1. 概述

**1.TCP/IP的概念**

TCP/IP中协议的概念就是需要实现某些功能和接口的软件规范。支持协议意味着该设备有实现协议的软件。协议族就是一系列相关协议的集合。TCP/IP就是1个协议族。

TCP/IP协议通常认为是1个4层协议系统。

（1）链路层，有时也称作数据链路层，通常包括计算机中的网络接口卡（NIC，网卡）和操作系统中对应的的设备驱动程序。

（2）网络层，用于处理分组在网络中的活动，如分组的选路。网络层协议包括IP协议、ICMP协议和IGMP协议。

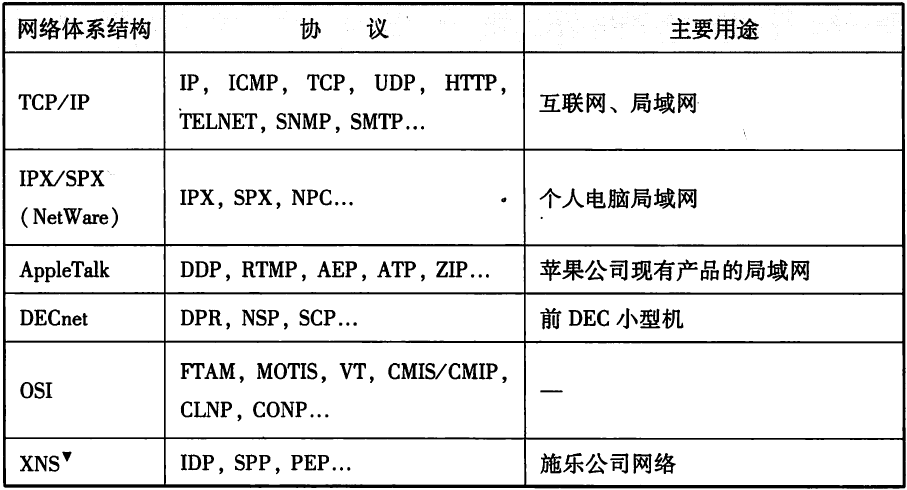
（3）运输层，主要为两台主机上的应用程序提供端到端的通信。包括TCP协议和UDP协议。

（4）应用层，负责处理特定的应用程序细节，是1个用户进程，并不关心数据在网络中的传输活动。

链路层、网络层和运输层通常在（操作系统）内核中执行。

协议的分层，如同软件开发中的模块化开发。每1层都接收由它下层提供的服务，并为自己的上层提供服务。上下层之间的交互所遵守的约定就是“接口”。同层之间交互所遵守的约定叫“协议”。无论使用何种下层协议，如何实现，但提供给上层的服务接口都是一样的。即使下层协议变化，上层协议实现也不用变化。

除了tcp/ip协议外，还有其他协议（只要有软件实现也硬件支持就行），有的用于其他网络，如AppleTalk，用于苹果产品的局域网；有的只是约定了一些概念，如OSI，实际并没有用。要在Internet中实现通信，就必须使用TCP/IP协议。



**2.网络分类**

（1）按传输方式，分为面向有连接型和面向无连接型。

面向有连接型中，在发送数据之前，需要在收发主机之间建立一条通信线路。面向无连接型中，发送端可以在任何时候自由发送数据，而接收端也不知道自己何时会从何处接受到数据。

有无连接是逻辑上的，物理上肯定都是有线路连接的。只是面向有连接型会有连接过程，进程保持会话直到断开，与网络层选择什么路由，走什么线路无关。

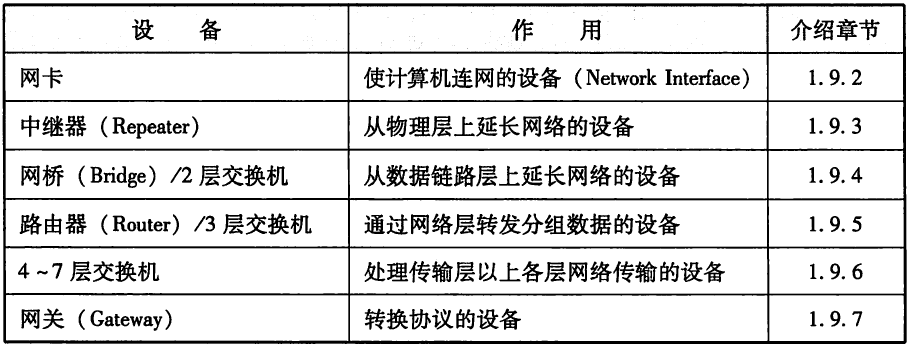
（2）电路交换和分组交换

电路交换主要用于过去的电话网络，需要通过交换机和目标主机建立通信线路，并一直通过这条线路通信直到线路断开。分组交换将数据分割为1个个较小的包进行传输，并非按特定线路传输。

（3）单播、广播、组播、任播

按接收端数量分类，其中单播是1对1通信；广播是发送给一个广播域中的所有主机；组播是以一组主机作为接受端；任播是在多台特定主机中任选1台作为其接收端，应用如DNS服务器。

**3.搭建网络的主要设备**



（1）网卡全称为网络接口卡（NIC）。任何计算机要连入网络，都必须使用网卡。

（2）交换机能按端口分为多个网段，从而区分广播数据传播范围。网桥能够识别数据链路层中的数据帧，并临时保存再重新生成新的信号作为1个全新的帧发送给相连的另1个网段。网桥、交换机检验数据正确性是通过帧的FCS位来判断的。

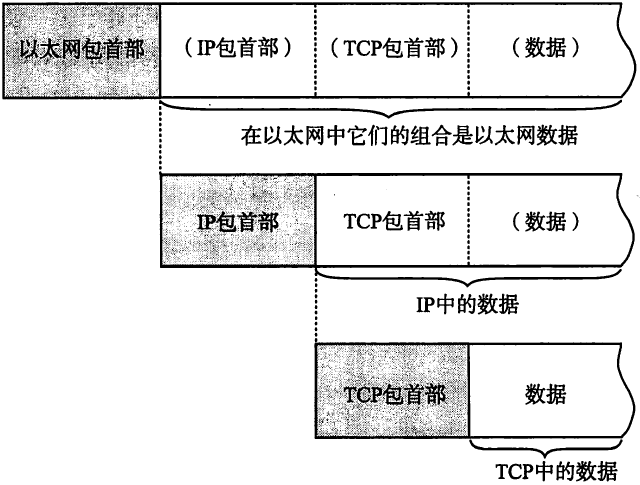
（3）历史上，网关指路由器。现在网关指表示应用层网关，是负责将传输层数据转换和转发到应用层的设备，如，互联网邮件和手机邮件在应用层使用的邮件协议可能不一样，在接收后会合理转换发送给应用层。防火墙也是1种网关。



（4）主机也可以有多个接口，也可以实现路由功能。但如果不是专门用于转发，主机一般不称作为路由器。

**4.包**

在TCP/IP协议中，包是描述数据的单位的统称。帧是数据链路层中包的单位。数据包是IP和UDP等网络层以上分层中包的单位。段则是指TCP数据流中的信息。



每1层，上层包都会被加上某种标识，以表明数据属于那一层。而且下层还会在包中专门用1个区域标注上层协议类型，如链路层将上层协议标注后，发送到对方主机，对方主机获取后，获取上层协议类型后，将数据发送给对应的上层协议处理接口处理。

**5.标准化**

（1）所有有关Internet正式标准都以RFC文档出版。

（2）TCP/IP协议在标准化过程中，不仅仅是制定协议规范，更注重实现通信的技术，即实现软件。

1. 数据链路层
   1. 概要

（1）数据链路可以视为网络中最小单位，其作用是为IP模块发送和接收IP数据包。

（2）数据链路层只是1段网络，跟对方网络对应部分没有任何关系。当这段数据链路接受到IP包时，没有任何上一链路的任何信息，只要上层IP协议的数据。当这段链路结束将数据转出时，出口设备也会去掉帧包装。所以，数据链路层选择任何协议都可以，只要传出的时候对方能够识别即可。

（3）数据链路层协议有许多，如以太网、无线网、PPP等，只要能实现收发数据就行。

* 1. 以太网

**1.概要**

（1）IEEE标准（应该也被收录到RFC中）中，前缀为802的标准规定了局域网和城域网的工作过程，其中802.3位以太网，802.11位无线局域网。

（2）以太网种类标识中，如10BASE-T，10表示10Mbps、100Mbps，后面的T、F等表示双绞线、单模/双模光纤等传输介质，因此会有多种组合。

（3）网络可分为共享介质和非共享介质型。

共享介质型网络中多个设备共享1个通信介质，通常是半双工通信，有争用方式和令牌方式2种对介质的访问控制方式。即使对于通过双绞线连接主机，但因每台主机的双绞线都汇到1条总线上，所以还是必须使用半双工。

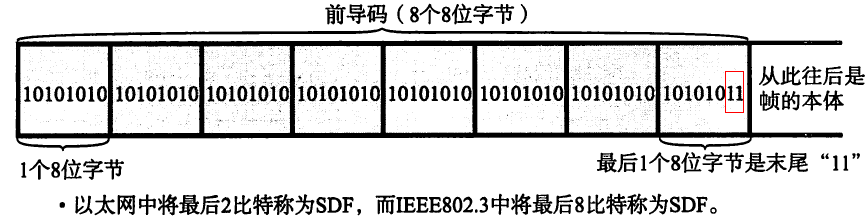
非共享介质网络中的每个站都直连交换机，通常采用全双工方式通信。

（4）以太网交换机就是有多个端口的网桥，根据数据帧中的目标MAC地址决定从哪个网络接口发送数据。交换机会在转发过程中自学MAC地址和接口的对应关系。

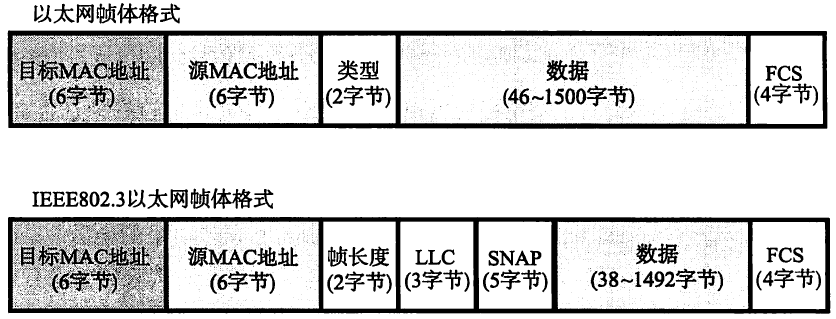
（5）交换机转发还可分为存储转发和直通转发。存储转发会检查数据帧末尾的FCS位后再转发。直通转发不会检查而直接发送。

**2.以太网帧格式**

（1）以太网帧前面会有1个前导码，有0、1交替组合而成，末尾是值为11的SFD。前导码后就是帧的本体。前导码也用于同步对方网卡。前导不是帧的组成部分。



（2）因历史原因，IEEE802.3和RFC894规定了2种不同的太网帧格式，如下：



因RFC中帧格式类型和802.3中帧格式长度值不会相同，因此可以进行区分。

（3）帧最小长度为64字节，数据长度最小为48字节。如果短于该长度，必须填充字节。规定最小长度的原因是为了载波侦听（以太网有最大物理线路长度规定，根据传播速度和最小帧大小，从而判断是否发生冲突）。

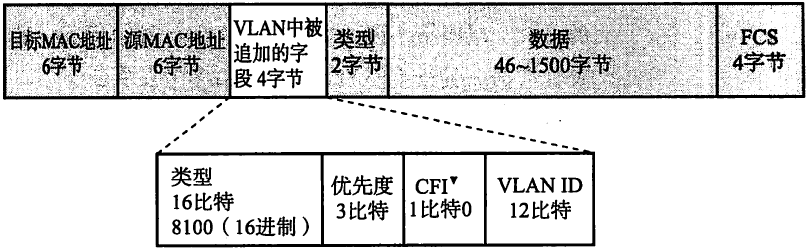
**3.虚拟局域网**

（1）虚拟局域网通过修改带有VLAN功能的交换机的网络结构，而不用实际修改网络布线，将1个局域网分割为多个不同的广播域。

（2）如果交换机配置了VLAN功能，在传输帧过程中，交换机会在以太网首部加入VID标签，并根据这个值决定将数据帧发送给那个网段。



（3）VLAN的以太网帧格式为（和前面的802.3以太网帧格式一样，类型为8100）：



**4.环路检测**

如果通过交换机/网桥连接的网络出现环路，数据帧可能在环路中一直被转发而造成网络瘫痪。解决环路的协议为生成树协议。

* 1. PPP

**1.概要**

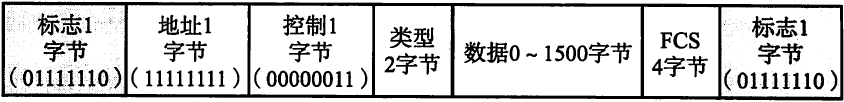
（1）PPP（Point-to-point Protocol）为点到点（1对1连接）信道间使用的协议。

（2）PPP协议可使用电话线、专线、ATM等作为连接物理层。

（3）PPP协议支持多种上层协议，提供身份认证、数据压缩、动态地址协商等功能，可用于用户与ISP通信等场景。

（4）PPP是一个协议集合，包括建立连接的基本方法链路控制协议（LCP），以及一系列网络层协议（NCP）。

**2.PPP协议帧格式**



（1）帧首尾都为1个字节的01111110标识码，用来区分帧的开始和结束。如果中间出现同样的字节码，需要转移。

（2）因为点到点连接，地址字段始终为0xFF，即所有地址。控制字段也始终为0x03。鉴于两个字段都是固定值，可以通过协商压缩。

（3）常见的类型字段有：0x0021：IP数据报文，0xC021：链路控制报文LCP，0xC023：PAP协议的认证报文，0x8021：NCP协议报文等。

（4）FCS为校验字段。

**3.工作过程**

（1）链路静止状态：当用户拨号接入ISP后，就建立了一条用户主机到ISP的物理连接。

（2）链路建立状态：使用LCP交换配置包，以便建立LCP连接。

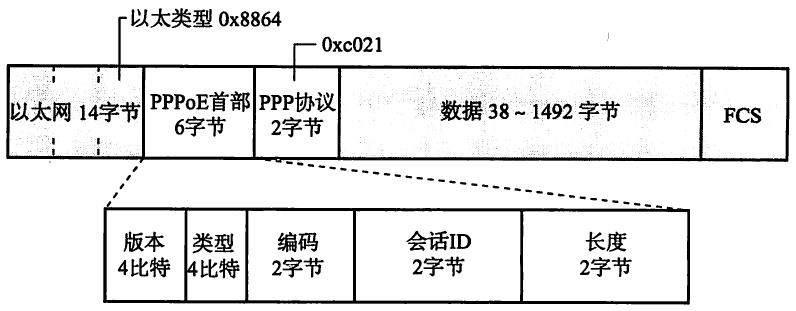
（3）认证阶段：可以使用PAP报文进行认证，也可以使用其他协议认证。

（4）网络层协议阶段：完成前面阶段后，就可以协商网络层配置（如一个临时的IP地址），传输网络层数据。



**4.PPPoE**

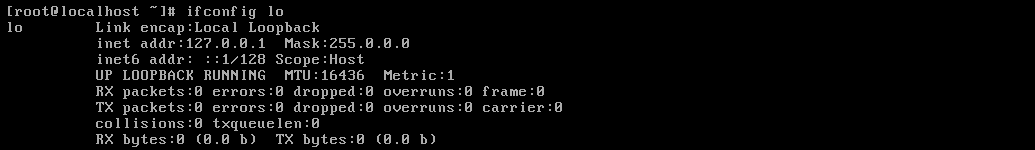
PPPoE可在以太网链路上提供PPP功能。尽管以太网不能提供认证等功能，但因以太网越来越普通，加之相关设备便宜，于是出现PPPoE技术。PPPoE帧仍然和以太网一样，只是帧类型为0x8864，并在以太网帧中携带PPP相关信息。



* 1. 环回

（1）用于客户端使用Internet协议（如TCP/IP）与同一台主机上的服务器通信。因此，操作系统提供了这样一个虚拟的环回网络接口实现该功能。

（2）linux系统中，环回接口被称为lo。



（3）如上，环回被分配的IP地址为127.0.0.1，子网掩码为255.0.0.0，是1个A类地址。

1. 网络层
   1. 概要

（1）网络层主要作用是实现端到端通信。由于端到端通信会跨越许多链路，因此需要进行路由选择。

（2）网络层主要有IP、ICMP、ARP、RARP等协议。

（3）网络层连接为无连接型，主要是为了简化和加速。如果需要连接，可以在上层实现。

* 1. IP地址

**1.概要**

（1）IP地址用于标识网络中主机通信地址。

（2）IPv4地址由32位二进制表示。为便于记忆，将其每8位转为10进制，在以.分隔。所以，每组地址最大值为255。

（3）一块网卡可以配置多个IP地址。

**2.最初的分类寻址**

最初将IP地址分为网络地址和主机地址，原因为当时还没有子网掩码，判断是否为同一链路就是通过网络地址。



（1）注意A类地址以0开头。

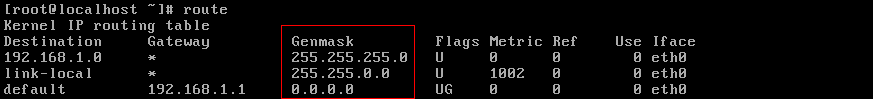
（2）主机地址不能全为0或1，前者表示地址不可知，后者表示广播地址。

**3.子网寻址**

（1）将IP地址分为A、B、C类导致地址浪费，因此后来不再使用网络地址、主机地址区分链路，而是使用子网掩码来区分。

（2）子网掩码网络部分全为1，主机部分全为0，通过和1个地址按位或，即可判断该地址的网络部分。

（3）使用子网掩码划分子网需要路由器层面支持，即无类别域间路由（CIDR）技术。无类别就是不再区分A、B、C类网络。路由表中也会有子网掩码，从而区分网络地址和主机地址。



CIDR使用斜线记法，即IP地址/网络前缀所占比特数。如172.1.1.1/26表示前26位是网络地址，后6位是主机地址。

（4）VLSM（可变长子网掩码）可以对子网进行层次化编址，以便最有效的利用现有的地址空间。如下，1个路由器地址为：，这个子网又可以划分为多个子网，划分的时候，不必等分划分，而是可以变长划分，如



反过来，这就是路由聚合，即，对外仍然为202.244.160.1/21这个路由地址，不会讲内部3个子网路由地址分发出去，从而有效降低路由表大小。

**4.全局地址和私有地址**

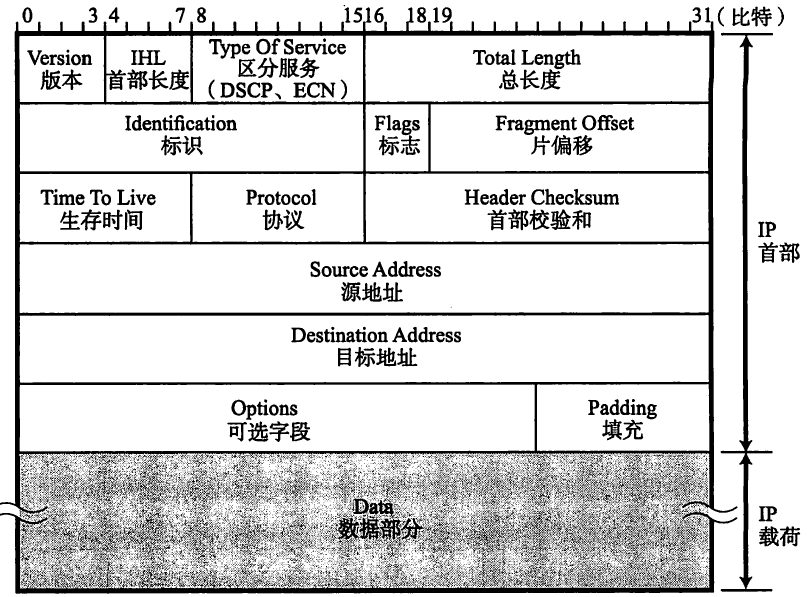
（1）A类、B类、C类的私有地址分别为10/8，172.16/12和192.168/16。

（2）私有IP地址最初用于互联网外的独立网络，后因NAT技术的出现，使得配置私有地址的主机通过地址转化可以和配置全局地址的主机实现通信。

* 1. IP协议

**1.IPv4首部**

当通过IP通信时，会在上层数据前面加入IP首部信息，其中IPv4首部信息如下：



（1）版本：4代表IPv4。

（2）首部长度：必须为4字节（32bit）的倍数。

（3）区分服务：没有任何软件支持。

（4）总长度：首部和数据部分的总长度，16bit，因此IP包最大长度为65535字节。使用总长度-首部长度就是数据长度。

（5）标识、标志、片偏移：用于分片重组。

（6）生存时间：即TTL。

（7）协议：即上层或本层协议，值可以为TCP/UDP、ICMP、IGMP等。

（8）首部校验和：校验首部。

（9）源地址、目标地址：IP地址。

（10）可选项：通常用于实验、军事等特殊。

（11）填充：因首部必须是4字节倍数，不足则填充。

**2.IP分片**

（1）IP数据包需要在链路上传输，每种链路层都有不同的最大传输单元（MTU），因此需要对IP数据包分片。

（2）整个传输路径最小的MTU被称为路径MTU。

（3）为了避免1个链路分片之后，因后面有更小的链路还再分片，需要先获取路径MTU。通过将IP首部分片标志设为禁止分片，如果链路不能处理将会将数据包丢弃并返回ICMP不可达消息，通过反复发送数据包，从而发现路径MTU。

（4）路由器虽然可以做分片，因为如果组装后面还可能需要分片。分片组装是在目的主机上进行。

* 1. 路由控制

**1.概要**

（1）用于指示下一跳路由或主机IP地址的表就是路由控制表。

（2）有2种方法生成路由表：手动设置或通过与其他路由交换信息自动生成，前者又称为静态路由，后者称为动态路由。

（3）用于生成路由表，控制路由器实现网络互联的协议就是路由协议。路由协议和IP协议不是一回事，但都在网络层工作。

（4）路由表中记录了网络地址和对应的下一跳路由器或主机地址。收到IP数据包时，会获取包首部中的目标地址，然后和路由表中的网络地址比对。比对原则为最长匹配原则，即优先主机地址，再网络地址，在默认地址。默认地址一般标记为0.0.0.0/0或default。

（5）静态路由配置工作会随着网络复杂度而急剧增加。动态路由会将自己已知的网络连接信息告知相邻的路由器，从而像接力一样依次传递，直到所有路由器都获知整个网络中的情况。此时路由表也制作完成。无论是静态路由还是动态路由，都不要只使用其中的一种，而是应该结合使用。

（6）内部制定有自己的路由策略，并对外部屏蔽内部网络细节，称为1个自治系统（AS）或路由选择域（RD）。AS内部的动态路由采用的协议称为域内路由协议，即IGP，又称为内部网关协议。AS之间的路由控制采用的协议称为域间路由协议，即EGP，又称为外部网关协议。

（7）由此，路由协议分为IGP和EGP两个层次，其中，IGP可使用RIP、RIP2、OSPF等协议。EGP可使用BGP协议。

（8）路由控制主要有2种算法：距离向量算法和链路状态算法。距离向量根据距离和方向决定目标位置，路由器之间互换目标网络方向及距离相关信息。每个路由器掌握的距离、方向等信息显然不同，因为距离目标网络不同。而使用链路状态算法的路由器必须保持一样的信息，即所有路由器保存的是整个网络拓扑结构。一般是从网络代理那里获取路由信息表，这一过程相当复杂。

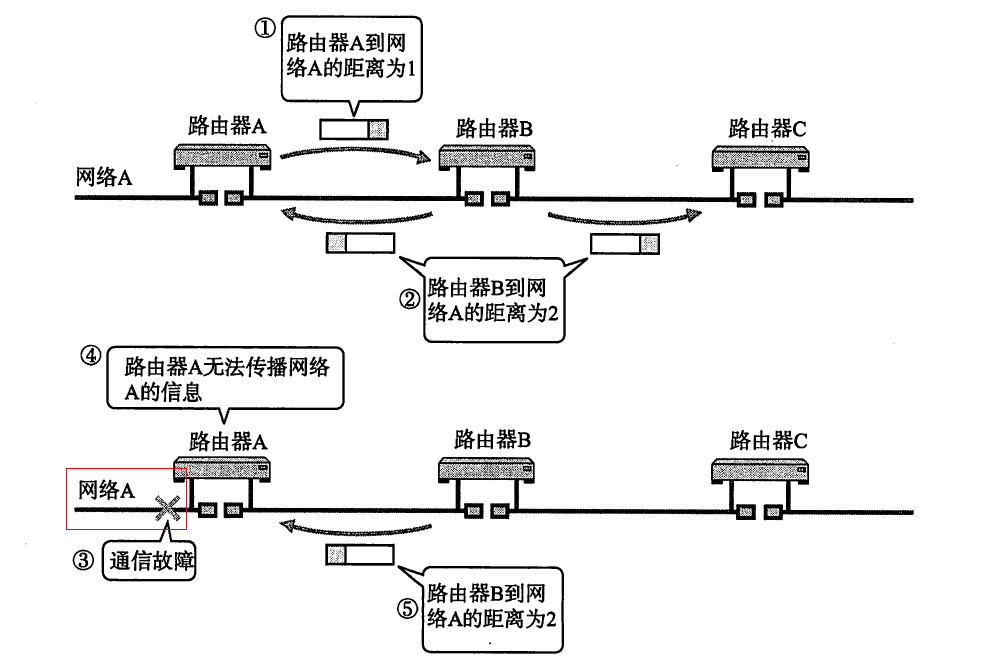
**2.RIP**

（1）基于距离向量算法，距离的单位为跳数。

（2）定期将路由信息发送给连接的路由器，相邻路由器获取后会将跳数+1。

（3）如果没有收到回复，连续5次视为断开。

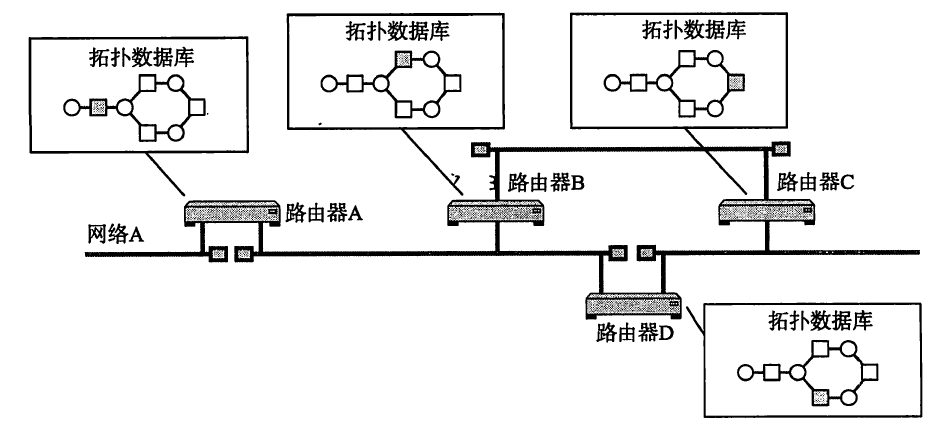
（4）可能存在一些问题，如网络A断开，但因路由器A曾经向路由器B发送过能连接网络A的信息，此时，路由B会将该信息返回给A，从而造成A认为B可以通过其他路径连接到A。解决方式是通常TTL为16，超过则丢弃包，返回无法送达消息。



RIP2是在RIP基础上，加入子网掩码、身份验证等功能。

**3.OSPF**

（1）使用链路状态型算法。路由器之间会交互链路状态，用于生成网络拓扑结构，进而生成路由表。



（2）OSPF会给每1条链路赋1个权重。从而判断出最小代价的路径。

**4.BGP**

（1）用于对整个互联网进行路由控制的协议。

（2）ISP、区域网络会将所辖网络划分为1个个AS，并给每个AS编号。BGP就是根据编号进行路由控制。

（3）BGP基于路径向量协议（不是距离向量算法）。路由器会保存对每个AS目标的路径信息访问列表，从而能够实现最短路由。

* 1. ARP协议

（1）ARP协议（地址解析协议）提供了一种根据IPv4地址到MAC地址的动态映射。

（2）ARP工作过程：同一链路上，源主机只有目标主机的IP地址，通过向所有链路上的主机广播发送1个ARP包，这些主机收到后如果自己是，会将自己的MAC地址放入ARP回应中返回源主机。

（3）无论是源主机还是目标主机（目标主机也会保存，因为返回数据也需要MAC地址），都有1个ARP信息表，当下次需要时就不用再次广播。缓存时间通常为20分钟。

（4）arp包格式与IP包不同。

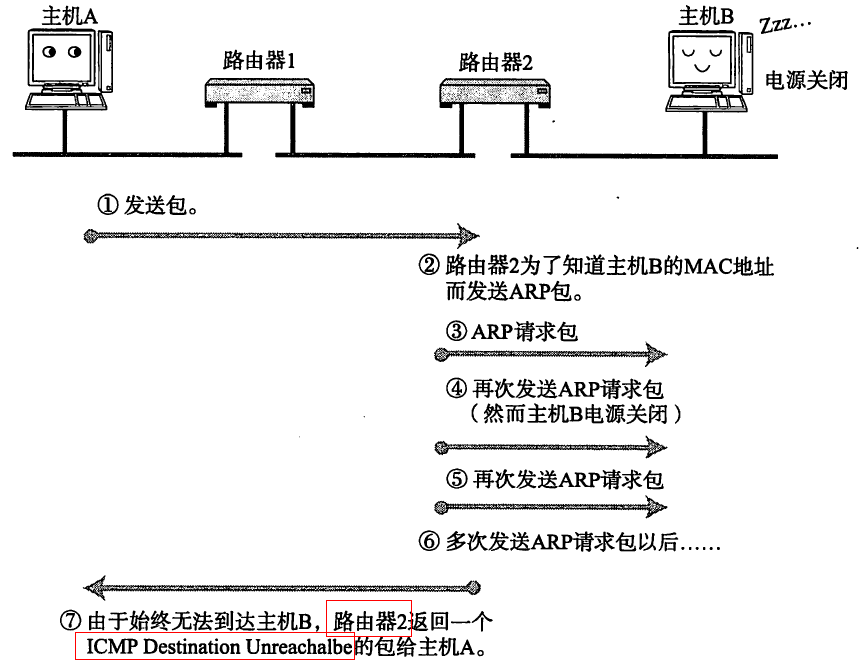
（5）可以使用arp命令获取ARP表。



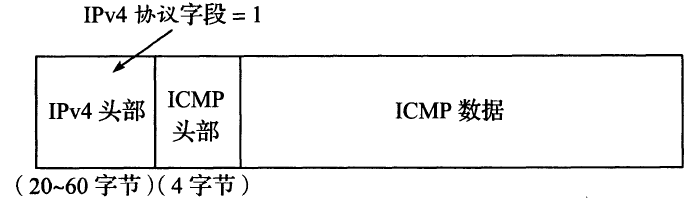
* 1. ICMP协议

（1）ICMP主要功能：提供IP层数据包处理或协议配置相关的诊断和控制信息。

（2）ICMP触发过程如下：



（3）ICMP通过IP数据报发送。



（4）ICMP消息可分为2类：通知出错原因的错误消息，和用于诊断的查询消息。



如果发送端主机选择次优路径发送数据，路由器将返回1个ICMP重定向（类型5），提示1个更合适的发送路由。

当TTL减为0，将发送1个超时（类型11）消息给发送端主机。

回送消息（类型0、8）用于判断通信的主机和路由器之间发送的数据包是否到达对方。ping命令就是通过这个消息实现。

类型9、10主要用于发现与自己相连网络中的路由器。

类型17、18主要用于了解主机或路由子网掩码情况，17为发送ICMP地址掩码请求消息，18为ICMP地址掩码应答消息。

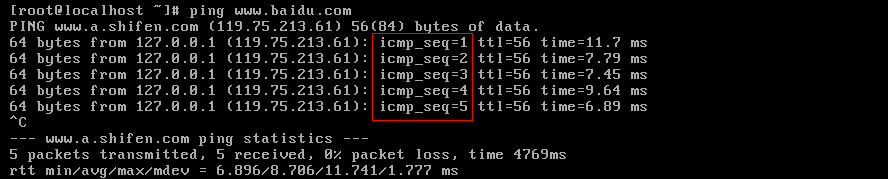
* 1. ping程序

（1）ping程序通过程序发送一份ICMP回显请求报文给主机，并等待返回ICMP回显应答，由此确定两台主机/路由之间是否连通、时延等信息。

（2）ping使用的ICMP报文格式如下：



类型为0或8。标识符为进程id，这样即使同时使用多个ping命令也能够区分。序号从0开始，每发送1次回显就加1。



（3）ping程序通过在报文数据中存放发送请求的时间来计算往返时间。

（4）linux中ping命令选项：-c数值：执行ping的次数；-s 数值：发送出去的ICMP封包大小；-t 数值：TTL的数值，预设是255；-W数值：等待回应对方主机的秒数。

* 1. traceroute程序

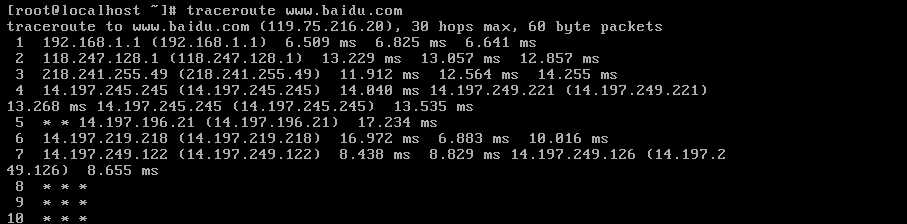
（1）用于获取IP数据包在网络中传输时经过的路由信息。

（2）为什么不是用ping而需要另开发1个程序？原因是：一些路由器配置不支持记录路由；记录路由需要单向，但ping从发送到接收会经过路由两次造成重复等原因。

（3）traceroute原理：发送一个TTL相当小的包，因TTL经过每一跳时会递减。当它减为0时，数据包就被丢弃。当TTL失效后，丢弃的路由器就会返回一个超时的ICMP。如果返回的路由器就是最终的目的地，停止trace。否则，TTL加1并返回到步骤1。

（4）因路由选择是动态的，特别是在广域网中，因此并不能途径的路由是同一个。

（5）如果路由关闭ICMP的回应或超时，则traceroute不会获得路由信息，会显示\*（但仍然会继续，TTL+1，trace下一个路由）。



（6）linux中traceroute命令选项：-n：不必解析主机名称，速度较快；-U：使用UDP侦测，端口预设为33434；-T：使用TCP侦测，端口预设为80；-I：使用ICMP方式侦测；-p：设置端口号；-w：若对方主机在几秒钟内没有回应就放弃，预设值是5秒。

1. 传输层
   1. 概要

（1）传输层主要使用的协议为TCP和UDP两个协议。TCP用于可靠传输，UDP主要用于广播和细节控制交给应用控制的通信。

（2）传输层使用端口号识别上一层应用。

（3）端口号确定方法：1）标准既定的端口号：一些广泛使用的应用协议使用的端口号为知名端口号，为0~1023之间，1024~49151之间的端口号也可以被正式注册；2）动态（时序）分配的端口号：主要用于分配客户端的端口号，取值范围在49152~65535之间。

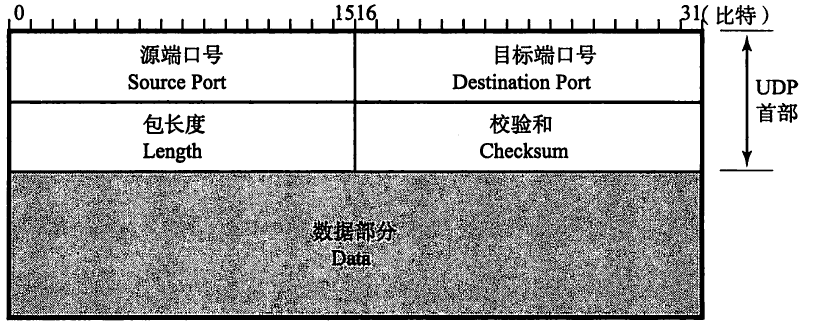
（4）不同传输协议可以使用相同的端口号。因IP层中会标识传输协议类型，然后分别发送给TCP或UDP，因此即使端口号相同也不会有影响。如53端口，TCP和UDP都被使用作为标识DNS服务。

* 1. UDP
     1. 概要

（1）UDP不提供复杂的控制机制。当收到应用层发来的数据后，就立即原样发送出去。即使网络拥堵，出现丢包情况，UDP也不负责重发。

（2）UDP主要用于如：包总量较少的通信，视频、音频等多媒体通信，广播通信，限定于LAN（LAN中用户通信，仍然会走上层协议）等特定网络中的通信。

（3）UDP首部如下：



包长度包括首部和数据部分。

* + 1. 单播、广播、组播

**1.概要**

（1）单播是对特定主机传输数据。广播是对某1个网络中的所有主机传输数据。组播是对一些特定主机发送数据（跟是不是1个网络没有关系，因为是特定地址，没有网络划分的概念）。

（2）主机地址的IP地址全为1位广播地址。IP地址以1110开头的地址为组播地址。除此之外的IP地址为单播地址。

（3）广播分为受限广播和定向广播。受限广播IP地址为255.255.255.255，任何路由器都不会转发，只能在本地使用。定向广播则由路由器配置决定。广播应该谨慎使用。

（4）应用层只能使用UDP发送广播、组播。

**2.主机对信道传来帧的处理过程**

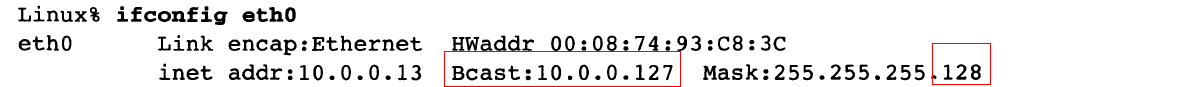
（1）网卡判断帧的目的地址是否是本机地址或广播地址，是才接收并转给网卡驱动程序。

（2）驱动程序过滤帧，如检查帧中是否有上层协议（IP、ARP等）等，检查没问题再传送给上层协议。

（3）上层协议（如IP）进行本层检查，没问题再传送给上层。

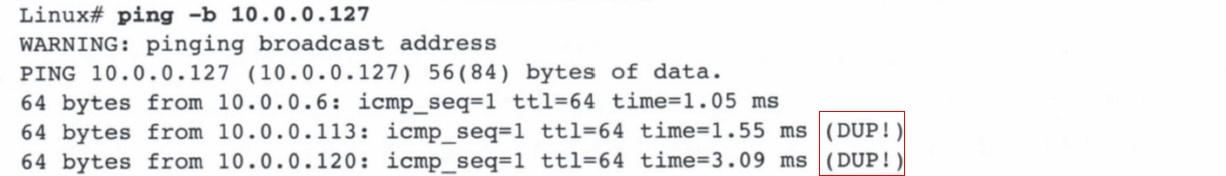
（4）上层协议（如UDP）检查端口，如没有进程使用该端口号，丢弃数据并生成1个ICMP报文。

**3.广播实例**

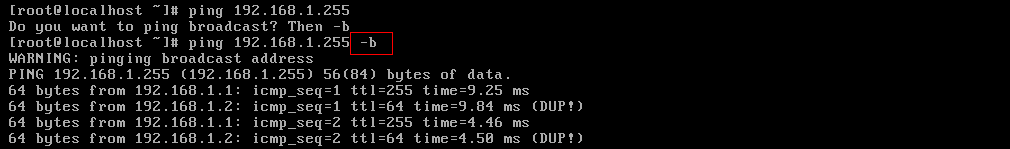


如上，广播地址不一定是255，但主机地址必须是全1。

使用ping该广播地址，将会受到多个相应（会用DUP!标示）。

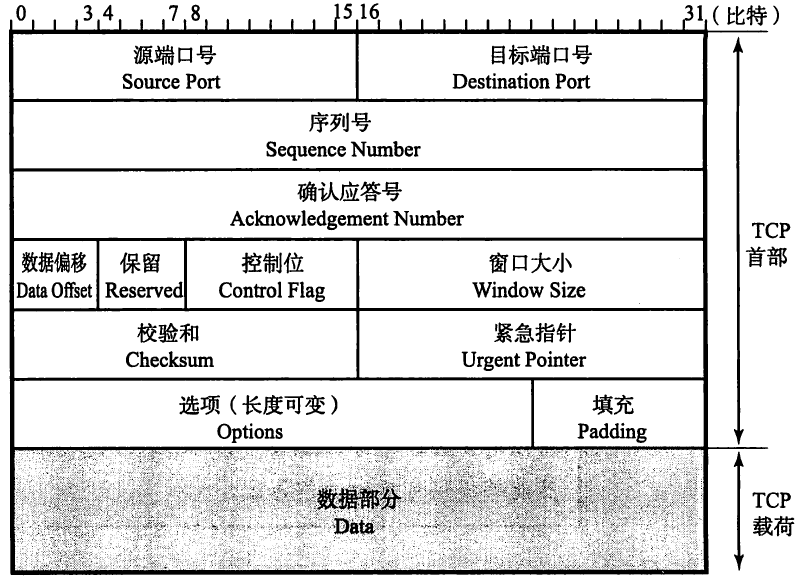


如果ping一个广播地址，会要求使用选项b。



* 1. TCP

**1.首部格式**



（1）序列号：发送数据的位置。每发送1次数据，就添加该数据字节数长度。字段长度为32位。注意，初始值并非0或1，而是在建立连接的时候生成的1个随机数，通过SYN包告诉给接收端，从而可以计算真正的长度。

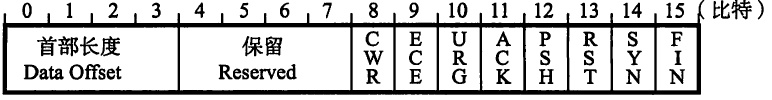
（2）确认应答号：下次应该收到的包中序列号的值。发送端收到这个值可以判断之前的数据是否都正常接收。

TCP中，当接收端收到数据后，会返回1个收到数据的通知，这个消息即应答（ACK）。TCP通过ACK实现可靠传输。如果收到肯定的应答，代表数据已成功到达对方。如果一定时间内没有收到应答，视为丢包，会重新发送。

如果数据已收到，但ACK因为超时导致重发，会使得接收端重复收到数据。为管理这些情况，需要使用序列号和确认应答号来进行控制。

（3）数据偏移：即TCP首部长度，表示数据部分从哪里开始。

（4）控制位：从左到右意义如下：



CWR和BCE：用于IP首部ECN字段，用于拥塞控制。

URG：如果值为1表示需要紧急处理。

ACK：值为1表示确认应答字段有效。根据规定，除最初建立连接时的SYN包之外的包，该位必须都为1。

PSH：值为1表示应立即传给上层协议，否则可以缓存。

RST：值为1时表示如果连接出现异常，应立即断开连接。

SYN：用于建立连接，值为1时表示希望建立连接，并设置序列号字段值。

FIN：值为1时表示希望断开连接。

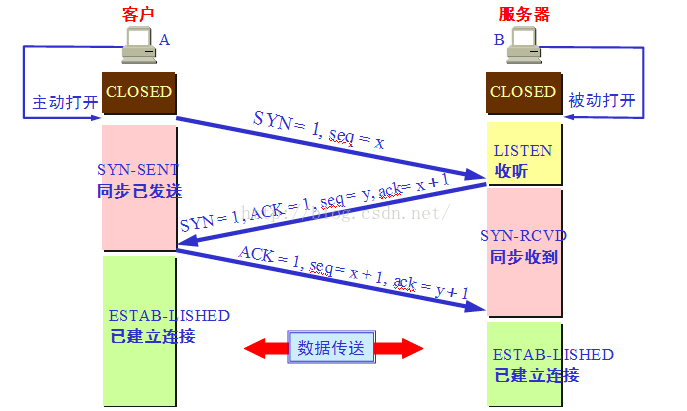
（5）窗口大小：用于窗口控制。

（6）选项：用于提高TCP性能的一些设定。

**2.面向连接**

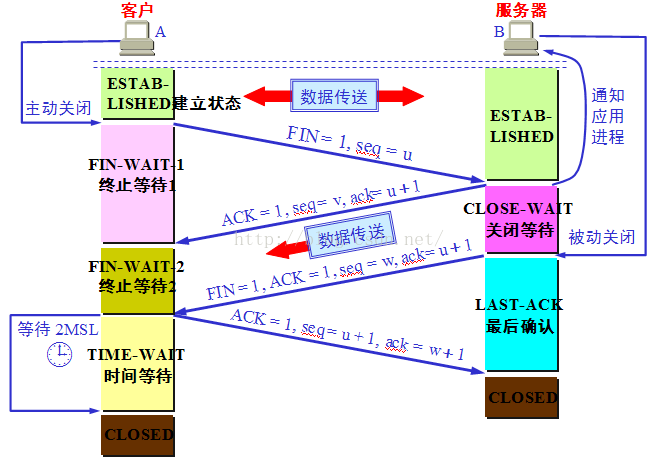
（1）面向连接是软件层面，物理层面肯定是连接的，要不没法通信。所谓面向连接，就是soket在连接建立后不会断开，除非要求断开。而面向无连接则是处理完即断开连接。

（2）3次握手：TCP建立连接时，会发送1个SYN包作为建立连接的请求等待应答。对方应答时ACK控制位为1。请求方在收到应答后还是针对该应答再次应答。注意这个过程还会交流序列号，序列号以服务端的为准。



之所以要3次握手，是为了防止已失效的连接请求发送到服务器端。如，A发送的请求延迟，于是A会再次发送。如果第2次发送正常，建立连接后，服务端收到了第1次的请求，服务端会误认为这是1个新的连接。但如果再确认1次，服务端会针对这次延迟的请求确认，请求端知道这不是新连接，会丢弃回应。

（3）4次挥手：断开连接会发送FIN包，过程如下：



之所以要4次挥手，原因是：1）等待最后1个报文回复到达服务器端，因为这个报文可能会延迟。2）防止一些延迟的报文影响下1次连接。

**3.可靠连接**

（1）可靠连接通过序列号和确认应答号实现。

（2）如果出现超时，会重新发送。超时时间会根据往返时间+超时偏差动态调整。

**4.其他**

（1）在建立连接的时候，就会协商最大包长度。通常为IP包不会分片的长度。

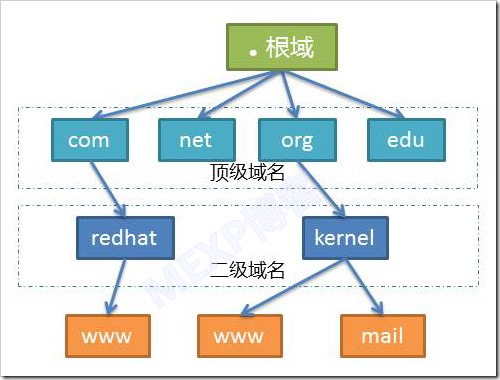
（2）可以使用窗口管理实现更快数据传输。

（3）还会通过流控制防止客户端出现丢弃。通过拥塞控制，防止传输过程中出现丢弃。

1. 应用层
   1. DNS
      1. 服务器设置

**1.概念**

（1）DNS（Domain Name System）：即域名系统，它使用层次结构的命名系统，将域名和IP地址相互映射，形成一个分布式数据库系统。

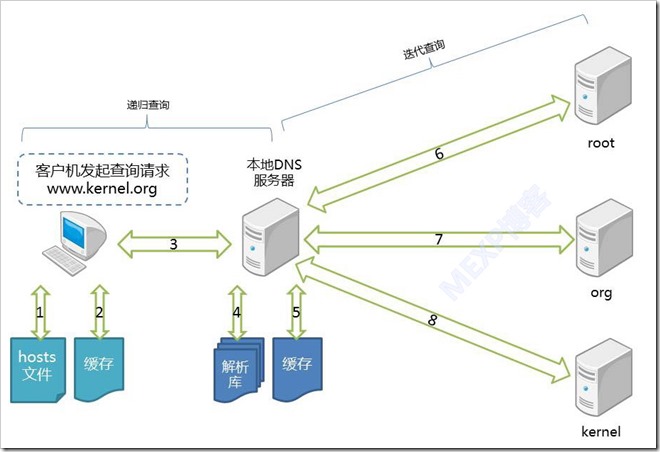


（2）FQDN（Fully Qualified Domain Name）：完全限定域名，是使用DNS的树状层级结构的完全路径域名来表示一个准确位置对应的主机，如www.kernel.org.就是1个FQDN。

（3）正解：从主机名到IP地址。反解：从IP地址到主机名。Zone：每个域名的记录。

反解主要用于垃圾邮件防护。邮件服务提供商需要通过反解进一步查询邮件发送方信息。反解需要固定公网IP地址和可用域名。能够反解的只有IP地址拥有者（ISP），需要申请。

**2.DNS访问模型**



客户端发起对www.kernel.org的解析请求，会经过如下过程：

（1）先检查本地hosts文件。

（2）检查本地缓存。

（3）向配置的DNS服务器（上图是本地DNS服务器）请求。

（4）服务器检查域名是否能够本地解析（zone）。

（5）服务器检查本地缓存。

（6）服务器向根服务器请求，获取org域服务器地址列表。

（7）服务器使用org域服务器IP发起DNS请求，获取kernel域服务器地址列表。

（8）服务器使用kernel域服务器IP发起DNS请求，获取www.kernel.org主机的IP地址。收到后缓存，并返回地址给客户端。

以上客户机和本地DNS服务器直接的查询方式，称为递归查询。本地DNS服务器多次重复查询的方式，称为迭代查询。

**3.软件安装**

（1）Bind是1款使用最为广泛的开放源码的DNS服务器软件。

（2）CentOS6.5中使用yum安装貌似没有chroot。

（3）默认配置文件在/var/named/目录下，需要将文件属组改为named。



bind-utils是bind提供的一组DNS工具包，内有一些DNS相关工具，如dig、host、nslookup、nsupdate等。

**4.本地配置**

（1）主配置文件/etc/named.conf

与服务器环境相关的配置在options中，包括：

listen-on：监听那个接口、网卡地址，设置为any表示全部接受；

directory：默认配置文件目录，后面的zone中的file就在目录下；

allow-query：允许那些主机来查询，设置为any表示全部接受；

recursion：递归查询，即当服务器自身不能解析时,会向其他DNS服务器发出查询请求，得到结果后转交给客户机，默认值为yes，需要保证服务器地址不出现在客户机的/etc/resolv.conf中；



include引入的文件，和这里的.zone都表示本地域名解析记录。

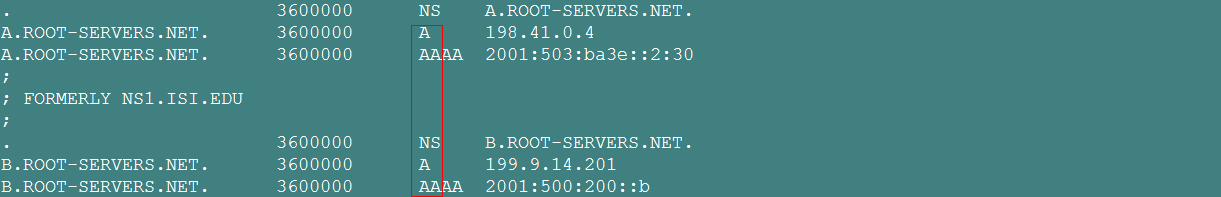


（2）/etc/named.rfc1912.zones

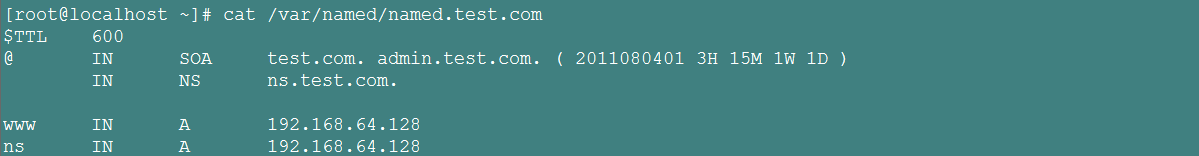


（3）/var/named/named.ca，以及个人测试用的文件named.test.com

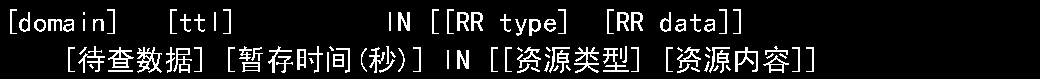
named.ca中是.根域名对应的服务器记录。



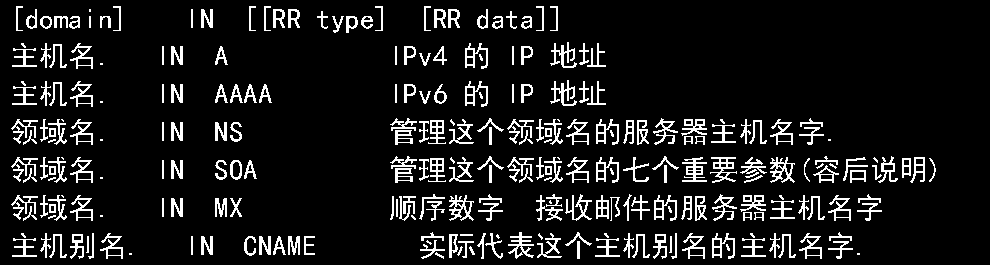
如上，主机记录可以对应多条解析结果，可以制定轮询方案。



（4）正解配置记录格式



domain应尽量使用FQDN。TTL表示保存到查询DNS服务器多长时间，可以由特定参数统一管理，因此该字段可以忽略。关键字IN是固定的。常用的RR记录格式如下：



RR type和RR data可以为：

SOA：zone授权起始记录，区域文件第一条记录，而且一个区域文件只能有一条。有7个字段：master，管理员email，序号，更新频率，失败重新尝试时间，失效时间，快取时间。master和管理员email貌似可以随便写。

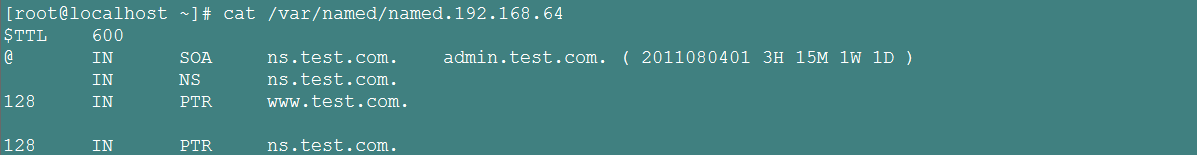
A和AAAA：对应被查询的IP记录。

NS：对应管理zone的DNS服务器主机名（如果向上级申请，这里就是申请服务器主机名）。

CNAME：别名。当IP地址用于多台主机使用时，如果地址变更，逐个修改会比较麻烦，而使用别名可以只修改1次。

MX：域名对应的邮件服务器主机名。服务器前面的数字表示有几台服务器收发邮件。

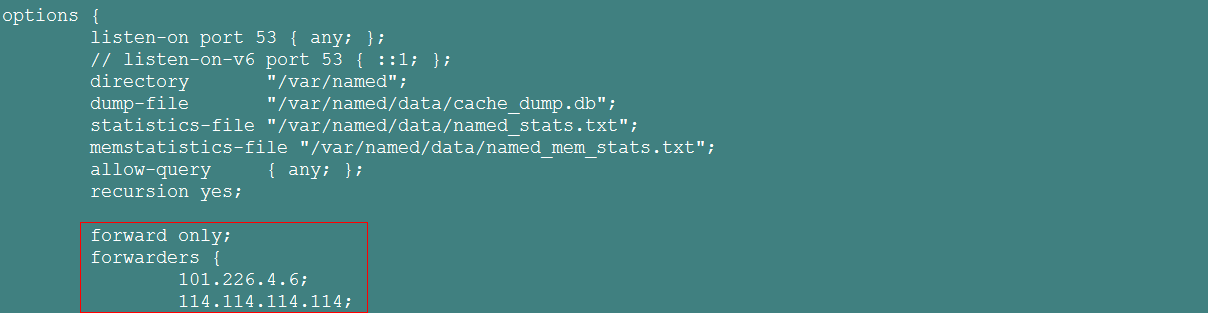
（5）反解记录和自用反解记录文件/var/named/named.192.168.64



无论是正解还是反解记录，@均代表zone，即在named.conf中的那个zone。如上，反解中前面的数字表示主机地址。

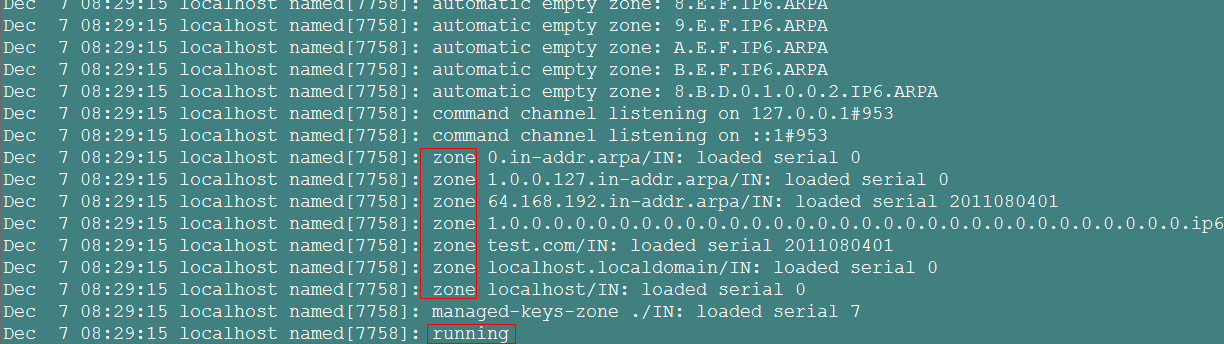
**5.forwarder转发**

即将本地DNS服务器设置为转发服务器。所有非本域的和在缓存中无法找到的域名查询都将转发到设置的DNS转发器上，由这台DNS来完成解析工作并做缓存。



**6./var/log/message**

可以在这里查询服务器启用失败等信息。



**7.其他**

（1）上面的配置只能本地使用，如果想要在外网访问服务器，就需要向域名上级申请解析。申请时填写DNS服务器域名和IP地址。

也可以直接向上层申请域名解析而不使用服务器。

（2）申请DNS服务器时，通常需要填写master/slave架构至少两台DNS服务器地址。

* + 1. 客户端设置

**1.配置文件**

（1）/etc/hosts：最早用来记录hostname和IP地址映射关系的文件。

（2）/etc/resolv.conf：记录DNS服务器IP。

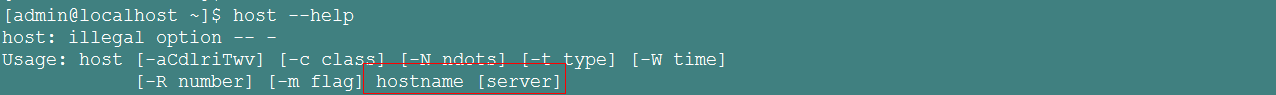


DNS服务器可以设置多台。当网络正常情况，只会使用第一台。如果使用DHCP，需要设置PEERDNS=no，否则会自动将/etc/resolv.conf文件重置。

（3）/etc/nsswitch.conf：记录优先使用（1）还是（2）配置。

**2.相关命令**

（1）host

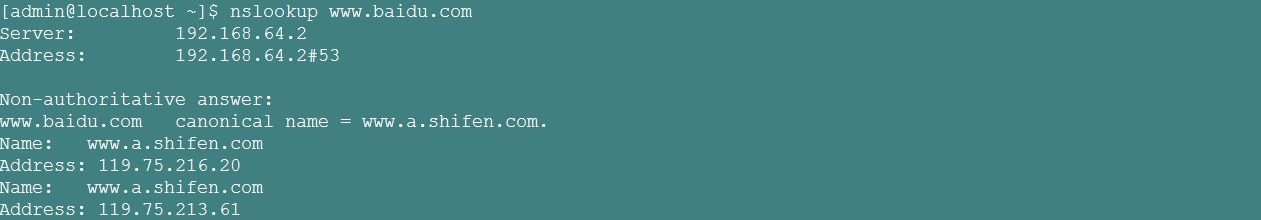


用于查询1个域名相关信息。-a查询所有信息。-l查询allow-transfer，通常用于服务器设置时管理，客户端用不到。server是另外指定DNS服务器查询。



（2）nslookup

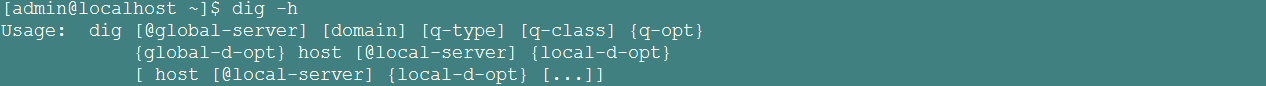
可以直接查询正解和反解。





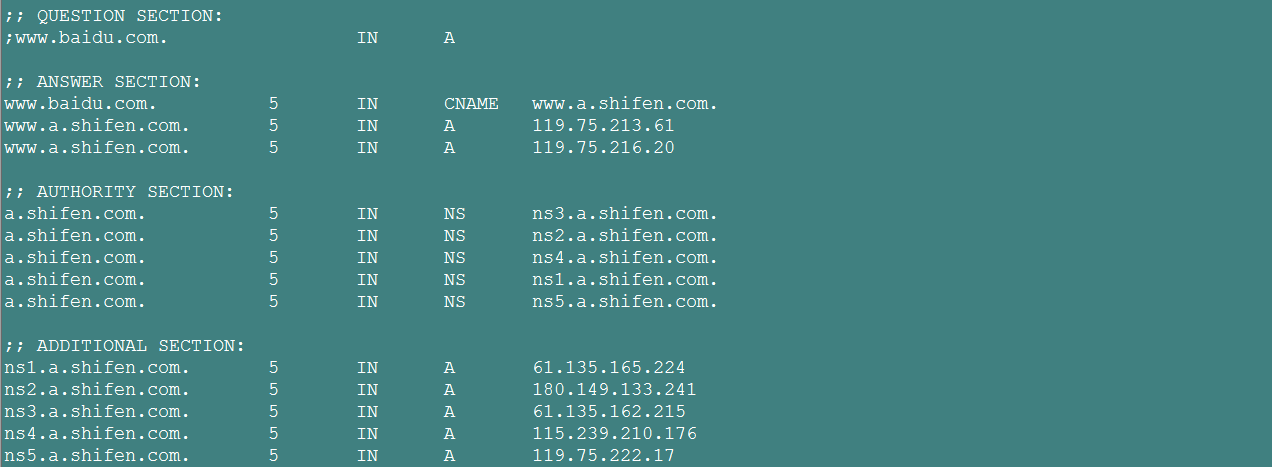
上面部分为DNS服务器信息。

（3）dig



选项有：+trace就是从.开始追踪。-t type查询mx、ns等类型。-x查询反解信息。





question为查询内容，answer为查询结果，AUTHORITY验证。

（4）whois

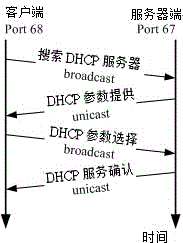
查询域名管理者信息。如果没有安装，可使用yum安装jwhois。

* 1. DHCP
     1. DHCP运作的原理

DHCP（Dynamic Host Configuration Protocol）：自动的将网络参数正确分配给网域中的每部计算机，让客户端的计算机可以在开机的时候就立即自动的设定好网络的参数值。这些参数值可以包括了IP、netmask、network、gateway与DNS地址等。

**1.运作方式**

DHCP通常是用于局域网络内的一个通讯协，通过由客户端传送广播封包给整个物理网段内的所有主机，若局域网络内有DHCP服务器时，才会响应客户端的IP参数要求。所以DHCP服务器与客户端是应该要在同一个物理网段内的。客户端取得IP参数过程可以简化如下：



（1）客户端：利用广播封包发送搜索DHCP服务器的封包

若客户端网络设定使用DHCP协议取得IP，则当客户端开机或者是重新启动网络卡时，客户端主机会发送出搜寻DHCP服务器的UDP封包给所有物理网段内的计算机。此封包的目标IP会是255.255.255.255。一般主机接收到这个封包后会直接丢弃，但若局域网络内有DHCP服务器时，则会开始后续行为。

（2）服务器端：提供客户端网络相关的租约以供选择

DHCP服务器在接收到这个客户端的要求后，会针对这个客户端的硬件地址与本身的设定数据来进行下列工作：

1）服务器检查配置文件，查看该用户之前是否曾经用过某个IP，如果有且该IP目前无人使用，则提供此IP给客户端。

2）若配置文件针对该MAC提供额外的固定IP时，则提供该固定IP给客户端。

3）若不符合上述两个条件，则随机取用目前没有被使用的IP给客户端，并记录下来，并设定租期。

（3）客户端：决定选择的DHCP服务器提供的网络参数租约并回报服务器

如果局域网络不止1台DHCP服务器，当决定好使用1台服务器的网络参数租约后，客户端便开始使用这组网络参数来设定自己的网络环境。此外，客户端也会发送一个广播封包给所有物理网段内的主机，告知已经接受该服务器的租约。此时若有第2台以上的DHCP服务器，则这些没有被接受的服务器会收回该IP租约。

（4）服务器端：记录该次租约行为并回报客户端已确认的响应封包信息

当服务器端收到客户端的确认选择后，服务器会回传确认的响应封包，并且告知客户端这个网络参数租约的期限，开始租约计时。出现如下情况服务端会回收ip：

1）客户端脱机：关闭网络接口（ifdown）、重新启动（reboot）、关机（shutdown）等，Server端会将该IP回收，并放到Server自己的备用区中，等待未来的使用。

2）客户端租约到期：当客户端IP到达期限规定的时间，而且没有重新提出DHCP的申请时，就会将IP收回，会造成断线，但用户会再向DHCP服务器要求再次分配IP（自动）。

**2.固定或动态**

可以设定DHCP服务器给予客户端的IP参数有两种方式：

（1）固定（Static）IP：DHCP可以根据MAC来给予固定的IP参数租约，通常用于作为服务器的主机设置。

（2）动态（dynamic）IP：Client端每次连上DHCP服务器所取得的IP不是固定的。

* + 1. 服务器端设置

如果没有安装，可以使用yum安装dhcp即可。命令位置为/usr/sbin/dhcpd，配置文件为/etc/dhcp/dhcpd.conf（centos6）或/etc/dhcpd.conf（centos5）。租约文件位置为/var/lib/dhcp/dhcpd.leases。dhcpd.conf中的设定主要分为2项：服务器运作的整体设定和IP设定模式（动态或固定）。

**1.dhcpd.conf的语法格式**

（1）#为批注符号。

（2）除了右括号）后面之外，其他的每一行设定最后都要以;做为结尾。

（3）设定项目语法主要为：<参数代号><设定内容>，例如：default-lease-time 259200。

（4）某些设定项目必须以option来设定，基本方式为option <参数代码> <设定内容>，例如：option domain-name "your.domain.name"。

**2.整体设定**

假设dhcpd只管理一个网段，那么除了IP之外的许多网络参数就可以放在整体设定的区域中，相关参数有：

（1）default-lease-time时间：默认租约时间，时间参数默认单位为秒。

（2）max-lease-time时间：规范使用者所能要求的最大租约时间。也就是说，使用者要求的租约时间若超过此设定值，则以此值为准。

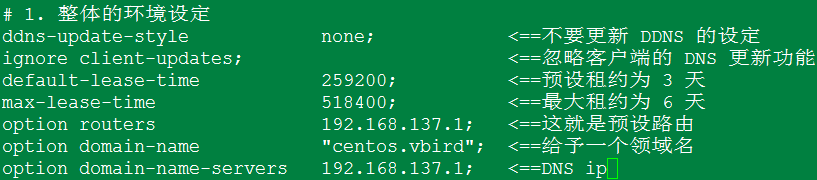
（3）option domain-name "域名"：如果在/etc/resolv.conf里设定了一个search google.com的话，这表示当你要搜寻主机名时，DNS系统会主动帮你加上这个域名。

（4）option domain-name-servers IP1,IP2：可以修改客户端的/etc/resolv.conf中nameserver后面接的那个DNS IP。

（5）ddns-update-style：因为DHCP客户端所取得的IP通常是一直变动的，所以某部主机的主机名与IP的对应就很难处理。此时DHCP可以通过ddns来更新主机名与IP的对应。

（6）ignore client-updates：与ddns-update-style设置相关，客户端可以通过dhcpd服务器来更新DNS，通常设定为ignore（忽略）。

（7）option routers路由器地址：设定客户端预设路由器的IP。

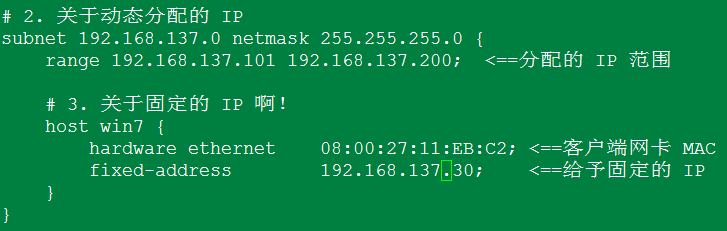


**3.IP设定模式（动态或固定）**

subnet NETWORK\_IP netmask NETMASK\_IP {...}，括号内的参数为：

（1）range IP1 IP2：ip区间。

（2）host主机名 {...}：host就是指定固定IP对应到固定MAC的设定值。如果不指定主机名，则大括号内就得要指定MAC与固定的IP，使用hardware ethernet硬件地址或fixed-address IP地址。

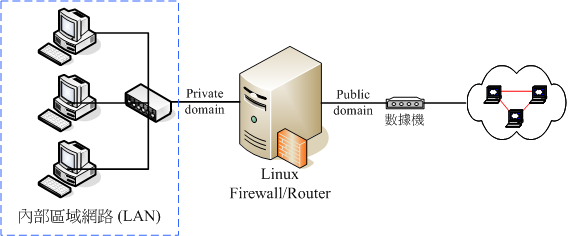


* 1. 防火墙
     1. 概要

防火墙是位于内部网络与外部网络之间的网络安全系统。防火墙依照特定的规则，允许或限制传输数据的通过。防火墙有网络层防火墙和应用层防火墙两类。网络层防火墙对应于OSI模型的3、4层。应用层防火墙其实是代理层的网关。防火墙不能很有效的抵挡病毒或木马程序，也难以抵御来自内部LAN的攻击。linux系统下的防火墙工具主要有Netfilter和TCP Wrappers两种。

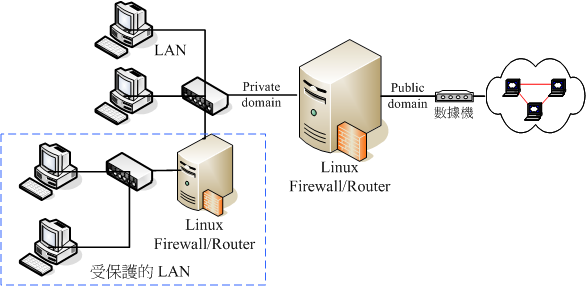
linux防火墙一般网路布线有以下几种：

（1）单一网域，仅有一个路由器

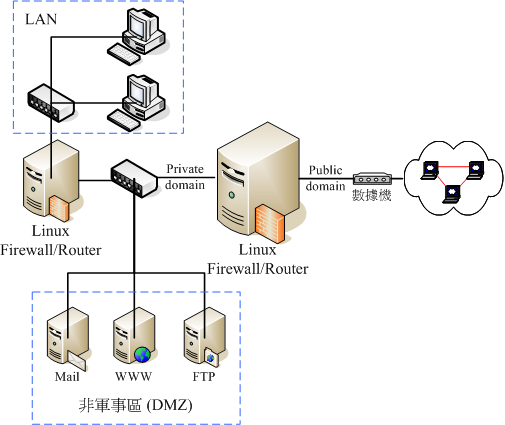


架设在路由器上，防火墙除了保护本机外，还可以的基本防护之外，还可以管控整个局域网的封包进出。优点：1）内外网域分开，安全维护在内部可以开放的权限较大；2）对外只能看到Linux防火墙主机，对内部可以达到有效的安全防护。

（2）用内部防火墙分隔需要更高安全性的子网，在LAN里再加设一个防火墙。



（3）在防火墙内架设网络服务器



优点：1）多台网络服务器的公网ip一样；2）网络服务器架设在两部服务器之间，即使内部网络出现问题也不会影响到服务器。

* + 1. TCP Wrappers

**1.工作原理**

通过TCP的守护进程tcpd管理。当有连接请求时，tcpd会截获请求，检查请求是否符合/etc/hosts.allow和/etc/hosts.deny配置规则来决定是否放行。过滤的配置规则主要针对应用名称。

只有由super daemon（xinetd）管理，且程序依赖libwrap.so模组的服务支持TCP Wrappers。这样的服务有sendmail、pop3、imap、sshd、telnet等。检查是否支持libwrap.so模组方法如下：



**2.配置规则**

首先先比对/etc/hosts.allow，符合规则就予放行。如果没有对应规则，再比对/etc/hosts.deny，符合规则就予以拒绝。如果两个文件均没有对应规则，都予放行。

/etc/hosts.allow和/etc/hosts.deny规则语法一样，如下：





可以使用关键字ALL、ALL EXCEPT，表示所有服务、ip。



* + 1. Netfilter

Netfilter是集成在Linux内核中的一个数据包处理模块，能够提供数据包过滤的防火墙功能，网络地址转换（NAT），以及数据包内容修改等功能。iptables位于/sbin/iptables，是Netfilter提供的编辑防火墙规则的命令工具。

**1.iptables的工作机制（4表5链）**

（1）规则

规则就是网络管理员预定义的条件。防火墙的作用就是比对数据报文是否匹配这些规则，然后执行对应的动作。匹配条件分为基本匹配条件与扩展匹配条件。基本匹配条件如Source IP、Destination IP等。扩展匹配条件如Source Port、Destination Port等。处理动作有：

|  |  |
| --- | --- |
| ACCEPT | 放行 |
| DROP | 丢弃 |
| REJECT | 拒绝，和丢弃类似，但会通知源主机 |
| LOG | 记录日志信息，并传递给下一条规则处理 |
| 其他 |  |

（2）表

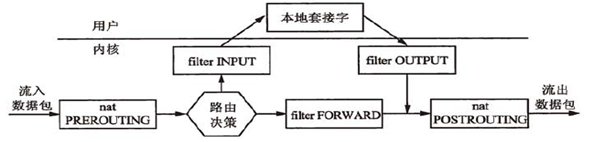
将具有共同功能的规则分类，就是表。不同功能的规则可以放置在不同的表中进行管理。iptables一共有4种表，4表按优先级排序如下：

|  |  |
| --- | --- |
| raw | 是否对该数据包进行状态跟踪，优先级最高。 |
| mangle | 拆解报文，做出修改，并重新封装（基本上用不着）。 |
| nat | 网络地址转换功能，可修改数据包中的源、目标ip地址和端口。 |
| filter | 是否放行该数据包（过滤），即防火墙功能。 |

（3）链

Netfilter在数据包传输路径必经的5个位置设置Hook进行监控。监控比对的规则可能有多条，会按顺序比对。将这些规则串到一个链条上的时候，就是链。如果数据包不符合链中任何规则，则按链的默认策略来处理数据包。

Netfilter中一共有5种链，如下图所示：

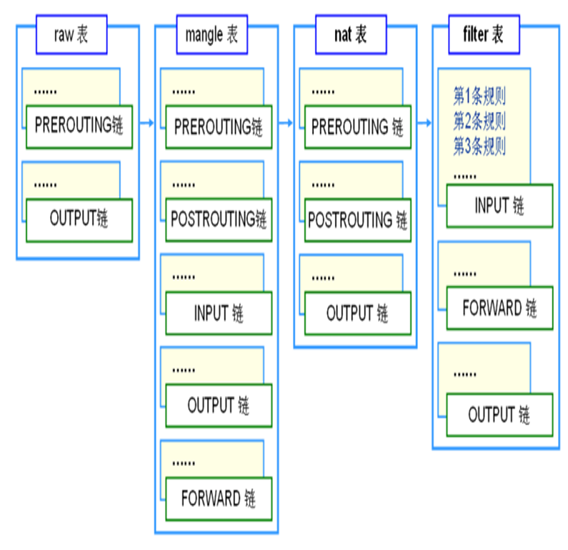


其中，PREROUTING链、FORWARD链、POSTROUTING链需要开启主机路由功能才会有。

设置防火墙时，要特别注意的是许多协议都需要双向通信，所以需要同时配置输入链和输出链。

（4）表和链的关系

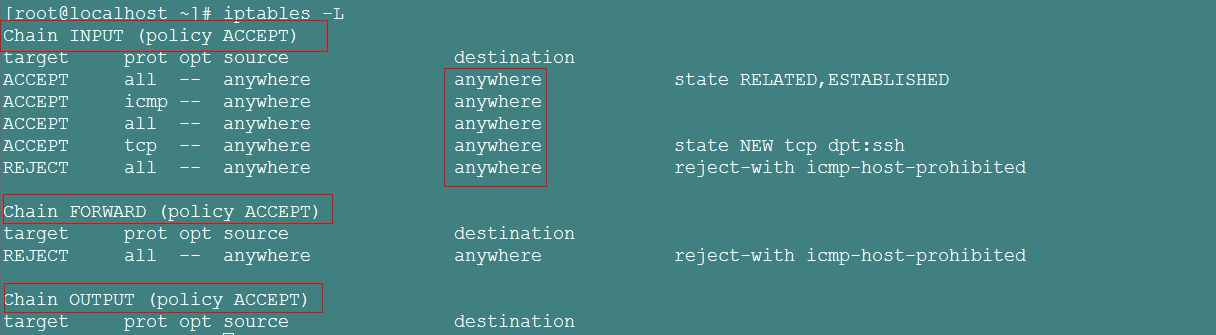
因功能目标不同，有的链不需要一些表的功能，即一些表只能存在某些链中。在实际使用中，往往是通过表作为操作入口。iptables中表和链的对应关系如下：

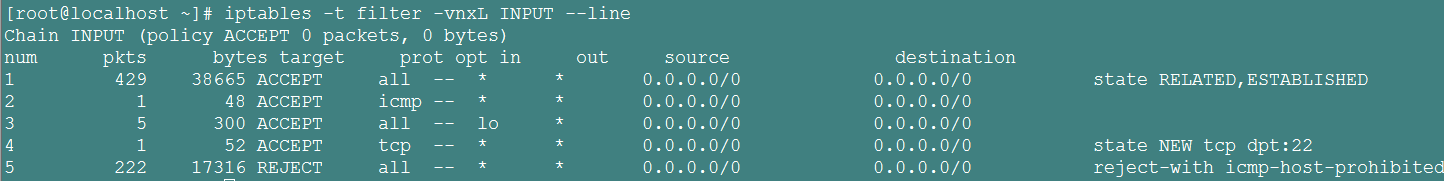


只有mangle表可以用在所有的链中，但它实际上基本用不着。fiter表负责防火墙功能的，出现在了INPUT、OUTPUT链中，还出现在了FORWARD链中。

**2.iptables命令基本语法**

（1）查看规则





-t：指定表名，如果省略，默认为filter。-L：查看链中的规则，如果后面不指定链，则显示所有的链。

-v：显示详细信息。-x：显示更精确的数字（如上bytes中显示为28229而不是28K）。-n：不解析ip地址为主机名。--line：显示行号。

显示的结果按链分类。其中的栏目含义为：pkts：规则匹配到的报文个数。bytes：规则匹配到的报文包大小总和。target：通常是规则对应的动作。prot：规则对应的协议，即只针对某些协议应用此规则。opt：规则对应的选项。in：匹配由哪个接口（网卡）流入的数据包。out：匹配由哪个接口（网卡）流出的数据包。source：源地址，可以是一个IP，也可以是一个网段。destination：目标地址。

在链的后面()中的policy为默认策略，packets和bytes为默认策略匹配到的包数量和大小。

（2）增加规则









-t：指定表名，如果没有指定，默认为filter表。

-I：insert，表示在链首部插入1条规则，如果后面有数字，则插入指定位置。A：append，表示在链尾追加1条规则。-P：为链指定默认策略。

-j：指定动作。-s：指定源地址。

（3）删除规则

删除规则有2种方法：1种是根据规则的行号删除。1种是根据匹配条件和动作删除。









-D：删除指定链中的规则。-F：清空链中的所有规则，如果不指定链，则清空所有链中的规则。

如果没有保存规则，删除规则需要慎重。

（4）修改

修改比较麻烦，可以用先删除，再增加来代替修改。

（5）保存

1）上述增删改默认是临时的，重启iptables服务或者重启服务器后，所做出的修改都将消失。在centos6中，可使用service iptables save命令保存规则。规则保存在/etc/sysconfig/iptables文件中。

2）iptables-save命令能够将当前的iptables规则以保存后的格式输出到屏幕上。因此也可以使用如下方式保存：

iptables-save > /etc/sysconfig/iptables

3）iptables-restore命令可以从指定文件中重载规则，重载规则时，现有规则将会被覆盖。如：

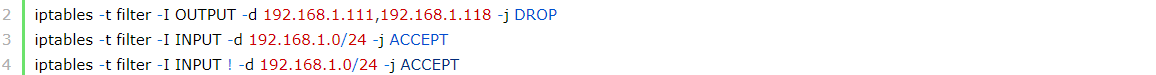
iptables-restore < /etc/sysconfig/iptables

**3.防火墙配置实例**

（1）匹配源/目标ip地址

可以使用-s匹配源地址，-d匹配目标地址。可以用,分割多个地址，可以匹配网段，可以取反。





（2）匹配协议类型

使用-p选项匹配报文的协议类型。



（3）匹配网卡接口

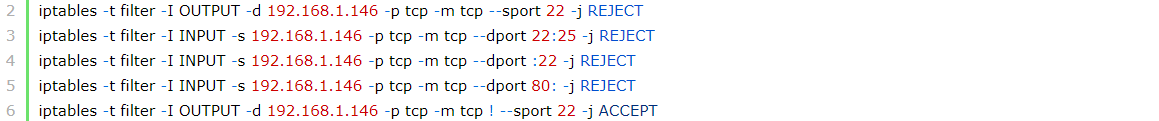
当本机有多个网卡接口时，可以使用-i选项匹配网卡流入，-o选项匹配网卡流出。



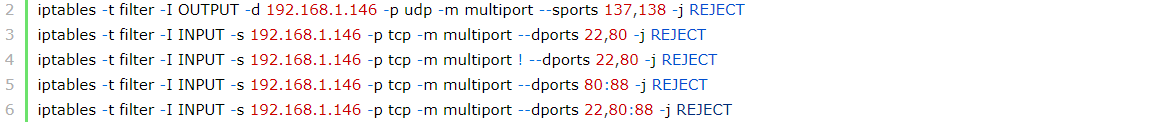


（4）匹配端口

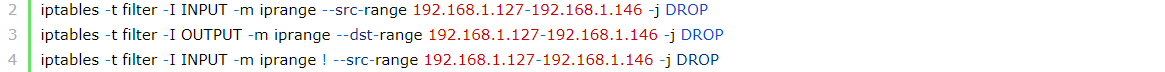
端口匹配属扩展匹配条件，需要扩展模块支持。可以使用--sport匹配源端口，--dprot匹配目标端口，可以匹配连续端口范围。匹配端口需要指定协议类型。如下，不仅用-p指定协议类型，还用-m指定了扩展模块名称。这里-m可以省略，则会使用-p指定协议对应的模块。



tcp模块只能使用连续端口，可以使用multiport模块不仅可以使用连续端口，还可以使用多个离散端口。



（5）使用iprange扩展模块，匹配连续ip地址

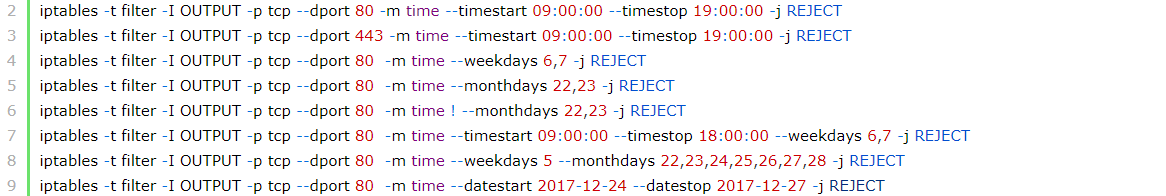


（6）使用string模块，匹配报文中的字符串



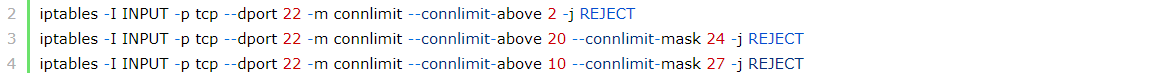
其中，algo为匹配算法，随便指定1个就行。

（7）使用time模块，根据时间段来匹配



（8）使用connlimit模块，指定每个ip连接server的数量

如果不指定ip，则为所有连接数量。如--connlimit-above 2表示限制每个IP的链接数量上限为2。结合--connlimit-mask 27表示掩码为27的这个网段下的30台机器中，只有2台可以连接。



（9）使用limit模块，限制报文速率。

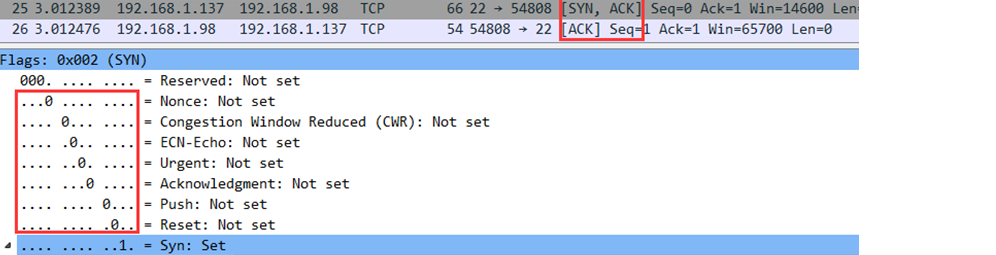
--limit 10/minute -j ACCEPT表示每分钟最多放行10个包。--limit-burst表示空闲时可放行的包的数量。



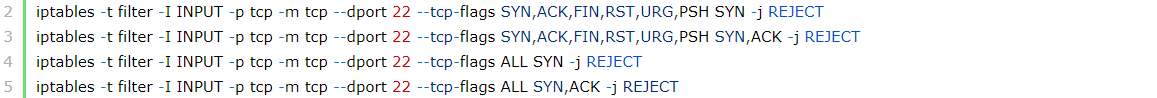
（10）使用tcp模块，匹配tcp的flags。

--tcp-flags匹配tcp头中的标志位。





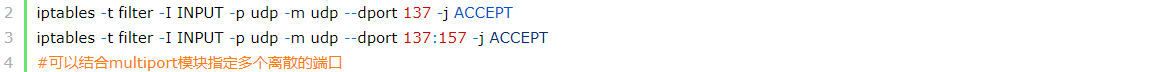
匹配报文的tcp头的标志位：



匹配tcp新建连接的请求报文，相当于使用"--tcp-flags SYN,RST,ACK,FIN SYN"：

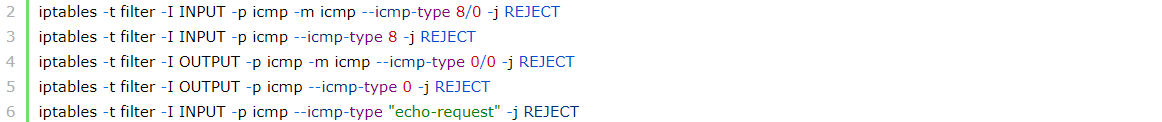


（11）使用udp扩展，匹配端口



（12）使用icmp扩展，匹配icmp报文类型

使用type/code匹配具体报文类型，如type3表示目标不可达，code1表示主机不可达报文。



（13）使用state扩展

当客户端访问服务器时，会开发1个临时端口等待对方相应。state扩展模块就是用于防止其中的风险。

尽管tcp/ip协议中，UDP没有状态，但state模块会视作有状态。state中的报文状态一共有5种：NEW、ESTABLISHED、RELATED、INVALID、UNTRACKED。其中，新连接的第1个包的状态为NEW。NEW状态包后面的包为ESTABLISHED。有关系的两个链接中的报文为RELATED，如果ftp。一个包没法被识别，或者这个包没有任何状态，那么这个包的状态就是INVALID。报文状态为Untracked时通常表示无法找到相关的连接。

问题的根源是：怎样判断报文是否是为了回应之前发出的报文。可以通过只要放行状态为RELATED或ESTABLISHED的报文即可，表示只有回应的报文能够通过防火墙，如果是别人主动发送过来的新的报文，则无法通过防火墙。

（14）iptables的黑白名单机制

按常理，如果设置链的默认规则为ACCEPT，那么就应该设置具体规则为DROP或REJECT，这样就是黑名单机制。如果设置默认规则为DROP，那么就应该设置具体规则为ACCEPT，这样就是白名单机制。白名单机制有1个问题，就是如果不小心执行iptables -F命令，管理员也无法登陆系统了。

实践中的做法是，设置默认规则为ACCEPT，然后设置放行规则，然后再设置拒绝其他一切请求。

**4.iptables自定义链**

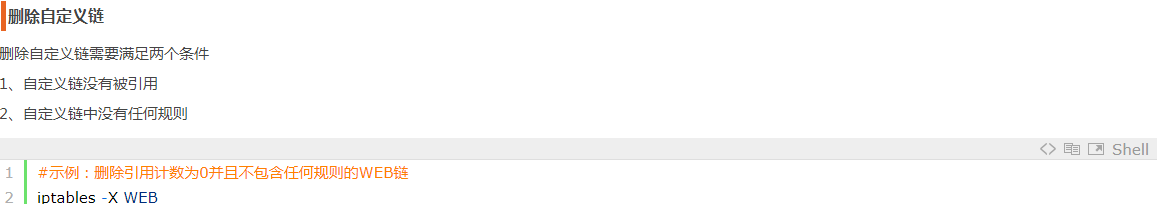
（1）使用场景：如果INPUT链中存放了200条规则，有针对httpd服务的，有针对sshd服务的，有针对私网IP的，有针对公网IP的，修改起来会比较麻烦。

（2）可以使用自定义链解决上述问题。自定义链是预设链的子集，必须在预设链下。



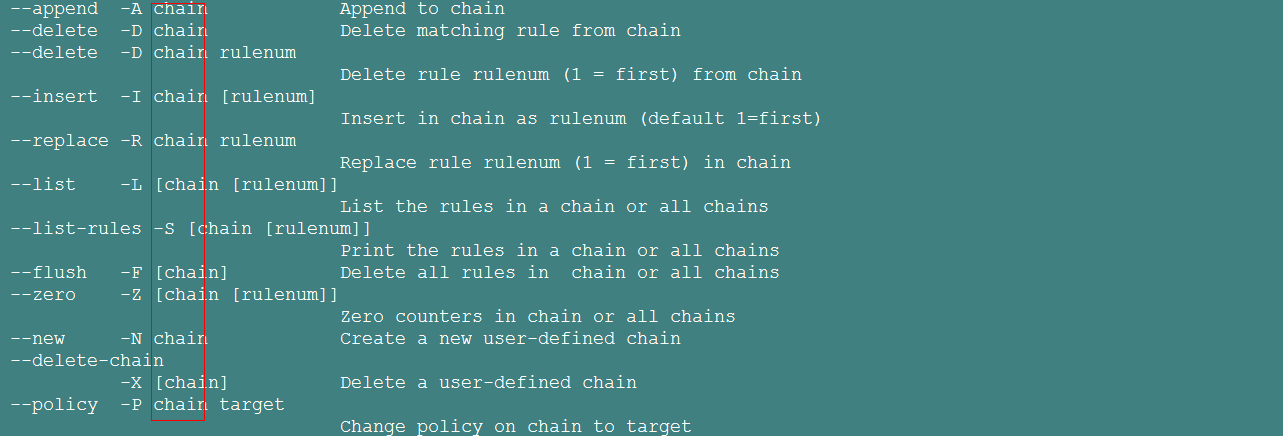






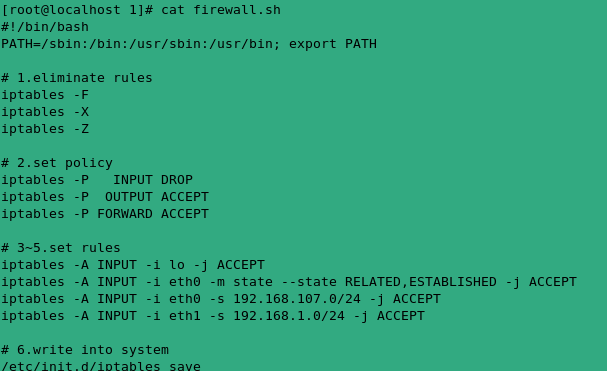
**5.iptables命令总结**

（1）链操作相关的都是大写



（2）除此之外都是小写。

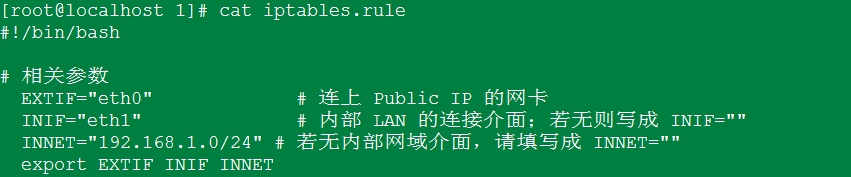
**6.最简单的iptables配置**



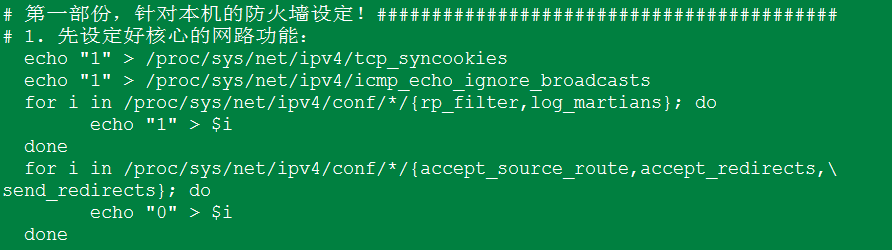


可以写成脚本形式，如下：

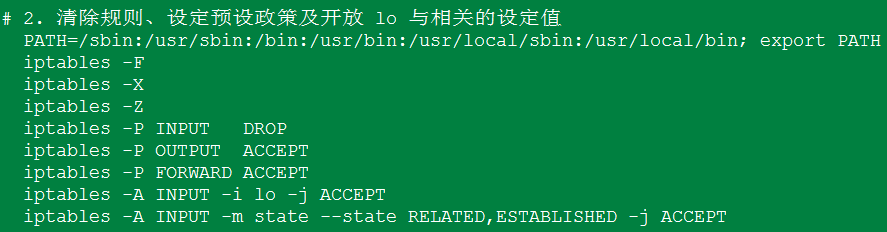
（1）定义变量



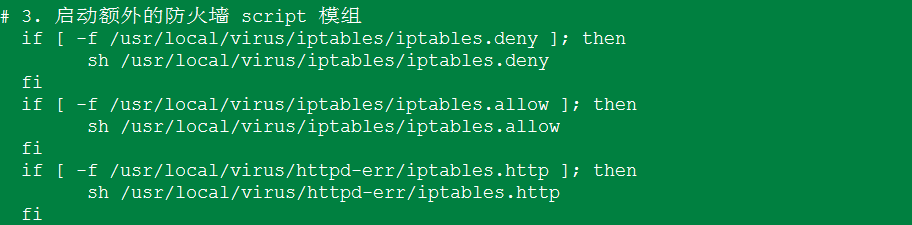
（2）配置核心网络功能



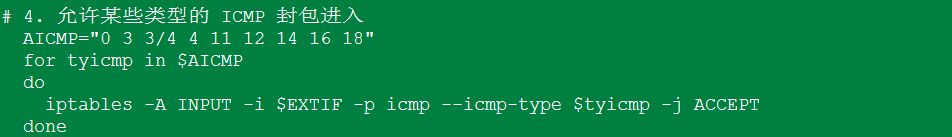
（3）清除规则，设置默认规则



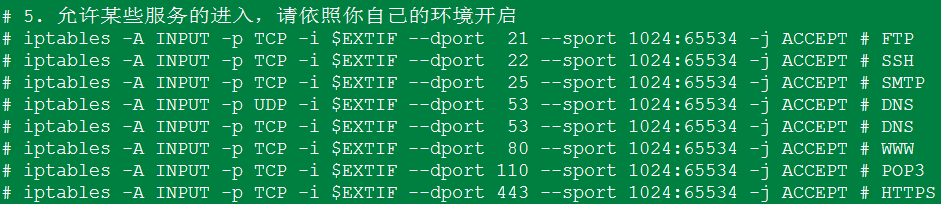
（4）启动额外的防火墙功能



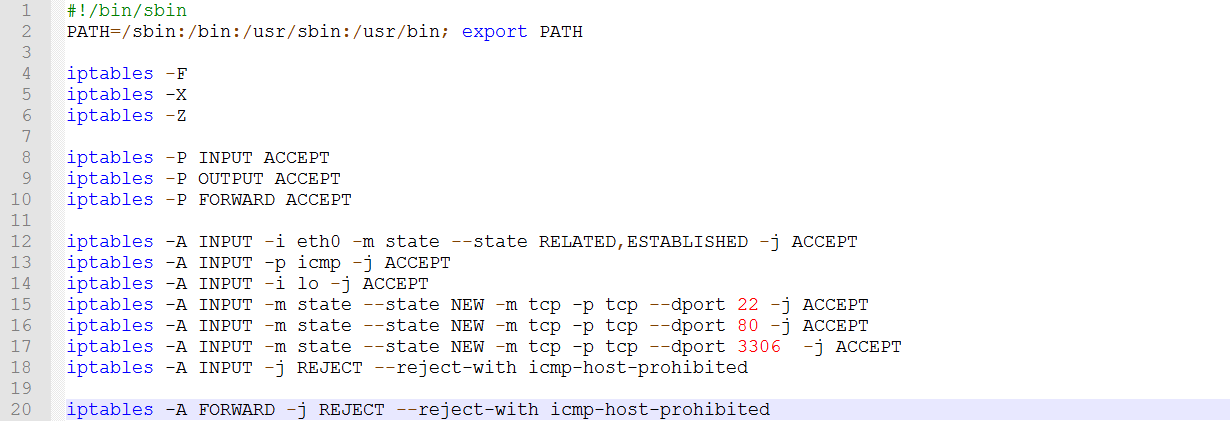
（5）允许某些类型的ICMP封包进入



（6）允许某些服务的进入，请依照你自己的环境开启



**7.LAMP防火墙设置**



* + 1. 其他防攻击机制

除了iptables外，linux内核还提供其他防攻击机制。因这些机制是核心的网络功能，所以相关配置文件都在/proc/sys/net/ipv4/目录下。

**1./proc/sys/net/ipv4/tcp\_syncookies**

阻断式服务（DoS）攻击法是利用TCP封包的SYN三次交握原理攻击，称为SYNFlooding。可以启用核心的SYNCookie模组，会在随机网络端口（1024-65535）即将用完时自动启动。如果启用SYNCookie，主机会在发送SYN/ACK确认封包前，要求Client端在短时间内回复1个序号，由此判断是否是正常访问。



不适合用在负载已经很高的服务器上使用，会让核心误判遭受SYNFlooding的攻击。

**2./proc/sys/net/ipv4/icmp\_echo\_ignore\_broadcasts**

发送分包大且数量多的ping的封包攻击称为ping flooding。可以通过取消ICMP类型8的ICMP封包回应解决。其中，icmp\_echo\_ignore\_broadcasts表示分包中有ping broadcast位址时才取消ping回应。icmp\_echo\_ignore\_all表示取消全部ping回应。通常设置icmp\_echo\_ignore\_broadcasts即可。



**3./proc/sys/net/ipv4/conf/网卡/\***

（1）rp\_filter

逆向路径过滤，通过分析网卡路由信息和封包来源ip判断封包是否合理。

（2）log\_martians

记录不合法ip来源访问日志。

（3）accept\_source\_route

过滤路由来源。

（4）accept\_redirects

路由重定向，建议关闭。

（5）send\_redirects

路由重定向，建议关闭。

* 1. NAT

**1.概要**

（1）NAT（Network Address Translation）即网络地址转换。

（2）NAT使用nat表，只能存在于PREROUTING链、POSTROUTING链、OUTPUT链中，通常在PREROUTING链和POSTROUTING链中修改ip地址等。

（3）需要启用主机路由功能。

**1.SNAT**

（1）使用场景1：LAN中有10台主机，但不想对外网暴露它们的ip。路由器如果开启NAT功能，会维护1张nat表，记录报文的内网ip和端口号。当报文经过路由器，会在POSTROUTING链中转换报文头为路由器的ip，并使用1个新端口向外网发出。接收到外网回应时，也会根据nat表进行ip和端口转回。



（2）使用场景2：IPv4地址紧张，内网只设置私网ip，通过NAT技术转为路由器的公网ip，缓解ip地址紧张。

（3）只需设置1条命令即可，不需要设置回传转换命令。



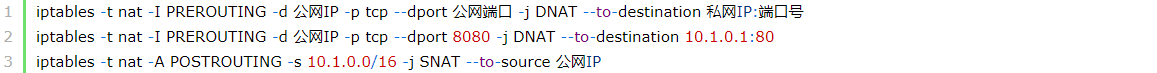
（4）如果路由器ip地址并非固定ip，可以使用MASQUERADE自动转为可用地址。



**2.DNAT**

（1）使用场景：公司只有1个公网ip，但内网中有很多服务器提供不同服务。可以将外网客户端发送过来的报文的目标地址与端口号和内网主机服务映射。如，报文目标IP与端口号为：公网IP+3306，就将报文的目标地址与端口改为：主机2的私网IP+3306；公网IP+80端口映射为主机1的私网IP+80端口。当私网中的主机回应时，再将回应报文的源地址从私网IP+端口号映射为公网IP+端口号，再由路由器发送给互联网中的主机。

（2）理论上来说，只要配置DNAT规则，不需要再配置对应的SNAT规则即可，但测试时可能会出错。可以先只配置DNAT规则，如果无法正常DNAT，再添加对应的SNAT规则。SNAT规则配置一条即可，DNAT规则需要根据实际情况配置不同的DNAT规则。



（3）因为是对外提供服务，公网ip应该是固定的，所以没有MASQUERADE。

**3.端口转发**

（1）可以使用REDIRECT动作在本机上进行端口映射。

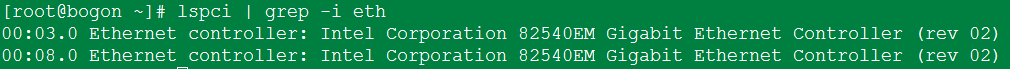
（2）将本机的80端口映射到本机的8080端口上：



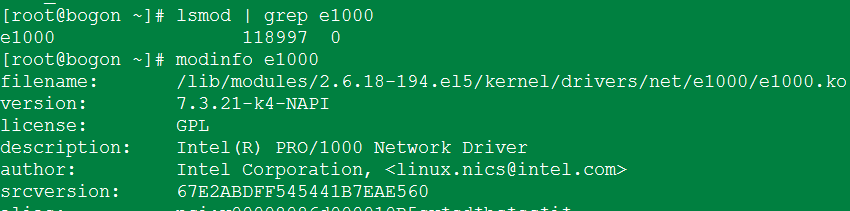
1. linux网络管理
   1. 网卡设置

**1.检查网卡**

一般都是以太网卡，可以使用lspci命令查看。

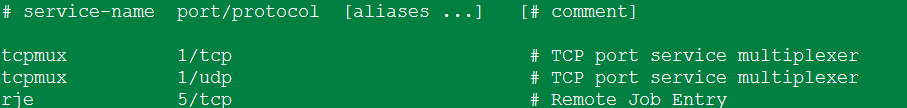


可以使用lsmod查看网卡驱动是否被系统加载，可以使用modinfo查看驱动详细情况。

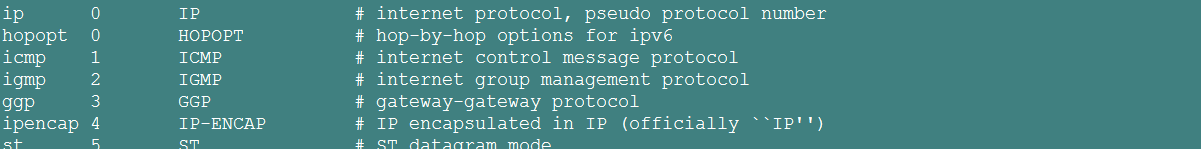


**2.相关配置文件**

（1）/etc/services：默认端口映射。



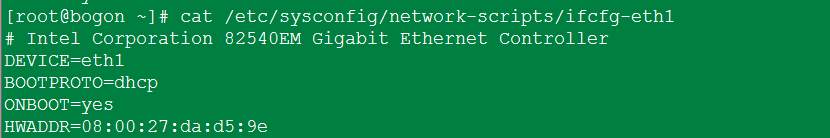
（2）/etc/protocols：网络层所有协议列表，包括协议名称、协议号和协议别名3项内容，通常不要修改。



（3）/etc/sysconfig/network：是否启动newwork，主机名设置，是否启用ipv6。如果在这里设置了网关，但在其它地方也重新配置网关并使其生效，则以其它地方为准。

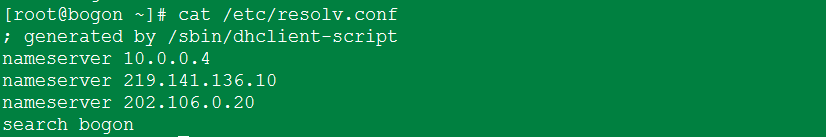


（4）/etc/sysconfig/network-scripts/ifcfg-eth[0-9]：网卡配置文件，修改设置后需要重启设备。



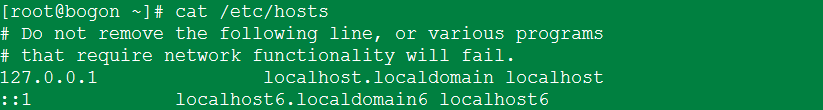
其中：DEVICE：网卡设备名称，应与文件名中设备代号相同，否则会找不到设备。ONBOOT：系统启动时是否默认启动本设备。BOOTPROTO：启动网络设备时使用的协议，如果是手动设置IP，输入static或none；如果是自动取得IP，输入dhcp（全小写）。IPADDR：IP地址。GATEWAY：默认网关，系统中只能配置1个（要不怎么叫默认）。HWADDR：MAC地址，不能和MACADDR同时使用。BROADCAST：广播地址。NETMASK：子网掩码。

（5）/etc/resolv.conf：DNS客户端设置。



nameserver为DNS服务器的IP地址，可指定多个。domain：定义本地域名。search：定义域名的搜索列表（当提供了一个不包括完全域名的主机名时，在该主机名后添加search后的域）。domainname和search可同时存在，也可只有一个。

（6）/etc/hosts：静态主机名和IP地址映射。

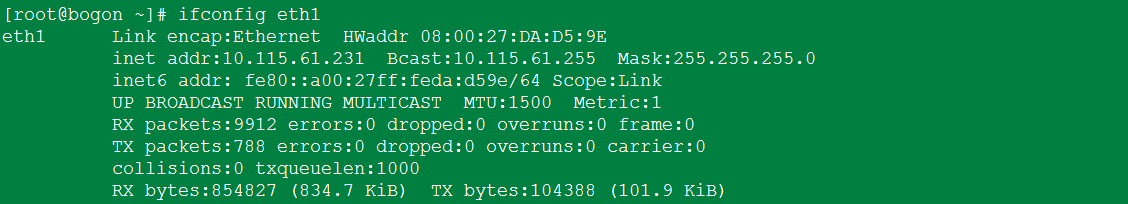


**3.网卡配置命令**

可以使用ifconfig查看、修改某个网卡配置。



（1）查看网卡配置情况



第1行：eth1：网卡设备名（lo为loopback）。Link encap：链路层封装。HWaddr：MAC地址。

第2行：inet addr：IPv4地址。Bcast：广播地址。Mask：掩码。inet6 addr：IPv6地址。

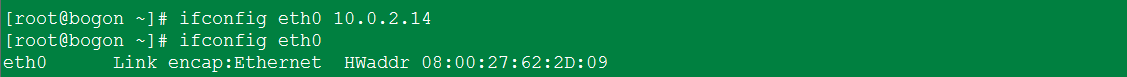
第3行：UP网卡已开启。BROADCAST支持广播。RUNNING网卡正在运行（已连接网线）。MULTICAST支持组播。MTU最大传输单元。Metric接口度量值，即发送1个分组的成本。

第4、5行：RX表示启动至今包接收情况，TX传送情况。后面的packets代表封包数，errors代表封包错误数，dropped代表封包遭丢弃数等。

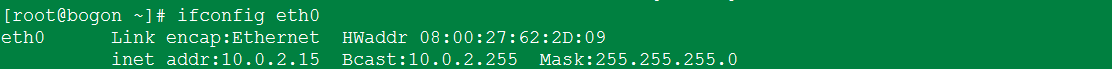
第6行：collisions：代表封包碰撞的情况，如果发生太多次，表示网路状况不太好。txqueuelen：代表用来传输资料的缓冲区的储存长度。

第7行：总接收、传送的位统计。

（2）修改网卡配置，但修改的配置在系统重启后不会保存。



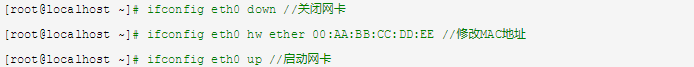








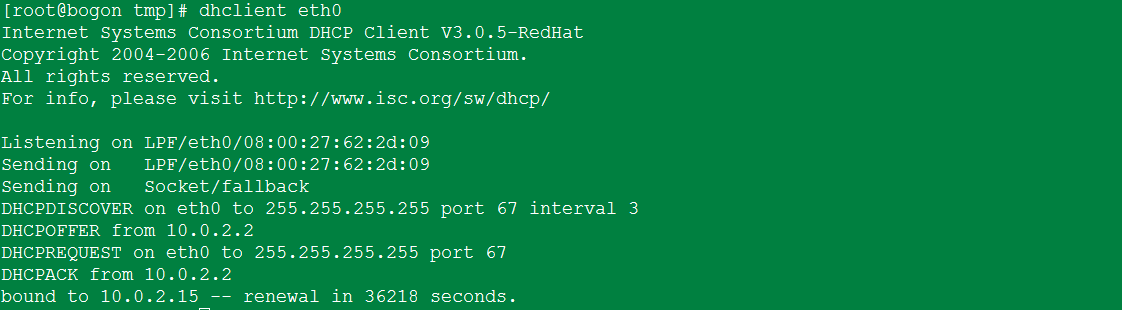
（3）启动关闭指定网卡。可以使用ifconfig up/down和ifup/ifdown，两者功能相同。



（4）/etc/init.d/network restart重新启动整个网络配置。

**4.手动使用DHCP取得IP**

-r为释放ip。

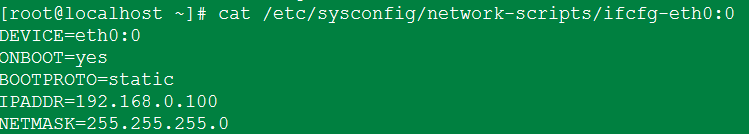


**5.单个网卡绑定多个ip**

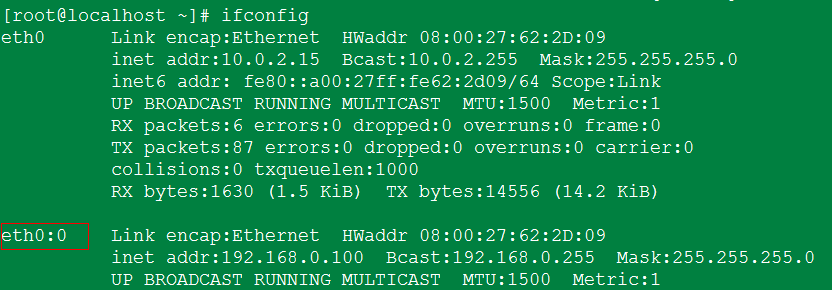
（1）所有的IP Alias都是由实体网卡模拟来的，所以当要启动eth0:0时，eth0必须要先被启动才行。而当eth0被关闭后，所以eth0:n的模拟网卡将同时也被关闭。

（2）1个网卡绑定多个ip通常只适用于手动设置ip，不能使用DHCP。

（3）配置方法：新建如/etc/sysconfig/network-scripts/ifcfg-eth0:0文件即可。







* 1. 路由设置

**1.一些概念**

（1）网关和默认网关

局域网中的路由器就是网关，用于连接外网。因此，一台主机想要联网，就必须设置自身网关地址为这台路由器的地址。

当一台主机有多个网卡时，如果每个网卡要连不同的局域网，通过不同的网关出去。此时设置默认网关的意义就是当没有指定目的地址对应的网关地址时，就走默认网关。1个系统的默认网关只能设置1个。

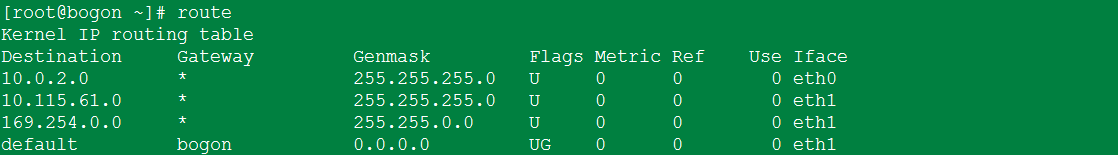
（2）主机默认没有开启路由转发功能。路由转发功能和上面的网关配置不是一回事。

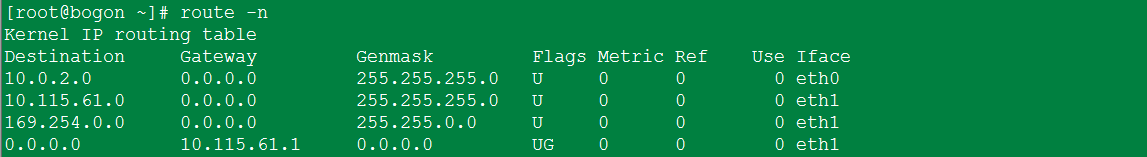
（3）路由3种类型：主机路由、网络路由、默认路由。

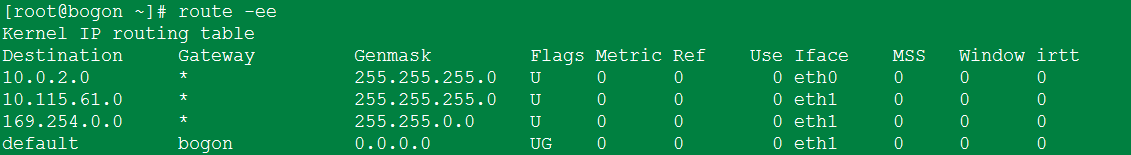
**2.使用route命令查看修改本机路由配置**

用于显示和操作IP路由表，系统重启后失效。

（1）可以直接使用route查看，或加选项ee查看更详细的信息，选项n表示Gateway用ip地址显示。







参数含义：Destination：目标网络或目标主机。Gateway：网关地址。Genmask：目标网络或目标主机的网络掩码。Flags：U表示路由为活动状态，G路由指向网关，H路由指向主机，R动态路由选项，D路由后台程序安装，M路由后台程序修改。Metric：路由距离。Ref：路由项引入次数，恒为0，不用。Use：此路由项被查找次数。Iface：路由项输出接口。

0.0.0.0表示路由表中没有特定指定目的主机和网络ip地址。如上，Destination中的1-3都没有配置网关，但因设置了默认网关（Destionation为0.0.0.0或default）因此，能通过默认路由（0.0.0.0/0.0.0.0），并最终通过10.115.61.1这个网关发送出去。

特别注意，不管在哪个网卡配置中设置的GATEWAY，在路由表中都是在ethN，N最大的上面。如下，给两个网卡配置相同的ip，两个网卡都能获取数据，但只会从eth1网卡发送出去，因为默认网关设置在它上面。



（2）添加/删除路由相关参数

route add/del [-net|-host] [网域或主机] netmask [mask] [gw|dev]



net|host为Destination。netmask和mask为掩码。gw为网关，default gw为默认网关。dev为对应的网卡。如果网关不通，则无法添加gw。如果没有设置网关，则默认为0.0.0.0。

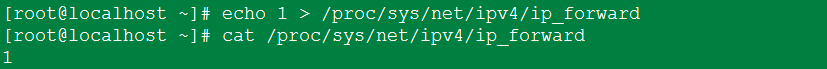
（3）让配置永久生效方法：1）将（2）中的命令写入/etc/rc.local中；2）在/etc/sysconfig/network中添加GATEWAY字段；3）创建/etc/sysconfig/static-router文件（默认没有）。

**3.主机路由设置**

（1）检查内核是否开启路由（转发）功能，即从1个接口收到数据包，从另外1个接口转发出去。这与普通主机网关设置不同，默认没有开启。



0代表没有打开，1代表打开。可以通过如下方法临时打开，重启系统后失效。

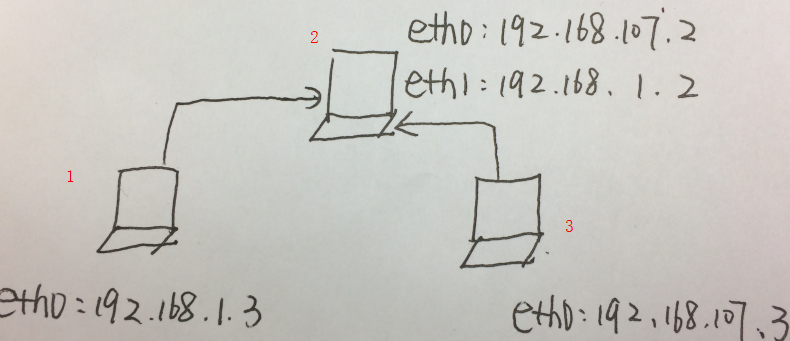


也可以修改系统配置文件/etc/sysctl.conf永久打开。



（2）静态路由设置

想布置如下网络环境如下，有2个子网192.168.1/24和192.168.107/24，中间的linux系统充当路由器。



需要如下配置：

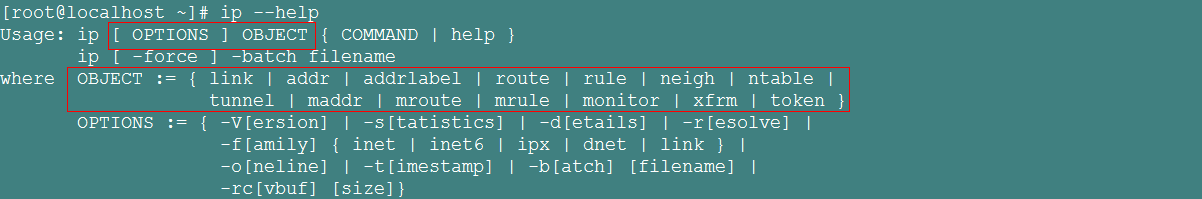
1）设置机器1、2、3的ip地址，并需要设置机器1的网关地址为192.168.107.2，设置机器2的网关地址为192.168.1.2。

2）开启机器2的路由转发功能，貌似就可以了。如果需要连其他路由，这台路由主机才需要设置网关。

需要特别说明的是，网络访问需要2个方向都能访问才行。如：192.168.1.3到192.168.1.2到192.168.107.2到192.168.107.3能访问，但同时反方向也要能通，这样才能实现最终访问（TCP是有连接的）。

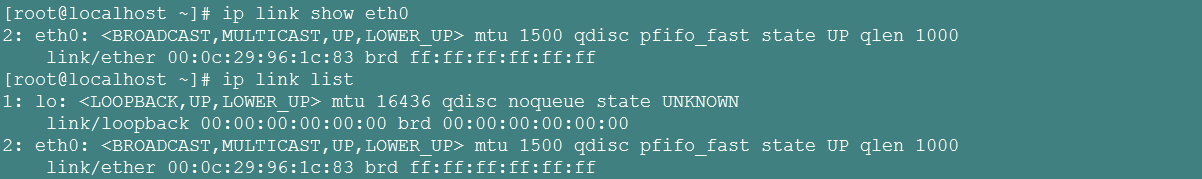
* 1. ip命令

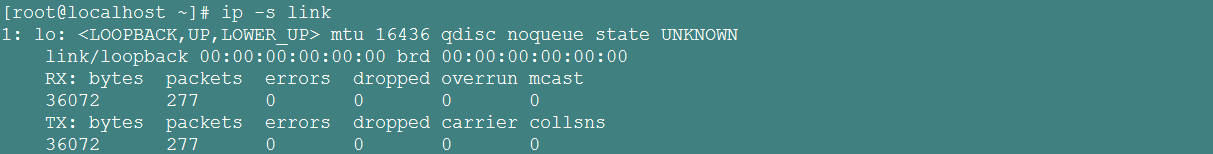
ip是个综合性的命令，可以替代ifconfig和route，并实现更细节的功能。



COMMAND是针对OBJECT的操作，有set、add、delete、show、list等。

（1）显示设备及属性（链路层）

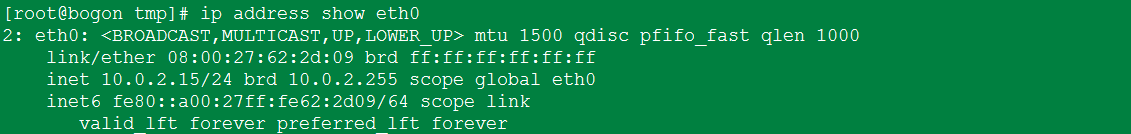




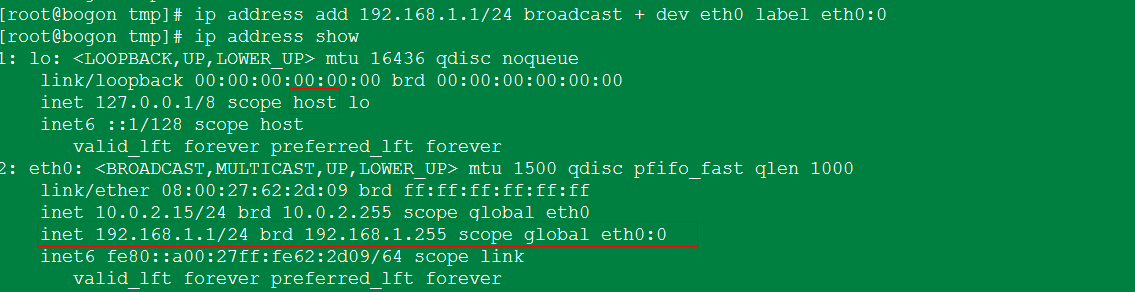
（2）改变设备属性，其中set up/down相当于之前的ifconfig up/down。其他设置属性最好不要修改。



（3）显示ip信息（网络层）



（4）设置ip信息（网络层）

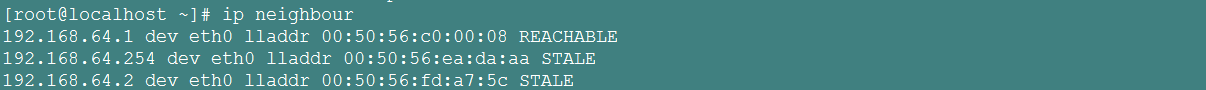


其中scope值有：global：允许来自所有来源的连线（默认）；host：仅允许本主机内部的连线；site：仅允许本主机内部的连线，但仅支持IPv6；link：仅允许本设备自我连线。

（5）显示路由信息



（6）查看局域网中的设备MAC地址



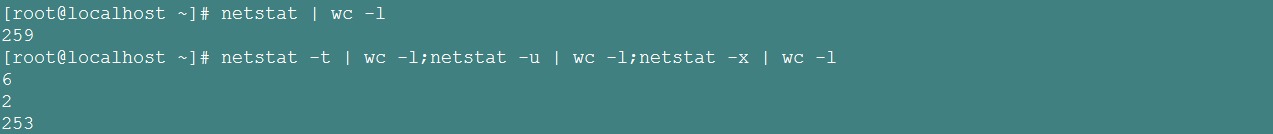
（7）查看网络设备的状态



* 1. netstat

**1.端口相关**

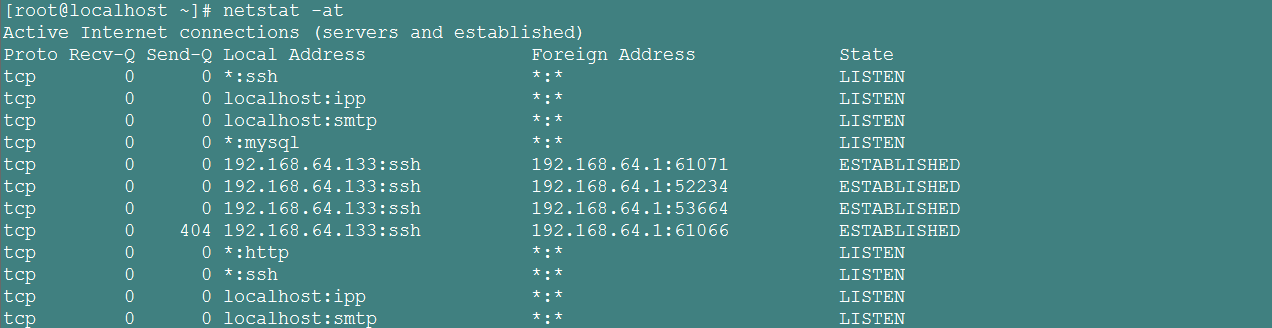
（1）-t：列出tcp端口；-u：列出udp端口；-x：列出UNIX端口。三者都显示已连接的端口。netstat不加任何参数，默认会显示所有已连接的端口，等价于t、u、x之和。



-l：列出所有处于监听的端口，因此，想查看tcp监听端口情况，如下：



-a：显示所有连接和监听的端口（等价于不加l和加l之和）。



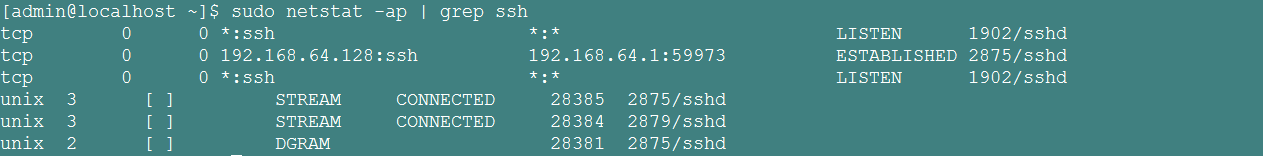
Proto：封包协定。

Recv-Q和Send-Q：为发送和接受队列，通常为0，除非数据包正在堆积。

Local Address：本地端的位址；Foreign Address：远端的主机IP与port。

stat：状态列，主要的状态含有：1）ESTABLISED：已建立连线；2）SYN\_SENT：发出主动连线（SYN标志）的连线封包；3）SYN\_RECV：接收到一个要求连线的主动连线封包；4）FIN\_WAIT1：该插槽服务（socket）已中断，该连线正在断线当中；5）FIN\_WAIT2：该连线已挂断，但正在等待对方主机回应断线确认的封包；6）TIME\_WAIT：该连线已挂断，但socket还在网路上等待结束；7）LISTEN：通常用在服务的监听port！可使用‘-l’参数查阅。

（2）-p和上面的选项一起使用可以显示进程信息。如，获取程序运行端口信息（最后为进程PID和进程名）：



（3）-n：不反解主机名。

**2.统计信息**

显示所有端口类别统计信息：-s。如显示tcp类端口统计信息，如下：

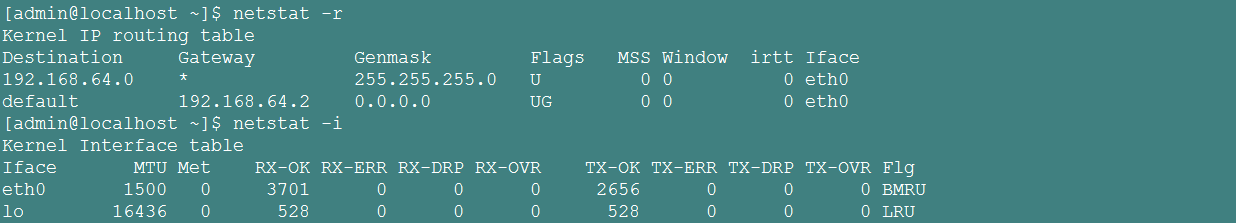


**3.持续输出netstat信息**

选项-c。

**4.网卡或路由**

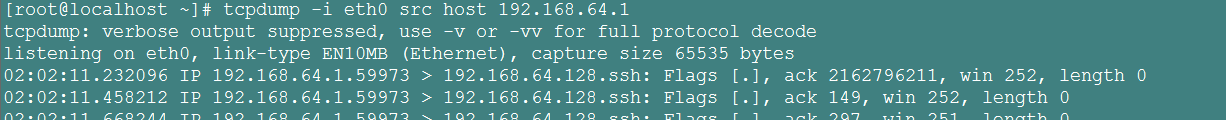
-i：显示网卡相关，结合-e可以显示更详细信息。-r：显示路由相关。



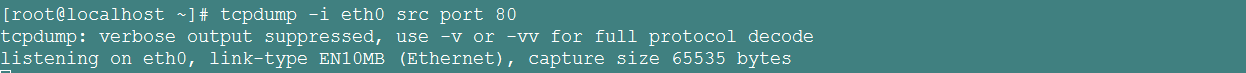
* 1. tcpdump

对网络数据包进行截获的包分析工具，可以监听数据包流向、内容等，是1个黑客软件。

（1）抓取所有经过eth0，目的或源地址是192.168.64.1数据。



（2）抓取所有经过eth0，目的或源端口是80数据。



（3）协议过滤



（4）其他选项

-A：封包的内容以ASCII显示，通常用来捉取WWW的网页封包资料。

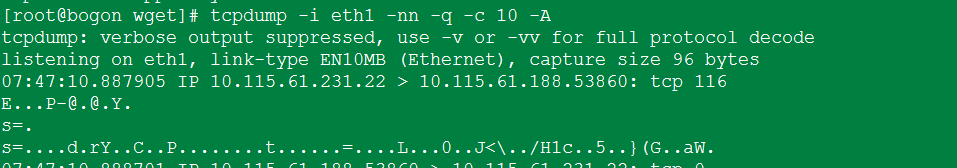
-e：使用数据链路层的MAC封包资料来显示。

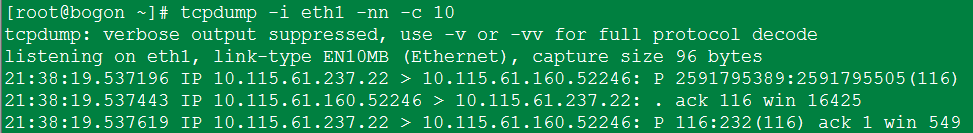
-nn：直接以IP及portnumber显示，而非主机名与服务名称。

-q：仅显示简短的封包信息。

-w：将监听所得的封包资料保存为文件，后接文件名。-r：读取由-w所创建的文件。

-c：监听的封包数，如果没有这个参数，tcpdump会持续不断的监听，直到使用者输入[ctrl]-c为止。





输出格式为：系统时间通信协议来源主机.端口>目标主机.端口数据包参数（其中P为封包带有push标志）

* 1. 网络使用命令

**1.curl**

curl是利用URL规则在命令行下工作的文件传输工具，支持HTTP、HTTPS、FTP等协议，还支持认证、上传、代理服务器、断点续传等，功能十分强大。常用操作：

（1）直接显示到屏幕上



（2）使用重定向功能保存



（3）-o（小写）：指定文件名保存保存网页



（4）-O（大写）：根据url中的文件名保存，url必须到文件名，否则无法下载，可同时获取多个文件





（5）测试网页返回码



（6）-x：使用代理



（7）-A：模仿浏览器



（8）-d：通过脚本发送post请求，顺便附带文本数据。-F：为带文件的形式发送post请求。



**2.wget**

（1）命令格式：wget [option] [网址]。

（2）使用HTTP，HTTPS和FTP协议自动下载文件的工具。所谓自动下载是指，wget可以在用户退出系统的之后在后台执行。而且还可以跟踪HTML页面上的链接依次下载，并在创建远程服务器的本地版本，能完全重建原始站点的目录结构，又常被称作“递归下载”。如果因网络原因下载失败，wget会不断的尝试，直到整个文件下载完毕。如果是服务器打断下载过程，会从停止的地方继续下载。

（3）不带参数，直接下载当前页。



（4）-r：递归下载服务器上的所有目录和文件，可能会有很多层，用-l num指定层级。



（5）支持批量下载，将url分行写在1个txt文件中，如wget -i download.txt下载。

（6）如果需要用户名、密码，可以使用--http-user=USER设置HTTP用户，--http-passwd=PASS设置HTTP密码。

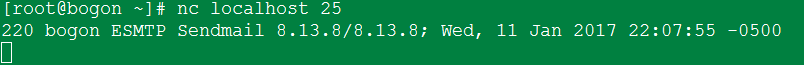
（7）可以使用http-proxy和ftp-proxy使用代理服务器下载。

**3.telnet**

telnet [host|IP [port]]：连接服务器。

**4.nc**

nc的功能和telnet类似，可以用来取代telnet，可以用来监听某些端口。



**5.links**

文字浏览器。



* 1. 网络排错

**1.网卡排错**

（1）网卡驱动

使用lspci和dmesg查询是否存在相关模组，如果没有，可能需要重新换张网卡。

（2）通过手动设置ip，检查能否设置ip。





**2.局域网排错**

（1）检查当前局域网段。

（2）检查网关和dns。

**3.检查ip设置**

通过ifconfig检查是否已自动获取ip。

**4.确定路由表规则**

（1）首先ping下路由器地址（网关）。

（2）通过route –n检查路由设置。

**5.检查dns配置**

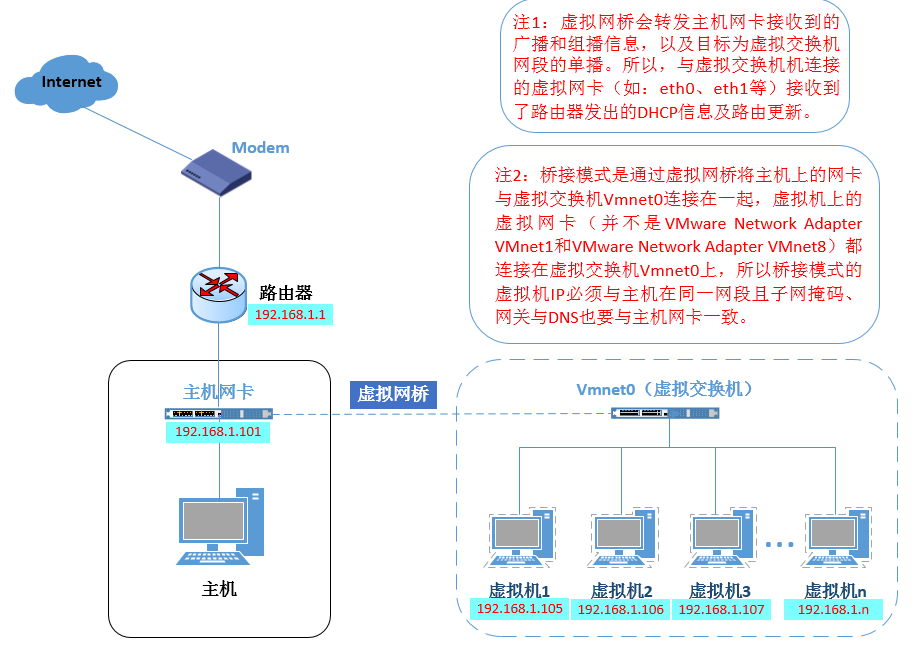
**6.检查NAT服务器配置**

1. VMWare网络模式

VMware提供三种网络模式：

**1.Bridged（桥接模式）**

桥接模式就是将主机网卡与虚拟机虚拟的网卡利用虚拟网桥进行通信。



桥接模式下，VMware虚拟出一个交换机，所有桥接设置的虚拟机连接到这个交换机的一个接口上，物理主机也同样插在这个交换机当中，所以所有桥接下的网卡与网卡都是交换模式的，相互可以访问而不干扰。

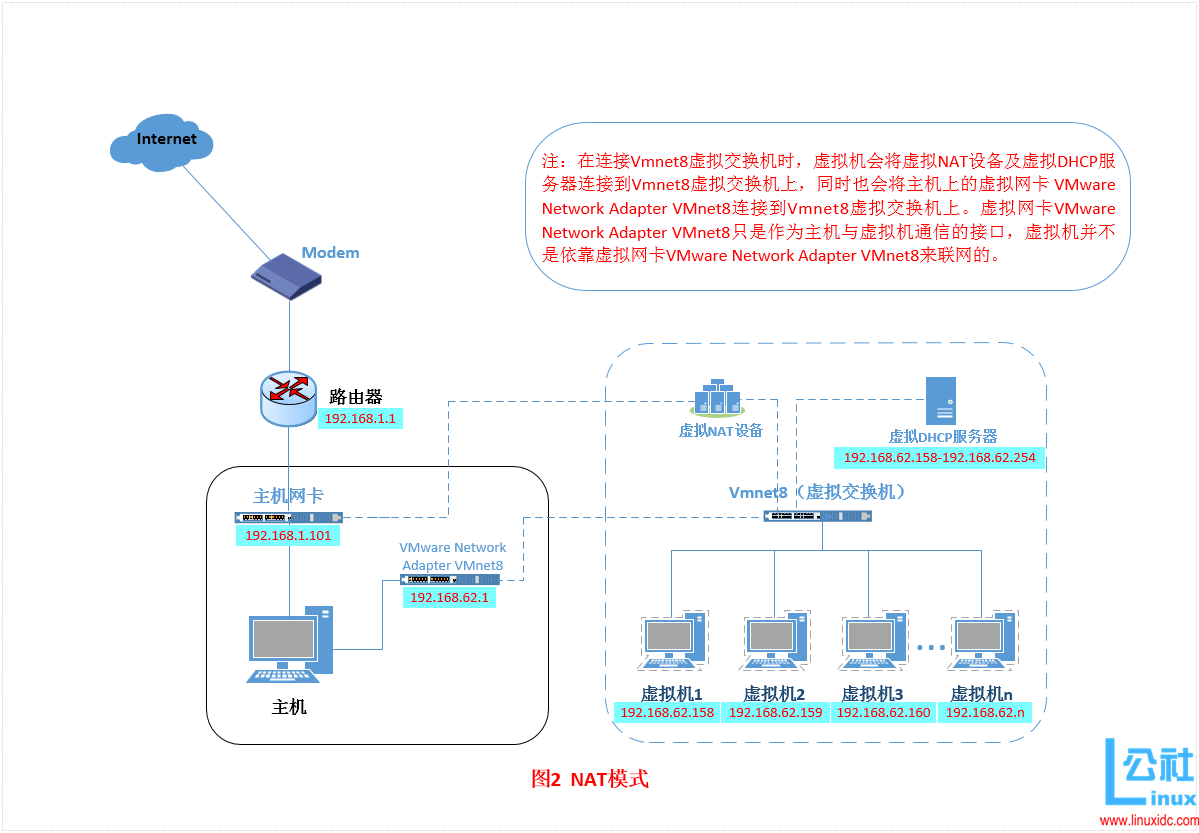
首先要设置虚拟网络编辑器的桥接模式，选择将虚拟机桥接到主机网卡上。



编辑虚拟机/etc/sysconfig/network-scripts/ifcfg-eth0文件，可以设为DHCP（如果主机网络就是自动分配的话），也可以手动设置ip，但虚拟机ip地址需要与主机在同一个网段，如果需要联网，则网关与DNS需要与主机网卡一致。

**2.NAT（地址转换模式）**

NAT模式借助虚拟NAT设备和虚拟DHCP服务器（VMware提供），使得虚拟机可以联网。其网络结构如下图所示：

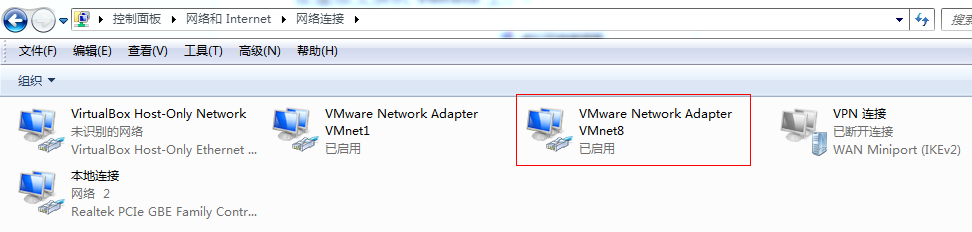


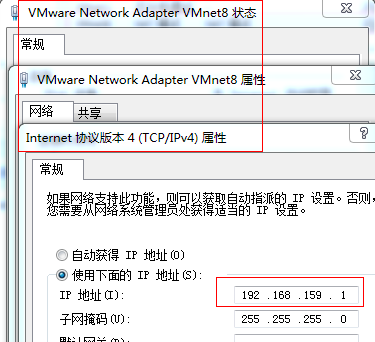
如上，VMware虚拟出1）虚拟NAT设备，2）虚拟DHCP服务器，3）VMware8（虚拟交换机），同时还为主机虚拟出VMware Network Adapter VMnet8虚拟网卡。主机网卡直接与虚拟NAT设备相连。VMware Network Adapter VMnet8和所有虚拟机直接和VMware8（虚拟交换机）连接，虚拟NAT设备与虚拟DHCP服务器也连接在虚拟交换机VMnet8上。

VMware提供的虚拟网络编辑器可以对NAT设备进行编辑，其中网关默认为192.168.159.2，可以自定义编辑。



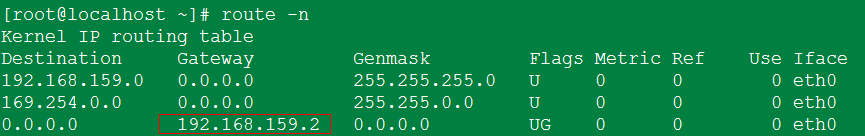
windows网络连接可以看到虚拟出的VMware Network Adapter VMnet8虚拟网卡，其中ip地址写死为192.168.159.1，可以人为更改，由此可见不是DHCP自动分配的。





查看虚拟机的ip和路由可见，已自动分配ip，默认网关就是NAT设备中设置的网关地址。





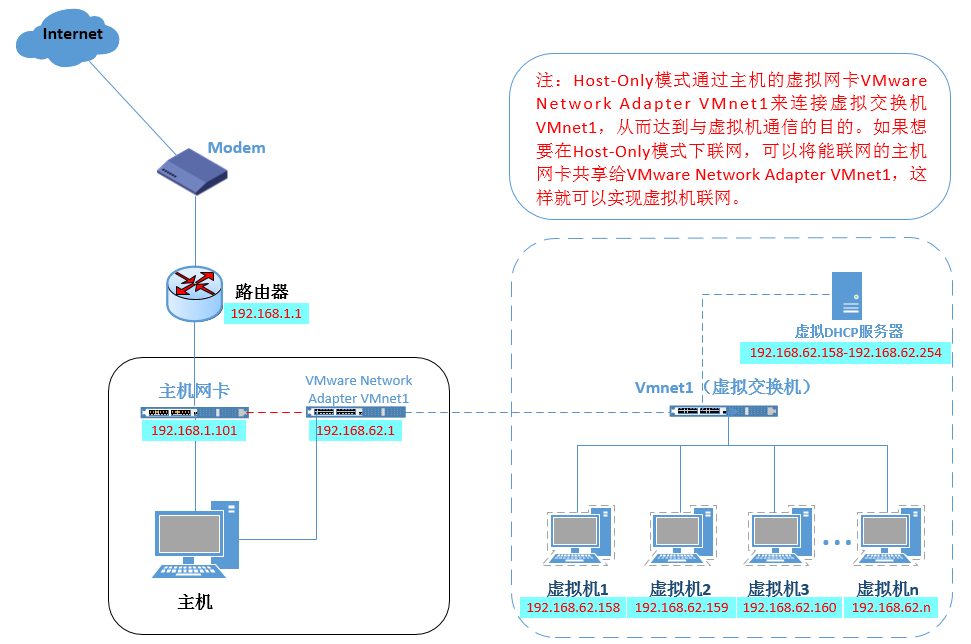
【注意】

1）如果关闭VMware Network Adapter VMnet8，主机和虚拟机显然无法通信。

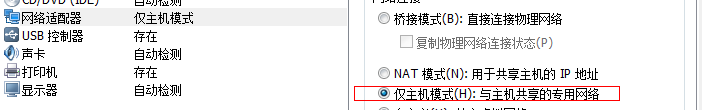
2）如果VMware Network Adapter VMnet8的ip地址和nat设备地址设为一样，主机和虚拟机仍然可以通信，但是虚拟机无法连接外网（局域网内地址有冲突）。

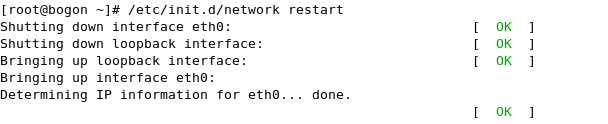
**3.Host-Only（仅主机模式）**

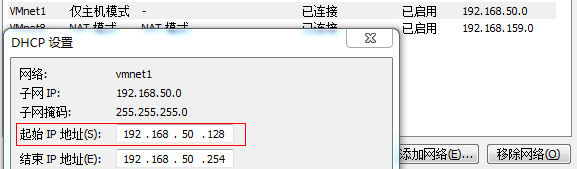
Host-Only模式其实就是NAT模式去除了虚拟NAT设备，然后使用VMware Network Adapter VMnet1虚拟网卡连接VMnet1虚拟交换机来与虚拟机通信的。Host-Only模式将虚拟机与外网隔开，使得虚拟机成为一个独立的系统，只与主机相互通讯。



将虚拟机连接模式设为主机模式后，重启系统网络，会获得新的ip地址。

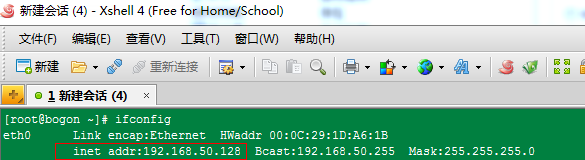


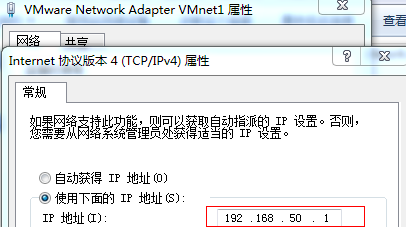


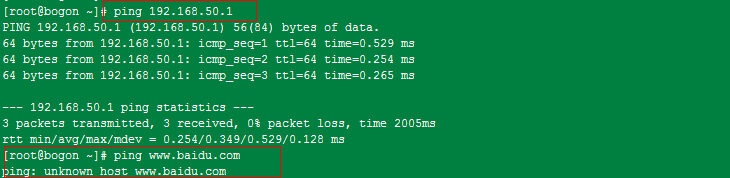




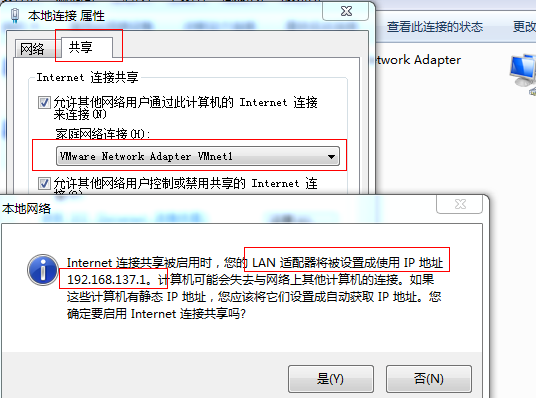
此时主机能连上虚拟机，虚拟机也能连接主机。



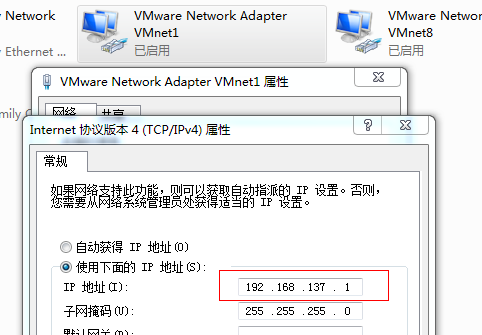




但是，虚拟机无法连接外网。可以通过将主机的网卡共享给VMware Network Adapter VMnet1虚拟网卡，并将该虚拟网卡作为虚拟机的路由，即可实现虚拟机连接外网。

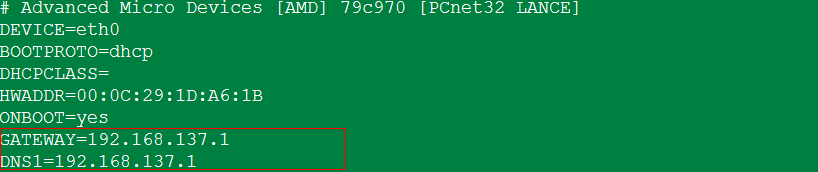


此时，VMware Network Adapter VMnet1虚拟网卡被作为路由器，地址被强制设置为192.168.137.1，因此，需要设置DHCP，将虚拟机地址也设为对应网段。

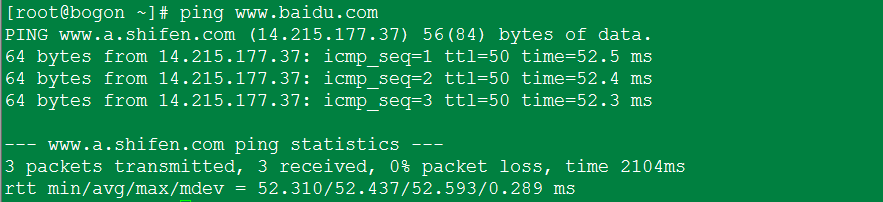




设置虚拟机路由：



重启后，虚拟机即可连接外网。



1. Wireshark使用

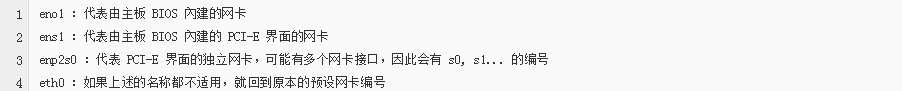
见https://wenku.baidu.com/view/2e44db52f46527d3240ce0d5.html。

1. CentOS7相关

**1.网卡配置**

（1）可能没有ifconfig，需要用ip命令。

（2）网卡命名规则：



前2个字符的含义：



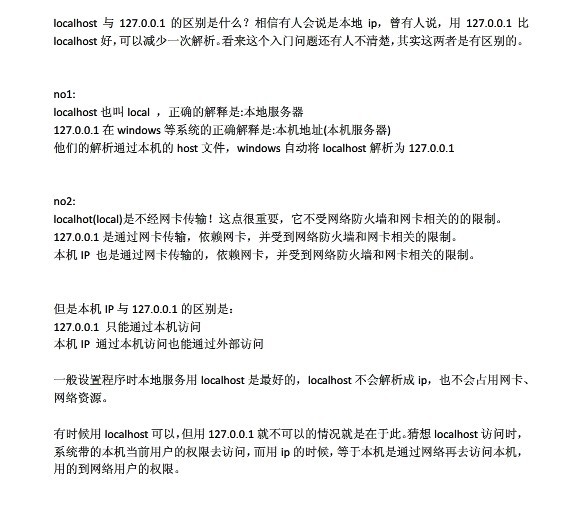
后面3个字符的含义：



1. practice

**1.localhost和127.0.0.1、0.0.0.0区别**

（1）区别



（2）vue需要将localhost改为ip地址，外网才能访问。

