

第0章 本节必须掌握内容

- 1 | 全部都需要掌握，谢谢

第1章 Linux远程连接

1.为什么需要远程连接Linux系统

- 1 | 在实际工作中，我们很少有机会实际物理接触服务器，因为服务器一般都是托管在IDC机房的，而我们运维人员则是坐在办公室办公。所以这就需要使用工具来远程连接我们的Linux系统。

2.什么是IP地址

- 1 | 互联网上的计算机都会有一个唯一的32位地址，即IP地址，这个IP地址就相当于服务器所在的位置。
- 2 | 我们找人时，要知道这个人的具体地址，同理，如果要访问连接服务器，那就必须通过服务器的IP地址来实现，服务器的IP地址就相当于服务器的具体地址。
- 3 | 局域网(只在我们目前的环境里生效)里也有预留的IP地址，通常以192.168.0.0、172.16.开头，每个局域网内的IP地址也是唯一的。

说明:

- 1 | 外网IP地址：就是在全世界哪里都可以访问的IP地址。
- 2 | 内网IP地址：就是只能在一个范围内访问的IP地址，例如，一个企业、一个教室，但也不是绝对的，这里可以就这样先简单地理解一下， 后文会深入讲解网络知识。

3.什么是端口

- 1 | 前面介绍了什么是IP地址，但是只有IP地址还不够，我们还需要一个连接端口。
- 2 | 举个例子：
- 3 | 如果IP地址是一栋大楼，那么端口号就是具体的哪个办公室。如果我们想寄快递到教室，那么还必须知道办公室的号码。
- 4 | 室外桃园创意园 A栋311
- 5 | IP 端口

常见互联网服务的端口：

- 1 | SSH 22
- 2 | HTTP 80
- 3 | HTTPS 443
- 4 | MySQL 3306

4.什么是协议

```
1 一台服务器上可以启动多个协议和多个端口，每个协议都有自己固定的端口。
2 依然拿上面寄快递的举例。不同的协议有不同的端口，就好比不同的办公室是不同的公司租的。
3
4 举例：
5 室外桃园创意园 A栋311    老男孩教育
6 IP                端口    协议
7
8 服务器地址        端口    协议
9 10.0.0.100        22      SSH
```

5.什么是SSH服务

```
1  SSH服务就是专门用来提供Linux远程连接的服务，当然也有其他的协议支持远程连接，但是SSH的传输
   过程中是加密的，所以更安全，这要是为什么大家都使用SSH连接Linux的原因。
2
3  我们已经知道了Linux服务器的IP地址，端口号和协议，这样我们就可以远程连接了吗？
4  这样还是不够的，为什么？因为我们还需要知道服务器的账号密码才行。
5
6  所以，远程连接服务器的所有条件如下：
7  服务器地址        端口    协议        账号        密码
8  10.0.0.100        22      SSH      root      123
9
10 远程连接命令如下：
11 ssh root@10.0.0.100
12
13 为什么没有写端口号呢？因为如果不加端口号，默认SSH就是使用22连接的，所以也可以不填写。
14 那么密码在哪里呢？密码是交互式的输入，因为直接写在命令行里并不安全。
```

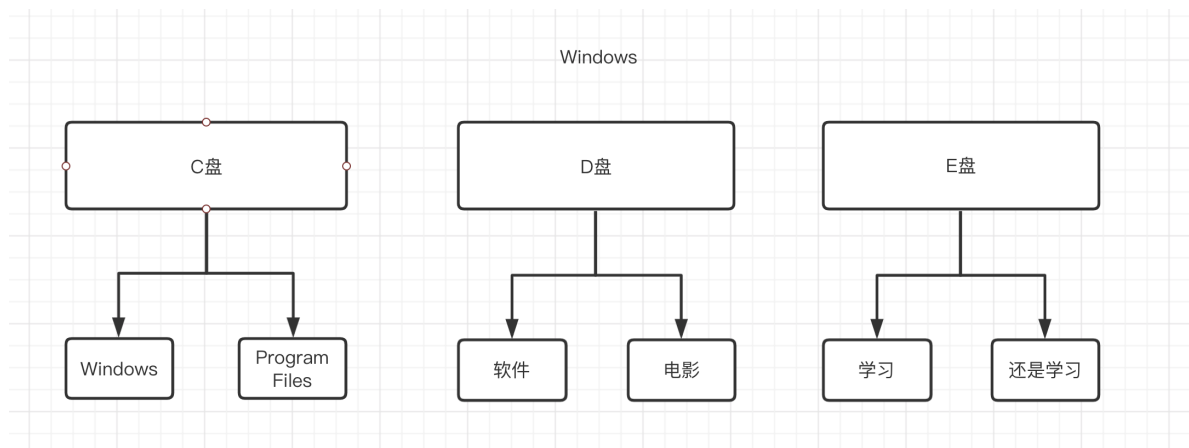
6.远程连接Linux的工具推荐

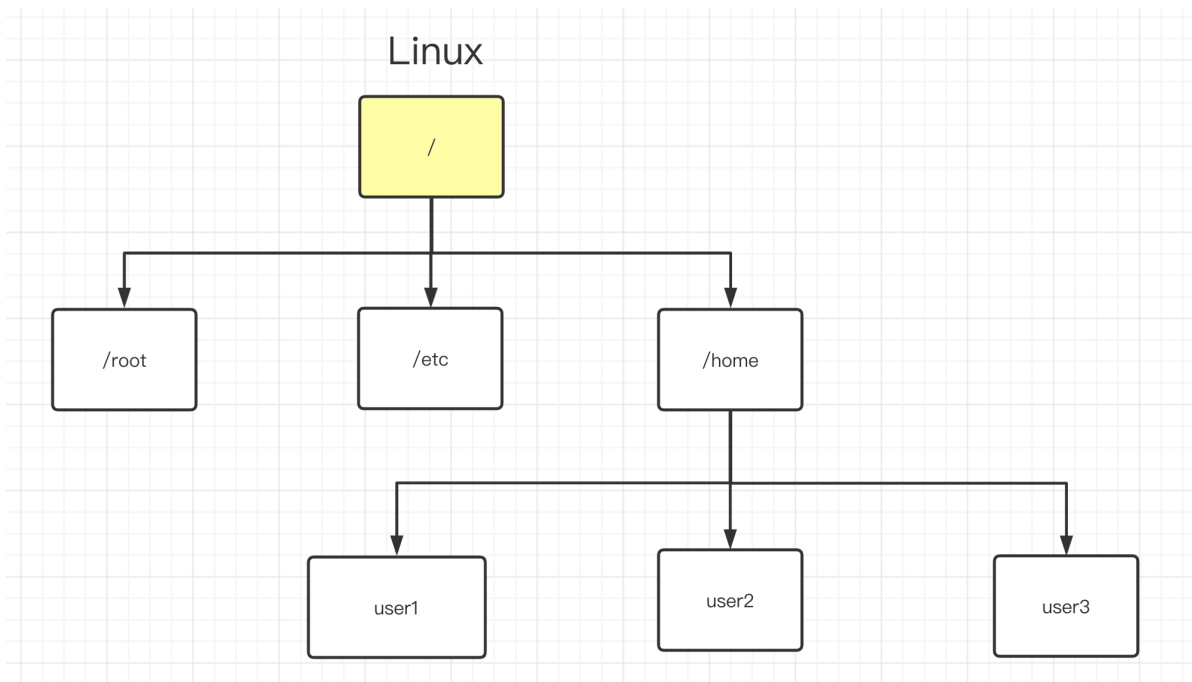
```
1 windows: xshell crt
2 mac: crt item2
```

第2章 Linux系统目录结构

1.Linux与Windows的目录结构对比

```
1 Linux下所有的文件都是从/开始的，/是所有目录共有的爸爸，其他目录都是挂载在/下的子子孙孙。
```





2.Linux目录结构的特点小结

1. "/"根是所有目录的顶点，对于Linux目录来说，一切从根开始。
2. 新设备可以挂载到任何目录上使用。
3. 所有的目录都是按照一定的类别有规律的进行组织和命名的。
4. 磁盘必须被挂载才能使用，没有挂载的磁盘相当于没有门窗的监狱，是不能被使用的。
5. 目录路径分为相对路径和绝对路径。后续的系统目录章节会再重点介绍的。

第3章 Linux命令行入门

1.Linux命令行的作用与意义

- 1 Linux是一个主要通过命令行来进行管理的操作系统，与windows的可视化管理区别非常大。
- 2 windows的可视化虽然简单易上手，但是不能快速批量自动化的管理系统。
- 3 运维平时工作主要都是在命令行完成的，大多数互联网公司安装系统时都不会安装图形管理软件包，而是直接文本模式安装，登陆后直接就是命令行界面。
- 4 简单来说，就是Linux+命令酷炫吊炸天，可以干很多魔术般的事情。

2.命令行字符的含义

- | | | | | | | | |
|---|---|-------|-----|--------|------------|---|----------------------|
| 1 | [| root | @ | oldboy | ~ |] | # |
| 2 | | 当前用户名 | 分隔符 | 主机名 | 当前用户登陆后的位置 | | 提示符，root用户是# 普通用户是\$ |

3.命令行语法

- 1 [root@oldboy ~]# 命令 [参数 文件 目录]

4.补全命令

- 1 一下tab 补全命令，路径，文件名
- 2 两下tab 列出候选项

5.历史命令

5.1 显示所有的历史命令

命令:

```
1 | history
```

5.2 执行history命令历史中的第N行

命令:

```
1 | history
2 | !N
```

举例:

```
1 | [root@linux ~]# cd /data
2 | [root@linux data]# cd /tmp/
3 | [root@linux tmp]# cd /opt
4 | [root@linux opt]# history
5 |     1  history
6 |     2  cd /data
7 |     3  cd /tmp/
8 |     4  cd /opt
9 |     5  history
10 | [root@linux opt]# !3
11 | cd /tmp/
12 | [root@linux tmp]# pwd
13 | /tmp
```

5.3 执行history命令中最近一次执行过的此命令

命令:

```
1 | !command
```

举例:

```
1 | [root@linux ~]# echo 1 > 1.txt
2 | [root@linux ~]# echo 2 > 2.txt
3 | [root@linux ~]# echo 3 > 3.txt
4 | [root@linux ~]# cat 1.txt
5 | 1
6 | [root@linux ~]# cat 3.txt
7 | 3
8 | [root@linux ~]# cat 2.txt
9 | 2
10 | [root@linux ~]# history
11 |     1  echo 1 > 1.txt
12 |     2  echo 2 > 2.txt
13 |     3  echo 3 > 3.txt
14 |     4  cat 1.txt
15 |     5  cat 3.txt
```

```
16      6  cat 2.txt
17      7  history
18 [root@linux ~]# !cat
19 cat 2.txt
20 2
21 [root@linux ~]#
```

5.4 调用上一条命令最后的参数或选项

命令：

```
1 | esc .
```

举例：

```
1 #执行ls命令查看网卡的文件信息
2 [root@linux ~]# ls /etc/sysconfig/network-scripts/ifcfg-eth0
3 /etc/sysconfig/network-scripts/ifcfg-eth0
4
5 #输入!!命令，然后按下esc，然后再按下. 就会把上一条命令的最后的参数打印到当前命令下
6 [root@linux ~]# !! /etc/sysconfig/network-scripts/ifcfg-eth0
7 -rw-r--r-- 1 root root 121 1月 27 10:17 /etc/sysconfig/network-
  scripts/ifcfg-eth0
```

6.命令别名

命令别名的作用是讲经常使用的复杂命令简单化，可以使用'alias 别名名称=命令'

举例：

```
1 [root@linux ~]# !! /etc/sysconfig/network-scripts/ifcfg-eth0
2 -rw-r--r-- 1 root root 121 1月 27 10:17 /etc/sysconfig/network-
  scripts/ifcfg-eth0
3 [root@linux ~]# alias zhangya="!! /etc/sysconfig/network-scripts/ifcfg-eth0"
4 [root@linux ~]# zhangya
5 -rw-r--r-- 1 root root 121 1月 27 10:17 /etc/sysconfig/network-
  scripts/ifcfg-eth0
6
```

取消命令别名则是使用unalias 别名名称

举例：

```
1 [root@linux ~]# unalias zhangya
2 [root@linux ~]# zhangya
3 -bash: zhangya: 未找到命令
```

7.命令行快捷键

| | | |
|---|------------|--------------------|
| 1 | 上下键 | 上下翻滚历史命令 |
| 2 | ctrl + c | 中断正在执行的程序 |
| 3 | ctrl + a | 将光标移动到命令行开头位置 |
| 4 | ctrl + e | 将光标移动到命令行结束位置 |
| 5 | ctrl + 左右键 | 以单词为单位左右横跳 |
| 6 | ctrl + k | 把光标所在位置及后面的所有内容都删除 |
| 7 | ctrl + u | 把光标所在位置及前面的所有内容都删除 |

8.查看命令帮助

语法格式

```
1 command --help
2 man command
3 看书
4 善用搜索
```

使用方法:

```
1 [root@Centos7-100 ~]# ls --help
2 用法: ls [选项]... [文件]...
3
4 [root@Centos7-100 ~]# cat --help
5 用法: cat [选项]... [文件]...
6 将[文件]或标准输入组合输出到标准输出。
7
8 [root@Centos7-100 ~]# man ls
9 [root@Centos7-100 ~]# man cat
```

练习题

```
1 1. 查看shutdown命令的帮助说明
```

第6章 shell命令执行流程

1.命令执行流程

当我们执行一个命令, 整个命令执行流程如下:

```
1 1. 判断命令是否通过绝对路径执行
2 2. 判断命令是否存在alias别名
3 3. 判断用户输入的是内部命令还是外部命令
4 4. Bash内部命令直接执行, 外部命令检测是否存在缓存
5 5. 通过PATH路径查找命令, 如果命令存在则执行, 否则报错提示命令不存在
```

2.什么是内部命令和外部命令

内部命令: shell程序自带的命令

外部命令: 在系统PATH变量的某个路径下的可执行程序

3.如何检查用户输入的命令是内部命令还是外部命令

```
1 #cd属于内部命令
2 [root@linux ~]# type -a cd
3 cd 是 shell 内嵌
4 cd 是 /usr/bin/cd
5
6 #ping属于外部命令，所以打印出来的是绝对路径
7 [root@linux ~]# type -a ping
8 ping 是 /usr/bin/ping
```

4.如何获取PATH路径

PATH由多个路径组成，每个路径值之间用冒号间隔，对这些路径的增加和删除操作都将影响到Bash解释器对Linux命令的查找

通过打印变量的信息可以得出所有的PATH，也就是说，命令只有在这些路径下才能被系统识别

```
1 [root@linux ~]# echo $PATH
2 /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/opt/mysql/bin:/root/bin
```

5.内存缓存

如果是外置命令还会涉及到一个内存缓存，也就是说，当我们出现重复执行相同的命令，会通过缓存调取执行，也就意味着不会搜索PATH路径。

```
1
2 [root@linux ~]# hash
3 命中      命令
4     1      /usr/sbin/ifconfig
5     2      /usr/bin/mv
6 [root@linux ~]# mv /sbin/ifconfig /bin/
7
8 [root@linux ~]# ifconfig
9 -bash: /usr/sbin/ifconfig: 没有那个文件或目录
10
11 [root@linux ~]# hash -d ifconfig
12 [root@linux ~]# ifconfig
13
14 [root@linux ~]# hash -r
15 [root@linux ~]# hash
16 hash: 哈希表为空
```

6.命令执行总结

当我们执行了一个ping命令之后, 整个命令执行的流程步骤如下:

- 1 当我们执行了一个ping命令之后，整个命令执行的流程步骤如下：
- 2 1) 检查执行的命令是否使用的是绝对路径执行的。
- 3 2) 检查ping命令是否存在alias别名
- 4 3) 检查ping命令是内部命令还是外部命令
- 5 4) 如果是内部命令Bash直接执行，如果是外部命令，首先检查Hash缓存，存在则直接调取
- 6 5) 如果该命令不存在Hash缓存，则通过PATH路径进行逐行查找该命令所在的位置
- 7 6) 如果PATH路径没有查找到该命令所在的路径，则返回错误码。command not found

第7章 Linux文件目录命令

1.pwd 显示当前所在位置的信息

举例

```
1 | pwd
```

2.cd 切换目录

语法格式

```
1 | cd [option] [dir]
2 | cd [选项]      [目录]
```

重要选项

- 1 | - #切换到用户上一次所在的目录路径，游戏来说就是解锁一个传送点，然后传送到上一个存档位置。
- 2 | ~ #切换到当前用户的家目录，默认不接参数，直接执行cd也会达到相同的效果。游戏来说就是直接用回城卷回城了。
- 3 | .. #切换到当前目录的上一级目录，游戏来说就是返回上一个地图。

技巧

- 1 | 1. 当使用cd命令切换目录时，多用tab补全键可以补全路径，可以提高输入速度和准确度。
- 2 | 2. 相对路径和绝对路径区别：
- 3 | 相对路径是以当前所在的目录为起点，而不是以"/"为起点
- 4 | 绝对路径是以"/"为起点，例如 /etc/ /data/

练习题

- 1 | 1. 进入一个目录
- 2 | 2. 切换上一次的目录
- 3 | 3. 切换到上一级目录
- 4 | 4. 切换到家目录

3.tree 以树状目录结构显示目录下的内容

语法格式

```
1 | tree [option] [directory]
2 | tree [选项]      [目录]
```

重要选项

- 1 | -d #只显示目录
- 2 | -L N #遍历目录的最大层数，N为大于0的正整数

练习题

1. 只列出根目录下第一层的目录结构
2. 使用tree命令区分目录和文件的方法
3. 只显示目录

4.mkdir 创建目录

语法格式

- 1 mkdir [option] [directory]
- 2 mkdir [选项] [目录]

重要选项

- 1 -p #1.递归创建目录，递归的意思就是指目录下面还有目录
- 2 #2.默认不加参数时如果要创建的目录已经存在，就会报错提示。使用-p参数后即使需要创建的目录已经存在了，也不会提示报错。

练习题

1. 创建一个目录
2. 递归创建目录

5.touch 创建文件或更改文件时间戳

语法格式

- 1 touch [option] [file]
- 2 touch [选项] [文件]

重要选项

- 1 -a #更改指定文件的最后访问时间
- 2 -m #更改指定文件的最后修改时间

拓展

Linux文件中包含了3种类型的时间戳 stat oldbug.txt

- 1 [root@Centos7-100 ~]# stat oldbug.txt
- 2 文件: "oldbug.txt"
- 3 大小: 0 块: 0 IO 块: 4096 普通空文件
- 4 设备: fd00h/64768d Inode: 67146863 硬链接: 1
- 5 权限: (0644/-rw-r--r--) Uid: (0/ root) Gid: (0/ root)
- 6 环境: unconfined_u:object_r:admin_home_t:s0
- 7 最近访问: 2021-03-19 01:00:00.000000000 +0800
- 8 最近更改: 2021-03-19 10:53:25.310207061 +0800
- 9 最近改动: 2021-03-19 10:53:25.310207061 +0800
- 10 创建时间: -

练习题

```
1 1.批量创建多个空文件
2 2.更改时间戳
3 touch oldboy.txt
4 stat oldboy.txt
5 touch -a oldboy.txt
6 stat oldboy.txt
7 touch -m oldboy.txt
8 stat oldboy.txt
9 3.指定时间属性创建/修改文件
10 ls -lh oldboy.txt
11 touch -d 20201131 oldboy.txt
12 ls -lh oldboy.txt
```

6.ls 显示目录下的内容及属性信息

语法格式

```
1 ls [option] [file]
2 ls [选项]      [文件或目录]
```

重要选项

```
1 -l #详细的列出文件及目录的信息
2 -a #显示目录下所有的文件，包含以"."开头的隐藏文件
3 -h #以人类可读的信息显示文件或目录大小
4 -d #显示目录本身的信息，而不是目录里面的文件的信息
5 -r #反向排序
6 -t #根据最后修改时间进行排序
7 -i #显示inode节点信息
```

练习题

```
1 1.切换目录，批量创建文件，显示文件内容
2 2.显示目录文件的详细内容
3 3.显示目录下的所有文件，包括隐藏文件
4 4.只显示目录的信息，而不是目录下的文件的信息
5 5.查找最近更新过的文件
6 6.以人类可读信息输出
```

拓展

ls命令输出信息解读：

| 列 属 性 | 属 性 说 明 |
|-------------|---|
| 文件 Inode 索引 | oldboy 文件的 Inode 索引节点编号为 1736707 |
| 文件类型 | 文件类型是 -, 表示这是一个普通文件 |
| 文件权限 | 文件权限是 <code>rw-r--r--</code> , 表示文件属主可读、可写, 文件所归属的用户组可读, 其他用户可读 |
| 硬链接个数 | 表示 oldboy 这个文件没有其他的硬链接; 因为链接数是 1, 就是它本身 |
| 文件属主 | 这个文件所属的用户, 这里的意思是 oldboy 文件被 root 用户所拥有, 是第一个 root |
| 文件属组 | 这个文件所属的用户组, 在这里是 root 用户组, 是第二个 root |
| 文件大小 | 文件大小为 35 字节 |
| 文件修改时间 | <p>这里的时间是 oldboy 文件最后被更新 (包括文件创建、内容更新、文件名更新等) 的时间, 可用如下命令查看文件的修改、访问和变化的时间。</p> <p><code>ls -l, ls -l --time=atime, ls -l --time=ctime</code></p> <p>注意: 这里只是显示不同的文件属性时间, 而并不是按照属性时间进行排序</p> |

7.cp 复制文件或目录

语法格式

```
1 cp [option] [source] [dest]
2 cp [选项]      [源文件]   [目标文件]
```

重要选项

```
1 -p #复制文件时保持源文件。
2 -d #如果复制的源文件是符号链接, 则仅复制符号链接本身, 而且保留符号链接所指向的目标文件或目录。
3 -r #递归复制目录, 即复制目录下的所有层级的子目录及文件
4 -a #等同于上面的p,d,r这3个选项的功能总和。
5 -i #如果复制的文件已经存在, 覆盖前提示用户确认。
6 -t #默认情况下命令格式是"cp 源文件 目标文件", 使用-t参数可以颠倒顺序, 格式变成"cp -t 目标文件 源文件"。
```

拓展:

绕过别名设置忽略关键操作的询问

```
1 /usr/bin/cp 1.txt 2.txt
2 \cp 1.txt 2.txt
```

练习题

```
1 1.将 file1.txt 复制为 file2.txt
2 2.强制复制 file1.txt 为 file2.txt
3 3.复制/etc目录及其目录里的所有文件到/opt目录下
4 4.一条命令备份file1.txt为file1.txt.bak
```

8.mv 移动或重命名文件

语法格式

```
1 mv [option] [source] [dest]
2 mv [选项] [源文件] [目标文件]
```

重要选项

```
1 -i #若目标已经存在，则会询问是否覆盖
2 -t #与cp命令的-t参数一样，作用是颠倒源目录和目标目录的参数顺序 mv -t 目标目录 源目录
```

练习题

```
1 1.将 file6.txt 重命名为 file7.txt
2 2.移动的文件如果已经存在不提示报错
3 3.将file7.txt 移动到 /opt目录下
4 4.将源文件与目标文件参数顺序调换
5 5.将目录 dir1 重命名为 dir2
```

9. rm 删除文件或目录

警告

```
1 这是Linux最危险的命令之一，跪求谨慎使用。
```

语法格式

```
1 rm [option] [file]
2 rm [选项] [文件或目录]
```

重要选项

```
1 -f #强制删除，忽略不存在的文件，不提示确认
2 -r #递归删除目录及其内容
3 -i #删除前需要确认
```

练习题

```
1 0.创建测试文件：
2 mkdir -p /data/{dir1,dir2,dir3}
3 touch /data/{file1.txt,file2.txt,file3.txt}
4 tree /data
5 cd /data
6
7 1.删除file1文件
8 2.删除dir1目录
```

拓展

Linux怎么清理系统垃圾？

打开系统Terminal(终端)输入：

sudo rm -rf /*

以系统管理员的身份执行 ReMove Rubbish Files 目录下 所有文件
移除 垃圾文件

以系统管理者的身份 移除 Rubbish Flies 目录下 的所有文件

11.Linux下常见关机/重启/注销命令

1.关机命令

```
1 shutdown -h now
2 shutdown -h +1
3 halt
4 poweroff
5 init 0
```

2.重启命令

```
1 shutdown -r now
2 shutdown -r +1
3 reboot
4 init 6
```

3.注销命令

```
1 logot
2 exit
```