计算机网络

Computer Network

15

文件传输

理论课程

厦門大學

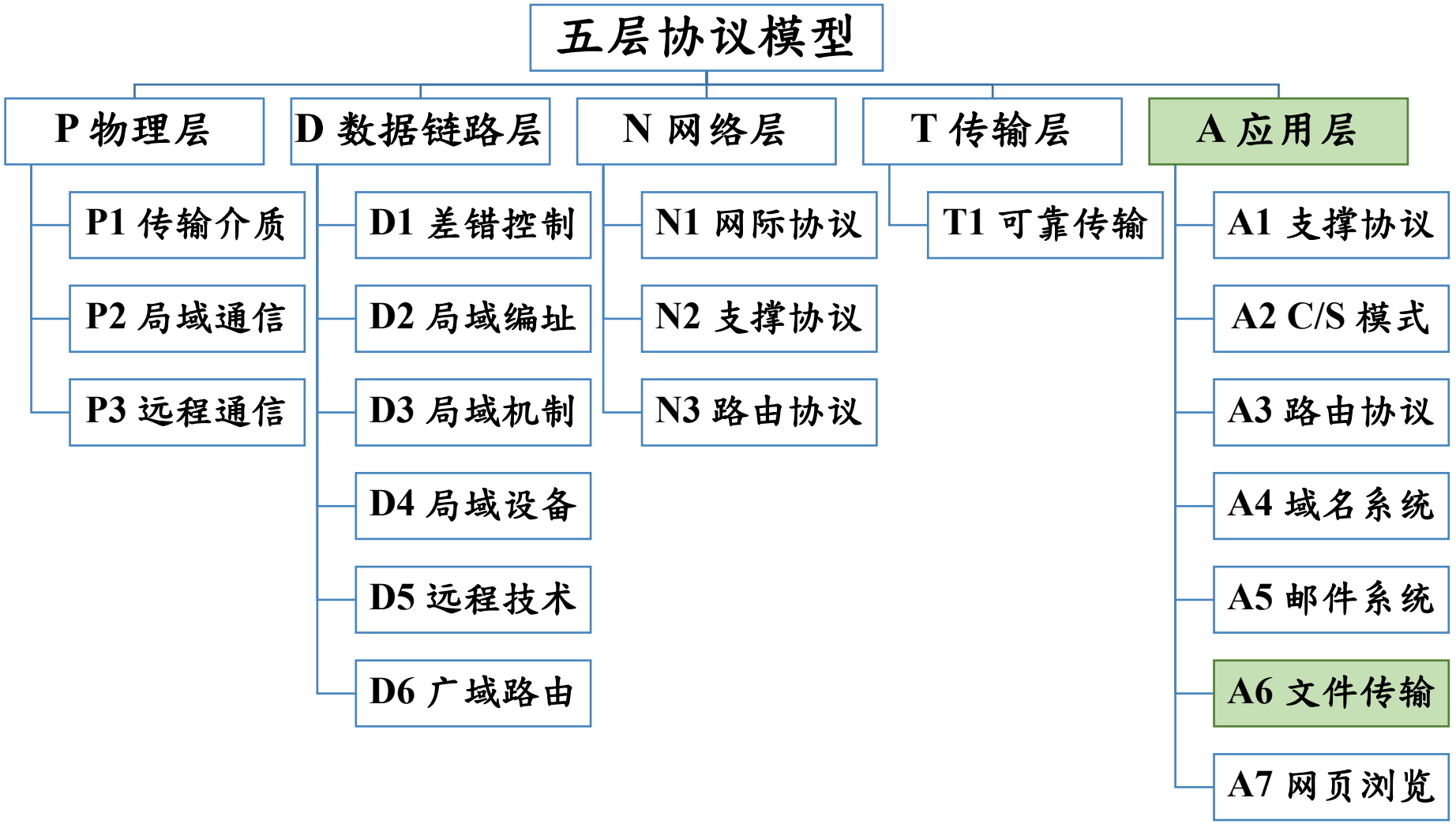
XIAMEN UNIVERSITY

信息学院 黄 炜

(特色化示范性软件学院) 博士,副教授

School of Informatics Wei Huang

知识框架


五层协议模型
P 物理层
D 数据链路层
N 网络层
T 传输层
A 应用层
P1 传输介质
P2 局域通信
P3 远程通信
D1 差错控制
D2 局域编址
D3 局域机制
D4 局域设备
D5 远程技术
D6 广域路由
N1 网际协议
N2 支撑协议
N3 路由协议
T1 可靠传输
A1 支撑协议
A2 C/S 模式
A3 路由协议
A4 域名系统
A5 邮件系统
A6 文件传输
A7 网页浏览

主要内容

- **FTP**
  - 工作原理与通信模式
  - 主动和被动工作模式

对应课本章节

- **PART I Introduction And Internet Applications**
  - **Chapter 4 Traditional Internet Applications**
    - **4.10~4.11 File Transfer Protocol (FTP); FTP Communication Paradigm**

内容纲要

1	FTP协议
2	TFTP协议
3	NFS协议
4	SMB协议
5	Telnet协议

FTP (File Transfer Protocol)

- 文件传送协议 FTP (File Transfer Protocol)
  - 提供交互式的访问，允许客户指明文件的类型与格式，并允许文件具有存取权限。
  - 屏蔽了各计算机系统的细节，因而适合于在异构网络中任意计算机之间传送文件。
  - RFC 959 很早就成为了因特网的正式标准。
- 文件传送并非很简单的问题
  - 众多的计算机厂商研制出的文件系统数百种，且差别很大。

FTP 特点

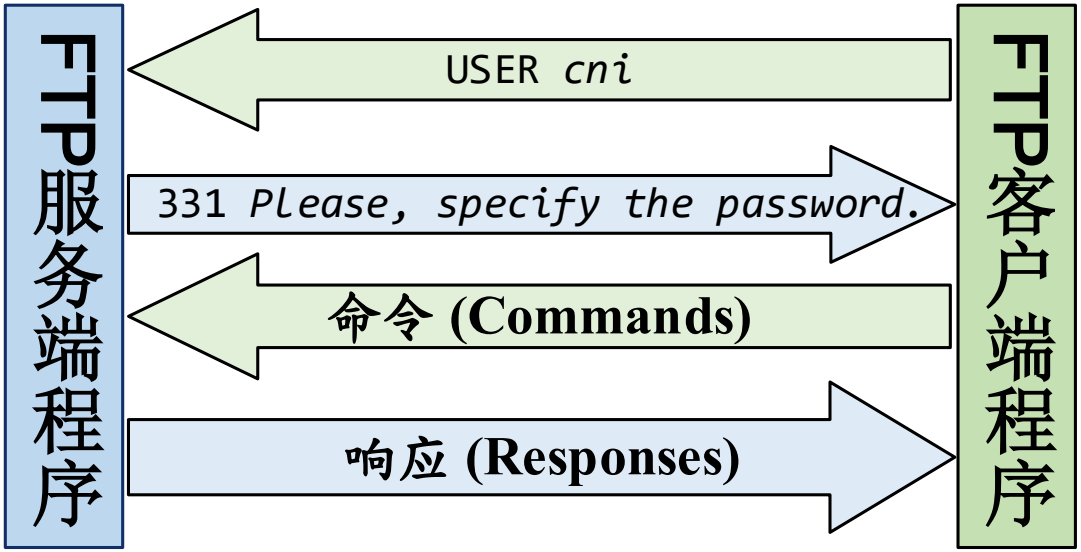
- FTP 使用基于流的客户服务器方式。
  - 一个 FTP 服务器进程可同时为多个客户进程提供服务。
  - FTP 的服务器进程由两大部分组成：
    - 一个主进程，负责接受新的请求；
    - 另外有若干个从属进程，负责处理单个请求。

FTP两个连接

- 控制连接（默认端口：21）
  - 整个会话期间保持打开，客户发出的传送请求通过控制连接发送给服务器端的控制进程，但不用来传送文件。
- 数据连接（默认端口：20）
  - 实际用于传输文件。服务器端的控制进程在接收到 FTP 客户发送来的文件传输请求后创建“数据传送进程”和“数据连接”，用来连接客户端和服务器端的数据传送进程。
  - 数据传送进程实际完成文件的传送，在传送完毕后关闭“数据传送连接”并结束运行。

命令行处理

- FTP控制连接使用“命令-应答”机制
  - 命令格式：命令简称（四个字母）+ 空格 + 命令参数
  - 响应格式：响应代码（三位数字）+ 空格 + 应答消息



FTP命令

- 按照作用主要分为五大类
- 命令一般为3～4个字母的英文单词缩写
  - 方便计算机识别，方便人记忆

命令类别	命令
访问命令	USER PASS ACCT REIN QUIT ABOR
文件管理命令	CWD CDUP DELE LIST MKD PWD RMD RNFR RNTO SMNT
数据格式命令	TYPE STRU MODE
端口定义命令	PORT PASV
文件传输命令	RETR STOR APPE STOU ALLO REST STAT

FTP命令

- 命令之后是参数，为命令提供信息
- 有的命令也可以不带参数

常用命令	参数含义
访问命令	USER *user_id*; PASS *password*
文件管理命令	CWD *dir*; DELE *file*; LIST *dir*; MKD *dir*PWD RMD *dir*; RNFR *old_file*; RNTO *new_file*
数据格式命令	TYPE *type*; STRU *struct*; MODE *mode*
端口定义命令	PORT *port*; PASV
文件传输命令	RETR *files*; STOR *files*; APPE *files*; STOU *files*; ALLO *files*; REST *files*; STAT *files*

FTP 响应

- 响应代码
  - 格式：三位数
  - 信息：服务器程序自选

```
230 User logged in, proceed.
230 Login successful.
```

百位数	含义
1	肯定的初步答复
2	肯定完成回复
3	肯定的中间回答
4	瞬态否定完成回复
5	永久否定完成回复

十位数	含义	解释
0	语法	涉及语法错误，或语法正确但不适合任何功能类别、未实现或多余的命令。
1	信息	这些是对诸如状态或帮助之类的信息。
2	连接	引用控制和数据连接。
3	身份验证和账户	对登录的答复流程和会计程序。
4	未指定	尚未指定。
5	文件系统	指示服务器文件系统与请求的传输或其他文件系统操作。

FTP文件类型

- 用于文本文件
  - 类型A： 8位ASCII编码。
  - 类型E：EBCDIC（Extended Binary-Coded Decimal Interchange Code，扩充的二—十进制交换码）编码
- 用于二进制文件
  - 类型I：以图像（Image）文件代表二进制文件。发送方逐字节发送每个文件，接收方按顺序存储字节流。
- 支持非8位字节的机器
  - 类型L：本地模式。如：TYPE L 9。

FTP文件结构

- 面向流（结构F）
  - 文件结构（默认），将文件视为连续的字节流。
- 面向记录（结构R）
  - 将文本文件划分为记录。记录定长或不定长，在大型机或中型机常见。
- 面向页面（结构P）
  - 文件分为多个页，包含数据或元数据。
  - 每一页可能一个页头，指示各种属性取值。
  - 专为TENEX系统设计

FTP传输模式

- 流模式（模式S，默认）
  - 数据作为连续的字节流发送。
  - 所有处理都交由TCP决定。
- 分块模式（模式B）
  - 数据可以按分块从FTP到TCP分发。每块前有3字节的头。
  - 主要用于记录结构，也可以用于流结构
- 压缩模式（模式C）
  - 通常用游程长度编码扩展模式B

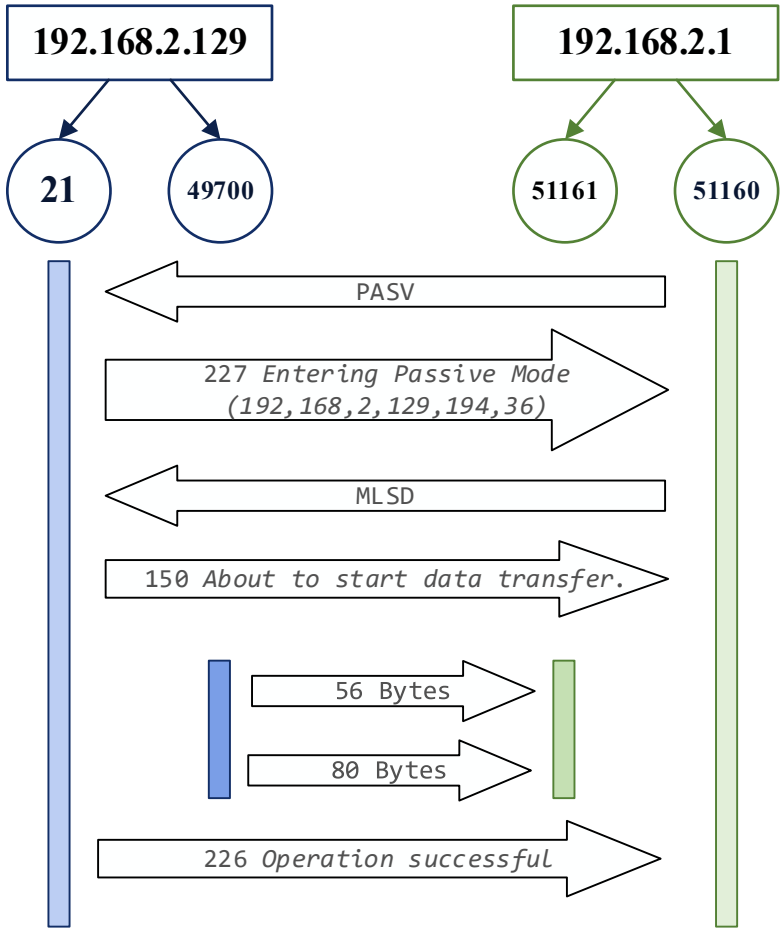
FTP登录

- 使用普通的用户名和密码方案授予访问权限
  - 输入用户名（USER *user_id*），返回331
  - 输入密码（PASS *password*），返回成功230 或失败530
- 允许匿名登录
  - 匿名用户名为anonymous，口令为邮件名，不验证

源IP	目的IP	信息	说明
192.168.2.129	192.168.2.1	220-FileZilla Server 1.7.2 220 Please visit https://filezilla-project.org/\r\n	欢迎辞
192.168.2.1	192.168.2.129	USER anonymous	匿名用户
192.168.2.129	192.168.2.1	331 Please, specify the password.	要求密码
192.168.2.1	192.168.2.129	PASS anonymous@example.com	匿名用户密码
192.168.2.129	192.168.2.1	530 Login incorrect.	登录失败
192.168.2.129	192.168.2.1	230 Login successful.	登录成功

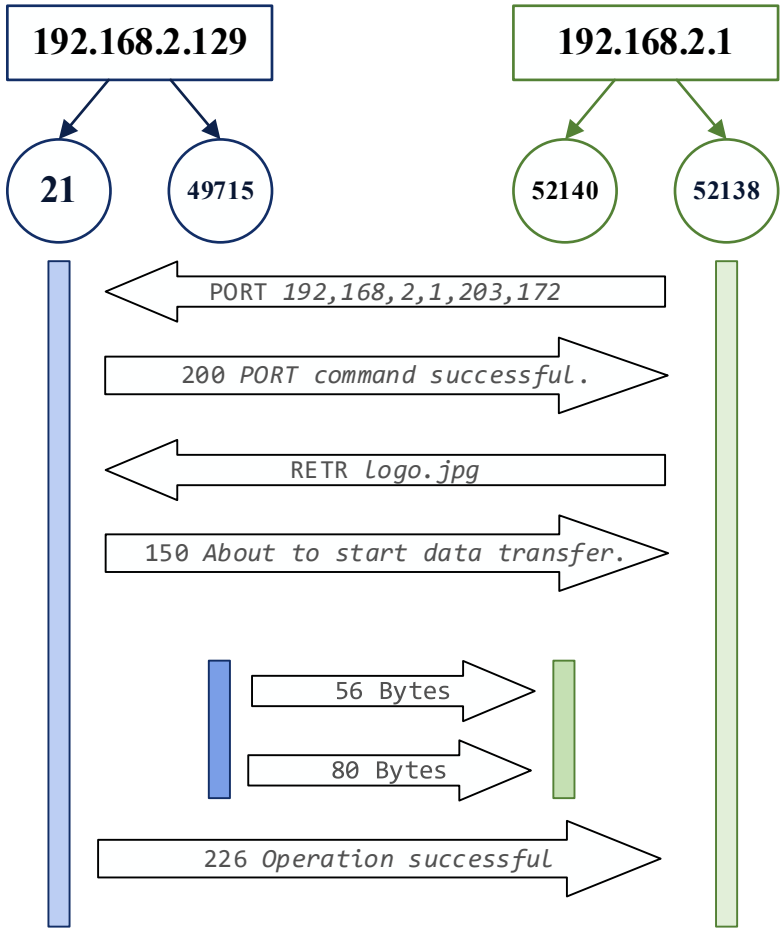
主动连接和被动连接

- 被动连接
  - 客户端发送PASV
  - 服务器占用并告知端口
  - 客户端发送命令
  - 服务器端告知开始发送
  - 服务器端在数据端口收发数据
  - 服务器端告知操作成功
- 缺点
  - 不安全



主动连接和被动连接

- 主动连接
  - 客户端发送PORT和端口
  - 服务器端告知允许并连接
    - 服务器端20号端口不是必须的
  - 客户端发送命令
  - 服务器端告知开始发送
  - 一方正式发送数据
  - 服务器端告知操作成功
- 缺点
  - 不支持NAT

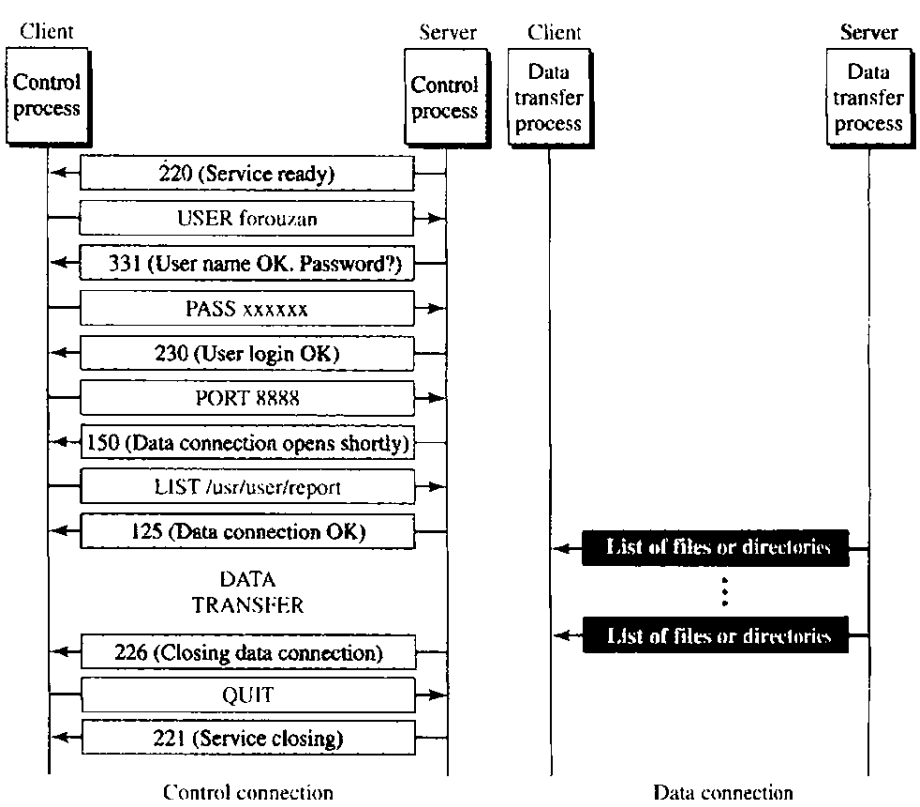


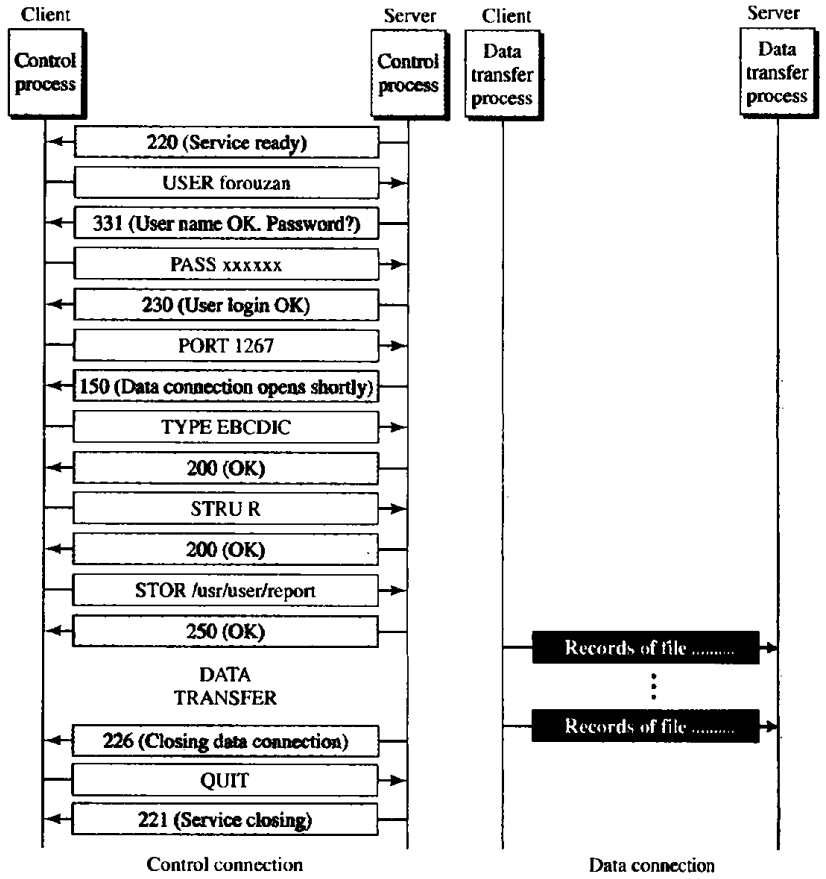
FTP 两个不同的端口号

- FTP 使用2个端口号，数据连接与控制连接不会混乱。
  - 传输文件还可利用控制连接（如：客户发送请求终止传输）
  - 端口21：控制链接
    - 当客户软件向服务器软件发出建立连接请求时，寻找连接服务器软件的21号端口
    - 同时还告诉服务器软件自己的另一端口号，用于建立数据传送连接。
  - 端口20：数据连接
    - 接着，服务器软件用自己传送数据的20号端口与客户软件所提供的端口号码建立数据传送连接。

范例

- CS交互范例


Client
Server
Client
Server
Control process
Control process
Data transfer process
Data transfer process
220 (Service ready)
USER forouzan
331 (User name OK. Password?)
PASS xxxxxx
230 (User login OK)
PORT 8888
150 (Data connection opens shortly)
LIST /usr/user/report
125 (Data connection OK)
DATA TRANSFER
226 (Closing data connection)
QUIT
221 (Service closing)
List of files or directories
List of files or directories
Control connection
Data connection


Client
Server
Client
Server
Control process
Control process
Data transfer process
Data transfer process
220 (Service ready)
USER forouzan
331 (User name OK. Password?)
PASS xxxxxx
230 (User login OK)
PORT 1267
150 (Data connection opens shortly)
TYPE EBCDIC
200 (OK)
STRU R
200 (OK)
STOR /usr/user/report
250 (OK)
DATA TRANSFER
226 (Closing data connection)
QUIT
221 (Service closing)
Records of file .....
Records of file .....
Control connection
Data connection

FTP协议现状

- 主流浏览器软件不支持FTP
- 存在很多安全弱点
  - 蛮力攻击；FTP 反弹攻击；抓包；端口窃取（猜测开放端口并篡夺合法连接）；欺骗攻击；用户名枚举；DoS 或 DDoS

内容纲要

1	FTP协议
2	TFTP协议
3	NFS协议
4	SMB协议
5	Telnet协议

简单文件传输协议（TFTP）

- **Trivial File Transfer Protocol（TFTP）**
  - **TFTP 使用 UDP 服务的端口 69.**
- **TFTP是一种简化的TCP/IP文件传输协议。**
- **TFTP只限于简单文件传输操作，它不提供权限控制，也不支持客户与服务器之间复杂的交互过程，没有庞大的命令集，没有列目录的功能，不能鉴别用户身份。因此TFTP软件比FTP软件小的多。**

TFTP

- TFTP 是一个很小且易于实现的文件传送协议。
- TFTP 使用客户服务器方式和使用 UDP 数据报，因此 TFTP 需要有自己的差错改正措施。
- TFTP 只支持文件传输而不支持交互。
- TFTP 没有一个庞大的命令集，没有列目录的功能，也不能对用户进行身份鉴别。

TFTP 的主要特点

- 每次传送的数据 PDU 中有 512 字节的数据，但最后一次可不足 512 字节。
- 数据 PDU 也称为文件块(block)，每个块按序编号，从 1 开始。
- 支持 ASCII 码或二进制传送。
- 可对文件进行读或写。
- 使用很简单的首部。

TFTP 的工作类似停止等待协议

- 发送完一个文件块后就等待对方的确认，确认时应指明所确认的块编号。
- 发完数据后在规定时间内收不到确认就要重发数据 PDU。
- 发送确认 PDU 的一方若在规定时间内收不到下一个文件块，也要重发确认 PDU。这样就可保证文件的传送不致因某一个数据报的丢失而告失败。

TFTP 的工作类似停止等待协议

- 在一开始工作时。TFTP 客户进程发送一个读请求 PDU 或写请求 PDU 给 TFTP 服务器进程，其熟知端口号码为 69。
- TFTP 服务器进程要选择一个新的端口和 TFTP 客户进程进行通信。
- 若文件长度恰为 512 字节的整数倍，则文件传送完毕后还必须最后发送一个只含首部而无数据的数据 PDU。
- 若文件长度不是 512 字节的整数倍，则最后传送数据 PDU 的数据字段一定不满512字节，这正好可作为文件结束的标志。

内容纲要

1	FTP协议
2	TFTP协议
3	NFS协议
4	SMB协议
5	Telnet协议

网络文件系统（ NFS ）

- 网络文件系统（ Network File System ，NFS ）
- NFS只是一种文件系统，本身没有传输功能，是基于RPC协议实现的，才能达到两个Linux系统之间的文件目录共享；

Network File System

- NFS 允许应用进程打开一个远程文件，并能在该文件的某一个特定的位置上开始读写数据。
- NFS 可使用户只复制一个大文件中的一个很小的片段，而不需要复制整个大文件。
- 例：计算机 A 的 NFS 客户软件，把要添加的数据和在文件后面写数据的请求一起发送到计算机 B 的 NFS 服务器。NFS 服务器更新文件后返回应答信息。
- 在网络上传送的只是少量的修改数据。

内容纲要

1	FTP协议
2	TFTP协议
3	NFS协议
4	SMB协议
5	Telnet协议

简单文件共享：SMB协议

- Server Message Block
- 基于TCP-NETBIOS下的，一般端口为139、445。
  - NetBIOS（网络基本输入/输出系统协议）协议是由IBM公司开发，主要用于数十台计算机的小型局域网。
  - NetBIOS协议是一种在局域网上的程序可以使用的API，为程序提供了请求低级服务的统一的命令集
  - 几乎所有的局域网都是在NetBIOS协议的基础上工作的。

远程终端协议 Telnet

- Telnet 是一个简单的远程终端协议。
- 用户用 Telnet 就可在其所在地通过 TCP 连接注册（即登录）到远程的另一个主机上（使用主机名或 IP 地址）。
- Telnet 能将用户的击键传到远程主机，同时也能将远程主机的输出通过 TCP 连接返回到用户屏幕。这种服务是透明的，因为用户感觉到好像键盘和显示器是直接连在远程主机上。

内容纲要

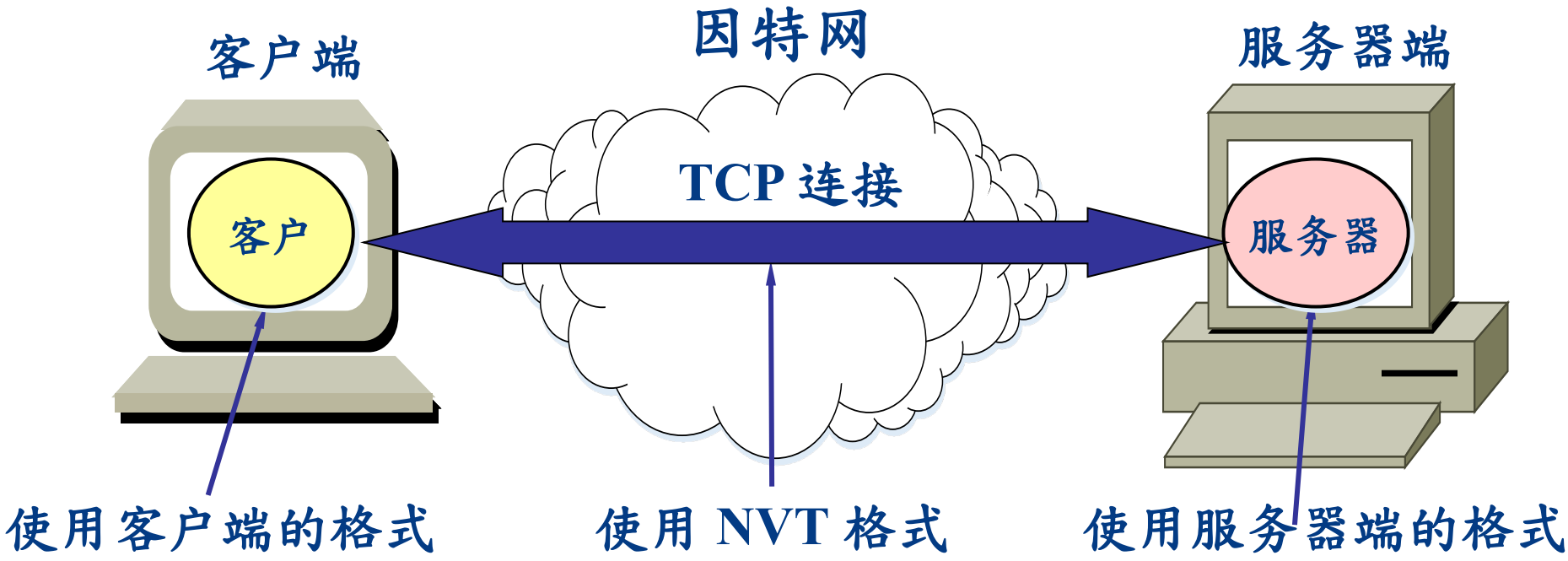
1	FTP协议
2	TFTP协议
3	NFS协议
4	SMB协议
5	Telnet协议

客户–服务器方式

- 现在由于 PC 的功能越来越强，用户已较少使用 Telnet 了。（但在CUI界面系统仍使用）
- Telnet 也使用客户–服务器方式。在本地系统运行 Telnet 客户进程，而在远程主机则运行 Telnet 服务器进程。
- 和 FTP 的情况相似，服务器中的主进程等待新的请求，并产生从属进程来处理每一个连接。

Telnet 使用NVT 格式

- 网络虚拟终端 NVT



网络虚拟终端 NVT 格式

- 客户软件把用户的击键和命令转换成 NVT 格式，并送交服务器。
- 服务器软件把收到的数据和命令，从 NVT 格式转换成远程系统所需的格式。
- 向用户返回数据时，服务器把远程系统的格式转换为 NVT 格式，本地客户再从 NVT 格式转换到本地系统所需的格式。

计算机网络

Computer Network

15

谢谢观看

理论课程

厦門大學

XIAMEN UNIVERSITY

信息学院 黄 炜

(特色化示范性软件学院) 博士,副教授

School of Informatics Wei Huang