

晴川のLinux笔记-Linux命令IV

用户管理

Linux 踢出其他正在 SSH 登陆用户

查看系统在线用户

```
1 [root@Linux ~]#w
2  20:40:18 up 1 day, 23 min,  4 users,  load average:
   0.00, 0.00, 0.00
3  USER      TTY      FROM          LOGIN@   IDLE   JCPU
   PCPU WHAT
4  root      tty1      -             Fri09    10:10m  0.34s
   0.34s -bash
5  root      pts/2     192.168.31.124 10:30    4:39m  0.99s
   0.99s -bash
6  root      pts/3     192.168.31.124 19:55    0.00s  0.07s
   0.00s w
7  root      pts/4     192.168.31.124 19:55    4:52   0.16s
   0.16s -bash
```

查看当前自己占用终端

```
1 [root@Linux ~]# who am i
2 root      pts/4          2016-10-30 19:55 (192.168.31.124)
```

用 pkill 命令剔除对方

```

1 [root@Linux ~]# pkill -kill -t pts/2
2 [root@Linux ~]# w
3  20:44:15 up 1 day, 27 min,  3 users,  load average:
  0.01, 0.03, 0.01
4  USER      TTY      FROM          LOGIN@   IDLE
  JCPU   PCPU WHAT
5  root      tty1      -             Fri09    10:14m
  0.34s   0.34s -bash
6  root      pts/3     192.168.31.124 19:55    51.00s
  1.43s   1.35s vim base.md
7  root      pts/4     192.168.31.124 19:55     0.00s
  0.21s   0.00s w

```

如果最后查看还是没有干掉，建议加上 -9 强制杀死。 [root@Linux ~]# pkill -9 -t pts/2

使用脚本创建有 sudo 权限的用户

```

1 cat >./create_user.sh <<-'EOF'
2 #!/bin/bash
3 arg="ceshi"
4 if id ${arg}
5 then
6     echo "the username is exist!"
7 else
8     useradd $arg
9     echo "ceshi_password" | passwd --stdin $arg
10    echo "${arg} ALL=(ALL) NOPASSWD:ALL" >
    /etc/sudoers.d/${arg}
11 fi
12 EOF

```

以上脚本会创建用户 `ceshi` 同时用户的密码为 `ceshi_password` , 并且此用户有 sudo 权限

无交互式修改用户密码

```
1 | echo "123456" | passwd --stdin root
```

网卡 bond

Linux 多网卡绑定

网卡绑定 mode 共有七种 (0~6) bond0、bond1、bond2、bond3、bond4、bond5、bond6

常用的有三种

- mode=0: 平衡负载模式, 有自动备援, 但需要"Switch"支援及设定。
- mode=1: 自动备援模式, 其中一条线若断线, 其他线路将会自动备援。
- mode=6: 平衡负载模式, 有自动备援, 不必"Switch"支援及设定。

需要说明的是如果想做成 mode 0 的负载均衡, 仅仅设置这里 options bond0 miimon=100 mode=0 是不够的, 与网卡相连的交换机必须做特殊配置 (这两个端口应该采取聚合方式), 因为做 bonding 的这两块网卡是使用同一个 MAC 地址。从原理分析一下 (bond 运行在 mode 0 下) :

mode 0 下 bond 所绑定的网卡的 IP 都被修改成相同的 mac 地址, 如果这些网卡都被接在同一个交换机, 那么交换机的 arp 表里这个 mac 地址对应的端口就有多 个, 那么交换机接受到发往这个 mac 地址的包应该往哪个端口转发呢? 正常情况下 mac 地址是全球唯一的, 一个 mac 地址对应多个端口肯定使交换机迷惑了。所以 mode0 下的 bond 如果连接到交换机, 交换机这几个端口应该采取聚合方式 (cisco 称为 ethernetchannel, foundry 称为 portgroup), 因为交换机做了聚合后, 聚合下的几个端口也被捆绑成一个 mac 地址。我们的解决办法是, 两个网卡接入不同的交换机即可。

mode6 模式下无需配置交换机, 因为做 bonding 的这两块网卡是使用不同的 MAC 地址。

其他设置

时区及时间

时区就是时间区域，主要是为了克服时间上的混乱，统一各地时间。地球上共有 24 个时区，东西各 12 个时区（东 12 与西 12 合二为一）。

UTC 和 GMT

时区通常写成 +0800，有时也写成 GMT +0800，其实这两个是相同的概念。

GMT 是格林尼治标准时间（Greenwich Mean Time）。

UTC 是协调世界时间（Universal Time Coordinated），又叫世界标准时间，其实就是 0000 时区的时间。

Linux 下调整时区及更新时间

修改系统时间

```
1 $ln -sf /usr/share/zoneinfo/Asia/Shanghai /etc/localtime
```

修改 /etc/sysconfig/clock 文件，修改为：

```
1 ZONE="Asia/Shanghai"
2 UTC=false
3 ARC=false
```

校对时间

```
1 $ntpdate cn.ntp.org.cn
```

设置硬件时间和系统时间一致并校准

```
1 $/sbin/hwclock --systohc
```

定时同步时间设置

凌晨 5 点定时同步时间

```
1 echo "0 5 * * * /usr/sbin/ntpdate cn.ntp.org.cn" >>
  /var/spool/cron/root
2 或者
3 echo "0 5 * * * /usr/sbin/ntpdate 133.100.11.8" >>
  /var/spool/cron/root
```

登录提示信息

修改登录前的提示信息

(1) 系统级别的设置方法 /etc/issue

使用此方法时远程 ssh 连接的时候并不会显示

```
1 在登录系统输入用户名之前，可以看到上方有 WELCOME..... 之类的信
  息，这里会显示 LINUX 发行版本名称，内核版本号，日期，机器信息等等
  信息，
2
3 首先打开 /etc/issue 文件，可以看到里面是这样一段"welcome to
  <LINUX 发行版本名称> -kernel 后接各项参数、"
4
5 参数的各项说明：
6 \r 显示 KERNEL 内核版本号；
7 \l 显示虚拟控制台号；
8 \d 显示当前日期；
9 \n 显示主机名；
10 \m 显示机器类型，即 CPU 架构，如 i386 等；
11
12 可以显示所有必要的信息：
13
14 welcome to <LINUX 发行版本名称> -kernel \r (\l) \d \n \m.
```

修改登录成功后的信息

motd(message of the day)

修改登录成功后的提示信息在此文件中添加内容即可：/etc/motd

如：

```
1 ///////////////////////////////////////////////////  
2 系统初始化配置提示  
3 xxxx  
4  
5 应用联系人: xxxx 联系方式: xxxx  
6 ///////////////////////////////////////////////////
```

CentOS 7 vs CentOS 6 的不同

运行相关

桌面系统

```
1 [CentOS6] GNOME 2.x  
2 [CentOS7] GNOME 3.x (GNOME Shell)
```

文件系统

```
1 [CentOS6] ext4  
2 [CentOS7] xfs
```

内核版本

```
1 [CentOS6] 2.6.x-x  
2 [CentOS7] 3.10.x-x
```

启动加载器

```
1 [CentOS6] GRUB Legacy (+efibootmgr)  
2 [CentOS7] GRUB2
```

防火墙

```
1 [CentOS6] iptables  
2 [CentOS7] firewalld
```

默认数据库

- 1 [CentOS6] MySQL
- 2 [CentOS7] MariaDB

文件结构

- 1 [CentOS6] /bin, /sbin, /lib, and /lib64 在 / 下
- 2 [CentOS7] /bin, /sbin, /lib, and /lib64 移到 /usr 下

主机名

- 1 [CentOS6] /etc/sysconfig/network
- 2 [CentOS7] /etc/hostname

时间同步

- 1 [CentOS6]
- 2 \$ ntp
- 3 \$ ntpq -p
- 4
- 5 [CentOS7]
- 6 \$ chrony
- 7 \$ chronyc sources

修改时区

- 1 [CentOS6]
- 2 \$ vim /etc/sysconfig/clock
- 3 ZONE="Asia/Shanghai"
- 4 UTC=fales
- 5 \$ sudo ln -s /usr/share/zoneinfo/Asia/Shanghai
- 6 /etc/localtime
- 7
- 7 [CentOS7]
- 8 \$ timedatectl set-timezone Asia/Shanghai
- 9 \$ timedatectl status

修改语言

```
1 [CentOS6]
2 $ vim /etc/sysconfig/i18n
3     LANG="en_US.utf8"
4 $ /etc/sysconfig/i18n
5 $ locale
6
7 [CentOS7]
8 $ localectl set-locale LANG=en_US.utf8
9 $ localectl status
```

重启关闭

```
1 1) 关闭
2 [CentOS6]
3 $ shutdown -h now
4
5 [CentOS7]
6 $ poweroff
7 $ systemctl poweroff
8
9 2) 重启
10 [CentOS6]
11 $ reboot
12 $ shutdown -r now
13
14 [CentOS7]
15 $ reboot
16 $ systemctl reboot
17
18 3) 单用户模式
19 [CentOS6]
20 $ init s
21
22 [CentOS7]
23 $ systemctl rescue
24
25 4) 启动模式
26 [CentOS6]
27 [GUI]
28 $ vim /etc/inittab
```



```
29 id:3:initdefault:
30 [CUIGUI]
31 $ startx
32
33 [CentOS7]
34 [GUICUI]
35 $ systemctl isolate multi-user.target
36 [CUIGUI]
37 $systemctl isolate graphical.target
38 默认
39 $ systemctl set-default graphical.target
40
41 [CentOS6]
42 $ chkconfig service_name on/off
43
44 [CentOS7]
45 $ systemctl enable service_name
46 $ systemctl disable service_name
```

服务一览

```
1 [CentOS6]
2 $ chkconfig --list
3
4 [CentOS7]
5 $ systemctl list-unit-files
6 $ systemctl --type service
```

强制停止

```
1 [CentOS6]
2 $ kill -9 <PID>
3
4 [CentOS7]
5 $ systemctl kill --signal=9 sshd
```

网络

网络信息

```
1 [CentOS6]
2 $ netstat
3 $ netstat -I
4 $ netstat -n
5
6 [CentOS7]
7 $ ip -s 1
8 $ ss
```

IP 地址 MAC 地址

```
1 [CentOS6]
2 $ ifconfig -a
3
4 [CentOS7]
5 $ ip address show
```

路由

```
1 [CentOS6]
2 $ route -n
3 $ route -A inet6 -n
4
5 [CentOS7]
6 $ ip route show
7 $ ip -6 route show
```