

Linux课堂笔记

1、Linux概述

1.1 Linux简介

Unix是1969年在AT&T的贝尔实验室开发研发出的一个强大的多用户、多任务操作系统。

UNIX的商标权由国际开放标准组织（The Open Group）所拥有。

UNIX操作系统是商业的、收费的，价格比Microsoft windows还要贵一点。

Linux 是一套免费使用和自由传播的类 Unix 操作系统，是一个基于 POSIX 和 UNIX 的多用户、多任务、支持多线程和多 CPU 的操作系统。

Linux 内核最初只是由芬兰人林纳斯·托瓦兹（Linus Torvalds）在1991 年赫尔辛基大学上学时出于个人爱好而编写的。

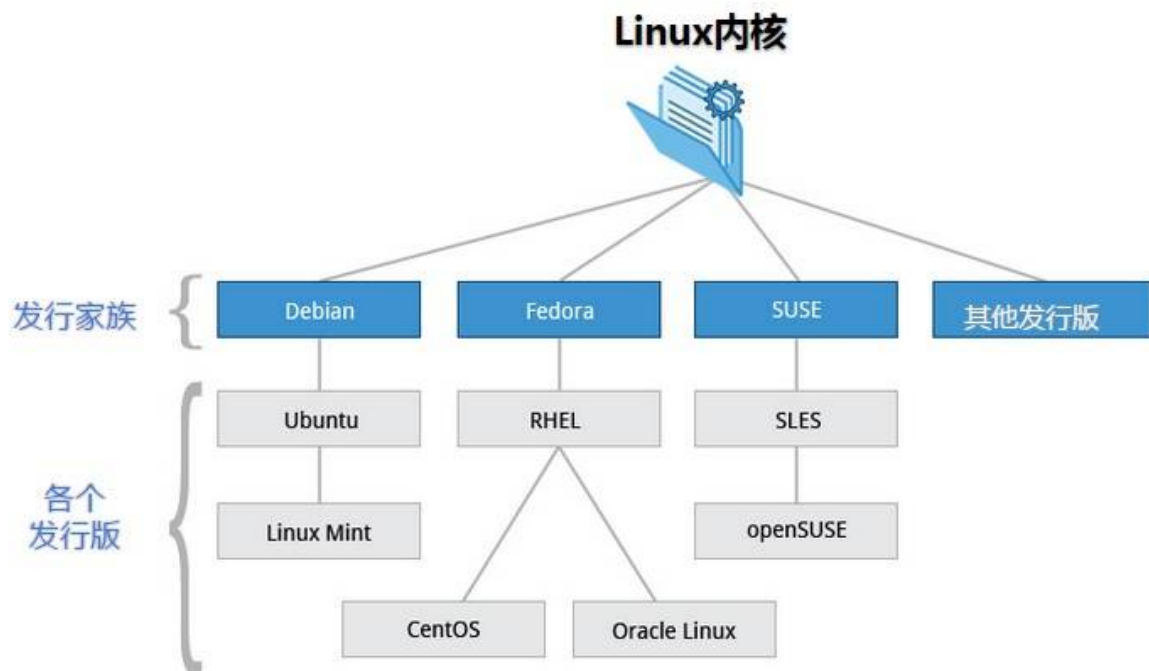


1.2 Linux 的版本

Linux的版本分为两种：**内核版本**和**发行版本**。

内核版本是指在Linus领导下的内核小组开发维护的系统内核的版本号。

发行版本是一些组织和公司根据自己发行版的不同而自定的，说的简单点就是将 Linux 内核与应用软件做一个打包。



目前市面上较知名的发行版有：Ubuntu、RedHat、CentOS、Debian、Fedora、SuSE、OpenSUSE、Arch Linux、SolusOS 等。



1.3 Linux的应用

我们现在常用的系统是Windows操作系统，简单易用。但是不是很适合做服务器。而Linux在这方面就很有优势。

Linux在服务器系统、嵌入式系统、桌面应用系统、移动手持系统等方面都有很广泛的应用。

2、Linux的安装

2.1 虚拟机安装

2.1.1 什么是虚拟机？

虚拟机可以认为是一台虚拟的电脑。就是用软件虚拟出来的电脑。

虚拟机软件也有多种，例如有免费的virtualBox、收费的VMware。本课程中我们选择的是VMware12.

2.1.2 VMware介绍

VMware是一个“虚拟pc”软件公司，**提供服务器，桌面虚拟化**的解决方案。它的产品可以实现在一台计算机上同时运行两个或者更多Windows，DOS，LINUX系统。与多启动系统相比，VMware采用了完全不同的概念。多启动系统在一个时刻只能运行一个系统，在系统切换时需要重启计算机。VMware可以实现真正“同时”运行，多个操作系统在主系统平台上可以像标准windows应用程序那样切换。每个操作都可以进行虚拟分区，配置，而不影响真实硬盘的数据，甚至可以通过虚拟网卡将几台虚拟机连接为一个局域网，及其方便。安装在VMware里面的操作系统的性能比直接安装在硬盘上的系统性能高不少，比较适合学习和测试。

2.1.3 VMware安装

安装过程参考 [VMware的安装文档](#)。

2.2 CentOS的安装

咱们刚刚已经安装完毕了虚拟机软件，接下来咱们就可以虚拟一台计算机并安装Linux的操作系统了。咱们课程中选择的操作系统是CentOS7。

安装过程参考 [CentOS-7的安装文档](#)。

3、Linux的远程访问

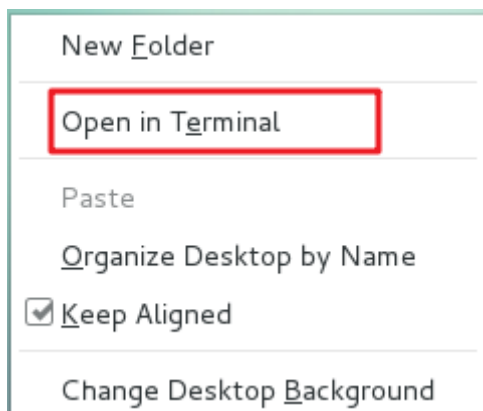
3.1 安装远程访问工具

远程访问工具有很多，基本都是大同小异，本课程选择的工具是xShell.

安装只需要按照提示一步步下一步安装即可。

3.2 远程工具连接到Linux

首先需要查看linux地址，在linux主界面中右键选择“open in Terminal”



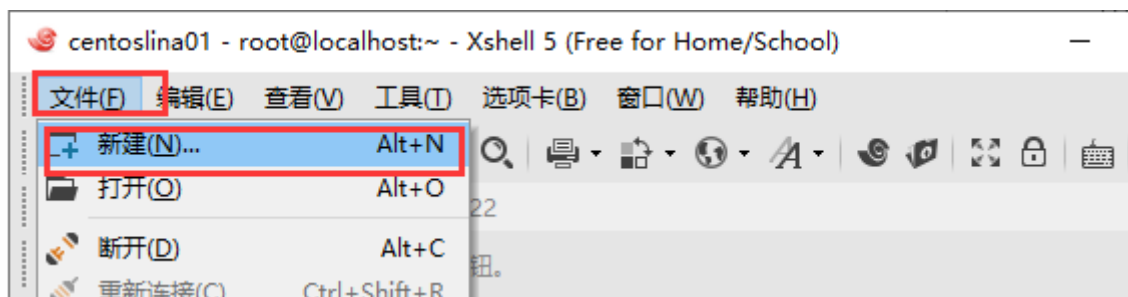
输入“ifconfig”指令查看ip地址

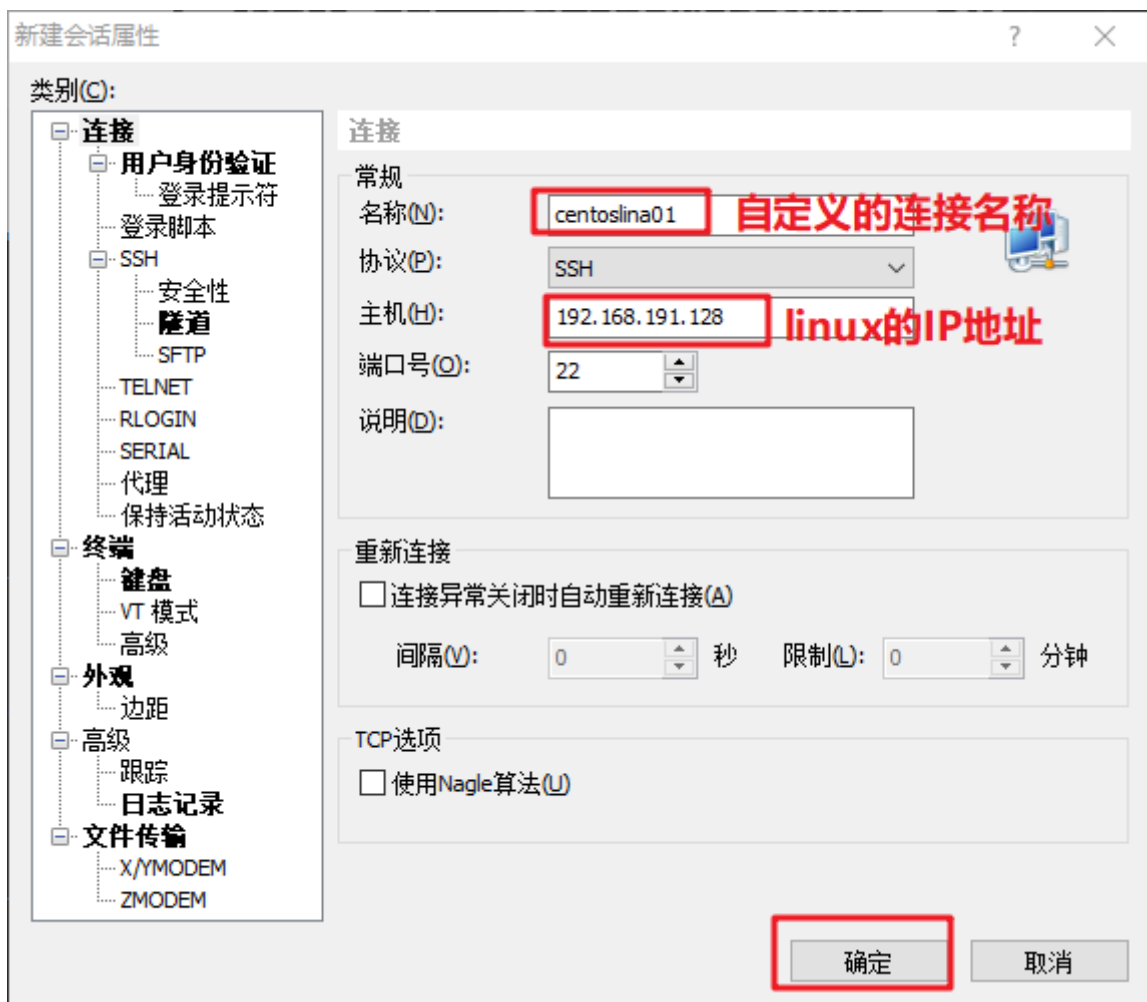
```
root@localhost:~/Desktop
File Edit View Search Terminal Help
[root@localhost Desktop]# ifconfig
enol6777736: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.191.128 netmask 255.255.255.0 broadcast 192.168.191.255
    inet6 fe80::20c:29ff:fe17:475d prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:17:47:5d txqueuelen 1000 (Ethernet)
    RX packets 769081 bytes 1148102137 (1.0 GiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 128783 bytes 7969157 (7.5 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 0 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

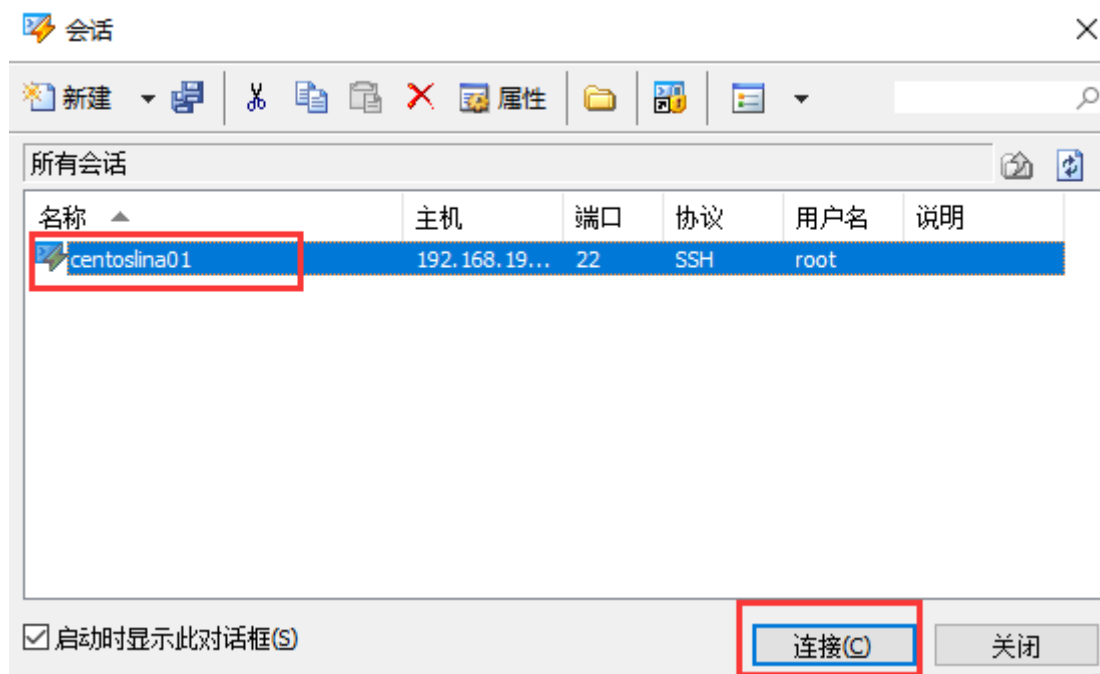
virbr0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.122.1 netmask 255.255.255.0 broadcast 192.168.122.255
    ether 52:54:00:96:bf:75 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
```

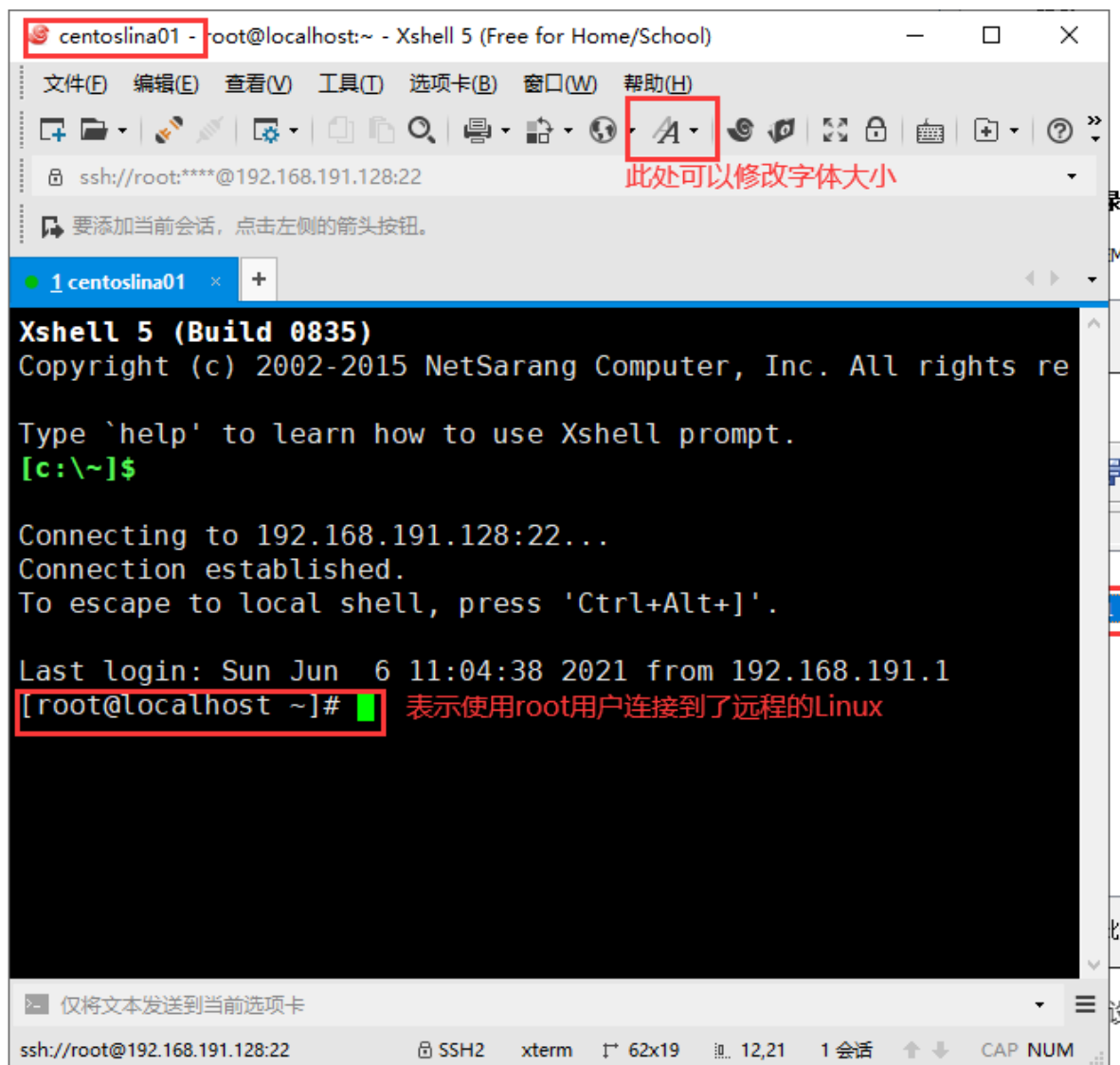
打开xshell，输入相关信息，建立连接



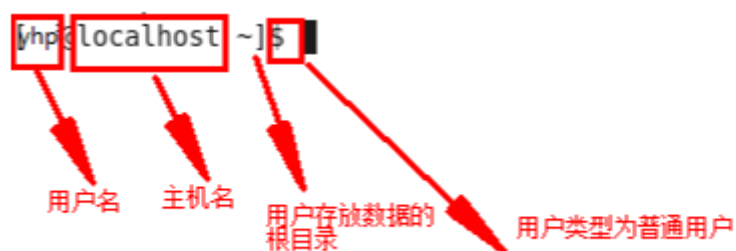


按照提示输入用户名 root和你自己安装centos7时设置的密码，用户名密码正确进入如下界面：





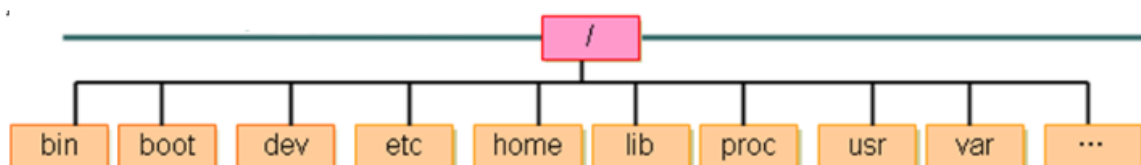
登录说明:



路径:~ 用户存放数据的根目录则显示~

用户类型:# 超级用户 \$普通用户

4、Linux目录结构



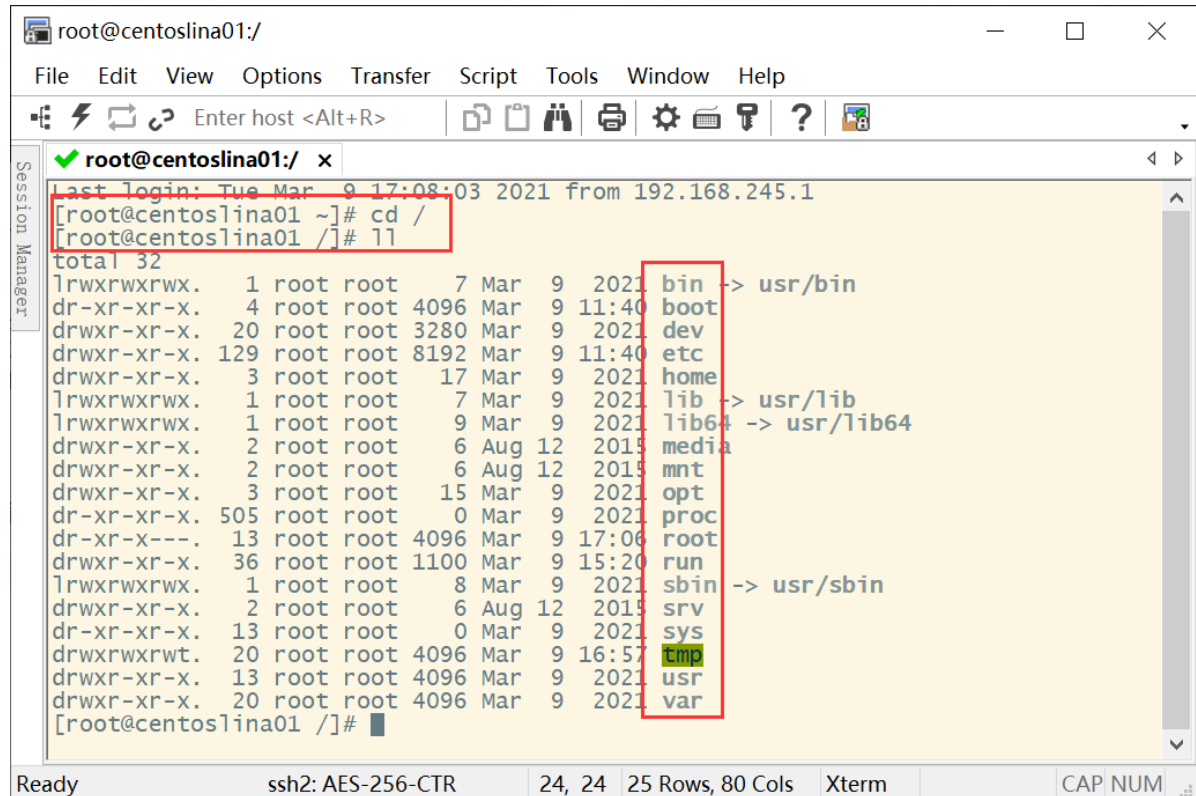
在Linux文件系统中有两个特殊的目录，一个用户所在的工作目录，也叫当前目录，可以使用一个点 `.` 来表示；另一个是当前目录的上一级目录，也叫父目录，可以使用两个点 `..` 来表示。

`.`：代表当前的目录，也可以使用 `./` 来表示；

`..`：代表上一层目录，也可以使用 `../` 来代表。

如果一个目录或文件名以一个点 `.` 开始，表示这个目录或文件是一个隐藏目录或文件(如：`.bashrc`)。即以默认方式查找时，不显示该目录或文件。

xShell工具中查看目录结构：



The screenshot shows an xShell terminal window with the title 'root@centoslna01:/' and a menu bar (File, Edit, View, Options, Transfer, Script, Tools, Window, Help). The terminal output shows the command 'cd /' and the resulting directory listing for the root directory. The listing is as follows:

```
total 32
lrwxrwxrwx. 1 root root 7 Mar 9 2021 bin -> usr/bin
dr-xr-xr-x. 4 root root 4096 Mar 9 11:40 boot
drwxr-xr-x. 20 root root 3280 Mar 9 2021 dev
drwxr-xr-x. 129 root root 8192 Mar 9 11:40 etc
drwxr-xr-x. 3 root root 17 Mar 9 2021 home
lrwxrwxrwx. 1 root root 7 Mar 9 2021 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 Mar 9 2021 lib64 -> usr/lib64
drwxr-xr-x. 2 root root 6 Aug 12 2019 media
drwxr-xr-x. 2 root root 6 Aug 12 2019 mnt
drwxr-xr-x. 3 root root 15 Mar 9 2021 opt
dr-xr-xr-x. 505 root root 0 Mar 9 2021 proc
dr-xr-xr-x. 13 root root 4096 Mar 9 17:06 root
drwxr-xr-x. 36 root root 1100 Mar 9 15:20 run
lrwxrwxrwx. 1 root root 8 Mar 9 2021 sbin -> usr/sbin
drwxr-xr-x. 2 root root 6 Aug 12 2019 srv
dr-xr-xr-x. 13 root root 0 Mar 9 2021 sys
drwxrwxrwt. 20 root root 4096 Mar 9 16:57 tmp
drwxr-xr-x. 13 root root 4096 Mar 9 2021 usr
drwxr-xr-x. 20 root root 4096 Mar 9 2021 var
```

The terminal status bar at the bottom shows 'Ready', 'ssh2: AES-256-CTR', '24, 24', '25 Rows, 80 Cols', 'Xterm', and 'CAP NUM'.

bin (binaries)存放二进制可执行文件

sbin (super user binaries)存放二进制可执行文件，只有root才能访问

etc (etcetera)存放系统配置文件

usr (unix shared resources)用于存放共享的系统资源

home 存放用户文件的根目录

root 超级用户目录

dev (devices) 用于存放设备文件

lib (library)存放跟文件系统中的程序运行所需要的共享库及内核模块

mnt (mount)系统管理员安装临时文件系统的安装点

boot存放用于系统引导时使用的各种文件

tmp (temporary)用于存放各种临时文件

var (variable)用于存放运行时需要改变数据的文件

开发人员关注目录:

配置目录etc: 软件安装后配置文件选择放入的地址。

软件安装目录: `usr/local`

安装应用程序目录: `opt`,部署的API程序一般放在这个目录。部署的API放入用户的目录:`/home/用户名`

注意: Linux系统的文件系统是以斜杠“/”开始, 不是windows中的盘符概念。

5、Linux的用户

Linux系统是一个多用户多任务的分时操作系统, 任何一个要使用系统资源的用户, 都必须首先向系统管理员申请一个账号, 然后以这个账号的身份进入系统。

用户的账号一方面可以帮助系统管理员对使用系统的用户进行跟踪, 并控制他们对系统资源的访问; 另一方面也可以帮助用户组织文件, 并为用户提供安全性保护。

每个用户账号都拥有一个唯一的用户名和各自的口令。

用户在登录时键入正确的用户名和口令后, 就能够进入系统和自己的主目录。

5.1 用户账号

用户账户一般分为两大类: 超级管理员和普通用户。

超级管理员: 用`root`表示, `root`用户在系统中拥有最高权限。

普通用户: 除了`root`之外的用户。例如我们之前创建的`lina`用户, 就是普通用户。

5.2 查看用户信息

#语法 `cat /etc/passwd`


```
[root@centos1lna01 ~]# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-bus-proxy:x:999:998:systemd Bus Proxy:/:/sbin/nologin
systemd-network:x:998:997:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:system message bus:/:/sbin/nologin
polkitd:x:997:996:User for polkitd:/:/sbin/nologin
abrt:x:173:173:/:etc/abrt:/sbin/nologin
unbound:x:996:995:Unbound DNS resolver:/etc/unbound:/sbin/nologin
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
colord:x:995:994:User for colord:/var/lib/colord:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin
geoclue:x:994:991:User for geoclue:/var/lib/geoclue:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
chrony:x:993:990:/:/var/lib/chrony:/sbin/nologin
radvd:x:75:75:radvd user:/:/sbin/nologin
qemu:x:107:107:qemu user:/:/sbin/nologin
avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
setroubleshoot:x:992:989:/:/var/lib/setroubleshoot:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
saslauthd:x:991:76:Saslauthd user:/run/saslauthd:/sbin/nologin
libstoragemgmt:x:990:988:daemon account for libstoragemgmt:/var/run/lsm:/sbin/nologin
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:42:/:/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:989:984:/:run/gnome-initial-setup:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/ssh:/sbin/nologin
postfix:x:89:89:/:/var/spool/postfix:/sbin/nologin
ntp:x:38:38:/:etc/ntp:/sbin/nologin
tcpdump:x:72:72:/:/sbin/nologin
lina:x:1000:1000:lina:/home/lina:/bin/bash
[root@centos1lna01 ~]#
```

- root:用户名
- x : 密码 : 已经加密 , 密码存放在 `vi /etc/shadow` 密码文件
- 0: 账号id , userId
- 0: 组id , group id

5.2 创建一个用户

```
# 创建一个用户
#语法: useradd 选项 用户名
#选项:
    -c comment 指定一段注释性描述。
    -d 目录 指定用户主目录, 如果此目录不存在, 则同时使用-m选项, 可以创建主目录。
    -g 用户组 指定用户所属的用户组。
    -G 用户组, 用户组 指定用户所属的附加组。
    -s Shell文件 指定用户的登录Shell。
    -u 用户号 指定用户的用户号, 如果同时有-o选项, 则可以重复使用其他用户的标识号。

#案例1:
useradd lina02
```

5.2 删除一个用户

```
# 删除一个用户
#语法: userdel 选项 用户名; 其中-r是常用的选项, 表示将用户的主目录一起删除。
#案例:
userdel -r lina02
```

5.3 用户口令的管理

用户口令的管理其实就是用户密码的管理。用户账号刚创建时没有口令, 但是被系统锁定, 无法使用, 必须为其指定口令后才可以使使用, 即使是指定空口令。

指定和修改用户口令的命令是 `passwd`。超级用户可以为自己和其他用户指定口令，普通用户只能用它修改自己的口令。

用户口令的管理

#语法: `passwd` 选项 用户名

可使用的选项:

- l 锁定口令，即禁用账号。
- u 口令解锁。
- d 使账号无口令。
- f 强迫用户下次登录时修改口令。

如果默认用户名，则修改当前用户的口令。

例如，切换到用户lina，则下面的命令修改该用户自己的口令：PS:linux操作系统中密码部分不予显示，所以看起来是空白的，实际已经输入了密码。

```
Last login: Wed Mar 10 17:55:19 2021 from 192.168.26.1
[root@centoslina01 ~]# su lina
[lina@centoslina01 root]$ passwd
Changing password for user lina.
Changing password for lina.
(current) UNIX password:
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[lina@centoslina01 root]$
```

如果是超级用户，可以用下列形式指定任何用户的口令：

```
Last login: Wed Mar 10 18:20:32 2021 from 192.168.26.1
[root@centoslina01 ~]# passwd lina
Changing password for user lina.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@centoslina01 ~]#
```

普通用户修改自己的口令时，`passwd`命令会要求先输入原口令，验证后再要求用户输入两遍新口令，如果两次输入的口令一致，则将这个口令指定给用户；而超级用户为用户指定口令时，就不需要知道原口令。

为了系统安全起见，用户应该选择比较复杂的口令，例如最好使用8位长的口令，口令中包含有大写、小写字母和数字，并且应该与姓名、生日等不相同。

6、Linux的常用命令-重要

6.1 Linux目录管理

Linux的目录结构为树状结构，最顶级的目录为根目录 `/`。

其他目录通过挂载可以将它们添加到树中，通过解除挂载可以移除它们。

绝对路径与相对路径：

绝对路径：由根目录 `/` 写起，例如：`/usr/share/doc` 这个目录。

相对路径：不是由 `/` 写起，例如由 `/usr/share/doc` 要到 `/usr/share/man` 底下时，可以写成：`cd ../man` 这就是相对路径的写法。

6.1.1 查看目录

Linux系统当中，ls 命令可能是最常被运行的。

语法：

```
ls :查看当前路径下的文件名称
ls -a :全部的文件，连同隐藏文件(开头为 . 的文件)一起列出来(常用)
ls -d :仅列出目录本身，而不是列出目录内的文件数据(常用)
ls -l :长数据串列出，包含文件的属性与权限等等数据：(常用)
ll :等价于ls -l
ls -al :目录下的所有文件列出来(含属性与隐藏档)
```

6.1.2 切换目录

cd是Change Directory的缩写，这是用来变换工作目录的命令。

语法：

```
cd [相对路径或绝对路径]
```

#使用绝对路径切换到 **local** 目录

```
cd /usr/local/
```

表示回到自己的家目录，亦即是 **/root** 这个目录

```
cd ~
```

表示去到目前的上一级目录，亦即是 **/root** 的上一级目录的意思；

```
cd ..
```

6.1.3 显示当前目录

pwd 是 **Print Working Directory** 的缩写，也就是显示目前所在目录的命令。

语法：

```
pwd
```

6.1.4 创建目录

mkdir (make directory)用来创建新的目录。

语法：mkdir [-mp] 目录名称

-m : 配置文件的权限喔！直接配置，不需要看默认权限的脸色

-p : 直接将所需要的目录(包含上一级目录)递归创建起来！

查看帮助 `mkdir --help`

示例：/tmp底下尝试创建数个新目录看看：

```
cd /tmp
```

```
mkdir lina <==创建一名为 lina 的新目录
```

```
mkdir test1/test2/test3/test4
```

```
mkdir: cannot create directory `test1/test2/test3/test4':
```

```
No such file or directory <== 没办法直接创建此目录啊！
```

```
mkdir -p test1/test2/test3/test4
```

加了这个 -p 的选项，可以自行帮你创建多层目录！

6.1.5 删除目录

`rmdir` 命令用来删除空的目录。

语法: `rmdir [-p] 目录名称`
-p : 连同上一级『空的』目录也一起删除

6.2 文件操作

6.2.1 查看文件内容

6.2.1.1 cat 查看所有内容

语法: `cat` 由第一行开始显示文件内容

6.2.1.2 more 一页一页的显示文件内容

一页一页翻动
语法: `more` 文件名称
例如: `more /etc/man_db.config`
....(中间省略)....
--More--(28%) <== 重点在这一行喔! 你的光标也会在这里等待你的命令
在 `more` 这个程序的运行过程中, 你有几个按键可以按的:
空白键 (space): 代表向下翻一页;
Enter : 代表向下翻『一行』;
:f : 立刻显示出档名以及目前显示的行数;
q : 代表立刻离开 `more`, 不再显示该文件内容。

6.2.1.3 less 一页一页翻动

语法: `less` 文件名称
例如: `less /etc/man.config`
....(中间省略)....
: <== 这里可以等待你输入命令!

`less`运行时可以输入的命令有:
空白键 : 向下翻动一页;
[pagedown]: 向下翻动一页;
[pageup] : 向上翻动一页;
q : 离开 `less` 这个程序;

6.2.2 创建文件

语法: `touch` 创建一个空白的普通文件

#示例:
[root@centoslina01 /]# `cd lina`
[root@centoslina01 lina]# `touch lina01.txt`
[root@centoslina01 lina]# `ll`
total 0
-rw-r--r--. 1 root root 0 Mar 11 10:08 lina01.txt
[root@centoslina01 lina]#

6.2.3 写入内容

echo 把内容重定向到指定的文件中，有则打开，无则创建

语法： echo '内容'>文件名称 覆盖模式
echo '内容'>>文件名称 追加模式

#示例：

```
[root@centoslina01 lina]# echo 'jiaobaoyu'>lina01.txt
[root@centoslina01 lina]# echo 'jiaobaoyu'>lina01.txt
[root@centoslina01 lina]# cat lina01.txt
jiaobaoyu
[root@centoslina01 lina]# echo 'jiaobaoyu'>>lina02.txt
[root@centoslina01 lina]# echo 'jiaobaoyu'>>lina02.txt
[root@centoslina01 lina]# cat lina02.txt
jiaobaoyu
jiaobaoyu
[root@centoslina01 lina]#
```

6.2.4 复制/拷贝文件

语法： cp [-adfilprsu] 来源档(source) 目标档(destination)

- a: 相当於 -pdr 的意思，至於 pdr 请参考下列说明；(常用)
- d: 若来源档为连结档的属性(link file)，则复制连结档属性而非文件本身；
- f: 为强制(force)的意思，若目标文件已经存在且无法开启，则移除后再尝试一次；
- i: 若clear问动作的进行(常用)
- l: 进行硬式连结(hard link)的连结档创建，而非复制文件本身；
- p: 连同文件的属性一起复制过去，而非使用默认属性(备份常用)；
- r: 递归持续复制，用於目录的复制行为；(常用)
- s: 复制成为符号连结档 (symbolic link)，亦即『捷径』文件；
- u: 若 destination 比 source 旧才升级 destination !

6.2.5 移动文件

mv 可以移动文件与目录，或修改名称。

语法：

```
mv [-fiu] source destination
```

- f : force 强制的意思，如果目标文件已经存在，不会询问而直接覆盖；
- i : 若目标文件 (destination) 已经存在时，就会询问是否覆盖！
- u : 若目标文件已经存在，且 source 比较新，才会升级 (update)

示例：复制一文件，创建一目录，将文件移动到目录中

```
cd /tmp
cp ~/.bashrc bashrc
mkdir mvtest
mv lina mvtest
```

将刚刚的目录名称更名为 mvtest2

```
[root@www tmp]# mv mvtest mvtest2
```

6.2.6 删除文件

rm 可以移除文件或目录。

语法： rm [-fir] 文件或目录

- f : 就是 **force** 的意思，忽略不存在的文件，不会出现警告信息；
- i : 互动模式，在删除前会询问使用者是否动作
- r : 递归删除啊！最常用在目录的删除了！这是非常危险的选项！！

6.3 打包与拆包

一般Linux 上常用的压缩方式是选用**tar** 将许多文件打包成一个文件，再以 **gzip**压缩命令压缩成紧xxx.tar.gz(或者xxx.tgz)的文件。

常用参数：

- c:创建一个新**tar**文件
- v:显示运行过程的信息
- f:指定文件名
- z:调用**gzip** 压缩命令进行压缩
- t:查看压缩文件的内容
- x:解开**tar** 文件

打包:tar -cvf xxx.tar 要打包的文件或目录的列表，用空格隔开

打包并且压缩:tar -zcvf xxx.tar.gz 要打包的文件或目录的列表，用空格隔开

解压:tar -xvf xx.tar

tar -zxvf xx.tar.gz -C /usr/lina 注意-C大写，防止拆包解压的路径找不到

```

root@centos1ina1: /test1/test2 x
[root@centos1ina01 ~]# cd /
[root@centos1ina01 /]# ls
bin boot dev etc home lib lib64 media mnt myself opt proc root run
[root@centos1ina01 /]# touch lina01.txt
[root@centos1ina01 /]# ls
bin dev home lib64 media myself proc run srv tmp var
boot etc lib lina01.txt mnt opt root sbin sys usr
[root@centos1ina01 /]# tar -cvf test.tar myself lina01.txt
myself/
myself/lina01.txx
lina01.txt
[root@centos1ina01 /]# ls
bin dev home lib64 media myself proc run srv test.tar usr
boot etc lib lina01.txt mnt opt root sbin sys tmp var
[root@centos1ina01 /]# tar -zcvf test.tar.gz myself lina01.txt
myself/
myself/lina01.txx          打包并压缩为指定文件
lina01.txt
[root@centos1ina01 /]# ls
bin dev home lib64 media myself proc run srv test.tar tmp v
boot etc lib lina01.txt mnt opt root sbin sys test.tar.gz usr
[root@centos1ina01 /]# ll
total 48
lrwxrwxrwx. 1 root root 7 Mar 12 12:28 bin -> usr/bin
dr-xr-xr-x. 4 root root 4096 Mar 12 12:47 boot
drwxr-xr-x. 20 root root 3240 Mar 12 12:46 dev
drwxr-xr-x. 129 root root 8192 Mar 12 17:29 etc
drwxr-xr-x. 4 root root 30 Mar 12 15:54 home
lrwxrwxrwx. 1 root root 7 Mar 12 12:28 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 Mar 12 12:28 lib64 -> usr/lib64
-rw-r--r--. 1 root root 0 Mar 12 18:58 lina01.txt
drwxr-xr-x. 2 root root 6 Aug 12 2015 media
drwxr-xr-x. 2 root root 6 Aug 12 2015 mnt
drwxr-xr-x. 2 root root 23 Mar 12 18:38 myself
drwxr-xr-x. 3 root root 15 Mar 12 12:34 opt
dr-xr-xr-x. 526 root root 0 Mar 12 12:46 proc
dr-xr-x---. 13 root root 4096 Mar 12 15:48 root
drwxr-xr-x. 36 root root 1100 Mar 12 15:45 run
lrwxrwxrwx. 1 root root 8 Mar 12 12:28 sbin -> usr/sbin
drwxr-xr-x. 2 root root 6 Aug 12 2015 srv
dr-xr-xr-x. 13 root root 0 Mar 12 12:46 sys
-rw-r--r--. 1 root root 10240 Mar 12 18:59 test.tar
-rw-r--r--. 1 root root 165 Mar 12 18:59 test.tar.gz
drwxrwxrwt. 20 root root 4096 Mar 12 18:59 tmp
drwxr-xr-x. 13 root root 4096 Mar 12 12:28 usr
drwxr-xr-x. 20 root root 4096 Mar 12 12:46 var
[root@centos1ina01 /]# mkdir test1
[root@centos1ina01 /]# tar -xvf test.tar -C /test1 打包为指定文件到指定位置
myself/
myself/lina01.txx
lina01.txt
[root@centos1ina01 /]# cd test1/
[root@centos1ina01 test1]# ls
lina01.txt myself
[root@centos1ina01 test1]# mkdir test2
[root@centos1ina01 test1]# tar -zxvf /test.tar.gz -C /test2
tar: /test2: Cannot open: No such file or directory
tar: Error is not recoverable: exiting now
[root@centos1ina01 test1]# tar -zxvf test.tar.gz -C /test1/test2
myself/
myself/lina01.txx          解压文件到指定位置
lina01.txt
[root@centos1ina01 test1]# ls
lina01.txt myself test2
[root@centos1ina01 test1]# cd test2/
[root@centos1ina01 test2]# ls
lina01.txt myself
[root@centos1ina01 test2]#

```

6.4 其它命令

6.4.1 查看进程快照

ps : - axu 显示当前进程的快照

查看Java进程: ps - axu | grep java

查看MySQL进程 ps - axu | grep mysql

6.4.2 管道 |

管道是Linux中比较重要且常用的一个内容，其作用是将一个命令的输出作为另一个命令的输入，组合使用。

```
ls --help | more    #分页查看帮助信息，按q退出
```

7、VIM编辑器-重要

7.1 VIM的模式

vim 共分为三种模式，分别是**命令模式 (Command mode)**，**输入模式 (Insert mode)** 和**底线命令模式 (Last line mode)**。

7.1.1 命令模式：

用户刚刚启动 vi/vim，便进入了命令模式。

此状态下敲击键盘动作会被vim识别为命令，而非输入字符。比如我们此时按下i，并不会输入一个字符，i被当作了一个命令。

以下是常用的几个命令：

- i 切换到输入模式，以输入字符。
- x 删除当前光标所在处的字符。
- : 切换到底线命令模式，以在最底一行输入命令。

若想要编辑文本：启动vim，进入了命令模式，按下i，切换到输入模式。
命令模式只有一些最基本的命令，因此仍要依靠底线命令模式输入更多命令。

7.1.2 输入模式

在命令模式下按下i就进入了输入模式。

在输入模式中，可以使用以下按键：

- 字符按键以及Shift组合，输入字符
- ENTER，回车键，换行
- BACK SPACE，退格键，删除光标前一个字符
- DEL，删除键，删除光标后一个字符
- 方向键，在文本中移动光标
- HOME/END，移动光标到行首/行尾
- Page Up/Page Down，上/下翻页
- Insert，切换光标为输入/替换模式，光标将变成竖线/下划线
- ESC，退出输入模式，切换到命令模式

7.1.3 底线命令模式

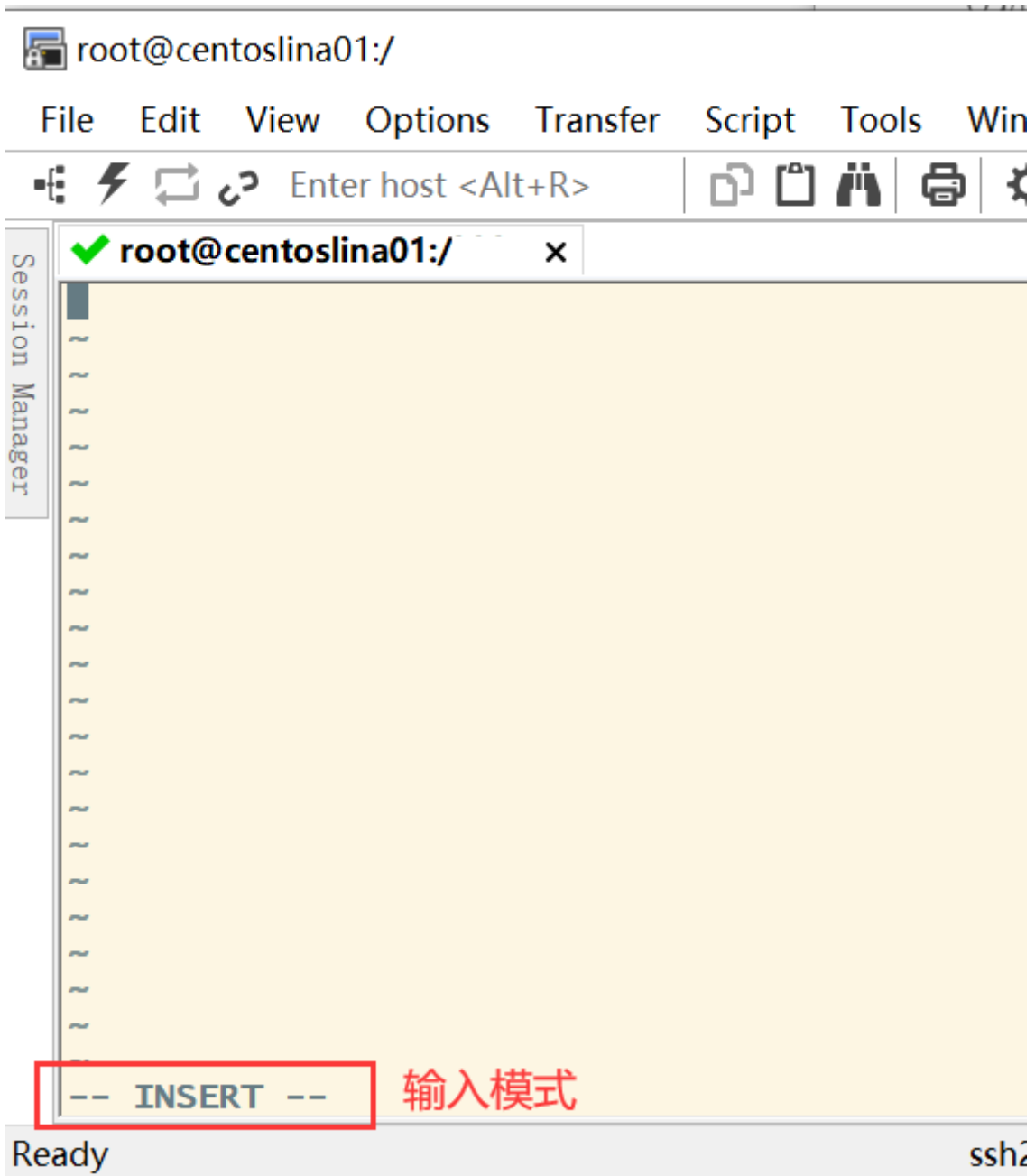
在命令模式下按下: (英文冒号) 就进入了底线命令模式。

底线命令模式可以输入单个或多个字符的命令，可用的命令非常多。

在底线命令模式中，基本的命令有（已经省略了冒号）：

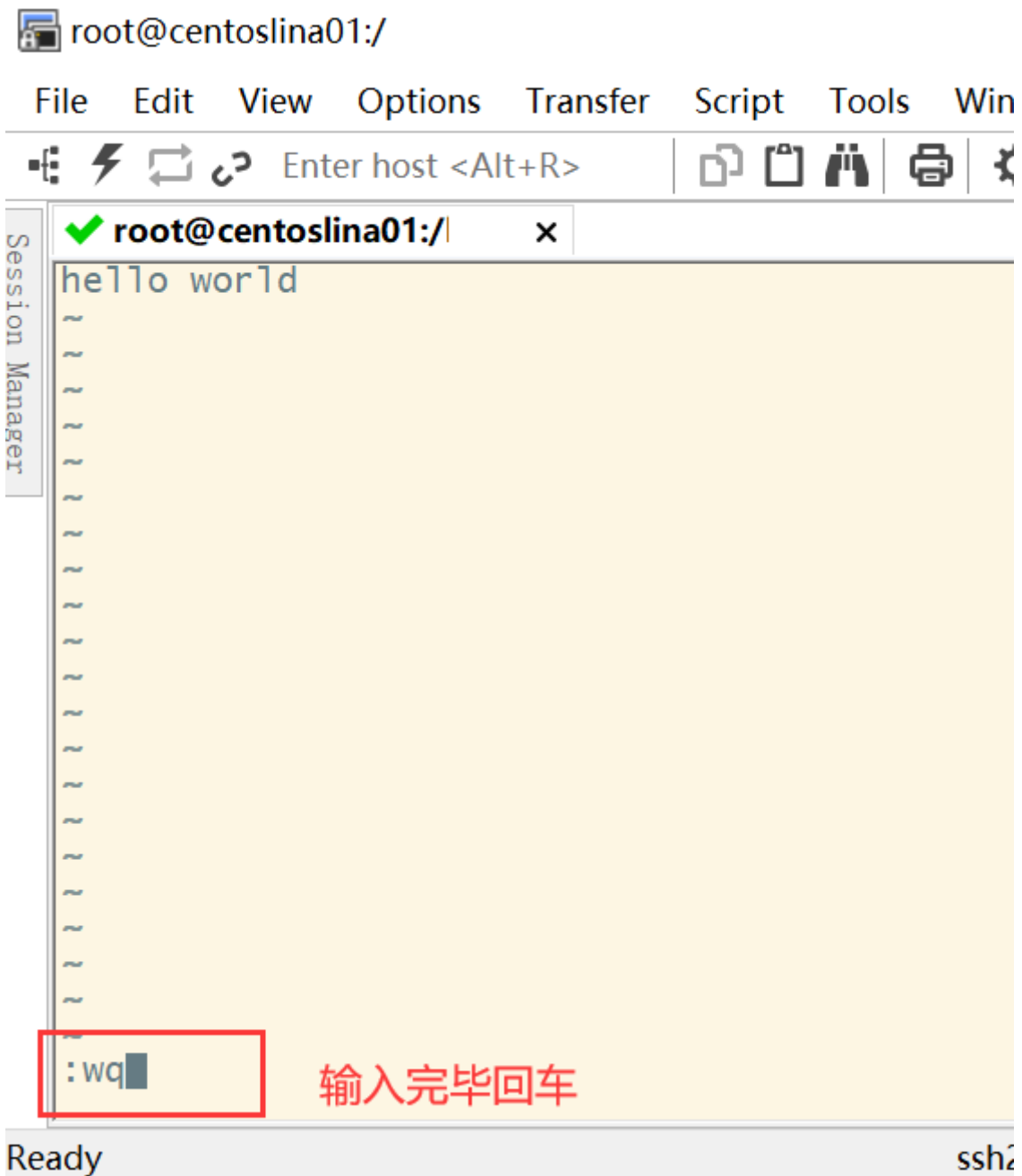
- q 退出程序
- w 保存文件

按ESC键可随时退出底线命令模式。



输入 hello world ,按下 ESC 按钮回到一般模式, 此时画面左下角的 - INSERT - 消失。

在一般模式中按下 **:wq** 即可实现保持文档并离开vim.



指令	说明
:w	将编辑的数据写入硬盘档案中(常用)
:w!	若文件属性为『只读』时，强制写入该档案。不过，到底能不能写入，还是跟你对该档案的档案权限有关啊！
:q	离开 vi (常用)
:q!	使用 ! 为强制离开不储存档案。
:wq	储存后离开，若为 :wq! 则为强制储存后离开 (常用)
:set number	显示行号
:set nonumber	取消行号显示

8、Linux的文件权限

Linux操作系统中对权限的管理很严格。Linux系统中不仅是对用户与组根据UID,GID进行了管理，还对Linux系统中的文件，按照用户与组进行分类，针对不同的群体进行了权限管理，用他来确定谁能通过何种方式对文件和目录进行访问和操作。

8.1文件权限



权限共有10个字符，我们将它分为4大部分来理解：

- - - - -

第1部分、表示文件的类型

- 表示是一个文件
- d** 表示是一个目录
- l** 表示是一个连接（理解为快捷方式）

第2部分、当前用户具有的对该文件的权限（owner 属主，缩写u）

第3部分、当前组内其他用户具有的对该文件的权限（group 属组，缩写g）

第4部分、其他组的用户具有的对该文件的权限（other 其他，缩写o）

r: Read 读

w: Write 写

x: execute 执行

针对目录加执行权限，文件不加执行权限（因文件具备执行权限有安全隐患）

对于文件和目录来说，**r**，**w**，**x**有着不同的作用和含义：

针对文件：**r**：读取文件内容

w：修改文件内容

x：执行权限对除二进制程序以外的文件没什么意义

针对目录：目录本质可看做是存放文件列表、节点号等内容的文件

r：查看目录下的文件列表

w：删除和创建目录下的文件

x：可以cd进入目录，能查看目录中文件的详细属性，能访问目录下文件内容（基础权限）

限）

PS: root账户不受文件权限的读写限制，执行权限受限制

权限项	文件类型	读	写	执行	读	写	执行	读	写	执行
字符表示	(d l c s p)	(r)	(w)	(x)	(r)	(w)	(x)	(r)	(w)	(x)
数字表示		4	2	1	4	2	1	4	2	1
权限分配		文件所有者			文件所属组用户			其他用户		

用户获取文件权限的顺序：先看是否为所有者，如果是，则后面权限不看；再看是否为所属组，如果是，则后面权限不看。

8.2 修改文件权限

chown 是change owner的意思，主要作用就是改变文件或者目录所有者。

chmod 修改文件和文件夹读写执行属性。 使用权限:所有使用者

chown 修改文件和文件夹的用户和用户组属性。使用权限:root

例如: `chown lina:lina lina.txt`

8.2.1 mode方式

语法: `chmod who opt per file`

who: u g o a (all) (u 用户user ,g用户组group,o其他用户,a所有用户默认)

opt: +添加某个权限 -取消某个权限 =赋予权限

per: r w x X

示例:

`chmod u=rwx,g=r lina03.txt`

`chmod u+x,g+w,o+w test.log` //r 读, w 写 x 执行

8.2.2 数字方式

语法: `chmod xxx file`

rwX rw- r--

421 420 400

7 6 4

0无权限,1表可执行=x,2表写入权限=w,4表可读权限=r

例如: `-rwxr--r-- . 1 root root 10 Oct 16 02:55 yhp.log`

用户权限= $rwx=4+2+1=7$

所属组权限= $r--=4+0+0=4$

其他用户权限= $r--=4+0+0=4$

组合:744

修改权限:

都加入写入权限:+2

给用户组加入写入权限:+2

给其他用户加入可执行权限:+1

`chmod 765 yhp.log`

```
Session Manager
root@centoslina01:/ x
total 52
lrwxrwxrwx. 1 root root 7 Mar 12 12:28 bin -> usr/bin
dr-xr-xr-x. 4 root root 4096 Mar 12 12:47 boot
drwxr-xr-x. 20 root root 3240 Mar 12 12:46 dev
drwxr-xr-x. 129 root root 8192 Mar 12 17:29 etc
drwxr-xr-x. 4 root root 30 Mar 12 15:54 home
lrwxrwxrwx. 1 root root 7 Mar 12 12:28 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 Mar 12 12:28 lib64 -> usr/lib64
-rw-r--r--. 1 root root 12 Mar 12 19:26 lina01.txt
drwxr-xr-x. 2 root root 6 Aug 12 2015 media
drwxr-xr-x. 2 root root 6 Aug 12 2015 mnt
drwxr-xr-x. 2 root root 23 Mar 12 18:38 myself
drwxr-xr-x. 3 root root 15 Mar 12 12:34 opt
dr-xr-xr-x. 527 root root 0 Mar 12 12:46 proc
dr-xr-x---. 13 root root 4096 Mar 12 19:31 root
drwxr-xr-x. 36 root root 1100 Mar 12 15:45 run
lrwxrwxrwx. 1 root root 8 Mar 12 12:28 sbin -> usr/sbin
drwxr-xr-x. 2 root root 6 Aug 12 2015 srv
dr-xr-xr-x. 13 root root 0 Mar 12 12:46 sys
drwxr-xr-x. 4 root root 48 Mar 12 19:01 test1
-rw-r--r--. 1 root root 10240 Mar 12 18:59 test.tar
-rw-r--r--. 1 root root 165 Mar 12 18:59 test.tar.gz
drwxrwxrwt. 20 root root 4096 Mar 12 20:20 tmp
drwxr-xr-x. 13 root root 4096 Mar 12 12:28 usr
drwxr-xr-x. 20 root root 4096 Mar 12 12:46 var
[root@centoslina01 /]# chown lina02 lina01.txt
[root@centoslina01 /]# ll
total 52
lrwxrwxrwx. 1 root root 7 Mar 12 12:28 bin -> usr/bin
dr-xr-xr-x. 4 root root 4096 Mar 12 12:47 boot
drwxr-xr-x. 20 root root 3240 Mar 12 12:46 dev
drwxr-xr-x. 129 root root 8192 Mar 12 17:29 etc
drwxr-xr-x. 4 root root 30 Mar 12 15:54 home
lrwxrwxrwx. 1 root root 7 Mar 12 12:28 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 Mar 12 12:28 lib64 -> usr/lib64
-rw-r--r--. 1 lina02 root 12 Mar 12 19:26 lina01.txt
drwxr-xr-x. 2 root root 6 Aug 12 2015 media
drwxr-xr-x. 2 root root 6 Aug 12 2015 mnt
drwxr-xr-x. 2 root root 23 Mar 12 18:38 myself
drwxr-xr-x. 3 root root 15 Mar 12 12:34 opt
dr-xr-xr-x. 527 root root 0 Mar 12 12:46 proc
dr-xr-x---. 13 root root 4096 Mar 12 19:31 root
drwxr-xr-x. 36 root root 1100 Mar 12 15:45 run
lrwxrwxrwx. 1 root root 8 Mar 12 12:28 sbin -> usr/sbin
drwxr-xr-x. 2 root root 6 Aug 12 2015 srv
dr-xr-xr-x. 13 root root 0 Mar 12 12:46 sys
drwxr-xr-x. 4 root root 48 Mar 12 19:01 test1
-rw-r--r--. 1 root root 10240 Mar 12 18:59 test.tar
-rw-r--r--. 1 root root 165 Mar 12 18:59 test.tar.gz
drwxrwxrwt. 20 root root 4096 Mar 12 20:22 tmp
drwxr-xr-x. 13 root root 4096 Mar 12 12:28 usr
drwxr-xr-x. 20 root root 4096 Mar 12 12:46 var
[root@centoslina01 /]# chown lina02:lina lina01.txt
[root@centoslina01 /]# ll
total 52
lrwxrwxrwx. 1 root root 7 Mar 12 12:28 bin -> usr/bin
dr-xr-xr-x. 4 root root 4096 Mar 12 12:47 boot
drwxr-xr-x. 20 root root 3240 Mar 12 12:46 dev
drwxr-xr-x. 129 root root 8192 Mar 12 17:29 etc
drwxr-xr-x. 4 root root 30 Mar 12 15:54 home
lrwxrwxrwx. 1 root root 7 Mar 12 12:28 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 Mar 12 12:28 lib64 -> usr/lib64
-rw-r--r--. 1 lina02 lina 12 Mar 12 19:26 lina01.txt
drwxr-xr-x. 2 root root 6 Aug 12 2015 media
drwxr-xr-x. 2 root root 6 Aug 12 2015 mnt
```

9、Linux常用网络操作

9.1 主机名操作

hostname :显示主机名

hostname XXX : 修改主机名, 不推荐, 临时生效

永久生效修改主机名需要修改/etc/sysconfig/network文件

9.2 查询系统完整信息

uname -a : 显示完整的系统信息

9.3 IP地址操作

查看IP地址: ipconfig 修改IP地址: 修改/etc/sysconfig/network-scripts/

```
Session Manager
✓ root@centoslina01:/etc/sysconfig/network-scripts x
TYPE="Ethernet"
BOOTPROTO="static" 静态IP
DEFROUTE="yes"
PEERDNS="yes"
PEERROUTES="yes"
IPV4_FAILURE_FATAL="no"
IPV6INIT="yes"
IPV6_AUTOCONF="yes"
IPV6_DEFROUTE="yes"
IPV6_PEERDNS="yes"
IPV6_PEERROUTES="yes"
IPV6_FAILURE_FATAL="no"
NAME="eno16777736"
UUID="c26d55bf-ccf9-431a-bc5a-3bb9732d889e"
DEVICE="eno16777736"
ONBOOT="yes"
IPADDR=192.168.26.129 本机地址
NETMASK=255.255.255.0 子网
GATEWAY=192.168.26.1 默认网关 (和主机保持一直)
DNS1=114.114.114.114
~
-- INSERT --
```

重启网络服务

service network restart

9.4 域名映射

修改 /etc/hosts文件

[illegible]

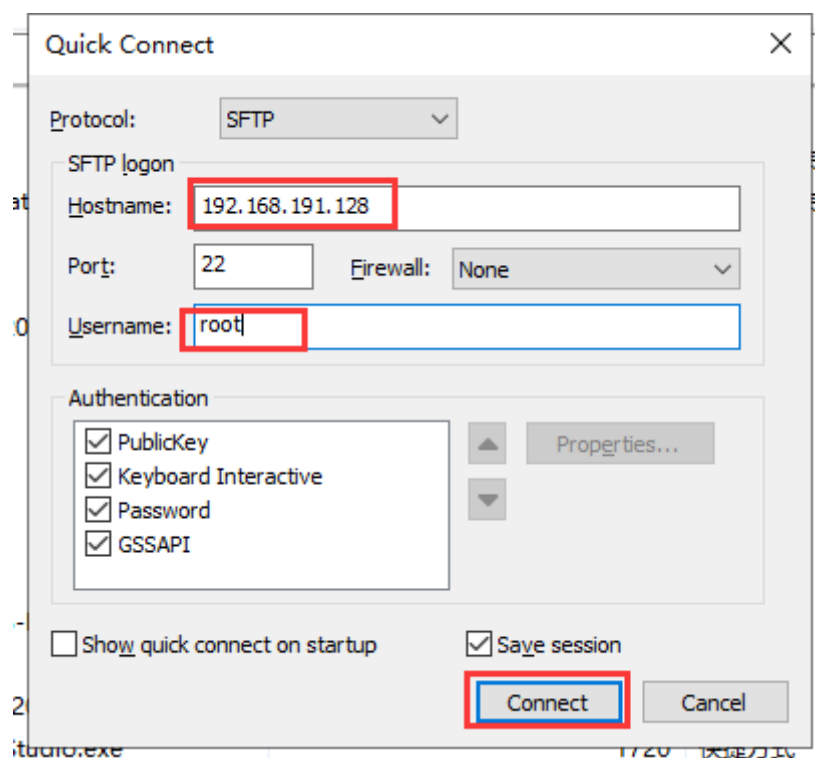
资料中提供了远程上传文件的工具

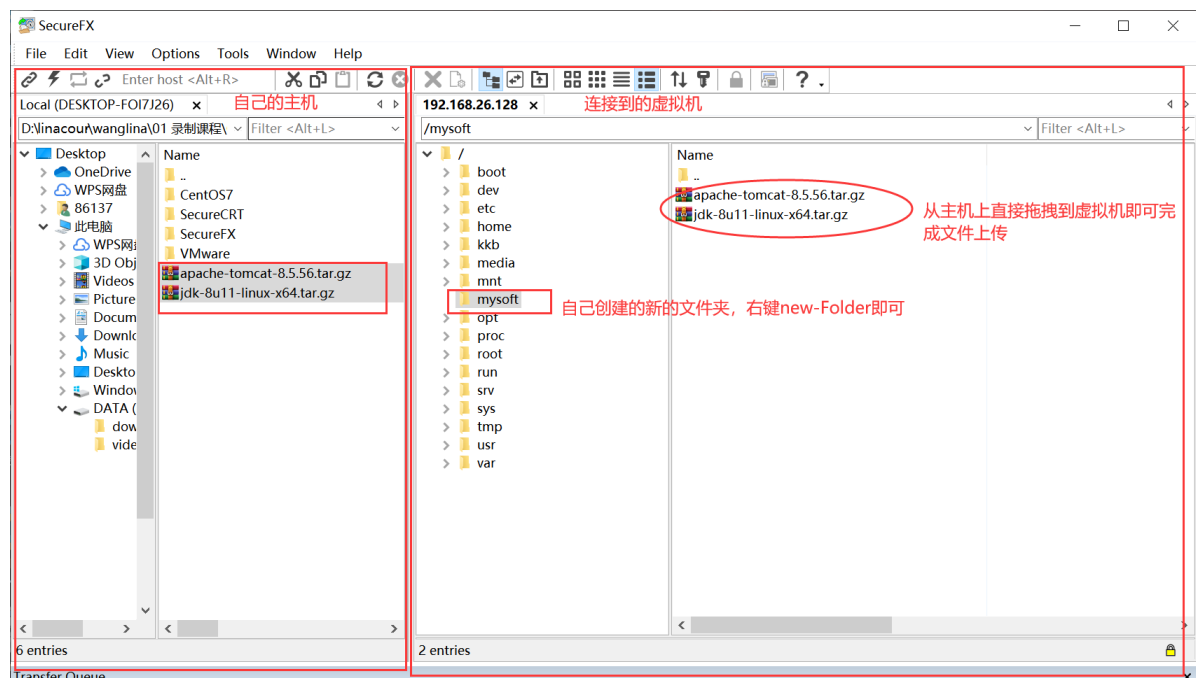
DATA (D:) > linacour > wanglina > 01 录制课程 > 01 Linux > 工具 > SecureFX

名称	修改日期	类
mfc140enu.dll	2019/2/5 8:18	应
mfc140u.dll	2019/2/5 8:18	应
msvcp100.dll	2019/2/5 8:17	应
msvcp140.dll	2019/2/5 8:18	应
msvcr100.dll	2019/2/5 8:17	应
SCP.dll	2019/2/5 8:53	应
SecureFX.chm	2019/3/17 21:37	编
SecureFX.exe	2019/4/18 20:28	应

可以发送一个快捷方式到桌面

双击打开





10.2 安装JDK

1、上传jdk文件到linux

使用SecureFX连linux后，将JDK压缩包上传到新建的mysoft文件夹。

2、解压压缩包

```
[root@centoslina01 mysoft]# tar -xzf jdk-8u11-linux-x64.tar.gz
```

3、测试jdk是否安装成功

```
[root@centoslina01 mysoft]# ll
total 165436
-rw-r--r--. 1 root root 10377460 Jun 17 2020 apache-tomcat-8.5.56.tar.gz
drwxr-xr-x. 8 10 143 4096 Jun 17 2014 jdk1.8.0_11
-rw-r--r--. 1 root root 159019376 May 8 2020 jdk-8u11-linux-x64.tar.gz
[root@centoslina01 mysoft]# cd jdk1.8.0_11/
[root@centoslina01 jdk1.8.0_11]# ll
total 25444
drwxr-xr-x. 2 10 143 4096 Jun 17 2014 bin
-r--r--r--. 1 10 143 3244 Jun 17 2014 COPYRIGHT
drwxr-xr-x. 4 10 143 4096 Jun 17 2014 db
drwxr-xr-x. 3 10 143 4096 Jun 17 2014 include
-rw-r--r--. 1 10 143 4673670 Jun 17 2014 javafx-src.zip
drwxr-xr-x. 5 10 143 4096 Jun 17 2014 jre
drwxr-xr-x. 5 10 143 4096 Jun 17 2014 lib
-r--r--r--. 1 10 143 40 Jun 17 2014 LICENSE
drwxr-xr-x. 4 10 143 44 Jun 17 2014 man
-r--r--r--. 1 10 143 159 Jun 17 2014 README.html
-rw-r--r--. 1 10 143 525 Jun 17 2014 release
-rw-r--r--. 1 10 143 21047086 Jun 17 2014 src.zip
-rw-r--r--. 1 10 143 110114 Jun 17 2014 THIRDPARTYLICENSEREADME-JAVAFX.txt
-r--r--r--. 1 10 143 178445 Jun 17 2014 THIRDPARTYLICENSEREADME.txt
[root@centoslina01 jdk1.8.0_11]# cd bin
[root@centoslina01 bin]# ./java -version
java version "1.8.0_11"
Java(TM) SE Runtime Environment (build 1.8.0_11-b12)
Java HotSpot(TM) 64-Bit Server VM (build 25.11-b03, mixed mode)
[root@centoslina01 bin]#
```

注意:这里实在jdk的bin目录下运行的。并且java命令前有"./"

4、配置环境变量

(1)编辑profile文件:

```
java hotspot(™) 64-Bit Server VM (build 25.65-b01, mixed mode)
[root@centoslina01 bin]# vim /etc/profile
```

(2)在profile文件中添加JAVA_HOME变量:

```
root@centoslina01:/mysoft/jdk1.8.0_11/bin x
fi
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL
# By default, we want umask to get set. This sets it for login shell
# Current threshold for system reserved uid/gids is 200
# You could check uidgid reservation validity in
# /usr/share/doc/setup-*/uidgid file
if [ $UID -gt 199 ] && [ "`id -gn`" = "`id -un`" ]; then
    umask 002
else
    umask 022
fi
for i in /etc/profile.d/*.sh ; do
    if [ -r "$i" ]; then
        if [ "${-#*i}" != "$-" ]; then
            . "$i"
        else
            . "$i" >/dev/null
        fi
    fi
done
unset i
unset -f pathmunge
export JAVA_HOME=/mysoft/jdk1.8.0_11
export PATH=$PATH:$JAVA_HOME/bin
-- INSERT --
```

(3)使环境变量及时生效

```
[root@centoslina01 /]# source /etc/profile
```

(4)查看变量是否更新成功:

```
[root@centoslina01 /]# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin:/mysoft/jdk1.8.0_11/bin
```

(5)在任意目录下测试环境是否配置成功

```
[root@centoslina01 /]# java -version
openjdk version "1.8.0_65"
OpenJDK Runtime Environment (build 1.8.0_65-b17)
OpenJDK 64-Bit Server VM (build 25.65-b01, mixed mode)
```

10.3 安装Tomcat

安装Tomcat之前要保证JDK安装成功哦!

1、上传Tomcat压缩包到根目录的mysoft文件夹，之前已经完成

2、解压: tar -xzf apache-tomcat-8.5.11.tar.gz

```
[root@centoslina01 mysoft]# tar -xzf apache-tomcat-8.5.56.tar.gz
```

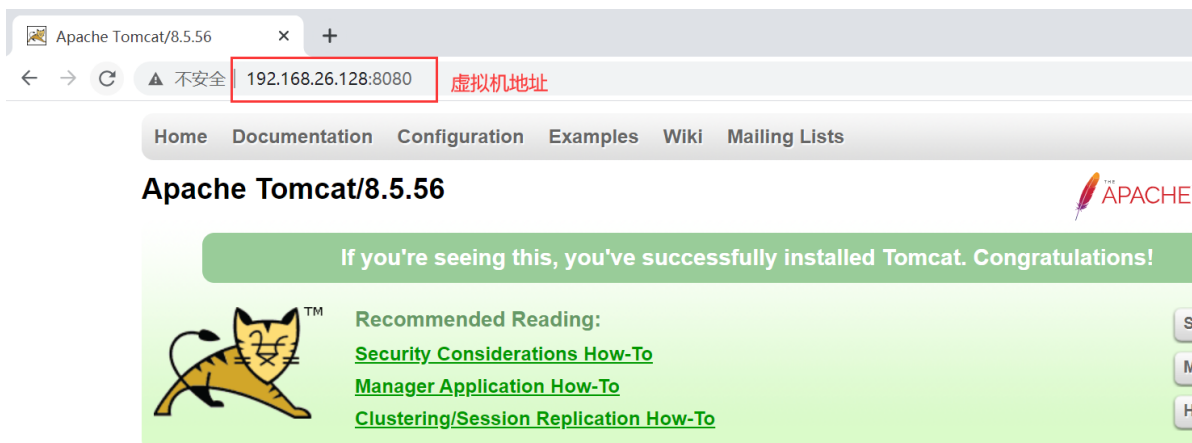
3、启动Tomcat

```
[root@centos1lna01 mysoft]# cd apache-tomcat-8.5.56/
[root@centos1lna01 apache-tomcat-8.5.56]# ll
total 128
drwxr-x---. 2 root root 4096 Mar 11 16:46 bin
-rw-r-----. 1 root root 19318 Jun 4 2020 BUILDING.txt
drwx-----. 2 root root 4096 Jun 4 2020 conf
-rw-r-----. 1 root root 5408 Jun 4 2020 CONTRIBUTING.md
drwxr-x---. 2 root root 4096 Mar 11 16:46 lib
-rw-r-----. 1 root root 57011 Jun 4 2020 LICENSE
drwxr-x---. 2 root root 6 Jun 4 2020 logs
-rw-r-----. 1 root root 1726 Jun 4 2020 NOTICE
-rw-r-----. 1 root root 3255 Jun 4 2020 README.md
-rw-r-----. 1 root root 7136 Jun 4 2020 RELEASE-NOTES
-rw-r-----. 1 root root 16262 Jun 4 2020 RUNNING.txt
drwxr-x---. 2 root root 29 Mar 11 16:46 temp
drwxr-x---. 7 root root 76 Jun 4 2020 webapps
drwxr-x---. 2 root root 6 Jun 4 2020 work
[root@centos1lna01 apache-tomcat-8.5.56]# cd bin
[root@centos1lna01 bin]# ll
total 868
-rw-r-----. 1 root root 35374 Jun 4 2020 bootstrap.jar
-rw-r-----. 1 root root 16608 Jun 4 2020 catalina.bat
-rwxr-x---. 1 root root 25272 Jun 4 2020 catalina.sh
-rw-r-----. 1 root root 1664 Jun 4 2020 catalina-tasks.xml
-rw-r-----. 1 root root 2123 Jun 4 2020 ciphers.bat
-rwxr-x---. 1 root root 1997 Jun 4 2020 ciphers.sh
-rw-r-----. 1 root root 25197 Jun 4 2020 commons-daemon.jar
-rw-r-----. 1 root root 206895 Jun 4 2020 commons-daemon-native.tar.gz
-rw-r-----. 1 root root 2040 Jun 4 2020 configtest.bat
-rwxr-x---. 1 root root 1922 Jun 4 2020 configtest.sh
-rwxr-x---. 1 root root 9127 Jun 4 2020 daemon.sh
-rw-r-----. 1 root root 2091 Jun 4 2020 digest.bat
-rwxr-x---. 1 root root 1965 Jun 4 2020 digest.sh
-rw-r-----. 1 root root 3460 Jun 4 2020 setclasspath.bat
-rwxr-x---. 1 root root 3708 Jun 4 2020 setclasspath.sh
-rw-r-----. 1 root root 2020 Jun 4 2020 shutdown.bat
-rwxr-x---. 1 root root 1902 Jun 4 2020 shutdown.sh
-rw-r-----. 1 root root 2022 Jun 4 2020 startup.bat
-rwxr-x---. 1 root root 1904 Jun 4 2020 startup.sh
-rw-r-----. 1 root root 51231 Jun 4 2020 tomcat-juli.jar
-rw-r-----. 1 root root 419572 Jun 4 2020 tomcat-native.tar.gz
-rw-r-----. 1 root root 4574 Jun 4 2020 tool-wrapper.bat
-rwxr-x---. 1 root root 5540 Jun 4 2020 tool-wrapper.sh
-rw-r-----. 1 root root 2026 Jun 4 2020 version.bat
-rwxr-x---. 1 root root 1908 Jun 4 2020 version.sh
[root@centos1lna01 bin]# sh startup.sh
Using CATALINA_BASE: /mysoft/apache-tomcat-8.5.56
Using CATALINA_HOME: /mysoft/apache-tomcat-8.5.56
Using CATALINA_TMPDIR: /mysoft/apache-tomcat-8.5.56/temp
Using JRE_HOME: /mysoft/jdk1.8.0_11
Using CLASSPATH: /mysoft/apache-tomcat-8.5.56/bin/bootstrap.jar:/mysoft/apache-tomcat-8.5.56/bin/tomcat-juli.jar
Tomcat started.
[root@centos1lna01 bin]#
```

4. 关闭防火墙之后测试访问

```
[root@centos1lna01 bin]# systemctl stop firewalld
[root@centos1lna01 bin]# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: enabled)
   Active: inactive (dead) since Thu 2021-03-11 16:50:09 CST; 14s ago
   Main PID: 961 (code=exited, status=0/SUCCESS)

Mar 10 15:08:59 centos1lna01 systemd[1]: Starting firewalld - dynamic firewall daemon...
Mar 10 15:09:02 centos1lna01 systemd[1]: Started firewalld - dynamic firewall daemon.
Mar 11 16:50:07 centos1lna01 systemd[1]: Stopping firewalld - dynamic firewall daemon...
Mar 11 16:50:09 centos1lna01 systemd[1]: Stopped firewalld - dynamic firewall daemon.
```



5. 关闭Tomcat

```
[root@centos1lna01 bin]# sh shutdown.sh
Using CATALINA_BASE: /mysoft/apache-tomcat-8.5.56
Using CATALINA_HOME: /mysoft/apache-tomcat-8.5.56
Using CATALINA_TMPDIR: /mysoft/apache-tomcat-8.5.56/temp
Using JRE_HOME: /mysoft/jdk1.8.0_11
Using CLASSPATH: /mysoft/apache-tomcat-8.5.56/bin/bootstrap.jar:/mysoft/apache-tomcat-8.5.56/bin/tomcat-juli.jar
[root@centos1lna01 bin]#
```

10.4 安装MySQL

YUM (Yellow dog Updater, Modified) 是一个在Fedora和RedHat以及CentOS中的Shell前端软件包管理器。它基于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,无须繁琐地一次次下载、安装。

步骤1: 检测系统是否自带安装mysql [root@centoslina01 ~]# yum list installed | grep mysql

```
[root@centoslina01 ~]# yum list installed | grep mysql
```

步骤2: 删除系统自带的mysql及其依赖 (防止后面安装发生冲突) [root@centoslina01 ~]# yum -y remove mysql-libs.x86_64

```
[root@centoslina01 ~]# yum -y remove mysql-libs.x86_64
Loaded plugins: fastestmirror, langpacks
No Match for argument: mysql-libs.x86_64
No Packages marked for removal
[root@centoslina01 ~]#
```

步骤3: 给CentOS添加rpm源, 并且选择较新的源,RPM软件包管理器缩写 [root@centoslina01 ~]# wget --no-check-certificate dev.mysql.com/get/mysql-community-release-el6-5.noarch.rpm

```
[root@centoslina01 ~]# wget --no-check-certificate dev.mysql.com/get/mysql-community-release-el6-5.noarch.rpm
--2021-03-11 17:01:22-- http://dev.mysql.com/get/mysql-community-release-el6-5.noarch.rpm
Resolving dev.mysql.com (dev.mysql.com)... 137.254.60.11
Connecting to dev.mysql.com (dev.mysql.com)|137.254.60.11|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://dev.mysql.com/get/mysql-community-release-el6-5.noarch.rpm [following]
--2021-03-11 17:01:23-- https://dev.mysql.com/get/mysql-community-release-el6-5.noarch.rpm
Connecting to dev.mysql.com (dev.mysql.com)|137.254.60.11|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://repo.mysql.com/mysql-community-release-el6-5.noarch.rpm [following]
--2021-03-11 17:01:24-- https://repo.mysql.com/mysql-community-release-el6-5.noarch.rpm
Resolving repo.mysql.com (repo.mysql.com)... 184.86.92.235
Connecting to repo.mysql.com (repo.mysql.com)|184.86.92.235|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5824 (5.7K) [application/x-redhat-package-manager]
Saving to: 'mysql-community-release-el6-5.noarch.rpm'

100%[=====] 5,824 --.-K/s in 0s

2021-03-11 17:01:24 (1.58 GB/s) - 'mysql-community-release-el6-5.noarch.rpm' saved [5824/5824]

[root@centoslina01 ~]# ^C
[root@centoslina01 ~]#
```

步骤4: 安装第一步下载的rpm文件 [root@centoslina01 ~]# yum install mysql-community-release-el6-5.noarch.rpm 此处需要输入"y"

```
[root@centoslina01 ~]# yum install mysql-community-release-el6-5.noarch.rpm
Loaded plugins: fastestmirror, langpacks
Examining mysql-community-release-el6-5.noarch.rpm: mysql-community-release-el6-5.noarch
Marking mysql-community-release-el6-5.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package mysql-community-release.noarch 0:el6-5 will be installed
--> Finished Dependency Resolution

base/7/x86_64 | 3.6 kB 00:00:00
extras/7/x86_64 | 2.9 kB 00:00:00
updates/7/x86_64 | 2.9 kB 00:00:00

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
mysql-community-release noarch el6-5 /mysql-community-release-el6-5.noarch 4.3 k
=====
Transaction Summary
=====
Install 1 Package

Total size: 4.3 k
Installed size: 4.3 k
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : mysql-community-release-el6-5.noarch 1/1
Verifying : mysql-community-release-el6-5.noarch 1/1

Installed:
mysql-community-release.noarch 0:el6-5

Complete!
```

[root@centoslina01 ~]# yum repolist enabled | grep mysql //列出你所有的yum repo文件

```
[root@centoslina01 ~]# yum repolist enabled | grep mysql
mysql-connectors-community/x86_64 MySQL Connectors Community 153
mysql-tools-community/x86_64 MySQL Tools Community 99
mysql56-community/x86_64 MySQL 5.6 Community Server 644
[root@centoslina01 ~]#
```

步骤5: 使用yum安装mysql [root@centoslina01 ~]# yum install mysql-community-server 此处输入3次"y"


```

Verifying : mysql-community-common-5.6.51-2.el6.x86_64 4/12
Verifying : perl-IO-Compress-2.061-2.el7.noarch 5/12
Verifying : 1:perl-Compress-Raw-Zlib-2.061-4.el7.x86_64 6/12
Verifying : mysql-community-libs-5.6.51-2.el6.x86_64 7/12
Verifying : mysql-community-client-5.6.51-2.el6.x86_64 8/12
Verifying : perl-DBI-1.627-4.el7.x86_64 9/12
Verifying : perl-Compress-Raw-Bzip2-2.061-3.el7.x86_64 10/12
Verifying : perl-PIRPC-0.2020-14.el7.noarch 11/12
Verifying : 1:mariadb-libs-5.5.44-2.el7.centos.x86_64 12/12

Installed:
mysql-community-libs.x86_64 0:5.6.51-2.el6 mysql-community-server.x86_64 0:5.6.51-2.el6

Dependency Installed:
mysql-community-client.x86_64 0:5.6.51-2.el6 mysql-community-common.x86_64 0:5.6.51-2.el6
perl-Compress-Raw-Bzip2.x86_64 0:2.061-3.el7 perl-Compress-Raw-Zlib.x86_64 1:2.061-4.el7
perl-DBI.x86_64 0:1.627-4.el7 perl-Data-Dumper.x86_64 0:2.145-3.el7
perl-IO-Compress.noarch 0:2.061-2.el7 perl-Net-Daemon.noarch 0:0.48-5.el7
perl-PIRPC.noarch 0:0.2020-14.el7

Replaced:
mariadb-libs.x86_64 1:5.5.44-2.el7.centos

Complete!
[root@centoslina01 ~]# █

```

安装完毕

步骤6: 启动mysql服务 [root@centoslina01 ~]# service mysqld start

```

[root@centoslina01 ~]# service mysqld start
Starting mysqld (via systemctl): [ OK ]
[root@centoslina01 ~]#

```

步骤7: 查看mysql是否自启动,并且设置开启自启动 [root@centoslina01 ~]# chkconfig --list | grep mysqld 0 1 2 3 4 5 6 代表centOS启动状态

```

[root@centoslina01 ~]# chkconfig --list | grep mysqld

Note: This output shows SysV services only and does not include native
systemd services. SysV configuration data might be overridden by native
systemd configuration.

If you want to list systemd services use 'systemctl list-unit-files'.
To see services enabled on particular target use
'systemctl list-dependencies [target]'.

mysqld 0:off 1:off 2:off 3:on 4:on 5:on 6:off

```

[root@centoslina01 ~]# chkconfig mysqld on

```

[root@centoslina01 ~]# chkconfig mysqld on

```

步骤8: 修改字符集为UTF-8 [root@centoslina01 ~]# vim /etc/my.cnf 在[mysqld]部分添加:
character-set-server=utf8 在文件末尾新增[client]段, 并在[client]段添加: default-character-set=utf8 esc退出编辑 :wq保存退出

```
✓ root@centoslina01:~ x
# For advice on how to change settings please see
# http://dev.mysql.com/doc/refman/5.6/en/server-configuration-defaults.html

[mysqld]
character-set-server=utf8
#
# Remove leading # and set to the amount of RAM for the most important data
# cache in MySQL. Start at 70% of total RAM for dedicated server, else 10%.
# innodb_buffer_pool_size = 128M
#
# Remove leading # to turn on a very important data integrity option: logging
# changes to the binary log between backups.
# log_bin
#
# Remove leading # to set options mainly useful for reporting servers.
# The server defaults are faster for transactions and fast SELECTs.
# Adjust sizes as needed, experiment to find the optimal values.
# join_buffer_size = 128M
# sort_buffer_size = 2M
# read_rnd_buffer_size = 2M
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock

# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0

# Recommended in standard MySQL setup
sql_mode=NO_ENGINE_SUBSTITUTION,STRICT_TRANS_TABLES

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid

[client]
default-character-set=utf8
```

修改完成后保存重启服务 [root@centoslina01 ~]# service mysqld restart

```
[root@centoslina01 ~]# service mysqld restart
Restarting mysqld (via systemctl): [ OK ]
[root@centoslina01 ~]#
```

步骤9: 修改默认配置 [root@centoslina01 ~]# mysql_secure_installation 按照提示进行配置, 这里默认root用户的密码空, 直接回车即可。此处输入内容:"y","y","n","y","y"

```
[root@centos1ina01 ~]# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current
password for the root user. If you've just installed MySQL, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MySQL
root user without the proper authorisation.

Set root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MySQL installation has an anonymous user, allowing anyone
to log into MySQL without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] n
... skipping.

By default, MySQL comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
- Dropping test database...
ERROR 1008 (HY000) at line 1: Can't drop database 'test'; database doesn't exist
... Failed! Not critical, keep moving...
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

All done! If you've completed all of the above steps, your MySQL
installation should now be secure.

Thanks for using MySQL!

Cleaning up...
[root@centos1ina01 ~]#
```

步骤10:授权远程登录

方案1:实现远程连接(授权法)- 将权限改为ALL PRIVILEGES [root@centos1ina01 ~]# mysql -uroot -p mysql> use mysql; Database changed mysql> flush privileges; //mysql 新设置用户或更改密码后需用flush privileges刷新MySQL的系统权限相关表, 否则会出现拒绝访问 mysql> select host,user,password from user; 这样机器就可以以用户名root密码root远程访问该机器上的MySQL.

```
[root@centoslna01 ~]# mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 5.6.51 MySQL Community Server (GPL)

Copyright (c) 2000, 2021, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use mysql
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)

mysql> select host,user,password from user;
+-----+-----+-----+
| host          | user  | password                                     |
+-----+-----+-----+
| localhost     | root  | *81F5E21E35407D884A6CD4A731AEBFB6AF209E1B |
| centoslna01   | root  | *81F5E21E35407D884A6CD4A731AEBFB6AF209E1B |
| 127.0.0.1     | root  | *81F5E21E35407D884A6CD4A731AEBFB6AF209E1B |
| ::1           | root  | *81F5E21E35407D884A6CD4A731AEBFB6AF209E1B |
+-----+-----+-----+
4 rows in set (0.00 sec)

mysql>
```

方案2:实现远程连接 (改表法) use mysql; update user set host = '%' where user = 'root' and host='localhost';

GRANT ALL PRIVILEGES ON . TO 'username'@'%' IDENTIFIED BY 'password' WITH GRANT OPTION;
/

flush privileges;

```
mysql> use mysql;
Database changed
mysql> update user set host = '%' where user = 'root' and host='localhost';
Query OK, 0 rows affected (0.01 sec)
Rows matched: 0  Changed: 0  Warnings: 0

mysql> update user set host = '%' where user = 'root' and host='localhost';
```

这样在远端就可以通过root用户访问Mysql

远程连接出问题:

看看防火墙是否关闭;

是否是授权问题: GRANT ALL PRIVILEGES ON . TO 'username'@'%' IDENTIFIED BY 'password' WITH GRANT OPTION;

flush privileges;