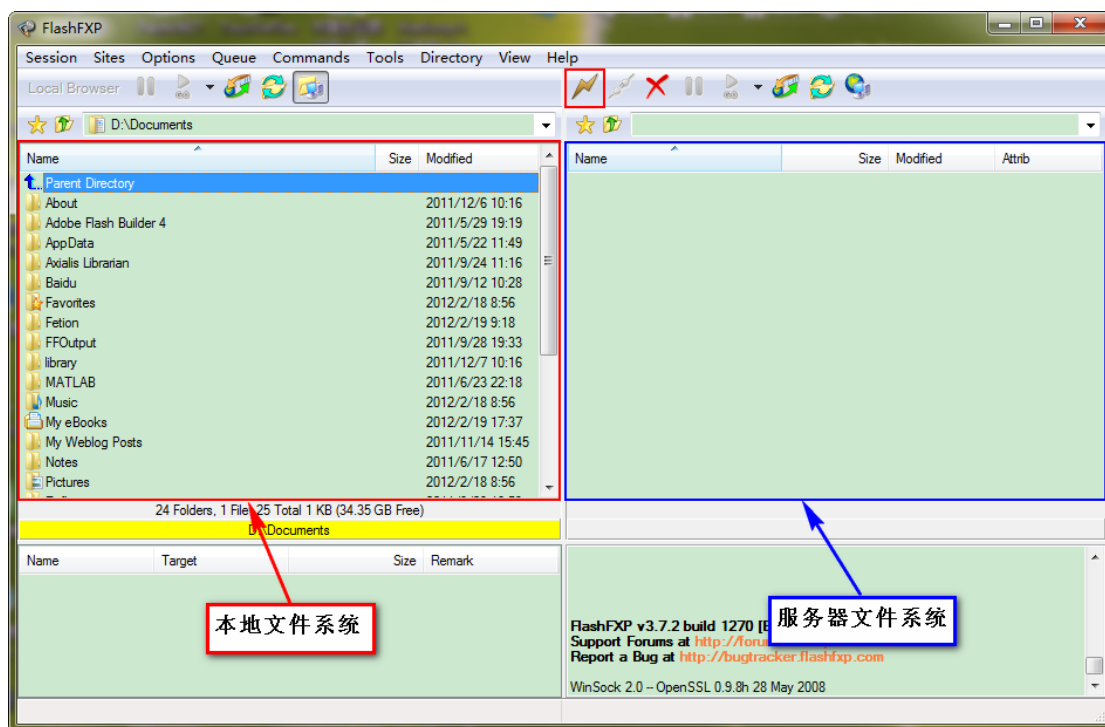
```
[root@Master ~]# service iptables restart
iptables: 清除防火墙规则: [确定]
iptables: 将链设置为政策 ACCEPT: filter [确定]
iptables: 正在卸载模块: [确定]
iptables: 应用防火墙规则: [确定]
[root@Master ~]#
```

到此为止 FTP 服务器就搭建完毕了，下面可以通过“FlashFXP.exe”软件进行访问了。

## 3、FlashFXP使用

### 3.1 FlashFXP简介

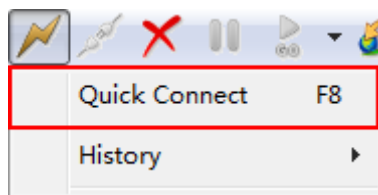
FlashFXP 是一款功能强大的 FXP/FTP 软件,集成了其它优秀的 FTP 软件的优点,如 CuteFTP 的目录比较,支持彩色文字显示;如 BpFTP 支持多目录选择文件,暂存目录;又如 LeapFTP 的界面设计。支持目录(和子目录)的文件传输,删除;支持上传,下载,以及第三方文件续传;可以跳过指定的文件类型,只传送需要的本件;可自定义不同文件类型的显示颜色;暂存远程目录列表,支持 FTP 代理及 Socks 3&4;有避免闲置断线功能,防止被 FTP 平台踢出;可显示或隐藏具有“隐藏”属性的文档和目录;支持每个平台使用被动模式等。



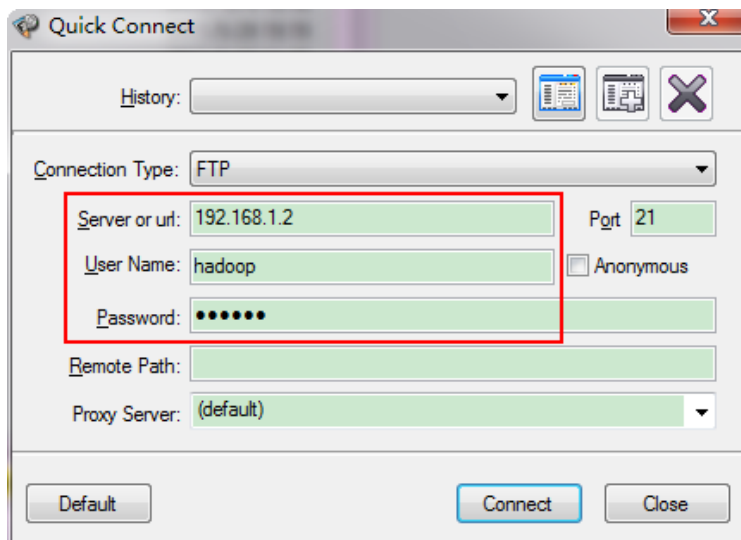
上图为 FlashFXP 开启界面，左面是显示的“本地文件系统”，后面是显示“服务器文件



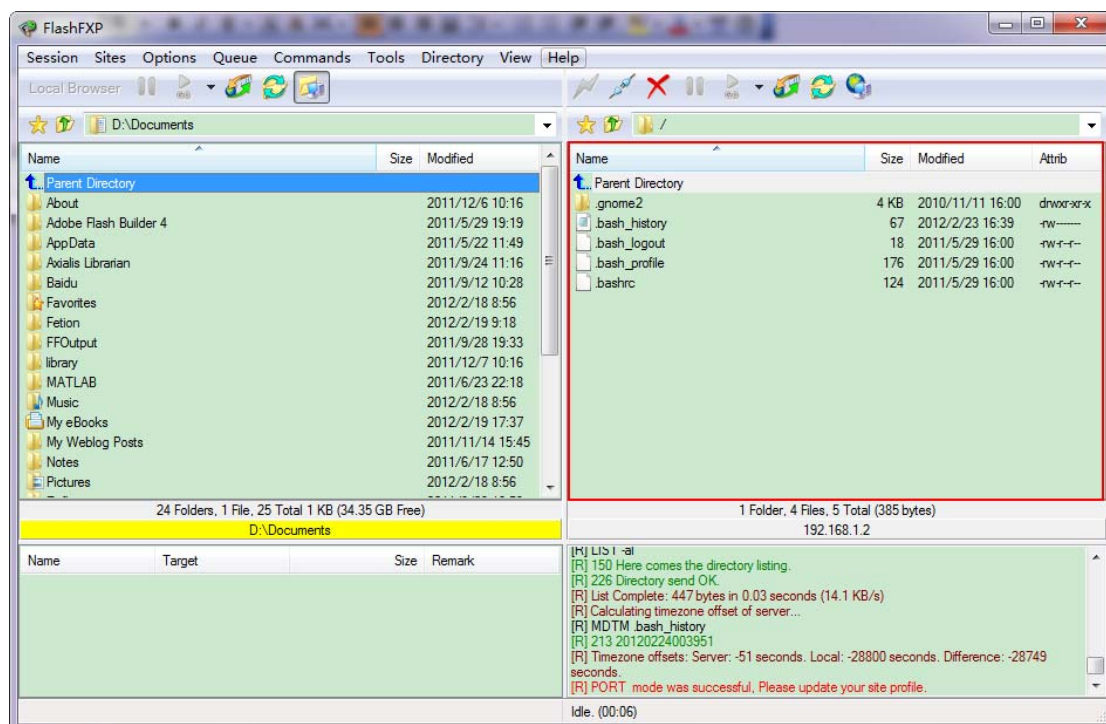
系统”，点击右侧上面的闪电图标，按照下图选择“Quick Connect”。



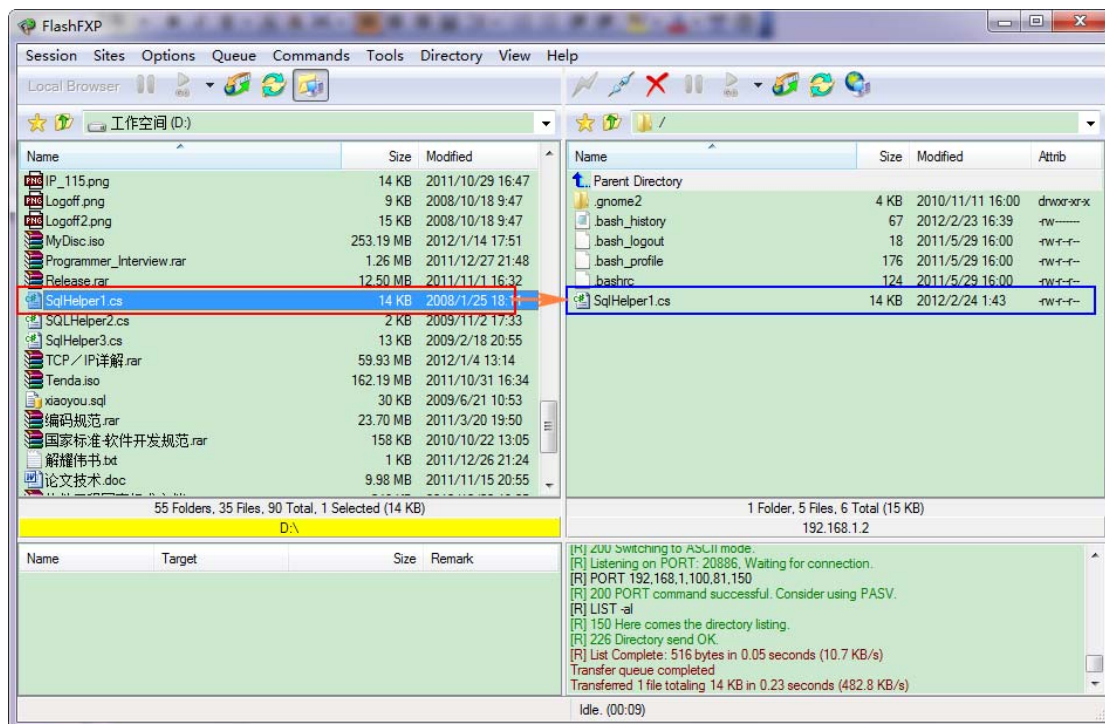
点击之后出现下图，并按照下图进行填写。例如：



## 3.2 示例上传



链接成功后，会显示上面的内容，可以尝试把左边的一个文件拖到右边，就能上传了，我们现在试一下。



最后登录一下看看是否在 linux 下面有这个文件。

```
[root@Master ~]# cd /home
[root@Master home]# ll
总用量 20
drwx-----. 3 hadoop hadoop 4096 2月 24 17:43 hadoop
drwx-----. 2 root root 16384 2月 24 04:35 lost+found
[root@Master home]# ls
hadoop lost+found
[root@Master home]# cd hadoop
[root@Master hadoop]# ls
SqlHelper1.cs
[root@Master hadoop]#
```

从上图中发现了我们刚才上传的文件“**SqlHelper1.cs**”文件，好了，到此为止，我们的 Linux 下的 FTP 服务器基本完成了，但是离真正的 FTP 服务器还差的很远，因为我们只是用于“远程上传”点文件，所以基本能满足我们的需要。

## 4、用到的Linux命令

### 4.1 chkconfig命令详解

**chkconfig** 命令主要用来**更新**（**启动**或**停止**）和**查询系统服务**的运行级信息。谨记

chkconfig 不是立即自动禁止或激活一个服务，它只是简单的改变了符号连接。

### 使用语法：

```
chkconfig [--add][--del][--list][系统服务] 或 chkconfig [--level <等级代号>][系统服务][on/off/reset]
```

c    **chkconfig** 在没有参数运行时，显示用法。如果加上服务名，那么就检查这个服务是否在当前运行级启动。如果是，返回 **true**，否则返回 **false**。如果在服务名后面指定了 **on**，**off** 或者 **reset**，那么 **chkconfig** 会改变指定服务的启动信息。**on** 和 **off** 分别指服务被启动和停止，**reset** 指重置服务的启动信息，无论有问题的初始化脚本指定了什么。**on** 和 **off** 开关，系统默认只对运行级 3，4，5 有效，但是 **reset** 可以对所有运行级有效。

### 参数用法：

**--add**    增加所指定的系统服务，让 **chkconfig** 指令得以管理它，并同时在系统启动的叙述文件内增加相关数据。

**--del**    删除所指定的系统服务，不再由 **chkconfig** 指令管理，并同时在系统启动的叙述文件内删除相关数据。

**--level<等级代号>**    指定读系统服务要在哪一个执行等级中开启或关毕。

等级 0 表示：表示关机  
 等级 1 表示：单用户模式  
 等级 2 表示：无网络连接的多用户命令行模式  
 等级 3 表示：有网络连接的多用户命令行模式  
 等级 4 表示：不可用  
 等级 5 表示：带图形界面的多用户模式  
 等级 6 表示：重新启动

需要说明的是，**level** 选项可以指定要查看的运行级而不一定是当前运行级。对于每个运行级，只能有一个启动脚本或者停止脚本。当切换运行级时，**init** 不会重新启动已经启动的服务，也不会再次去停止已经停止的服务。

**chkconfig --list [name]**：显示所有运行级系统服务的运行状态信息（**on** 或 **off**）。如果指定了 **name**，那么只显示指定的服务在不同运行级的状态。

**chkconfig --add name**：增加一项新的服务。**chkconfig** 确保每个运行级有一项启动(S)或者杀死(K)入口。如有缺少，则会从缺省的 **init** 脚本自动建立。

**chkconfig --del name**：删除服务，并把相关符号连接从/etc/rc[0-6].d 删除。

**chkconfig [--level levels] name**：设置某一服务在指定的运行级是被启动，停止还是重置。

### 运行级文件：

每个被 **chkconfig** 管理的服务需要在对应的 **init.d** 下的脚本加上两行或者更多行的注释。第一行告诉 **chkconfig** 缺省启动的运行级以及启动和停止的优先级。如果某服务缺省不在任

何运行级启动，那么使用 - 代替运行级。第二行对服务进行描述，可以用\ 跨行注释。

例如，random.init 包含三行：

```
# chkconfig: 2345 20 80
# description: Saves and restores system entropy pool for \
# higher quality random number generation.
```

#### 使用范例：

```
chkconfig --list          #列出所有的系统服务
chkconfig --add httpd     #增加 httpd 服务
chkconfig --del httpd     #删除 httpd 服务
chkconfig --level httpd 2345 on    #设置 httpd 在运行级别为 2、3、4、5 的情况下都是 on（开启）的状态
chkconfig --list          #列出系统所有的服务启动情况
chkconfig --list mysqld   #列出 mysqld 服务设置情况
chkconfig --level 35 mysqld on    #设定 mysqld 在等级 3 和 5 为开机运行服务，--level 35 表示操作只在等级 3 和 5 执行，on 表示启动，off 表示关闭
chkconfig mysqld on       #设定 mysqld 在各等级为 on，“各等级”包括 2、3、4、5 等级
```

#### 如何增加一个服务：

1.服务脚本必须存放在/etc/ini.d/目录下；

2.chkconfig --add servicename

在 chkconfig 工具服务列表中增加此服务，此时服务会被在/etc/rc.d/rcN.d 中赋予 K/S 入口了；

3.chkconfig --level 35 mysqld on

修改服务的默认启动等级。

## 4.2 yum命令详解

**yum**（全称为 Yellow dog Updater, Modified）是一个在 Fedora 和 RedHat 以及 SUSE 中的 **Shell 前端软件包管理器**。基於 RPM 包管理，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体包，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，而且命令简洁而又好记。

yum 的命令形式一般是如下：yum [options] [command] [package ...]

其中的[options]是可选的，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）等等。[command]为所要进行的操作，[package ...]是操作的对象。

#### 概括了部分常用的命令包括：

自动搜索最快镜像插件： `yum install yum-fastestmirror`  
安装 yum 图形窗口插件： `yum install yumex`  
查看可能批量安装的列表： `yum grouplist`

## 1 安装

`yum install` 全部安装  
`yum install package1` 安装指定的安装包 `package1`  
`yum groupinstall group1` 安装程序组 `group1`

## 2 更新和升级

`yum update` 全部更新  
`yum update package1` 更新指定程序包 `package1`  
`yum check-update` 检查可更新的程序  
`yum upgrade package1` 升级指定程序包 `package1`  
`yum groupupdate group1` 升级程序组 `group1`

## 3 查找和显示

`yum info package1` 显示安装包信息 `package1`  
`yum list` 显示所有已经安装和可以安装的程序包  
`yum list package1` 显示指定程序包安装情况 `package1`  
`yum groupinfo group1` 显示程序组 `group1` 信息 `yum search string` 根据关键字 `string` 查找安装包

## 4 删除程序

`yum remove package1` 删除程序包 `package1`  
`yum groupremove group1` 删除程序组 `group1`  
`yum deplist package1` 查看程序 `package1` 依赖情况

## 5 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; yum clean oldheaders)` 清除缓存目录下的软件包及旧的 `headers`

比如，要安装游戏程序组，首先进行查找：

#： `yum grouplist`

可以发现，可安装的游戏程序包名字是” Games and Entertainment “，这样就可以进行安装：

```
# : yum groupinstall "Games and Entertainment"
```

所有的游戏程序包就自动安装了。在这里 Games and Entertainment 的名字必须用双引号选定，因为 linux 下面遇到空格会认为文件名结束了，因此必须告诉系统安装的程序包的名字是 “Games and Entertainment” 而不是 “Games”。

此外，还可以修改配置文件/etc/yum.conf 选择安装源。可见 yum 进行配置程序有多方便了吧。更多详细的选项和命令，当然只要在命令提示行下面：man yum

## 4.3 SELinux两个命令

getsebool 与 setsebool 工具

说明：SELinux 规范了许多 boolean 数值清单档案，提供开启或关闭功能存取项目，而这些值都存放在/selinux/booleans/目录内相关档案，这些档案里的值只有两种：1（启用）或 0（关闭）

### 1) getsebool

说明：?出所有 selinux bool 数值清单?表与内容

使用方式：getsebool [ -a ]

例如以下范例：

```
#getsebool ftpd_disable_trans
ftpd_disable_trans -> off
#getsebool -a
NetworkManager_disable_trans -> off
allow_cvs_read_shadow -> off
allow_daemons_dump_core -> on
allow_daemons_use_tty -> off
allow_execheap -> off
allow_execmem -> on
allow_execmod -> off
.....
```

### 2) setsebool

说明：设定 selinux bool 数值清单?表与内容

使用方式：setsebool [ -P ] boolean value | bool1=val1 bool2=val2 bool3=val3.....

参数配置：-P 表示设定该项目永久套用

使用范?：

```
setsebool ftpd_disable_trans=on ( on 或者 1 )
setsebool -P ftpd_disable_trans=off ( off 或者 0 )
```



## 4.4 vim命令详解

vi 编辑器是所有 Unix 及 Linux 系统下标准的编辑器，它的强大不逊色于任何最新的文本编辑器，这里只是简单地介绍一下它的用法和一小部分指令。由于对 Unix 及 Linux 系统的任何版本，vi 编辑器是完全相同的，因此您可以在其他任何介绍 vi 的地方进一步了解它。Vi 也是 Linux 中最基本的文本编辑器，学会它后，您将在 Linux 的世界里畅行无阻。

### 1、vi 的基本概念

基本上 vi 可以分为三种状态，分别是命令模式(command mode)、插入模式(Insert mode)和底行模式(last line mode)，各模式的功能区分如下：

#### 1) 命令行模式 command mode)

控制屏幕光标的移动，字符、字或行的删除，移动复制某区段及进入 Insert mode 下，或者到 last line mode。

#### 2) 插入模式 (Insert mode)

只有在 Insert mode 下，才可以做文字输入，按「ESC」键可回到命令行模式。

#### 3) 底行模式 (last line mode)

将文件保存或退出 vi，也可以设置编辑环境，如寻找字符串、列出行号……等。

不过一般我们在使用时把 vi 简化成两个模式，就是将底行模式(last line mode)也算入命令行模式 command mode)。

### 2、vi 的基本操作

#### a) 进入 vi

在系统提示符号输入 vi 及文件名称后，就进入 vi 全屏幕编辑画面：

```
$ vi myfile
```

不过有一点要特别注意，就是您进入 vi 之后，是处于「命令行模式 (command mode)」，您要切换到「插入模式 (Insert mode)」才能够输入文字。初次使用 vi 的人都会想先用上下左右键移动光标，结果电脑一直哔哔叫，把自己气个半死，所以进入 vi 后，先不要乱动，转换到「插入模式 (Insert mode)」再说吧！

#### b) 切换至插入模式 (Insert mode) 编辑文件

在「命令行模式 (command mode)」下按一下字母「i」就可以进入「插入模式 (Insert mode)」，这时候你就可以开始输入文字了。

#### c) Insert 的切换

您目前处于「插入模式 (Insert mode)」，您就只能一直输入文字，如果您发现输错了字！想用光标键往回移动，将该字删除，就要先按一下「ESC」键转到「命令行模式 (command mode)」再删除文字。

#### d) 退出 vi 及保存文件

在「命令行模式 (command mode)」下，按一下「:」冒号键进入「Last line mode」，例如：

:w filename (输入「w filename」将文章以指定的文件名 filename 保存)

:wq (输入「wq」，存盘并退出 vi)

:q! (输入 q!，不存盘强制退出 vi)

### 3、命令行模式 (command mode) 功能键

#### 1) 插入模式

按「i」切换进入插入模式「insert mode」，按"i"进入插入模式后是从光标当前位置开始输入文件；

按「a」进入插入模式后，是从目前光标所在位置的下一个位置开始输入文字；

按「o」进入插入模式后，是插入新的一行，从行首开始输入文字。

#### 2) 从插入模式切换为命令行模式

按「ESC」键。

#### 3) 移动光标

vi 可以直接用键盘上的光标来上下左右移动，但正规的 vi 是用小写英文字母「h」、「j」、「k」、「l」，分别控制光标左、下、上、右移一格。

按「ctrl」+「b」：屏幕往"后"移动一页。

按「ctrl」+「f」：屏幕往"前"移动一页。

按「ctrl」+「u」：屏幕往"后"移动半页。

按「ctrl」+「d」：屏幕往"前"移动半页。

按数字「0」：移到文章的开头。

按「G」：移动到文章的最后。

按「\$」：移动到光标所在行的"行尾"。

按「^」：移动到光标所在行的"行首"

按「w」：光标跳到下个字的开头

按「e」：光标跳到下个字的字尾

按「b」：光标回到上个字的开头

按「#l」：光标移到该行的第#个位置，如：5l,56l。

#### 4) 删除文字

「x」：每按一次，删除光标所在位置的"后面"一个字符。

「#x」：例如，「6x」表示删除光标所在位置的"后面"6个字符。

「X」：大写的 X，每按一次，删除光标所在位置的"前面"一个字符。

「#X」：例如，「20X」表示删除光标所在位置的"前面"20个字符。

「dd」：删除光标所在行。

「#dd」：从光标所在行开始删除#行



#### 5) 复制

「yw」：将光标所在之处到字尾的字符复制到缓冲区中。

「#yw」：复制#个字到缓冲区

「yy」：复制光标所在行到缓冲区。

「#yy」：例如，「6yy」表示拷贝从光标所在的该行"往下数"6行文字。

「p」：将缓冲区内的字符贴到光标所在位置。注意：所有与"y"有关的复制命令都必须与"p"配合才能完成复制与粘贴功能。

#### 6) 替换

「r」：替换光标所在处的字符。

「R」：替换光标所到之处的字符，直到按下「ESC」键为止。

#### 7) 回复上一次操作

「u」：如果您误执行一个命令，可以马上按下「u」，回到上一个操作。按多次"u"可以执行多次回复。

#### 8) 更改

「cw」：更改光标所在处的字到字尾处

「c#w」：例如，「c3w」表示更改 3 个字

#### 9) 跳至指定的行

「ctrl」+「g」列出光标所在行的行号。

「#G」：例如，「15G」，表示移动光标至文章的第 15 行行首。

### 4、Last line mode 下命令简介

在使用「last line mode」之前，请记住先按「ESC」键确定您已经处于「command mode」下后，再按「:」冒号即可进入「last line mode」。

#### A) 列出行号

「set nu」：输入「set nu」后，会在文件中的每一行前面列出行号。

#### B) 跳到文件中的某一行

「#」：「#」号表示一个数字，在冒号后输入一个数字，再按回车键就会跳到该行了，如输入数字 15，再回车，就会跳到文章的第 15 行。

#### C) 查找字符

「/关键字」：先按「/」键，再输入您想寻找的字符，如果第一次找的关键字不是您想要的，可以一直按「n」会往后寻找到您要的关键字为止。

「?关键字」：先按「?」键，再输入您想寻找的字符，如果第一次找的关键字不是您想要的，可以一直按「n」会往前寻找到您要的关键字为止。

#### D) 保存文件

「w」：在冒号输入字母「w」就可以将文件保存起来。

#### E) 离开 vi

「q」：按「q」就是退出，如果无法离开 vi，可以在「q」后跟一个「!」强制离开 vi。

「qw」：一般建议离开时，搭配「w」一起使用，这样在退出的时候还可以保存文件。

## 5、vi 命令列表

### 1、下表列出命令模式下的一些键的功能：

h	左移光标一个字符
l	右移光标一个字符
k	光标上移一行
j	光标下移一行
^	光标移动至行首
O	数字"0"，光标移至文章的开头
G	光标移至文章的最后
\$	光标移动至行尾
Ctrl+f	向前翻屏
Ctrl+b	向后翻屏
Ctrl+d	向前翻半屏
Ctrl+u	向后翻半屏
i	在光标位置前插入字符
a	在光标所在位置的后一个字符开始增加
o	插入新的一行，从行首开始输入
ESC	从输入状态退至命令状态
x	删除光标后面的字符
#x	删除光标后的#个字符
X	(大写 X)，删除光标前面的字符
#X	删除光标前面的#个字符
dd	删除光标所在的行
#dd	删除从光标所在行数的#行
yw	复制光标所在位置的一个字
#yw	复制光标所在位置的#个字
yy	复制光标所在位置的一行
#yy	复制从光标所在行数的#行
p	粘贴
u	取消操作
cw	更改光标所在位置的一个字
#cw	更改光标所在位置的#个字

### 2、下表列出行命令模式下的一些指令

w filename	储存正在编辑的文件为 filename
wq filename	储存正在编辑的文件为 filename，并退出 vi
q!	放弃所有修改，退出 vi
set nu	显示行号
/或?	查找，在/后输入要查找的内容

**n** 与/或?一起使用，如果查找的内容不是想要找的关键字，按 **n** 或向后（与/联用）或向前（与?联用）继续查找，直到找到为止。

对于**第一次**用 vi，有几点注意要提醒一下：

1、用 vi 打开文件后，是处于「命令行模式（command mode）」，您要切换到「插入模式（Insert mode）」才能够输入文字。切换方法：在「命令行模式（command mode）」下按一下字母「i」就可以进入「插入模式（Insert mode）」，这时候你就可以开始输入文字了。

2、编辑好后，需从插入模式切换为命令行模式才能对文件进行保存，切换方法：按「ESC」键。

3、保存并退出文件：在命令模式下输入:**wq** 即可！（别忘了 wq 前面的:）

## 4.5 touch命令详解

linux 中新建文件命令 touch 用法详解：

linux 中的 touch 命令一般用来修改文件的时间，或者新建一个不存在的文件。

语法如下：

```
touch [-acm][-r ref_file(参照文件)]-t time(时间值) file(文件名)
```

例子：

```
touch file1.txt 更新 file1.txt 的存取和修改时间
touch -c file1.txt 如果 file1.txt 不存在，不创建文件
touch -r ref_file file1.txt 更新 file1.txt 的时间戳和 ref+file 相同
touch -t 0911252234.40 file1.txt 设定文件的时间为 09 年 11 月 25 日 22 点 34 分 40 秒
```

个参数说明如下：

-a 修改文件 file 的存取时间。  
 -c 不创建文件 file。  
 -m 修改文件 file 的修改时间  
 -r ref\_file 将参照文件 ref\_file 相应的时间戳记的数值作为指定文件 file 时间戳记的新值。  
 -t time 使用指定的时间值 time 作为指定文件 file 相应时间戳记的新值。此处的 time 规定为如下形式的十进制数： **[[CC]YY]MMDDhhmm[.SS]**

这里，CC 为年数中的前两位，即“世纪数”；YY 为年数的后两位，即某世纪中的年数。如果不给出 CC 的值，则 touch 将把年数 CCYY 限定在 1969--2068 之内。MM 为月数，DD 为天将把年数 CCYY 限定在 1969--2068 之内。MM 为月数，DD 为天数，hh 为小时数(几点)，mm 为分钟数，SS 为秒数。此处秒的设定范围是 0--61，这样可以处理闰秒。这些数字组成的时间是环境变量 TZ 指定的时区中的一个时间。由于系统的限制，早于 1970 年 1 月 1 日的时间是错误的。

## 参考文献

---

感谢以下文章的编作者，没有你们的铺路，我或许会走得很艰难，参考不分先后，贡献同等珍贵。

【1】Red Hat Linux 服务器管理及配置实战详解

【2】CentOS 安装 FTP 配置总结

地址: <http://bolg.malu.me/html/2011/1798.html>

【3】CentOS 安装 VSFTPD FTP 服务器

地址: <http://www.embest.net/index.php/archives/23>

【4】CentOS vsftpd 虚拟用户创建步骤

地址: <http://peishuangcai.blog.51cto.com/1316844/463538>

【5】centos 安装 VSFTP

地址: <http://home.phpchina.com/blog-1628-36583.html>

【6】CentOS6 vsFTP 安装与配置

地址: <http://icedot.blog.51cto.com/61369/772452>