

远程管理常用命令

目标

- 关机/重启
 - shutdown
- 查看或配置网卡信息
 - ifconfig
 - ping
- 远程登录和复制文件
 - ssh
 - scp

01. 关机/重启

序号	命令	对应英文	作用
01	shutdown 选项 时间	shutdown	关机 / 重新启动

1.1 shutdown

- shutdown 命令可以 安全 关闭 或者 重新启动系统

选项	含义
-r	重新启动

提示：

- 不指定选项和参数，默认表示 1 分钟之后 关闭电脑
- 远程维护服务器时，最好不要关闭系统，而应该重新启动系统
- 常用命令示例

```
# 重新启动操作系统，其中 now 表示现在
$ shutdown -r now

# 立刻关机，其中 now 表示现在
$ shutdown now

# 系统在今天的 20:25 会关机
$ shutdown 20:25

# 系统再过十分钟后自动关机
$ shutdown +10
```

```
# 取消之前指定的关机计划
$ shutdown -c
```

02. 查看或配置网卡信息

序号	命令	对应英文	作用
01	ifconfig	configure a network interface	查看/配置计算机当前的网卡配置信息
02	ping ip地址	ping	检测到目标 ip地址 的连接是否正常

2.1 网卡 和 IP 地址

网卡

- 网卡是一个专门负责网络通讯的硬件设备
- **IP 地址**是设置在网卡上的地址信息

我们可以把 **电脑** 比作 **电话**，**网卡** 相当于 **SIM 卡**，**IP 地址** 相当于 **电话号码**

IP 地址

- **每台联网的电脑上都有 IP 地址，是保证电脑之间正常通讯的重要设置**

注意：每台电脑的 IP 地址不能相同，否则会出现 IP 地址冲突，并且没有办法正常通讯

提示：有关 **IP 地址**的详细内容，在就业班会详细讲解！

2.2 ifconfig

- **ifconfig** 可以查看 / 配置计算机当前的网卡配置信息

```
# 查看网卡配置信息
$ ifconfig

# 查看网卡对应的 IP 地址
$ ifconfig | grep inet
```

提示：一台计算机中有可能会有一个 **物理网卡** 和 **多个虚拟网卡**，在 Linux 中物理网卡的名字通常以 **ensxx** 表示

- **127.0.0.1** 被称为 **本地回环/环回地址**，一般用来测试本机网卡是否正常

2.3 ping

```
# 检测到目标主机是否连接正常
$ ping IP地址

# 检测本地网卡工作正常
$ ping 127.0.0.1
```

- `ping` 一般用于检测当前计算机到目标计算机之间的网络 **是否通畅**，数值越大，速度越慢

- `ping` 的工作原理与潜水艇的声纳相似，`ping` 这个命令就是取自 **声纳的声音**
- 网络管理员之间也常将 `ping` 用作动词 —— **ping 一下计算机X，看他是否开着**

原理：网络上的机器都有 **唯一确定的 IP 地址**，我们给**目标 IP 地址**发送一个数据包，对方就要返回一个数据包，根据返回的数据包以及时间，我们可以确定目标主机的存在

提示：在 Linux 中，想要终止一个终端程序的执行，绝大多数都可以使用 `CTRL + C`

03. 远程登录和复制文件

序号	命令	对应英文	作用
01	ssh 用户名@ip	secure shell	关机 / 重新启动
02	scp 用户名@ip:文件名或路径 用户名@ip:文件名或路径	secure copy	远程复制文件

3.1 ssh 基础 (重点)

在 Linux 中 SSH 是 **非常常用** 的工具，通过 **SSH 客户端** 我们可以连接到运行了 **SSH 服务器** 的远程机器上

数据传输是**加密的**，可以**防止信息泄漏**
数据传输是**压缩的**，可以**提高传输速度**



- **SSH 客户端**是一种使用 `Secure Shell (SSH)` 协议连接到远程计算机的软件程序
- `SSH` 是目前较可靠，**专为远程登录会话和其他网络服务** 提供安全性的协议
 - 利用 `SSH 协议` 可以有效**防止远程管理过程中的信息泄露**
 - 通过 `SSH 协议` 可以对所有传输的数据进行加密，也能够防止 DNS 欺骗和 IP 欺骗
- `SSH` 的另一项优点是传输的数据可以是经过压缩的，所以可以加快传输的速度

1) 域名 和 端口号

域名

- 由一串 **用点分隔** 的名字组成，例如：`www.itcast.cn`
- 是 **IP 地址** 的别名，方便用户记忆

端口号

- **IP 地址**：通过 **IP 地址** 找到网络上的 **计算机**
- **端口号**：通过 **端口号** 可以找到 **计算机上运行的应用程序**

- **SSH 服务器** 的默认端口号是 `22`，如果是默认端口号，在连接的时候，可以省略
- 常见服务端口号列表：

序号	服务	端口号
01	SSH 服务器	22
02	Web 服务器	80
03	HTTPS	443
04	FTP 服务器	21

提示：有关 **端口号** 的详细内容，在就业班会详细讲解！

2) SSH 客户端的简单使用

```
ssh [-p port] user@remote
```

- `user` 是在远程机器上的用户名，如果不指定的话默认为当前用户
- `remote` 是远程机器的地址，可以是 **IP / 域名**，或者是 **后面会提到的别名**
- `port` 是 **SSH Server 监听的端口**，如果不指定，就为默认值 `22`

提示：

- 使用 `exit` 退出当前用户的登录

注意：

- `ssh` 这个终端命令只能在 `Linux` 或者 `UNIX` 系统下使用
- 如果在 `Windows` 系统中，可以安装 `Putty` 或者 `xshell` 客户端软件即可

提示：

- 在工作中，SSH 服务器的端口号很有可能**不是 22**，如果遇到这种情况就需要使用 `-p` 选项，指定正确的端口号，否则无法正常连接到服务器

3) Windows 下 SSH 客户端的安装

- `Putty` <http://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>
- `xshell` <http://xshellcn.com>

建议从官方网站下载正式的安装程序

3.2 scp（掌握）

- `scp` 就是 `secure copy`，是一个在 `Linux` 下用来进行 **远程拷贝文件** 的命令
- 它的**地址格式与 ssh 基本相同**，**需要注意的是**，在指定端口时用的是大写的 `-P` 而不是小写的



```
# 把本地当前目录下的 01.py 文件 复制到 远程 家目录下的 Desktop/01.py
# 注意: `:` 后面的路径如果不是绝对路径, 则以用户的家目录作为参照路径
scp -P port 01.py user@remote:Desktop/01.py

# 把远程 家目录下的 Desktop/01.py 文件 复制到 本地当前目录下的 01.py
scp -P port user@remote:Desktop/01.py 01.py

# 加上 -r 选项可以传送文件夹
# 把当前目录下的 demo 文件夹 复制到 远程 家目录下的 Desktop
scp -r demo user@remote:Desktop

# 把远程 家目录下的 Desktop 复制到 当前目录下的 demo 文件夹
scp -r user@remote:Desktop demo
```

选项	含义
-r	若给出的源文件是目录文件, 则 scp 将递归复制该目录下的所有子目录和文件, 目标文件必须为一个目录名
-P	若远程 SSH 服务器的端口不是 22, 需要使用大写字母 -P 选项指定端口

注意:

- `scp` 这个终端命令只能在 `Linux` 或者 `UNIX` 系统下使用
- 如果在 `Windows` 系统中, 可以安装 `PuTTY`, 使用 `pscp` 命令行工具或者安装 `FileZilla` 使用 `FTP` 进行文件传输

FileZilla

- 官方网站: <https://www.filezilla.cn/download/client>
- `FileZilla` 在传输文件时, 使用的是 `FTP` 服务 而不是 `SSH` 服务, 因此端口号应该设置为 `21`

3.3 SSH 高级 (知道)

- 免密码登录
- 配置别名

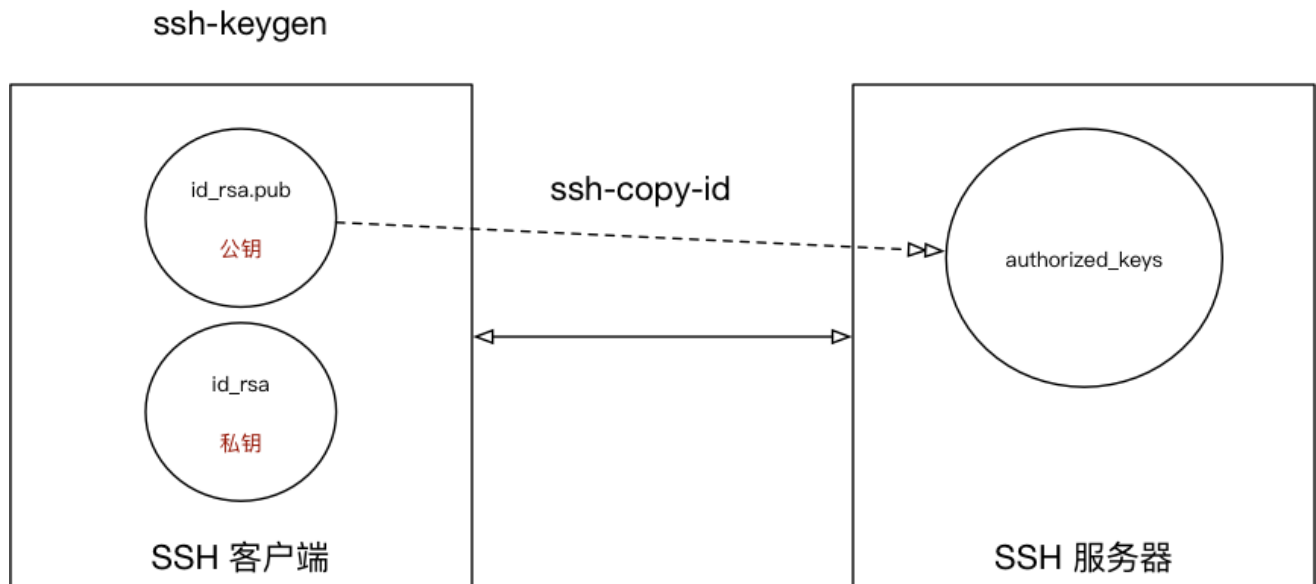
提示: 有关 SSH 配置信息都保存在用户家目录下的 `.ssh` 目录下

1) 免密码登录

步骤

- 配置公钥
 - 执行 `ssh-keygen` 即可生成 SSH 钥匙，一路回车即可
- 上传公钥到服务器
 - 执行 `ssh-copy-id -p port user@remote`，可以让远程服务器记住我们的公钥

示意图



本地 使用 私钥 对数据进行加密 / 解密
服务器 使用 公钥 对数据进行加密 / 解密

非对称加密算法

- 使用 公钥 加密的数据，需要使用 私钥 解密
- 使用 私钥 加密的数据，需要使用 公钥 解密

2) 配置别名

每次都输入 `ssh -p port user@remote`，时间久了会觉得很麻烦，特别是当 `user`，`remote` 和 `port` 都得输入，而且还不好记忆

而 配置别名 可以让我们进一步偷懒，譬如用：`ssh mac` 来替代上面这么一长串，那么就在 `~/.ssh/config` 里面追加以下内容：

```
Host mac
  HostName ip地址
  User itheima
  Port 22
```

保存之后，即可用 `ssh mac` 实现远程登录了，`scp` 同样可以使用