



Linux 基础

刘浩

2023.9.16

目录

- linux 介绍
- linux 基本概念
- linux 远程登录
- linux 常用命令详解

操作系统

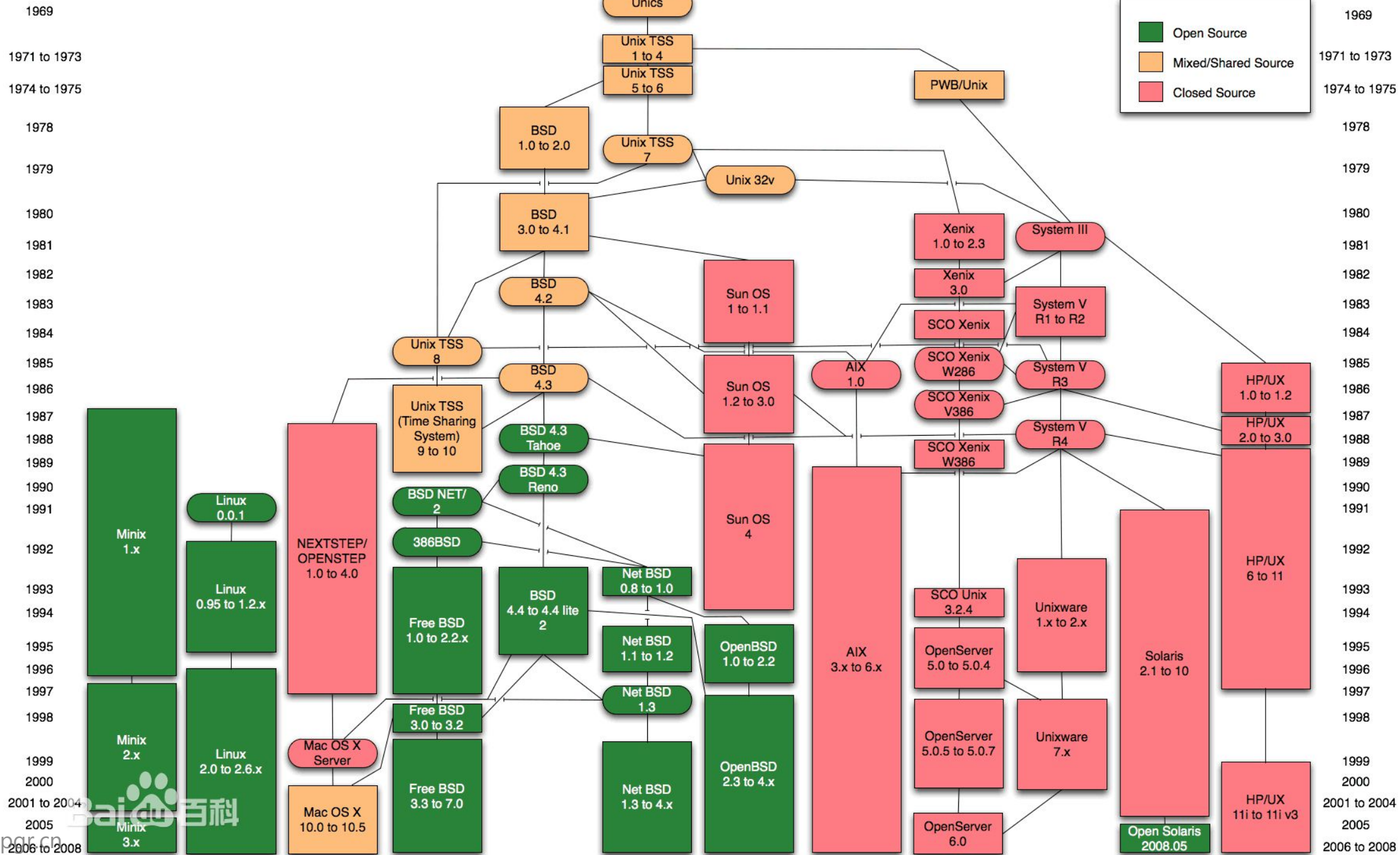
计算机是一台机器，它按照用户的要求接收信息、存储数据、处理数据，然后再将处理结果输出（文字、图片、音频、视频等）。计算机由硬件和软件组成：

- 硬件是计算机赖以工作的实体，包括显示器、键盘、鼠标、硬盘、CPU、主板等；
- 软件会按照用户的要求协调整台计算机的工作，比如 Windows、Linux、Mac OS、Android 等操作系统，以及 Office、QQ、微信等应用程序。

操作系统（Operating System，OS）是软件的一部分，它是硬件基础上的第一层软件，是硬件和其它软件沟通的桥梁。

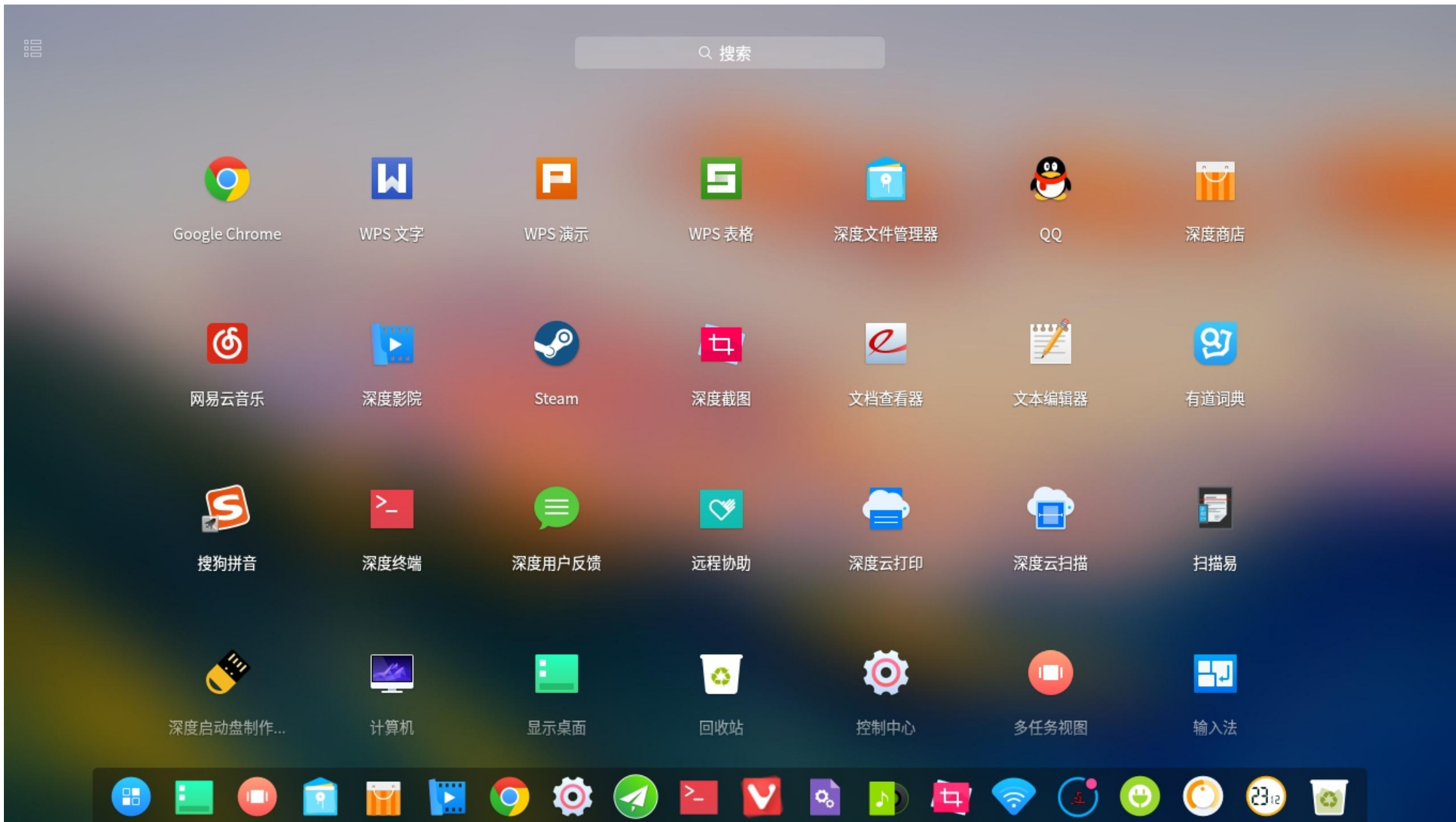
Linux-Unix

- 1970年，Unix 诞生于贝尔实验室，开放源码，此后诞生了AIX、Solaris、HP-UX、BSD等Unix系统
- 1979年，从Unix V7开始，贝尔实验室严格了版本，禁止其源码用于教学
- 1987年，Tanenbaum 发布了其从头开发、兼容Unix V7的操作系统 Minix(mini-UNIX)用于教学
- 1991年，Minix功能有限，Linus借鉴Minix开发了Linux(内核)
- 1983年，Stallman发起GNU(GNU is not Unix)计划，目标是创建一套完全自由的操作系统，并陆续开发了Emacs、GCC等各种工具，但缺乏操作系统内核



Linux

- GNU计划和Linux内核结合, 形成现代Linux生态环境
- 从2017年11月份开始, 世界Top500超级计算机全部使用Linux系统
- 根据不同用途, 整合Linux内核和各类软件形成不同的操作系统, 称之为Linux发行版
- 常用发行版本:Ubuntu, Debian, RHEL, CentOS, Fedora, SuSE Linux, Gentoo, Arch Linux, Kali Linux, Deepin, LFS
- 生信Linux版本推荐:服务器CentOS, 个人PC Ubuntu Desktop
- CentOS不再发行, 使用Rocky Linux替换

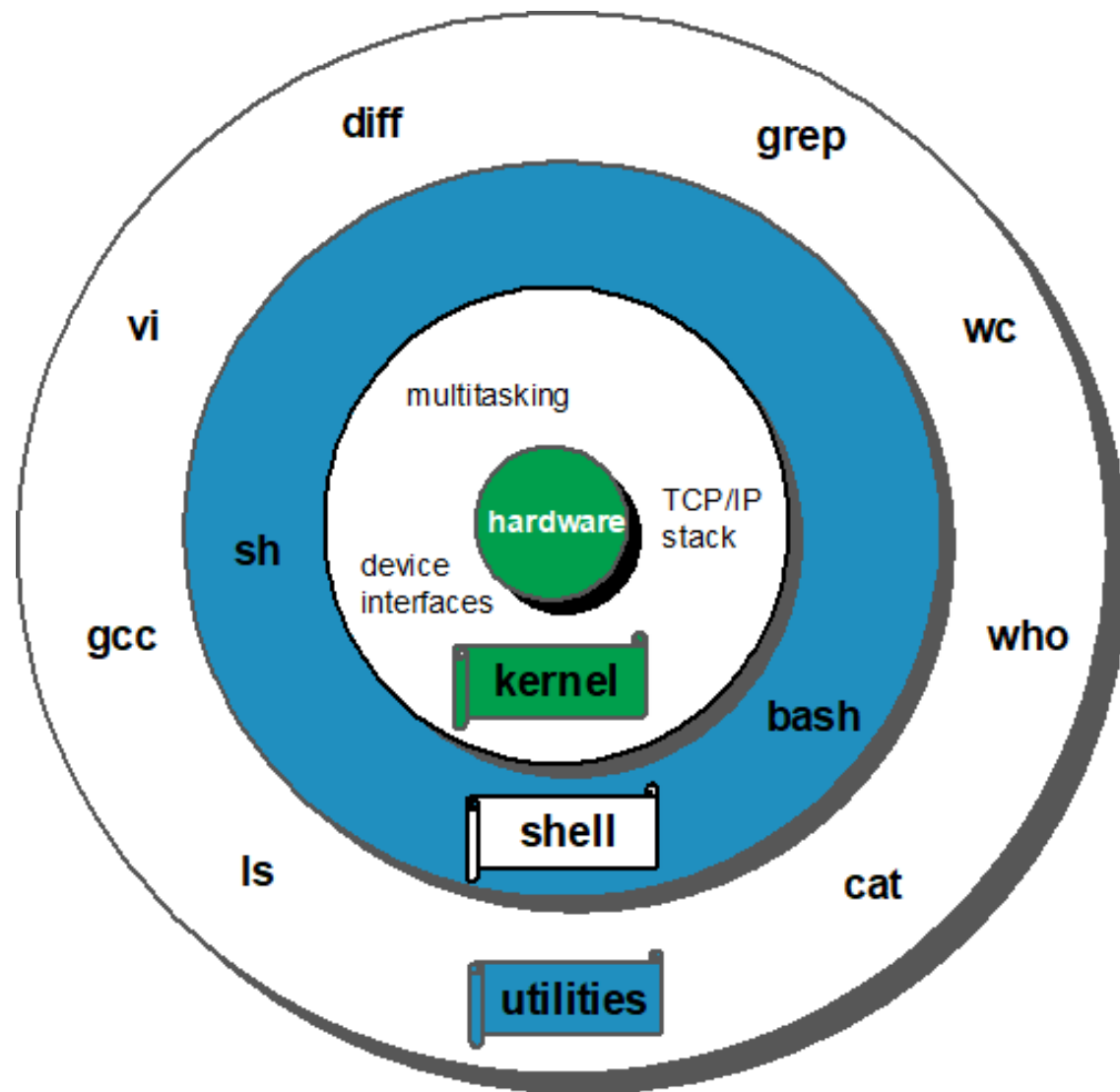


Linux 特点

- 免费
 - 一切皆文件
 - 由目的单一的小程序组成，组合小程序完成复杂任务
 - 多任务、多用户系统
 - 大量的可用软件
 - 良好的可移植性及灵活性
 - 优秀的稳定性和安全性

Linux系统结构

- Kernel(内核)
 - 系统启动时将内核装入内存
 - 管理系统各种资源
- Shell(应用接口)
 - 保护操作系统
 - 用户界面, 提供用户与内核交互处理接口, 各种命令运行的地方
 - bash, ash, pdksh, tcsh, ksh, sh, csh, zsh....
- Utility(应用程序)
 - 提供各种管理工具, 应用程序



Linux基本概念-文件

- linux哲学: 一切皆文件
- 对所有文件（目录、块设备、套接字、进程等）操作，读写都可用 `fopen()/fclose()/fwrite()/fread()` 等函数进行处理。屏蔽了硬件的区别，所有设备都抽象成文件，提供统一的接口给用户。
- 文件具有多种属性，如类型、所属用户、用户组、权限、修改时间、大小等。
- 所有文件的起始为 `/` (根目录)，采用树形结构存放。各类文件存放目录遵循FHS (Filesystem Hierarchy Standard)。
- 文件的位置称之为路径，如 `/var/log/messages` 。
 - 从 `/` 写起的为 绝对路径，如 `/var/log/messages`
 - 从当前位置写起的为 相对路径，如 `log/messages`

/—	— bin	# binaries(二进制文件), 存放着最常用的程序和指令
	— boot	# 系统启动相关文件, 如内核, initrd, 以及grub (BootLoader)
	— dev	# device, 设备文件 – 体现了Linux的“一切皆文件”思想
	— etc	# 配置文件。大多数为纯文本文件
	— home	# 用户的家目录
	— lib	# library, 公共库文件 (不能单独执行, 只能被调用)
	— media	# 挂载点目录, 挂载移动设备 (如U盘)
	— mnt	# 挂载额外的临时文件 (如第二块硬盘)
	— opt	# optional, 存放额外安装软件
	— proc	# processes, 伪文件系统, 内存映射文件, 系统启动后才出现文件, 关机就空
	— root	# 管理员的家目录
	— sbin	# superuser binaries, 存放系统管理所需要的命令
	— tmp	# temporary, 临时文件, 所有用户都可以操作, 但只能删自己的, 不能删别人的
	— usr	# unix shared resources, 存放只能读的命令和其他文件
	— var	# variable, 存放运行时需要改变数据的文件, 如日志等

Linux基本概念-用户和用户组

- 多个用户可以在同一时间内登录同一个Linux系统执行各自不同的任务，互不影响。
- 每个用户有唯一的用户名、用户id(uid)，并通过各自的密码登录使用。
- root用户(超级用户)拥有系统最高权限。
- 用户组是具有相同特征用户的逻辑集合。将用户分组是Linux 系统中对用户进行管理
及控制访问权限的一种手段，通过定义用户组，简化了管理工作。
- 每个用户组有唯一的名称和id(gid)。

用户和用户组的对应关系有：

- 一对一：即一个用户可以存在一个组中，也可以是组中的唯一成员
- 一对多：即一个用户可以存在多个用户组中。那么此用户具有多个组的共同权限
 - 每个用户拥有唯一的主用户组(primary group)
 - 每个用户可以拥有多个附加用户组(supplementary groups)
- 多对一：多个用户可以存在一个组中，这些用户具有和组相同的权限
- 多对多：多个用户可以存在多个组中。其实就是上面三个对应关系的扩展

Linux基本概念-权限

- 在Linux中，多用户之间通过权限实现资源的隔离，以保障整个系统的安全
- 权限的主体为用户，对象为文件
- 拥有权限即拥有对文件进行操作的权利，基本权限：可读(r)、可写(w)、可执行(x)
- 用户对自己的文件有绝对的权限，本用户的文件默认禁止其它用户访问，可通过更改文件权限向其它用户共享特定文件，使其它用户可以访问和修改该文件
- root用户对所有文件拥有访问和修改的权限

Linux基本概念-命令

- 命令(command)是在终端上运行的程序或实用程序
- **命令提示符(prompt)**: 登陆成功后显示的符号, 如 `[username@localhost ~]$` , 其中 `#` :root用户, `$` :普通用户
- **命令格式**: `命令 -选项 参数`
 - 命令: shell传递给内核执行
 - 选项(options): 命令所含的选项, 通过选项控制命令的行为 (可选, 非必须)
 - 短选项: -character, 多个选项可以组合, 可以写 `ls -l -a` 或者 `ls -la`
 - 长选项: --word, 不能组合, 要分开
 - 参数(arguments), 多数用于指向命令的指向目标 (可选, 非必须) , 多个参数要用空格隔开, 参数顺序决定命令的作用顺序

- **命令类型：**
 - 内置命令 (shell builtin 内置)
 - 外部命令：某个路径下有一个与命令名称相应的可执行文件
- **命令补全：**使用 `tab`键 补全功能，可提升输入效率，减少输入错误
- **命令历史：**shell会记录执行过的命令，退出后写入 `~/.bash_history`。使用 `history` 命令查看使用过的命令
 - `↑` 和 `↓` 可以上下翻看使用过的命令
 - `ctrl + r` 输入字符，可以快速定位到使用过的命令
- **命令帮助：**
 - `man 命令` 打开命令手册信息，包含命令的详细用法及各选项解释；
 - `命令 -h` 显示命令用法

- **快捷键:**

```
ctrl+a: 跳到命令行首  
ctrl+e: 跳到命令行尾  
ctrl+u: 删除光标至命令行首的内容, 命令行下清除输错的密码很管用  
ctrl+k: 删除光标至命令行尾的内容  
ctrl+l: 清屏
```

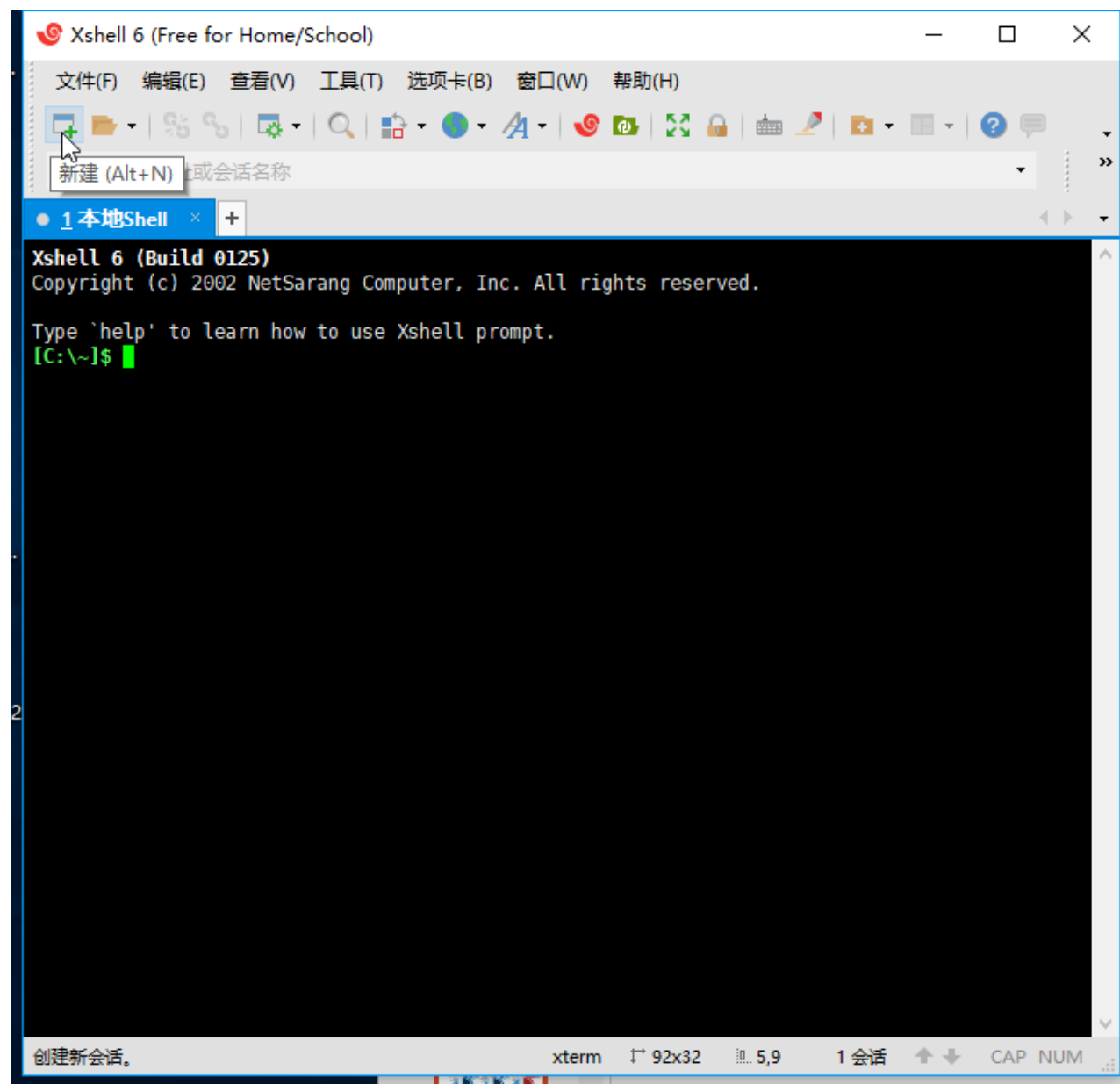
- **命令路径:**

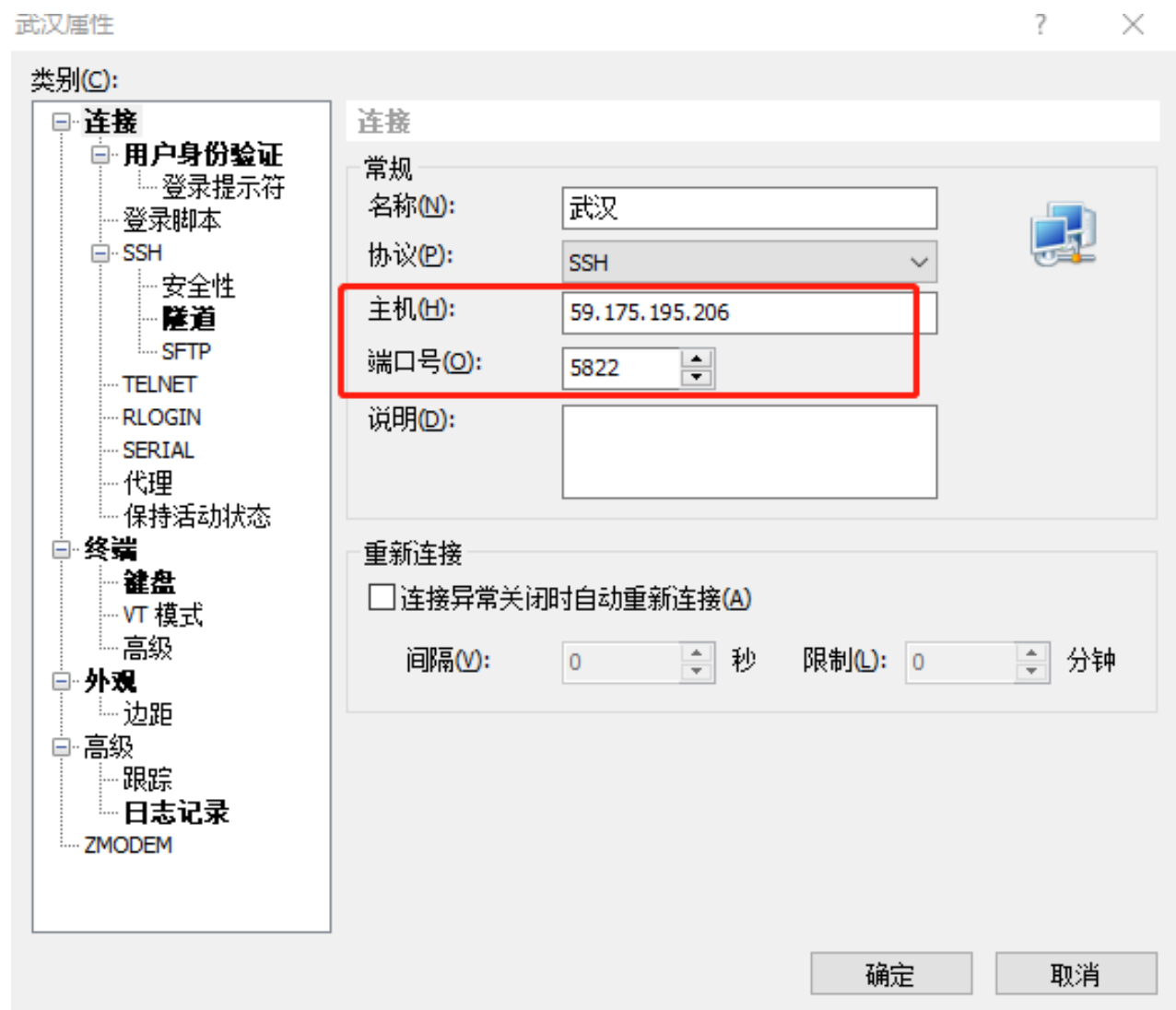
命令存放在 `$PATH` 的路径中, 命令执行时会在这些路径中从前往后逐一搜索, 并执行第一个搜索的命令。可以使用绝对路径准确执行命令。

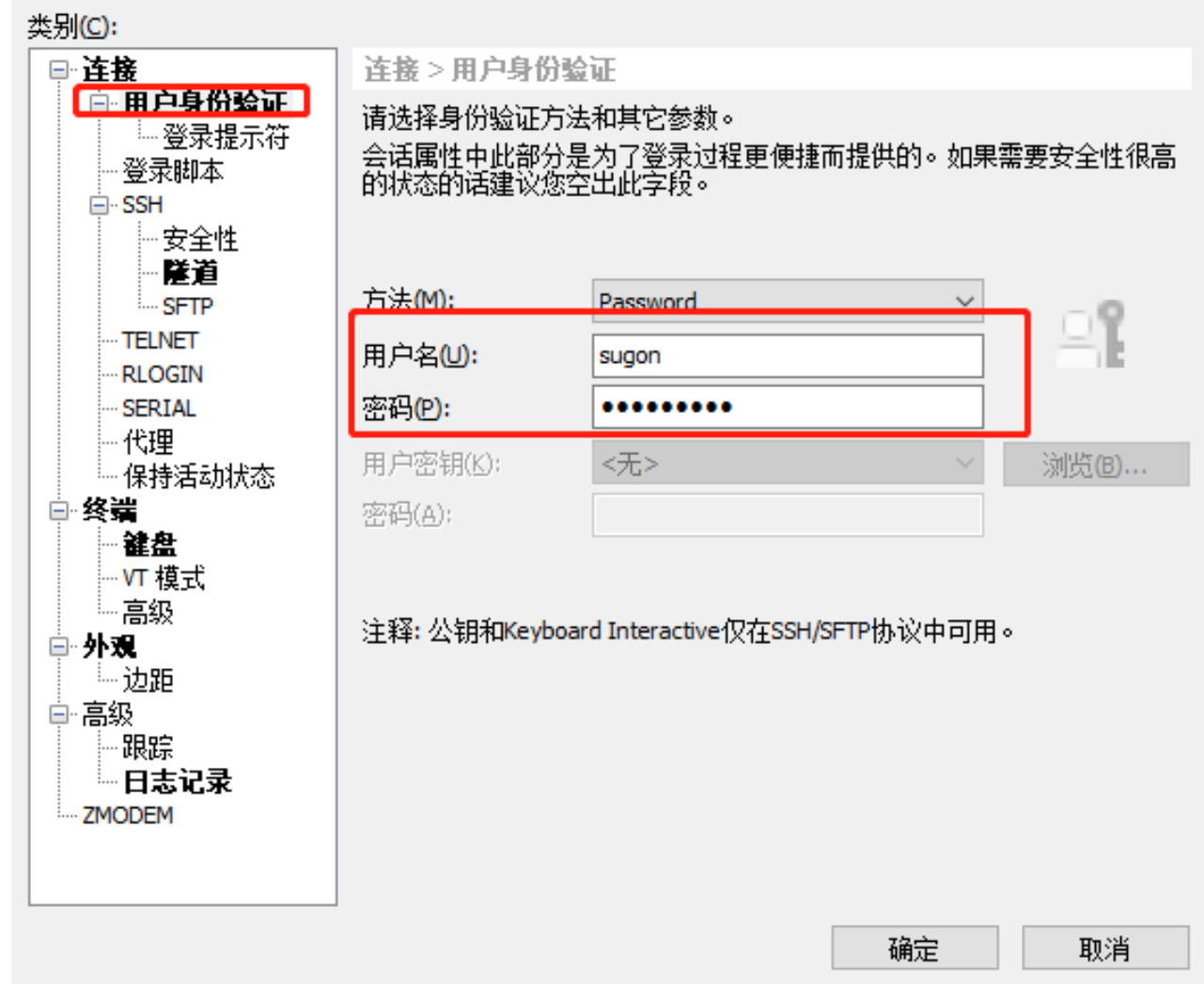
```
$ echo $PATH  
/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin
```

远程登录Linux系统

- 必须信息:
 - ip地址, 如 192.168.11.22
 - 端口号(port), 默认22
 - 用户名和密码/秘钥/动态口令
- Linux/mac: `$ ssh -p port username@ip`
- Windows:
 - Xshell, Putty, MobaXterm, Xshell, tabby

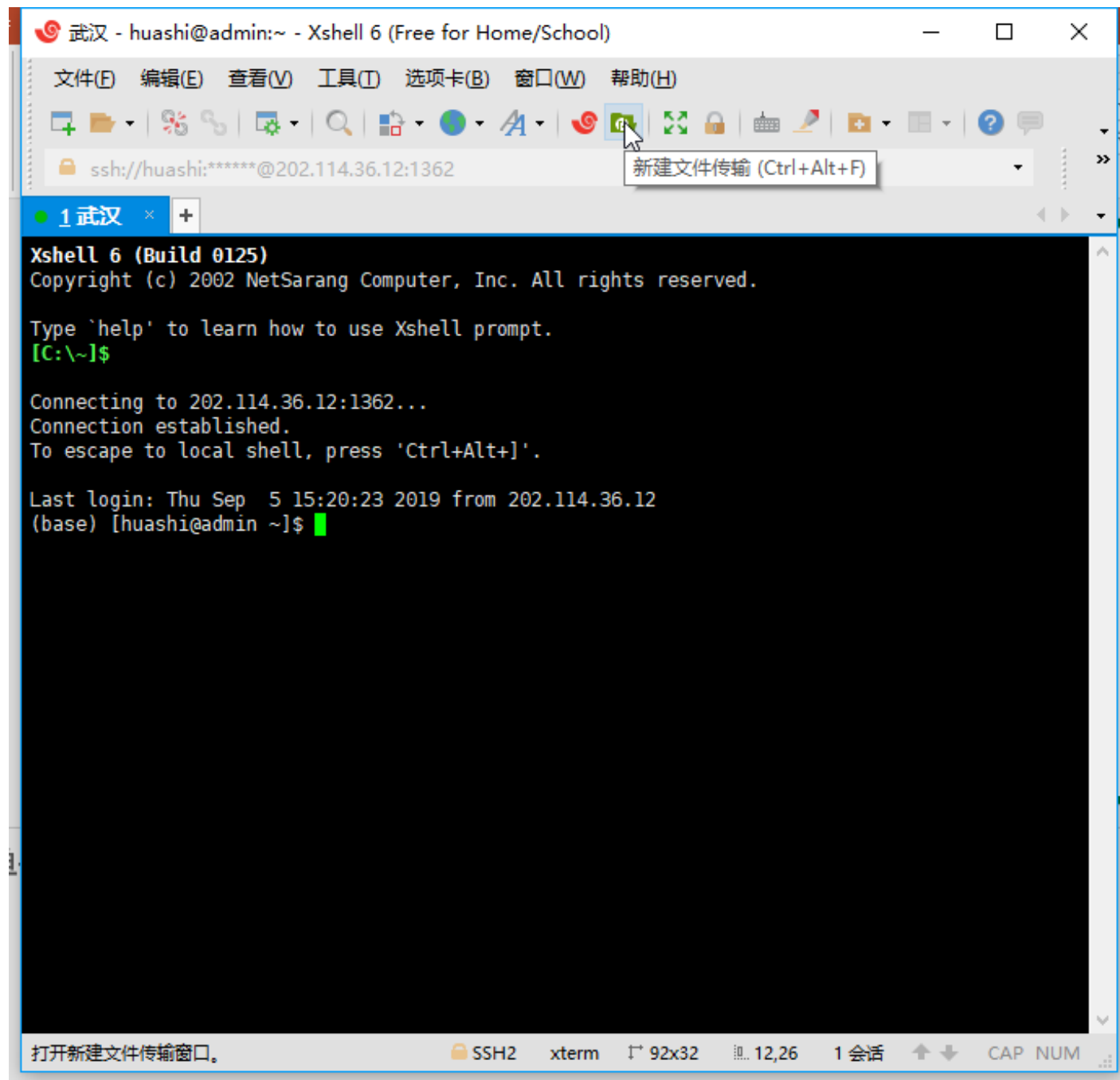


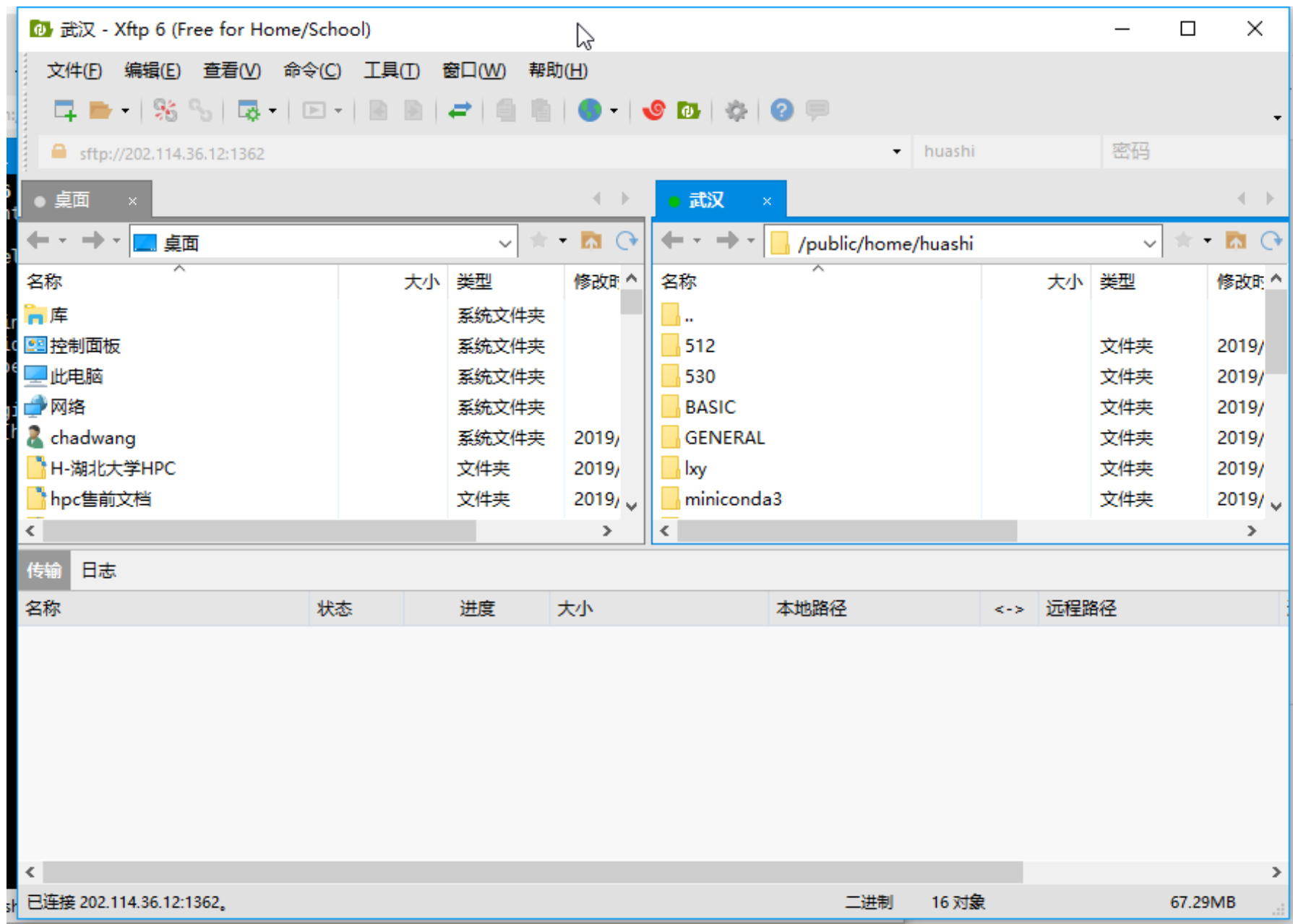




远程文件传输

- Linux:
 - `$ scp -p port 本地文件 IP:/远程目录/`
 - `$ rsync -avP 本地文件 IP:/远程目录/`
- Windows:
 - winscp, Xftp, filezilla





学习环境

- `git bash`, 在Windows上模拟bash环境, 允许用户在Windows上使用大多数Linux命令
- `cygwin`, 在windows上运行的linux模拟环境, 可以使用linux命令、编译安装软件
- `WSL`, Windows子系统(WSL)可让开发人员按原样运行 GNU/Linux 环境
- 虚拟机, 使用virtualbox或VMware Workstation Player安装linux虚拟机, 可以体验完整的linux从安装到完整系统的使用
- 云服务器, 阿里云, 腾讯云, 华为云等

<http://hpc.ncpgr.cn/linux/093-linux-exercise-env/>