

Linux 02

今日学习目标

学习进度

一天总结

磁盘指令

网络命令

系统管理

系统配置

系统时间,日期

更新系统时间

环境变量

重定向与管道

命令执行控制

信息黑洞

遇到的问题以及解决方案

问题1 NAT和桥接不清楚

问题2 Mac下使用rz报错

今日学习目标

- ✓ 磁盘管理
- ✓ 网络管理
- ✓ 系统管理
- ✓ 重定向和信息黑洞

学习进度

Linux(11/20)

一天总结

磁盘指令

1.查看硬盘信息

df 命令可以输出一下信息：默认是以 KB为单位显示，-h 可以以友好的方式展示。（人类更易于阅读）磁盘信息，使用率以及挂载的位置。

```
[root@node01 dev]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2        18G   859M   16G    6% /
tmpfs            491M     0   491M    0% /dev/shm
/dev/sda1        194M   28M   157M   15% /boot
```

2.查看文件或者目录的大小

du 命令 + 文件或者目录,默认单位为 kb , 同上 加上 -h 也会以更易阅读的方式展示。du 命令查看的大小，不足1M的按照1M显示。

还有一个参数 **--max-depth=n**，其中 n 代表一个数字，用来查看目录的层级。

网络命令

查看网络网卡信息：ifconfig

测试连通性：ping

查看网络信息：**netstat -anp**（显示程序名） -t(TCP) -u (UDP) -l (监听)

在后面加上管道符 | 过滤 grep 端口号 可以查看某个端口是否被占用

使用 lsof 命令也可以查看端口是否被占用：

lsof -i:port 没有这个命令的话，可以使用 **yum install lsof** 安装

```
[root@node01 ~]# lsof -i:22
COMMAND  PID  USER  FD  TYPE  DEVICE  SIZE/OFF  NODE  NAME
sshd      1025 root    3u  IPv4  11433      0t0  TCP  *:ssh (LISTEN)
sshd      1025 root    4u  IPv6  11438      0t0  TCP  *:ssh (LISTEN)
sshd      1272 root    3r  IPv4  12761      0t0  TCP  192.168.33.11:ssh->192.168.33.1:56660 (ESTABLISHED)
```

curl -x 网址 可以模拟 HTTP 请求，可以使用 -X 或者 --request 执行特性的 HTTP METHOD
可以试一下 curl www.baidu.com

系统管理

1.用户操作

在 root 下面新建用户。

useradd 名字，就会新增一个用户

passwd 名字，修改该用户的密码

userdel -r 名字，删除该用户(以及目录),注意如果想要删除该用户的home目录。必须要加上参数 r

usermod -l newname oldname 修改名字

usermod -L 用户 锁定用户

usermod -U 用户 解锁用户

注意：这里的虽然锁定用户了，但是还是可以从 root 用户su到被锁的普通用户。

更多与锁定有关的资料可以参考：[几种锁定用户的方式](#)

查看当前用户是谁：whoami 输出 用户名

查看当前登录的是谁以及使用的终端号：who am i 有空格的、会输出

```
[xixi@node01 root]$ who am i
root      pts/1      2018-11-01 21:09 (192.168.33.1)
```

2.用户组操作

查看用户所在的组：groups 用户名 ,下面可以看到 xixi 用户属于 root 用户组。

```
[xixi@node01 root]$ groups xixi
xixi : root
```

也可以使用 id 命令来查看用户的更多信息：

```
[xixi@node01 root]$ id
uid=500(xixi) gid=0(root) groups=0(root) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[xixi@node01 root]$
```

3.权限操作

查看：使用 `ll` 命令可以查看目录或者文件的权限，红色圈出来的区域

```
[xixi@node01 root]$ ll
total 32
-rwx-----. 1 root root 27 Nov 2 2018 11
-rw-r--r--. 1 root root 42 Nov 1 20:06 123
-rw-----. 1 root root 912 Oct 31 22:44 anaconda-ks.cfg
-rw-r--r--. 1 root root 8815 Oct 31 22:44 install.log
-rw-r--r--. 1 root root 3384 Oct 31 22:44 install.log.syslog
-rwx-----. 1 root root 25 Nov 2 2018 test
```

权限类别：有三种，分别是读r,写w,执行x。对应的数字是 4 2 1

UGO模型。user group other 分别是三个不同的权限组、

上面说了这么多权限的概念，下面开始修改权限：（注意是 root 下面执行）

（1）修改所有者与所属组

格式：chown 用户：用户组 文件或目录

如果单独修改一个的话，另一个缺省就可以。（这句话不全对）

比如：

```
chown xix:root 123.txt
```

```
chown xixi: 123.txt(错)，这种都会改。所有者和所属组都会变成 xixi
```

```
chown :root 123.txt (可以的)
```

加上 -R 就是递归的修改！！

（2）修改文件或目录权限

使用 chmod 命令，后面可以使用 ugo+rw 或者 ugo-rw 或者 755 这样的。

上面 + 号代表授权，- 号代表取消授权

系统配置

1.用户组信息配置 `/etc/group`

2.用户信息配置

一个是 只有root可以看到的 `/etc/shadow` ，普通用户是没有权限查看这个文件的。

```
[xixi@node01 ~]$ cat /etc/shadow
cat: /etc/shadow: Permission denied
```

另一个是 普通用户可以看的 `/etc/passwd`

3.系统服务初始化配置

查看 `/etc/inittab` 文件。

```
1. root@node01:~ (ssh)
# ADDING OTHER CONFIGURATION HERE WILL HAVE NO EFFECT ON YOUR SYSTEM.
#
# System initialization is started by /etc/init/rcS.conf
#
# Individual runlevels are started by /etc/init/rc.conf
#
# Ctrl-Alt-Delete is handled by /etc/init/control-alt-delete.conf
#
# Terminal gettys are handled by /etc/init/tty.conf and /etc/init/serial.conf,
# with configuration in /etc/sysconfig/init.
#
# For information on how to write upstart event handlers, or how
# upstart works, see init(5), init(8), and initctl(8).
#
# Default runlevel. The runlevels used are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:
"/etc/inittab" 26L, 884C                                     3,1 Bot
```

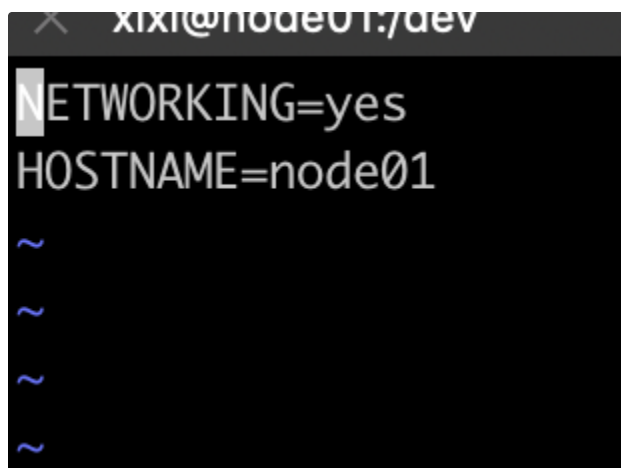
Linux 默认的级别是 3，不能把 `initdefault` 设置为0 或者 6 。

如图所示，`init` 后面有7种数字，常用的有 0，3，5，6

- `init 0` 🚫 慎用关机！！！！
- `init 3` 默认多用户模式
- `init 5` 图形化界面
- `init 6` 重启服务器

4.主机名配置

在 `/etc/sysconfig/network` 下面配置 `hostname = xxx`。

A terminal window with a dark background. The title bar shows 'xixi@node01:/dev'. The terminal content shows 'NETWORKING=yes' and 'HOSTNAME=node01' on two lines. Below these, there are four blue tilde '~' characters, each on a new line.

```
xixi@node01:/dev
NETWORKING=yes
HOSTNAME=node01
~
~
~
~
```

5.配置 DNS

在 `/etc/hosts`里面做映射

或者在 `/etc/resolv.conf` 设置 `nameserver`

`"/etc/resolv.conf" 2L, 25C`

```

xixi@node01:~ (ssh)
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)        ALL
xixi    ALL=(root)      /sbin/service

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, DRIVERS

```

比如上面新增的：**xixi ALL=(root) /sbin/service** 就是给xixi 用户添加sudo权限可以执行 **service** 命令

如果不想每次使用sudo的时候输入密码，可以加入 **NOPASSWD:** 来配置不需要密码！！

```

## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)        ALL
xixi    ALL=(root) NOPASSWD: /sbin/service

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, D

```

使用 **sudo -l** 可以查看该用户所有的 **sudo** 权限。

系统时间,日期

```

date
cal
cal 2018
cal 10 2018

```

更新系统时间

使用服务 **ntp** 来同步时间。

yum install -y ntp 来安装。

nptdate cn.ntp.org.cn 来更新同步时间。

如果网络有问题或者太慢的，可以来这个网站找到Ip池里面提供的ip. ntp.org.cn ❤️

ntp.org.cn/pool.php

写作 服务器商 搜索 工具 好的博主 问题解答 技术官网 面试 阅读 书单 金融 扩展程序 JVM虚拟机 UI 资源查找 名企招聘

NTP.ORG.CN
NTP授时快速域名服务

首页 池 关于 更新 捐助 EN

池
提供更快更稳定的NTP服务

中国 · 中国教育网 · 中国香港 · 中国台湾 · 美国 · 新加坡 · 韩国 · 日本 · 德国 · 印度尼西亚 · 其他

区域	域名	IP池
中国	cn.ntp.org.cn	118.24.4.66 自有节点
		202.108.6.95
		120.25.108.11 由*阿里云*提供
		182.92.12.11 由*阿里云*提供
		203.107.6.88 由*阿里云*提供
		120.25.115.20 由*阿里云*提供

环境变量

全局环境变量在 /etc/profile 里面配置，慎用。
局部变量在 ~/.bash_profile 里面配置。主要是设置一些命令路径等

```
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=/root/:$PATH:$HOME/bin

export PATH

~
~
"~/.bash_profile" 12L, 183C
```

重定向与管道

1.输出重定向:

> 覆盖

>> 追加

2.输入重定向

< 重定向到一个程序: `cat < 1.txt`

3.

标准输出重定向: `1>` 等价于 `>` , 只会输出正确的信息。

错误输出重定向: `2>` , 只会输出错误的信息。

```
[root@node01 ~]# cat sdsdashdk > 123
cat: sdsdashdk: No such file or directory
[root@node01 ~]# vim 123
[root@node01 ~]# cat sdsdashdk 2> 123
[root@node01 ~]# vim 123
```

如上，cat 一个不存在的文件，则会报错，如果使用 1> 则不会把错误信息重定向到 123 文件中。
使用 2> 则会把错误信息重定向到 123 文件中。

4.结合使用 2>&1 专业说法：将标准错误输出重定向到标准输出。这种比较常用，因为我们在开发的过程中肯定需要完善的日志信息，错误的、正确的信息都将会保存下来。

5.管道 | 一般结合 grep 使用。

命令执行控制

1.&& 只有前面的命令执行正确，才会执行下一个命令。

2.|| 只有前面的命令执行错误，才会执行下一条命令

信息黑洞

/dev/null

使用 ll /dev/null 可以看出 null 是一个设备块文件。

```
[root@node01 dev]# ll | grep null
crw-rw-rw-. 1 root root      1,   3 Nov  1 16:44 null
```

使用 du /dev/null 可以看到 null 的大小为 0

使用 echo "some thind" >> /dev/null 之后，再次查看 null 文件的大小，依然为 0

```
[root@node01 dev]# du null
0      null
```

遇到的问题以及解决方案

问题1 NAT和桥接不清楚

【问题描述】

对于NAT和桥接模式不是很清楚，懵懵懂懂的那种

【问题思路】

查资料，老师讲解。以教室的网络为例。

【解决方案】

NAT是**物理主机模拟一个子网出来**以供虚拟机里面的服务器使用，使用vmnet8网卡。

桥接是物理主机和虚拟机服务器公用同一个子网，使用vmnet0网卡。

NAT模式下可以访问外部网络，但是外部网路不能访问我的虚拟机。

桥接模式下虚拟机可以访问外部网络，同时外部网络也可以访问虚拟机。

综上，NAT 节省IP且安全。桥接模式浪费IP

问题2 Mac下使用rz报错

问题描述】

使用expect脚本自动登录的ssh远程的服务器，再次使用rz的时候iterm2卡死。

【问题思路】

进程交互的问题

【解决方案】

expect和rz只能同时选择一个，还是使用 scp或者**rsync吧**。

参考：<https://blog.csdn.net/pkueecser/article/details/46491133>