Linux系统优化应用指南

目录

[1.1添加一个普通用户账号 3](#_Toc11518)

[1.2 使用更新源的国内镜像 3](#_Toc25934)

[1.3 安装必要的软件包 4](#_Toc841)

[1.4 安装后的基本调优及安全设置 5](#_Toc23237)

[1.5 精简开机系统启动的服务为 6](#_Toc28027)

[1.6 更改SSH服务远程登录的配置 6](#_Toc8586)

[1.7 修改中文显示 9](#_Toc22968)

[1.8 服务器时间同步 10](#_Toc10573)

[1.9 加大服务器文件描述符 10](#_Toc12965)

[1.10 调整内核参数/etc/sysctl.conf 12](#_Toc19718)

[1.10 定时清理clientmqueue目录垃圾文件防止占满磁盘空间 16](#_Toc16025)

[1.11 隐藏系统版本号和版本信息 17](#_Toc15337)

[1.12 锁定关键文件 17](#_Toc20811)

[1.13 设置Linux的命令行记录数 17](#_Toc7905)

[1.14 Linux系统优化小结 18](#_Toc11231)

# 1.1添加一个普通用户账号

|  |
| --- |
| 使用如下命令添加一个普通用户账号，并为其设置口令:  useradd oldboy  passwd oldboy  提示  [oldboy@oldboylinux ~]$ <== 普通用户的提示符  [root@oldboylinux ~]# <==超级管理员的提示符  su - oldboy #切换当前用户到oldboy  whoami #显示当前终端的系统用户  提示：两个普通用户之间进行切换是要输入对方用户的密码，而root切换到任何普通用户下，都不用输入密码,普通用户只能改自己的密码  免交换模式下添加普通用户账号  echo "123456" | passwd --stdin oldboy && history -d 171 |

# 1.2 使用更新源的国内镜像

Linux下房本安全软件的优秀工具叫做YUM工具，Linux的二进制软件包一般是rpm包.类似Windows下exe程序

通过yum工具来安装软件，默认获取rpm包的软件的配置从国外centos官方源下载，因此，我们yum安装软件速度回比较慢，因此需要把默认获取rpm包的配置从国外官方源，改成国内的优秀yum源

当centOS安装完毕，首先应该修改更新源，并升级到最小的系统。

|  |
| --- |
| 国内速度较快的常用更新源如下:  ----http://mirrors.suhu.com/  ----http://mirrors.163.com/  #163yum源安装配置方法  cd /etc/yum.repos.d/  /bin/mv CentOS-Base.repo Centos-Base.repo.ori  wget http://mirrors/.help/CentOS6-Base-163.repo  /bin/mv CentOS6-Base-163.repo CentOS-Base.repo  运行yum makecache 生产缓存  yum upgrade #给系统打补丁  说明:我们现在使用的互联网上的门户网站提供的yum源，将来我们也可以把iso镜像或光盘配置成yum源，你还可以自己配置一个像门户网站提供的这种yum源的服务器。如果不能上网，你还可以将系统的ISO配置成本地yum仓库  #suhu yum源安装配置方法 |

# 1.3 安装必要的软件包

安装必要的软件工具:

#新系统安装常用命令

|  |
| --- |
| yum install tree telnet dos2unix sysstat lrzsz -y  yum install lrzsz sysstat -y #这些也可以在安装时选上的  rpm -aq sysstat lrzsz |

另外，如果在安装的时候落下了安装需要的软件包组(或者公司例别人装的系统),可以在这里

执行如下命令来安装

|  |
| --- |
| #yum grouplist  #yum groupinstall "Development Tools"  #yum groupinstall "x software development"  提示：yum的用法的详细介绍请参见其他相关文档 |

# 1.4 安装后的基本调优及安全设置

|  |
| --- |
| a:简单优化Linux  1:关闭Selinux  sed -i 's/SELINUX=enforcing/SELINUX=diabled/' /etc/selinux/config  <=修改配置文件则永久生效，但是必须要重启系统  grep SELINUX=disabled /etc/selinux/config  setenforce 0 <=临时生效的命令  getenforce <=查看SELinux当前状态  2:设定runlevel为3  grep 3:initdefault /etc/inittab  提示：工作中常用3级别，即文本模式  作业：  /etc/inittab文件的详细注释，  Linux的7中运行模式总结，  Linux系统启动流程总结  命令总结：  1)runlevel #查看当前系统的运行级别  2)init #切换运行级别 |

# 1.5 精简开机系统启动的服务为

命令的处理方法:

1)ntsysv

2)setup-system service.

3)脚本一键完成处理.

|  |
| --- |
| 操作命令  方法一:  LANG=en  for oldboy in `chkconfig --list|grep 3:on|awk '{print $1}';do chkconfig -level 3 $oldboy off;done  for oldboy in crond network syslog sshd; do chkconfig --level 3 $oldboy on;done  chkconfig --list|grep 3:on  方法二:  for oldboy in `chkconfig --list|grep 3:on|awk '{print $1} | grep -vE cron|network|syslog|sshd'`;do chkconfig --level 3 $oldboy off;done  4)临时关闭iptables防火墙  /etc/init.d/iptables stop  将来学了iptables后再开启，一般有外网ip的服务器需要开启防火墙，但是高并发的业务不能开，只能在前端加硬件防火墙 |

# 1.6 更改SSH服务远程登录的配置

Windows:默认远程端口和用户

管理员:administrator,普通用户:guset

默认端口号:3389

linux：远程连接默认端口和超级管理员

远程连接默认端口:22端口

超级管理员:root

|  |
| --- |
| ls /etc/ssh/sshd\_config  cp /etc/ssh/sshd\_config /etc/ssh/sshd\_config.ori  vim /etc/ssh/ssh\_config修改配置文件内容如下  Port 52113 #修改SSH默认登录端口  sed -i 's@#PermitRootLogin yes@PermitRootLogin no@' /etc/ssh/sshd\_config  #禁止ROOT通过SSH登录系统  sed -i 's@#UseDNS yes@UseDNS no@' /etc/ssh/ssh\_config  #禁止使用DNS功能  sed -i 's@#PermitEmptyPasswords no@PermitEmptyPasswords no@ /etc/ssh/ssh\_config  方法二:  cat << /etc/ssh/ssh\_config >> EOF  Port 52113  PermitRootLogin no  UseDNS no  PermitEmptyPasswords no  EOF  #禁止空密码登录  #修改完成后，保存退出，重新启动SSH服务，配置就生效了  /etc/init.d/sshd reload 或者 /etc/init.d/ssh restart  #别忘了关闭防火墙  /etc/init.d/iptables stop  netstat -an | grep 192.168.100.140 |

#相关参数说明

|  |  |
| --- | --- |
| 参数 | 说明 |
| Port | 指定sshd(8)守护进程监听的端口号，默认为22，可以指定使用多条指令监听多个端口,默认将在本机的所有网络端口上监听，但是可以通过ListenAddress指定只是某个特定的接口上监听，端口范围:0-65535之间，不能和已有的服务端口冲突,议用稍微大点的端口 |
| PermitEmptyPassword | 是否允许密码为空的用户远程登录，默认为"no" |
| PermitRootLogin | 是否允许root登录，可用值如下:"yes(默认)表示允许,no表机制使用,"without-password"表示机制使用密码认证登录 "forced-commands-only"表示只有在指定了command选项的情况下才允许使用公钥认证登录，同时其他认证防范全部禁止，这个值常用于做远程备份之类的事情 |
| UseDNS | 指定sshd(8)是否应该对远程主机名进行反向解析，以检查此机名是否与其IP地址真实对应，默认值为"yes" |

提示:在重启后，当前的crt仍然可以连接服务器，直到退出重新登录为止.

6) 将oldboy普通账户加入到sudo管理

为了安全及管理的方便，可将需要有root权限的用户名加入sudo管理。这样用户通过自己的普通账户登录。就可以将root的权限来管理整个系统，而不需要有root账户及密码，

|  |
| --- |
| 执行如下命令，就可打开visodu的配置文件  visudo #相当于直接编辑/etc/sudoer,但用命令方式更安全，推荐使用  比如：  oldboy ALL=(ALL) /usr/sbin/useradd  oldboy ALL=(ALL) NOPASSWD: ALL #将oldboy模仿成和root一样的权限使用，  oldboy ALL=(ALL) /usr/sbin/useradd,/usr/sbin/userdel  提示：  #NOPASSWD:免密码登陆和切换  用户 机器=(授权那个角色的权利) 授权的权限名称  小技巧:使用"行号gg",可直接却换到指定的行  添加环境变量的方法：PATH=/oldboy/:$PATH |

# 1.7 修改中文显示

什么是字符集

简单的说就是一套文字符号及其编码，常用的字符集有:

GBK 定长 双字节 不是国际标准，支持的系统不少

UFT-8 非定长 1-4字节 广泛支持 MySQL也使用utf-8

通过快捷的命令方式在/etc/sysconfig/i18n中添加如下内容

|  |
| --- |
| cat /etc/sysconfig/i18n  LANG="en\_US.UTF-8"  cp /etc/sysconfig/i18n /etc/sysconfig/i18n.ori  echo 'LANG="zh\_CN.UTF-8"'>/etc/sysconfig/i18n  cat /etc/sysconfig/i18n  source /etc/sysconfig/i18n |

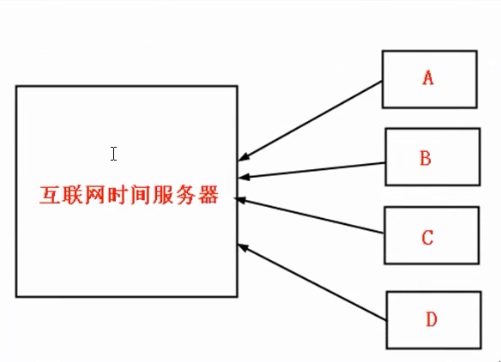
#提示：这个中文你现实配置要跟你自己的SSH客户端的配置一致，默认情况下就是我演示的配置

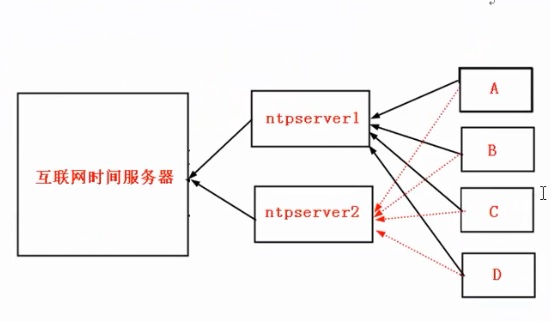
# 1.8 服务器时间同步

|  |
| --- |
| 1)手动同步  /usr/sbin/ntpdate time.nist.gov  2)定时任务  echo 'time sysnc by oldboy at 2010-2-1' >>/var/spool/cron/root  echo '\*/5 \* \* \* \* /usr/sbin/ntpdate time.nist.gov'>/dev/null 2>&1 >>/var/spool/cron/root  #这个命令其实就是写一个定时任务，相当于执行crontab -e 然后加入内容:\*/5 \* \* \* \*  /usr/sbin/ntpdate time.windows.com >/dev/null 2>&1保存退出，有关crontab定时任务  后文有详细讲解，大家不必再这里纠缠 |

crontab -l #查看定时任务的配置信息

时间服务器架构图





作业:ntp服务器的配置

# 1.9 加大服务器文件描述符

|  |
| --- |
| vim /etc/security/limits.conf  在末尾加上如下配置  \* - nofile 65535 或者直接执行echo '\* - nofile 65535'>> /etc/security/limits.conf  tail -1 /etc/security/limits.conf  注明:配置完成后，重新登陆即可查看  ulimit -n  提示:也有人把ulimit -SHn 65535 命令加入到/etc/rc.local 然后每次重起生效该方法可用于centos5以上版本。但是在6以上版本，就不能用了  cat >>/etc/rc.local<< EOF  #open files  ulimit -HSn 65535  #stack size  ulimit -s 65535  EOF  文件描述符  对于内核而言，所有打开的文件都通过文件描述符引用，文件描述符是一个非负整数.当打开一个现有文件或创建一个新文件时，内核向进程返回一个文件描述符，当读或写一个文件时，使用open或creat返回的文件描述符标识该文件，将其作为参数传递给read或write。  按照惯例，unix系统shell使用描述符0余进程的标准输入相关联，文件描述符1与标准输出相关联，文件描述符2余标准出错相关联。这时各种shell以及很多应用程序使用的惯例，而与unix内核无关，如果不遵守这种惯例，那么很多Unix系统应用程序就不能正常工作。  文件描述符是由无符号整数表述的句柄，进程使用它来标识打开的文件，文件描述符与包括相关信息(如文件的打开模式，文件的位置类型，文件的初始化类型等)的文件对象相关联，  这些信息被称为文件的上下文，文件描述符的有效范围是0到OPEN\_MAX |

# 1.10 调整内核参数/etc/sysctl.conf

linux服务器内核参数优化

说明：本优化适合Apache，Nginx，squid多种WEB应用，特殊的业务也可能需要略做调整所谓内核优化，主要是在Linux系统中针对业务服务应用而进行的系统内很参数优化，优化并无特殊标准，

|  |
| --- |
| 下面以常见生产环境Linux的内核优化为例讲解，仅供大家参考:  net.ipv4.tcp\_fin\_timout = 2  net.ipv4.tcp\_tw\_reuse = 1  net.ipv4.tcp\_tw\_recycle =1  net.ipv4.tcp\_syncookies = 1  net.ipv4.tcp\_keepalive\_time = 600  net.ipv4.ip\_local\_port\_range = 4000 65000  net.ipv4.tcp\_max\_syn\_backlog = 16384  net.ipv4.tcp\_max\_tw\_buckets = 36000  net.ipv4.route.gc\_timeout = 100  net.ipv4.tcp\_syn\_retries = 1  net.ipv4.tcp\_synack\_retries = 1  net.core.somaxconn = 16384  net.core.netdev\_max\_backlog = 16384  net.ipv4.tcp\_max\_orphans = 16384  #以上参数是对iptables防火墙的优化，防火墙不开会提示，可忽略不理  防火墙的优化在5.8上是  net.ipv4.ip\_conntrack\_max = 25000000  net.ipv4.netfilter.ip\_conntrack\_max = 25000000  net.ipv4.netfilter.ip\_conntrack\_tcp\_timeout\_established = 180  net.ipv4.netfilter.ip\_conntrack\_tcp\_timeout\_time\_wait = 120  net.ipv4.netfilter.ip\_conntrack\_tcp\_timeout\_close\_wait = 60  net.ipv4.netfilter.ip\_conntrack\_tcp\_timeout\_fin\_wait = 120  防火墙的优化，在6.4或6.5上是  net.nf\_conntrack\_max = 25000000  net.netfilter.nf\_conntrack\_max = 25000000  net.netfilter.nf\_conntrack\_tcp\_timeout\_established = 180  net.netfilter.nf\_conntrack\_tcp\_timeout\_time\_wait = 120  net.netfilter.nf\_conntrack\_tcp\_timeout\_close\_wait = 60  net.netfilter.nf\_conntrack\_tcp\_timeout\_fin\_wait = 120  将上面的内核参数值加入/etc/sysctl.conf文件中，然后执行如下命令使之生效，sysctl -p,6.5下面，必须开启iptables,才不会报error |

下面是对sysctl.conf内核文件中的参数含义进行介绍

|  |  |
| --- | --- |
| 参数 | 说明 |
| net.ipv4.tcp\_fin\_timout | 表示如果套接字由本端要求关闭，这个参数决定了它保存在FIN-WAIT-2状态的时间，默认值为60秒,该参数对应系统路径为该参数对应系统路径为:/proc/sys/net/ipv4/tcp\_tw\_reuse 0. |
| net.ipv4.tcp\_tw\_reuse | 表示开启重用,允许TIME-WAIT socket重新用于新的TCP连接, 默认值为0，表示关闭，该参数对应系统路径为:/proc/sys/net/ipv4/tcp/tcp\_tw\_reuse 0 |
| net.ipv4.tcp\_tw\_recycle | 表示开启TCP连接中TIME-WAIT socket的快速回收,该参数对应系统路径为:/proc/sys/net/ipv4/tcp\_tw\_recycle 默认值为0表示关闭,提示:reuse和recycle两个参数为了防止生产环境下web,squid等，time\_wait过多设置的 |
| net.ipv4.tcp\_syncookies | 表示开启SYN Cookies功能，当出现SYN等待队列溢出时，启用cookies来处理，可防范少量SYN攻击,centos5系列默认值为1表示开启，因此这个参数也可以不添加,该参数对应系统路径 为:/proc/sys/net/ipv4/tcp\_keepalive\_time 默认值为7200 |
| net.ipv4.tcp\_keepalive\_time  net.ipv4.tcp\_keepalive\_time | 表示当keepalive起用的时候,TCP发送keepalive消息的频度，缺省值是2个小时，改为10分钟，该蚕食对应系统路径为:/proc/sys/net/ipv4/tcp\_keepalive\_time,默认值为7200秒 |
| ip\_local\_port\_range | 选项用来设定允许系统打开的端口范围.既用于向外连接的端口范围，该蚕食对应路径为:/proc/sys/net/ipv4/ip\_local\_range 32768 61000 |
| net.ipv4\_max\_syn\_backlog | 表示syn队列的长度，默认值为1024，加大队列的长度为8192 |
| net.ipv4.tcp\_max\_tw\_buckets | 标识系统同时保持TIME\_WAIT套接字的最大数量，如果超过这个数字，TIME\_WAIT套接字将立刻被清除并打印警告默认值为180000，对于Apache,Nginx等服务器来说可以调整低一些，如:改为5000-30000；不同业务的服务器也可以给大一点，比如lvs,suqid.上几行的参数可以很好解决TIME\_WAIT套接字数量,但是对于suqid,效果却不大此项参数可以控制TIME\_WAIT套接字的数据,避免suqid服务器被大量的TIME\_WAIT套接字拖死.该参数对应系统路径为:/proc/sys/net/ipv4/tcp\_synack\_retries默认值为5 |
| net.ipv4.tcp\_max\_orphans | 该选项用于设定系统中最多有多少个TCP套接字不被关联到任何一个用户文件句柄上,如果超过这个数字,孤立连接将立即被复位并打印警告信息,这个限制只是为了防止简单的Dos攻击,不能过分依靠这个限制甚至人为较少这个值,更多的情况是增加这个值，该参数对应系统路径为:/proc/sys/net/ipv4/tcp\_max\_orphans 65535 |
| net.core.somaxconn | 选项默认值为128,这个参数用于调节系统同时发起的TCP连接数,在高并发的请求中,默认的值可能导致链接超时或者重传,因此,需要结合并发请求数来调节此值,该参数对应系统路径为:/proc/sys/net/core/somaxconn 128 |
| net.core.netdev\_max\_backlog | 选项表示当每个网络接口接受数据包的速率比内核处理这些包的速率低时,允许发送到队列的数据包的最大数目该参数对应系统路径为:/proc/sys/net/core/ netdev\_max\_backlog默认值1000 |

sysctl.conf内核文件中的参数含义1.jpg

http://yangrong.blog.51cto.com/6945369/1321594

# 1.10 定时清理clientmqueue目录垃圾文件防止占满磁盘空间

/var/spool/clientmqueue/sendmail 邮件服务邮件临时存放地点,c5.8默认就有sendmailc6.4默认没有sendmail

|  |
| --- |
| 1)手动清理方法  find /var/spoll/clientmqueue/ -type f |xargs rm -rf  ll /var/spool/clientmqueue|wc -l  2)定时清理方法  echo "find /var/spool/clientmqueue/ -type f |xarge rm -rf" >/server/scripts/del\_sys\_file.sh  cat /server/scripts/del\_sys\_file.sh  echo "00 00 \* \* \* /bin/sh /server/scripts/del\_sys\_file.sh >/dev/null 2>&1" >>  /var/spool/cron/root  echo "00 00 \* \* 0 /bin/sh /server/scripts/del\_sys\_file.sh >/dev/null 2>&1" >>  /var/spool/cron/root  crontab -l|tail -2  提示：使用命令df -i来查看硬盘各个挂载点的innodes是否已满，如果%IUse%参数的值是100%,说明innodes的值已经满了， |

# 1.11 隐藏系统版本号和版本信息

|  |
| --- |
| >/etc/issue 或者cat /dev/null >/etc/issue |

1.12 锁定关键文件

|  |
| --- |
| chattr +i /etc/passwd  chattr +i /etc/shadow  chattr +i /etc/group  chattr +i /etc/gshadow  chattr +i /etc/inittab  #解锁关键文件  chattr -i /etc/passwd  chattr -i /etc/shadow  chattr -i /etc/group  chattr -i /etc/gshadow  chattr -i /etc/inittab |

提示：如果再想提升安全度的话，你也可以将/usr/bin/chattr移动到其他地方，或是直接删除，mv /usr/bin/chattr /usr/bin/oldboy1

lsattr /etc/passwd #查看/etc/passwd里的内容是否加锁了

# 1.13 设置Linux的命令行记录数

设置Linux命令行的记录数实例命令如下，注意此处的配置仅临时生效

|  |
| --- |
| export HISTSIZE=5  history |

设置用户的命令行历史记录文件(~/.bash\_history)记录指定命令数量的实例命令如下，注意此处的配置仅仅临时生效

|  |
| --- |
| export HISTFILESIZE=5  cat ~/.bash\_history  连接的超时时间控制变量  TMOUT=10(生产环境中可以设置为10分钟或是20分钟)  上述系统控制变量永久生效的配置:  echo 'export TMOUT=300' >>/etc/profile  echo 'export HISTSIZE=5' >>/etc/profile  echo 'export HISTFILESIZE=5' >>/etc/profile  tail -3 /etc/profile  source /etc/profile #使得配置文件生效 |

# 1.14 Linux系统优化小结

1)不用root,添加普通用户，通过sudo授权管理

2)更改默认的远程连接ssh服务端口及禁止root用户远程连接

3)定时自动更新服务器时间，

4)配置yum更新源,从国外改到国内更新源下载安装rpm包

5)关闭selinux及iptables(iptables工作场景如果有wan ip一般要打开，高并发除外)

6)调整文件描述符的数量,进程及文件的打开都会消耗文件描述符

7)定时自动清理/var/spool/clientmqueue/目标垃圾文件，防止inodes节点被占满(c6.4

默认没有sendmail,因此，可以不配)

8)精简开机自启动服务(crond,ssh,network,syslog(rsyslog)C5.8是syslog)

9)linux内核参数优化/etc/sysctl.conf,执行sysctl -p生效

10)更改字符集，支持中文，但建议还是用英文字符集，防止乱码问题，不要使用中文

11)锁定关键系统文件

12)隐藏系统的版本和系统信息