# Linux网络管理

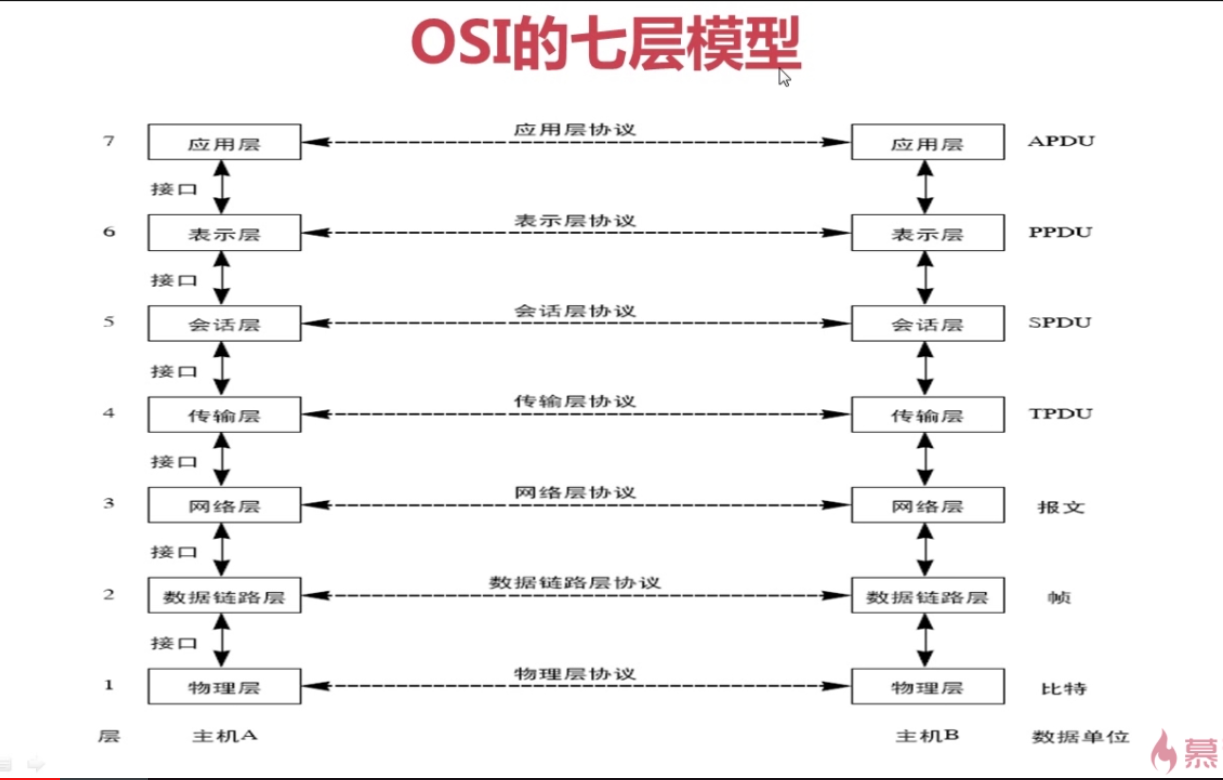
## 网络基础

### 1.1 iso/osi七层模型

ISO:国际化标准组织

OSI:开放系统互联组织

IOS:苹果操作系统…



分层的目的 是为了 便于管理

上 三层 是给 用户用的

下 四层 才是真正传递数据的

Ipconfig /all

Mac地址负责局域网通信

Ip地址负责网络通信

#### 物理层

设备之间比特流(0101)的传输,物理接口,电气特性等.

(网线有八根线,1 3 2 6四根线传输数据) 网线 网卡

#### 数据链路层

成帧,用mac地址访问媒介,错误检测与修正.

#### 网络层

提供逻辑地址(ip地址 源Ip与目标ip),选路

#### 传输层

可靠与不可靠的传输,传输前的错误检测,流控

(TCP (可靠)/ UDP(传输速度快,有可能丢失)传输协议)

传输层 为了确定端口号 通过端口号确定 响应的是哪个服务

#### 会话层

对应会话的管理,同步

(确定数据是否会向下层传递, 是 保存文件到硬盘 还是 进行网络传输 )

#### 表示层

数据的表现形式,特定功能的实现,如 加密

英文 通过 ASCLL码翻译成 0101

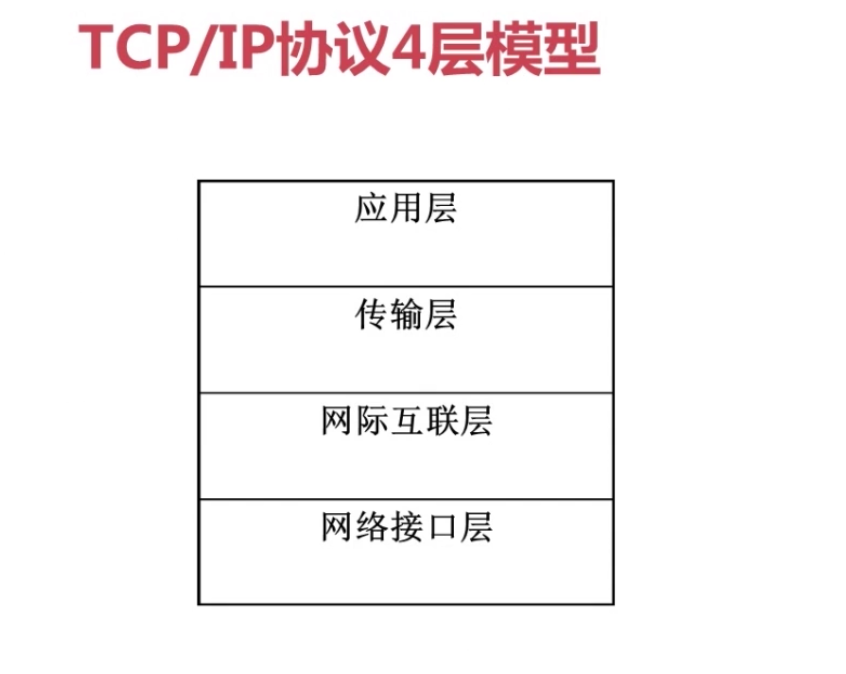
中文 通过 GB2321 翻译成 0101

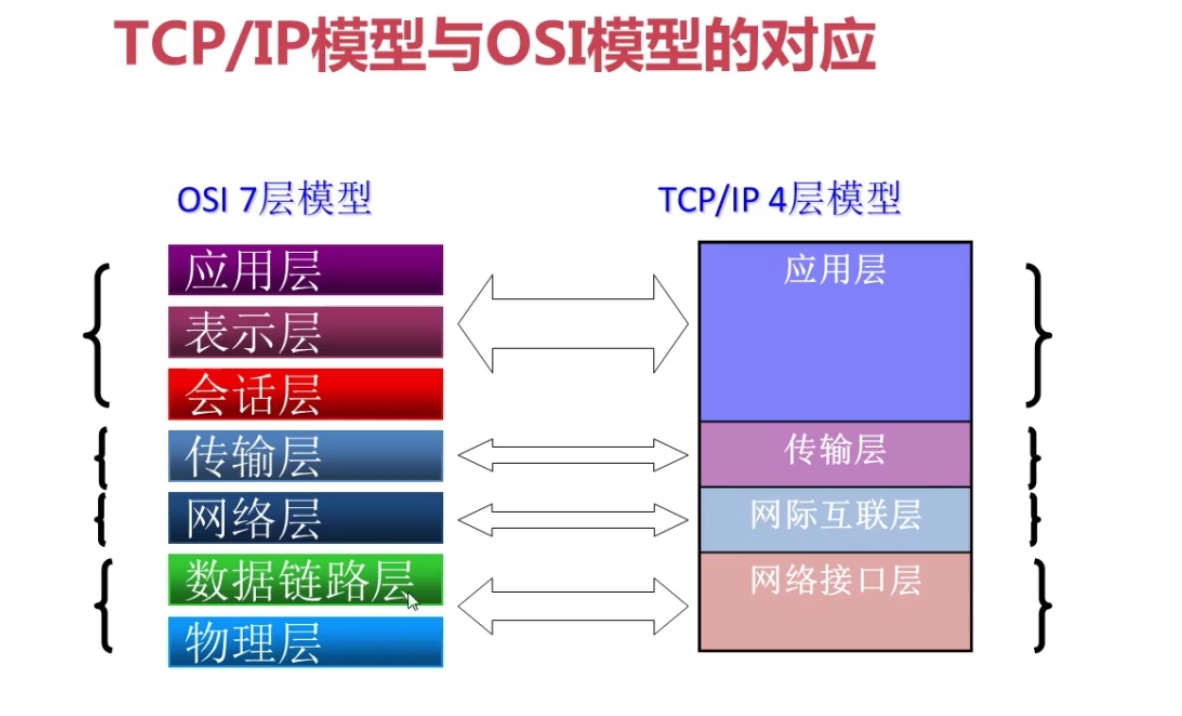
Jpg,mp3 等 文件格式 是为了让计算机知道对应额的解码方式

#### 应用层

用户接口

### 1.2 TCP/IP四层模型

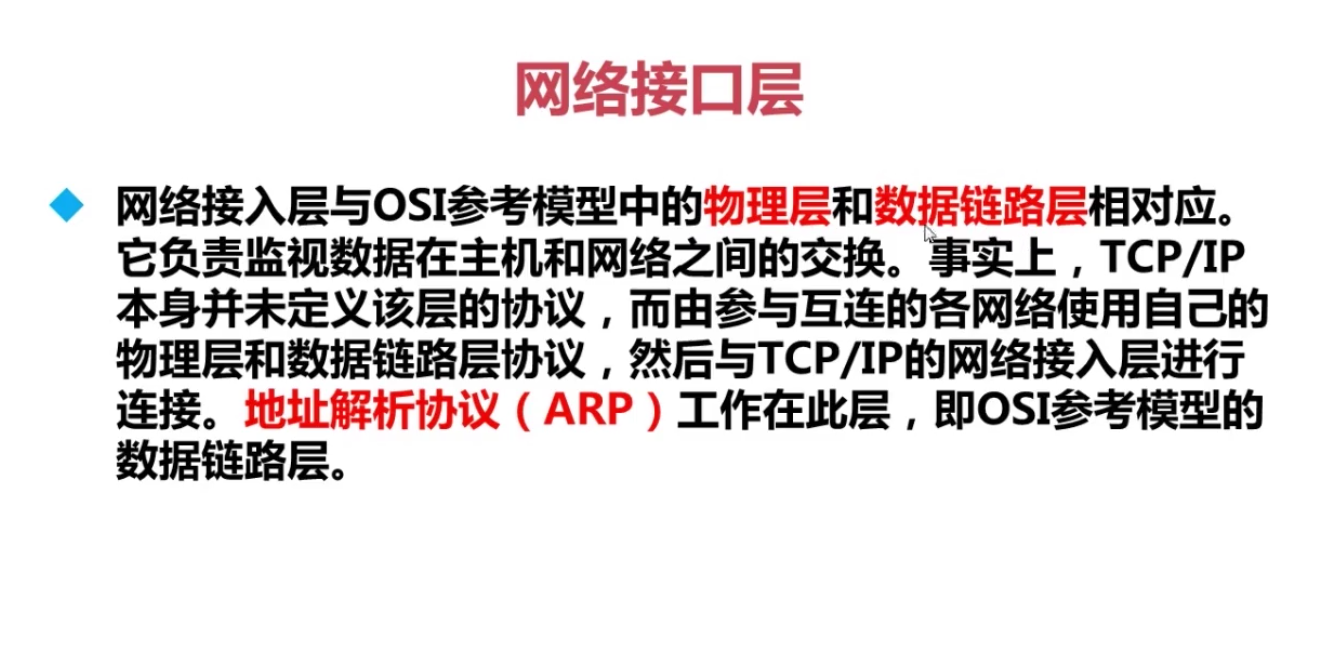


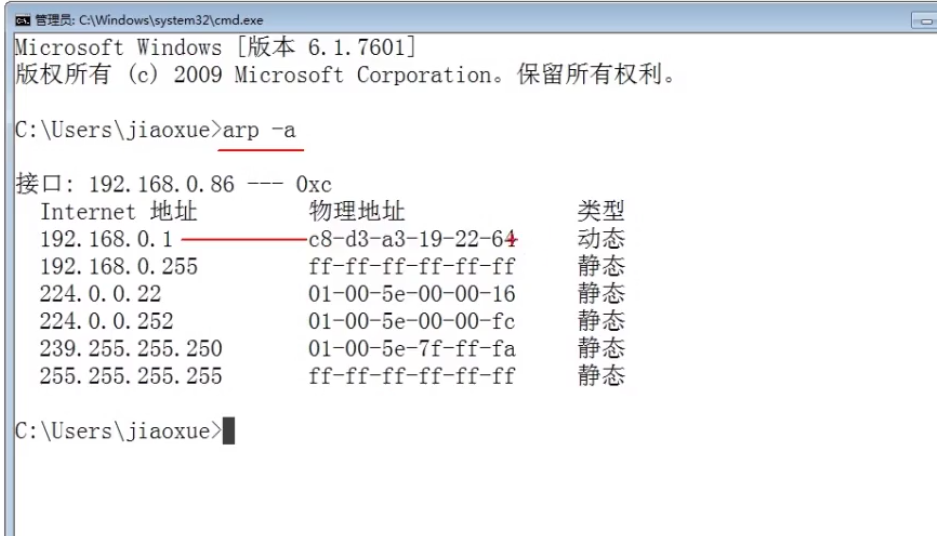


OSI 7层模型,TCP/IP 4层 原型车

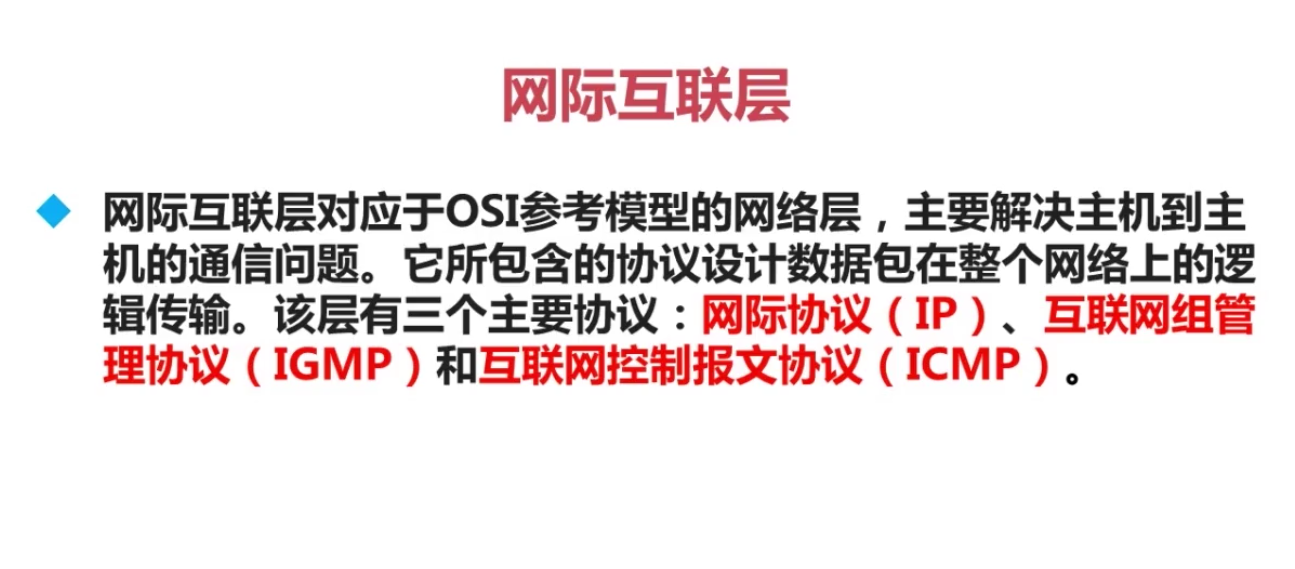
#### 网络接口层

网络接口层与OSI参考模型中的 物理层 和 数据链路层 相对应.它负责监视数据在主机和网络之间的交换. 事实上,TCP/IP本身并未定义该层的协议, 而由参与 互联的 各网络使用自己的物理层和数据链路层协议, 然后与TCP/IP的网络接入层进行连接. 地址解析协议(ARP) 工作在此层, 即OSI参考模型的数据链路层.

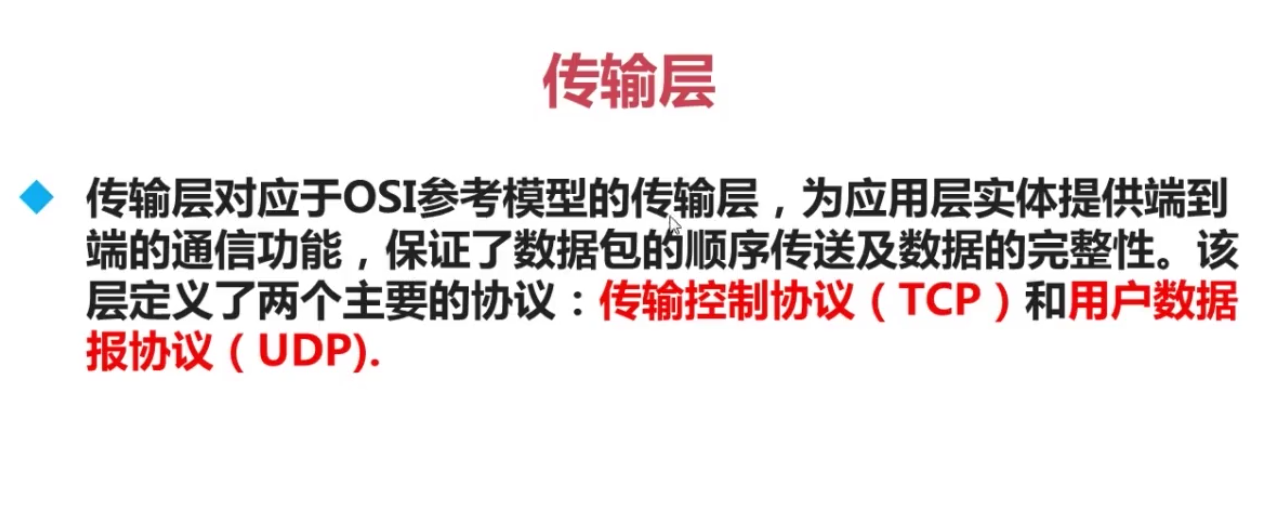


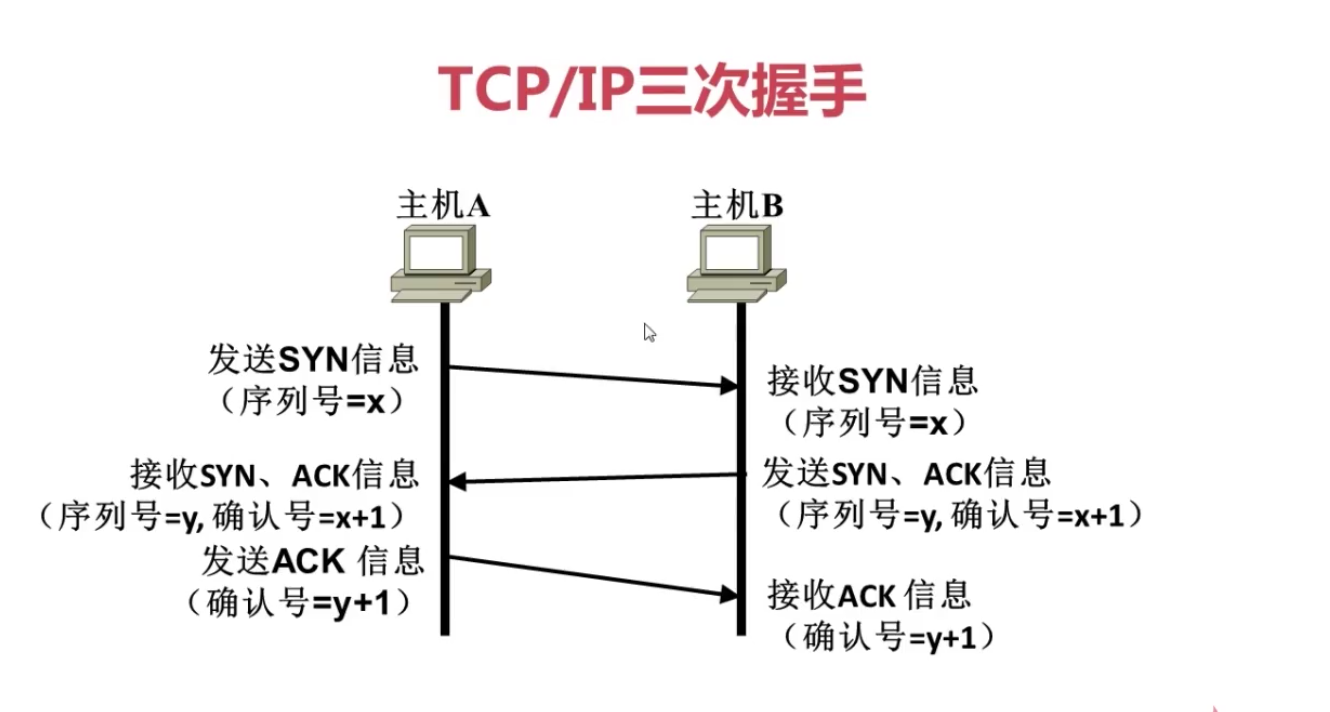


#### 网际互联层



#### 传输层



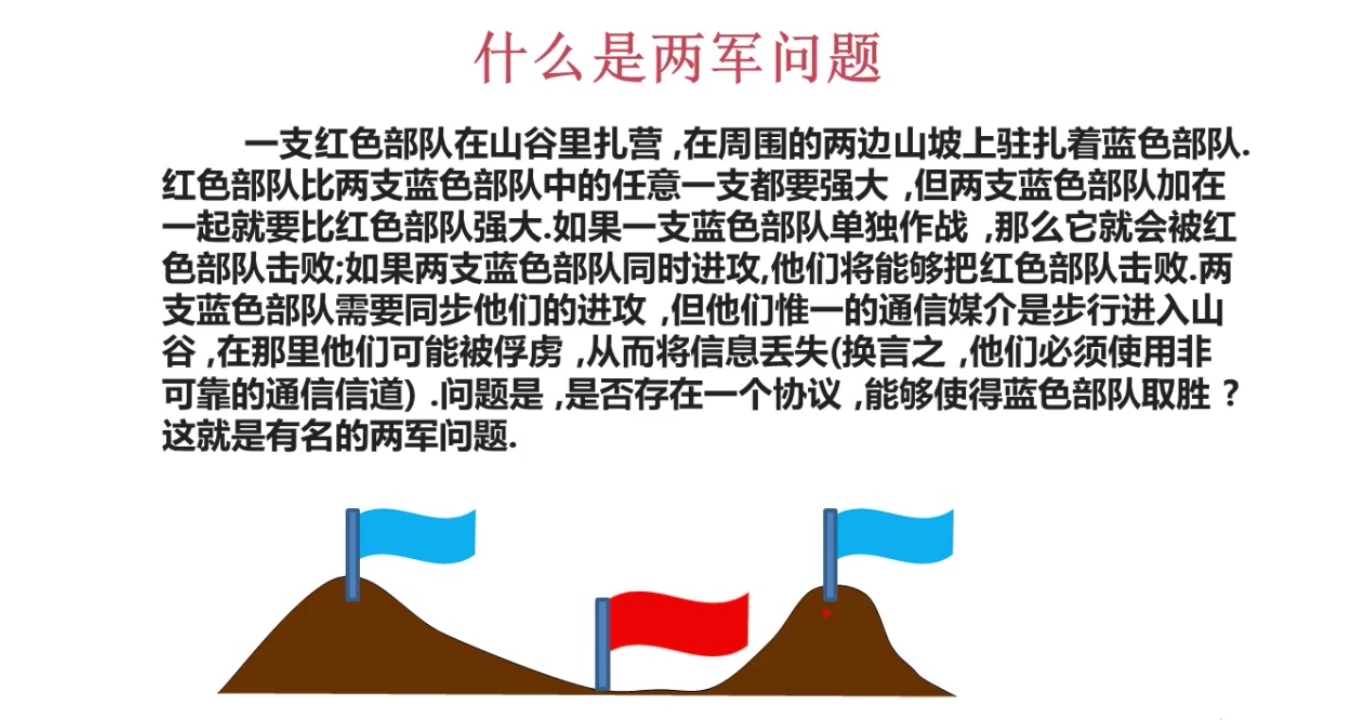


TCP三次握手 面向连接 可靠 的协议

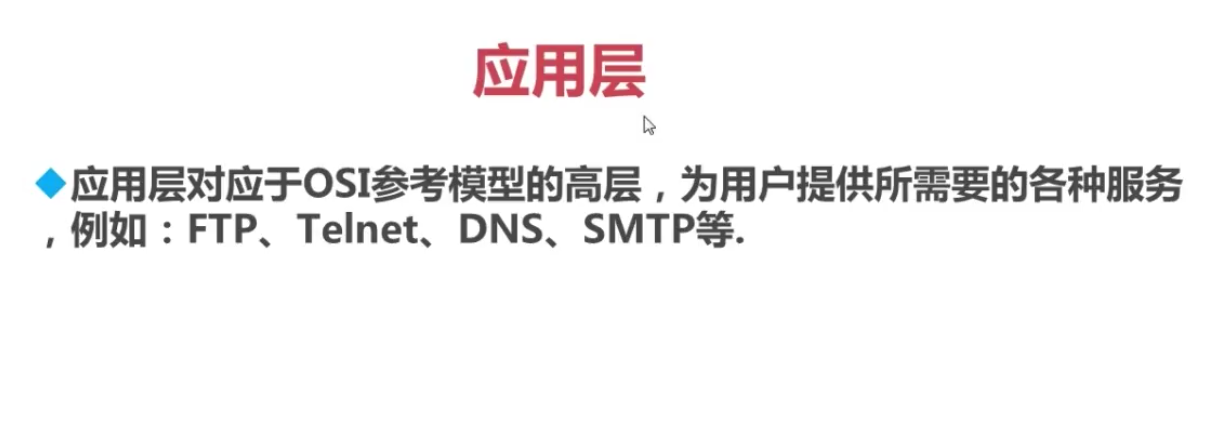
Eg:打电话

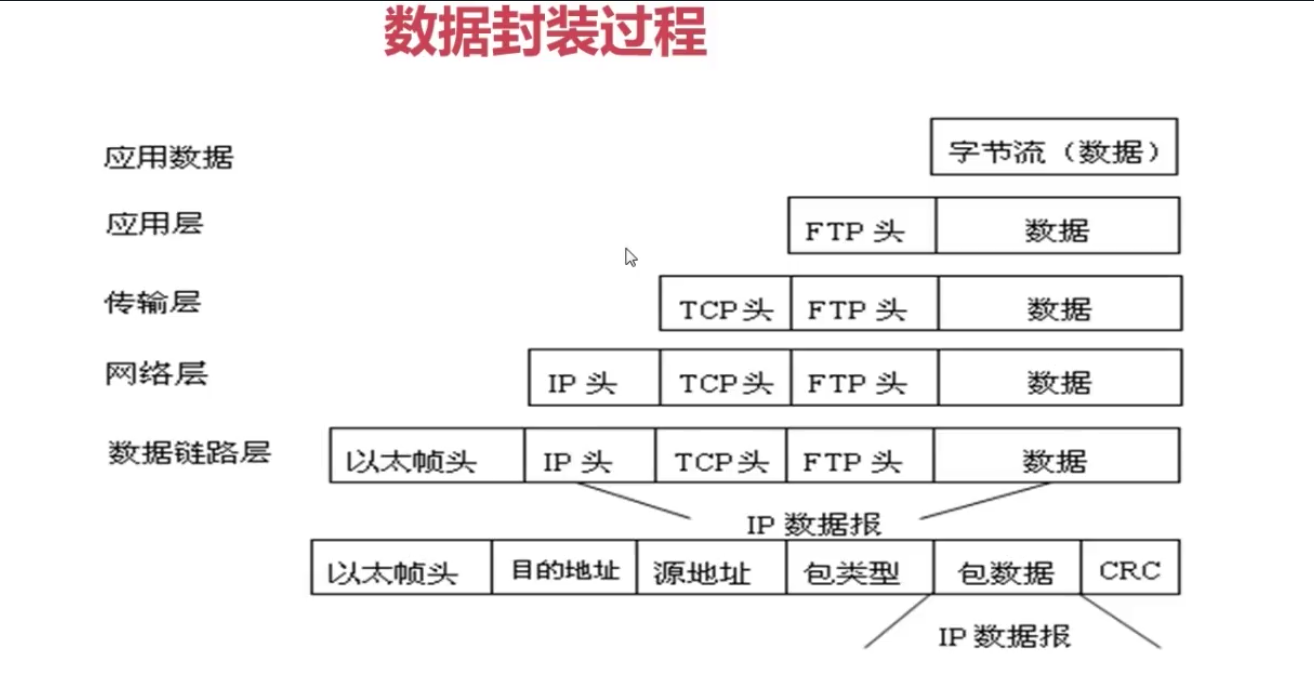
UDP直接发送 协议

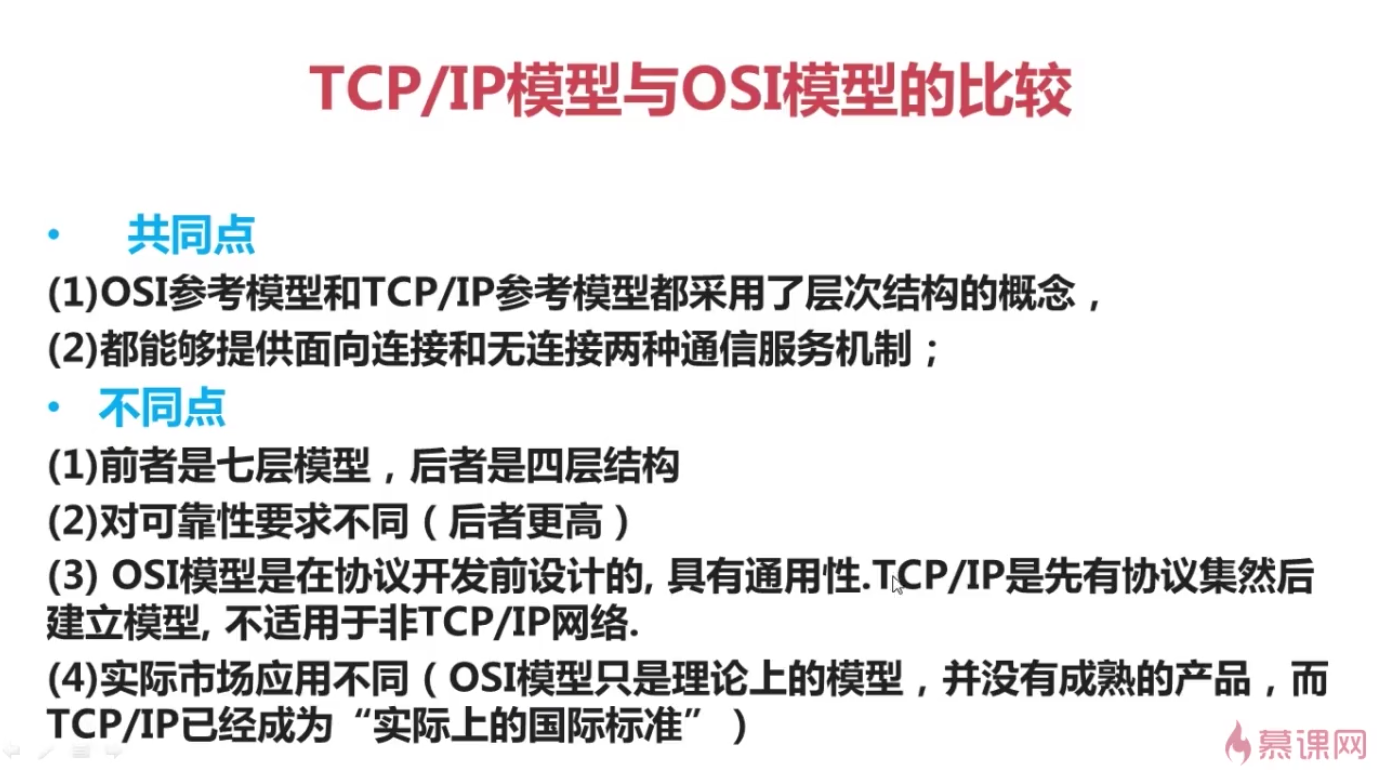
Eg:发短信



#### 应用层



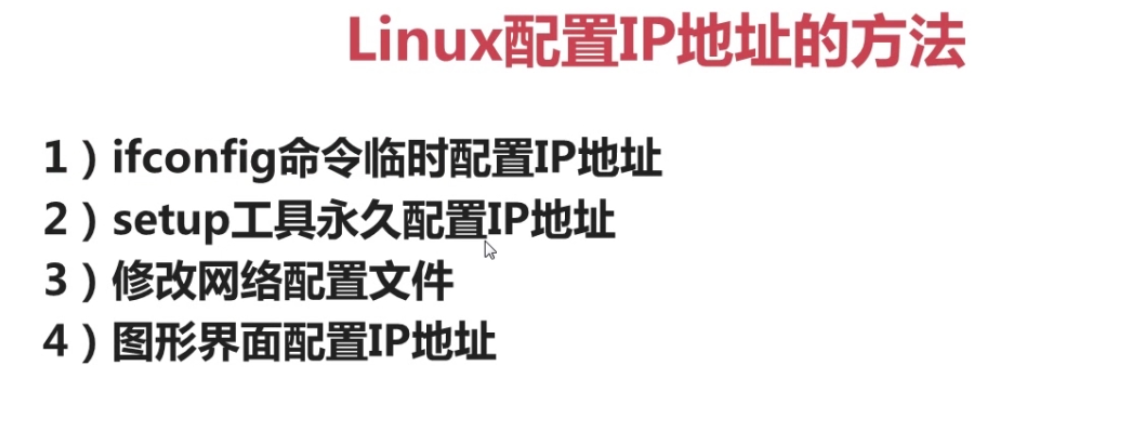




* 1. IP地址
  2. 端口作用
  3. DNS作用
  4. 网管作用

## linux网络配置

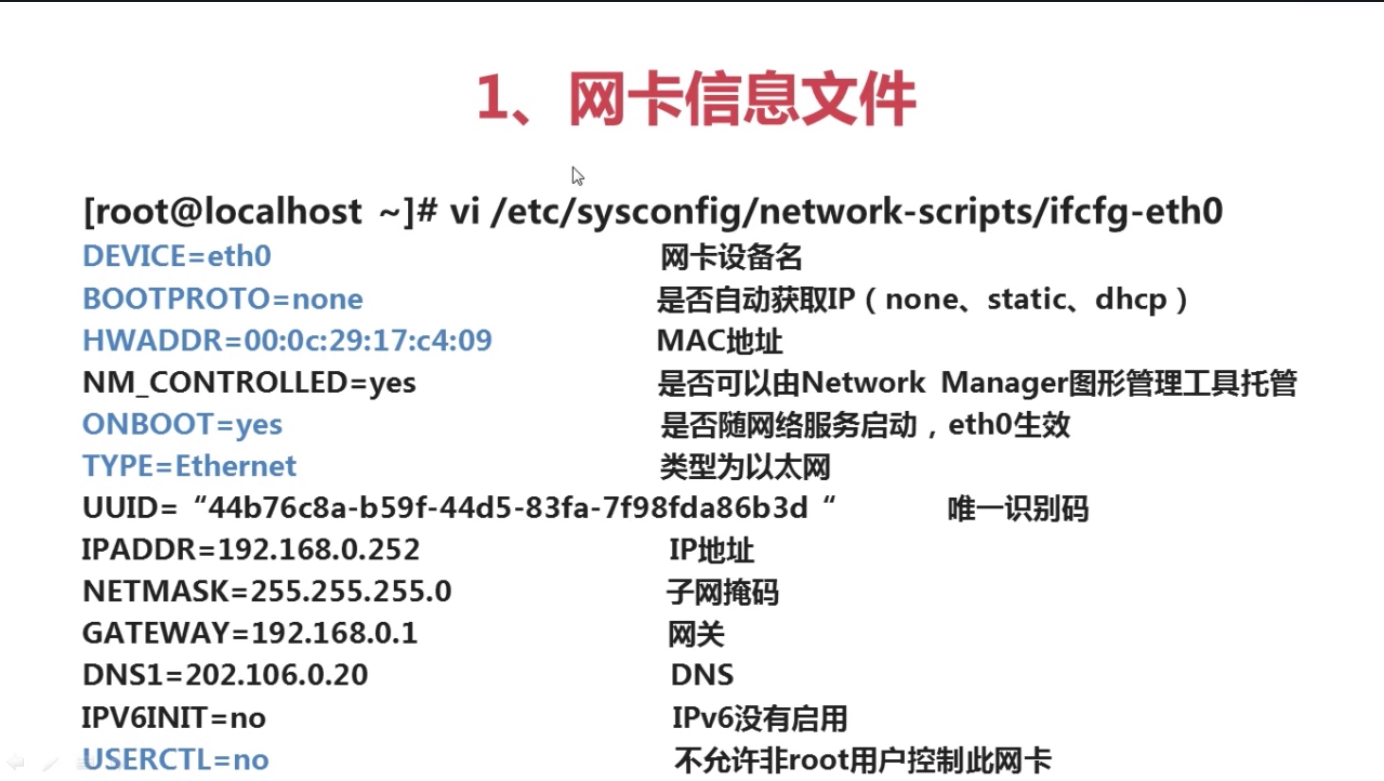
* 1. Linux配置ip地址的方法

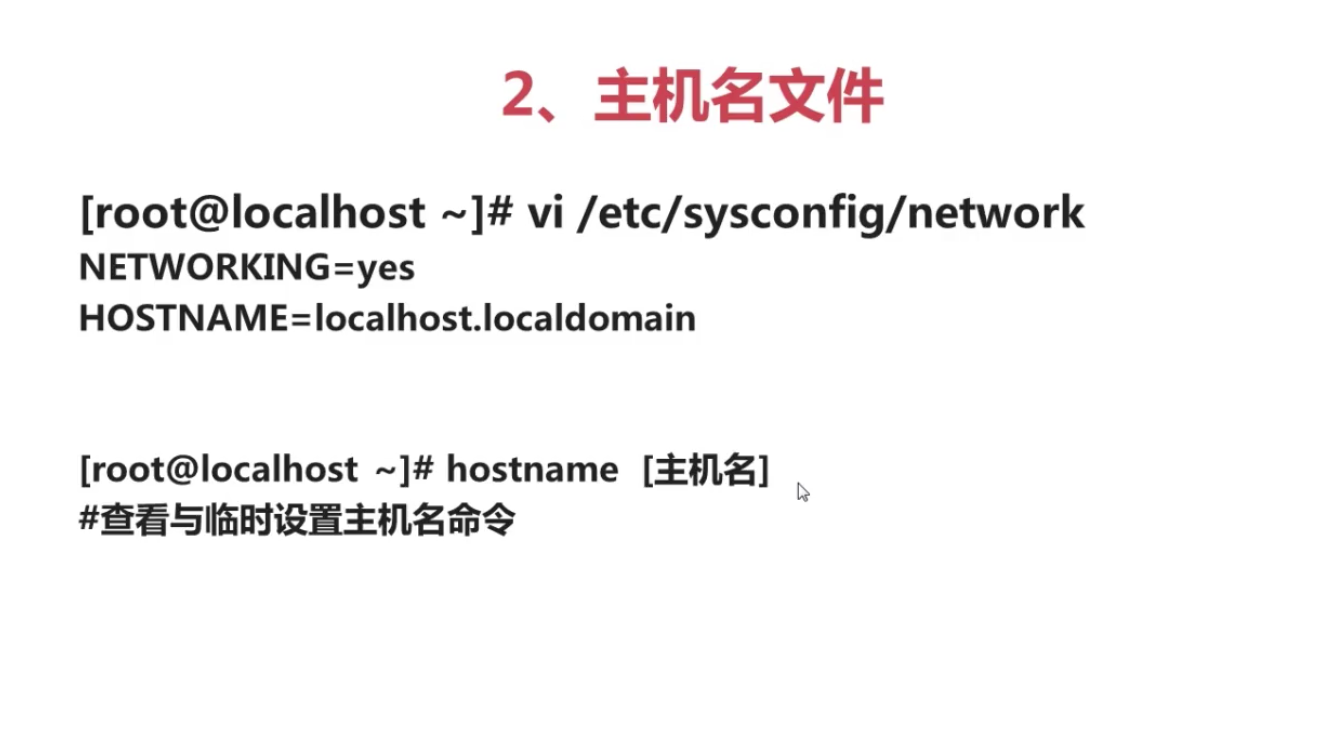


1. ifconfig命令临时配置Ip地址



1. setup 工具永久配置ip地址
2. 修改网络配置文件







1. 图形界面配置Ip地址

开放防火墙端口号命令:

Iptables -I INPUT -p tcp --dport 3306 -j ACCEPT

### CentOS7防火墙

CentOS 7.0默认使用的是firewall作为防火墙，这里改为iptables防火墙。

**1、关闭firewall：**

systemctl stop firewalld.service #停止firewall

systemctl disable firewalld.service #禁止firewall开机启动

**2、安装iptables防火墙**

yum install iptables-services #安装

vi /etc/sysconfig/iptables #编辑防火墙配置文件

# Firewall configuration written by system-config-firewall

# Manual customization of this file is not recommended.

\*filter

:INPUT ACCEPT [0:0]

:FORWARD ACCEPT [0:0]

:OUTPUT ACCEPT [0:0]

-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

-A INPUT -p icmp -j ACCEPT

-A INPUT -i lo -j ACCEPT

-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT

-A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT

-A INPUT -m state --state NEW -m tcp -p tcp --dport 3306 -j ACCEPT

-A INPUT -j REJECT --reject-with icmp-host-prohibited

-A FORWARD -j REJECT --reject-with icmp-host-prohibited

COMMIT

:wq! #保存退出

systemctl restart iptables.service #最后重启防火墙使配置生效

systemctl enable iptables.service #设置防火墙开机启动

## linux网络命令

## 远程登录