0. 学习资源分享

Windows版Kali：https://github.com/arch3rPro/Pentest-Windows

1. 学习目标

1. 初步建立网络框架体系
2. 掌握计算机基础知识
3. 掌握ARP欺骗攻击

注意：主要通过这次学习掌握一个数据包在网络中是怎么走的？要经过哪些协议？在脑海中初步建立一个网络体系（重点不在ARP欺骗）

2. 前置知识

2.1 MAC地址

MAC地址也叫物理地址、硬件地址。MAC地址的长度为48位(6个字节)；前24位为厂商地址，后24位厂商自主分配

二层交换机根据MAC地址转发

可以通过dos命令`ipconfig /all`查看

```
MAC 地址查询网站：https://itool.co/mac
```

2.2 IP地址

2.2.1.什么是IP地址?

IP地址（Internet Protocol Address 互联网国际地址）是一种在Internet上的给主机编址的方式，它主要是为互联网上的每一个网络和每一台主机分配一个逻辑地址，以此来屏蔽物理地址的差异。

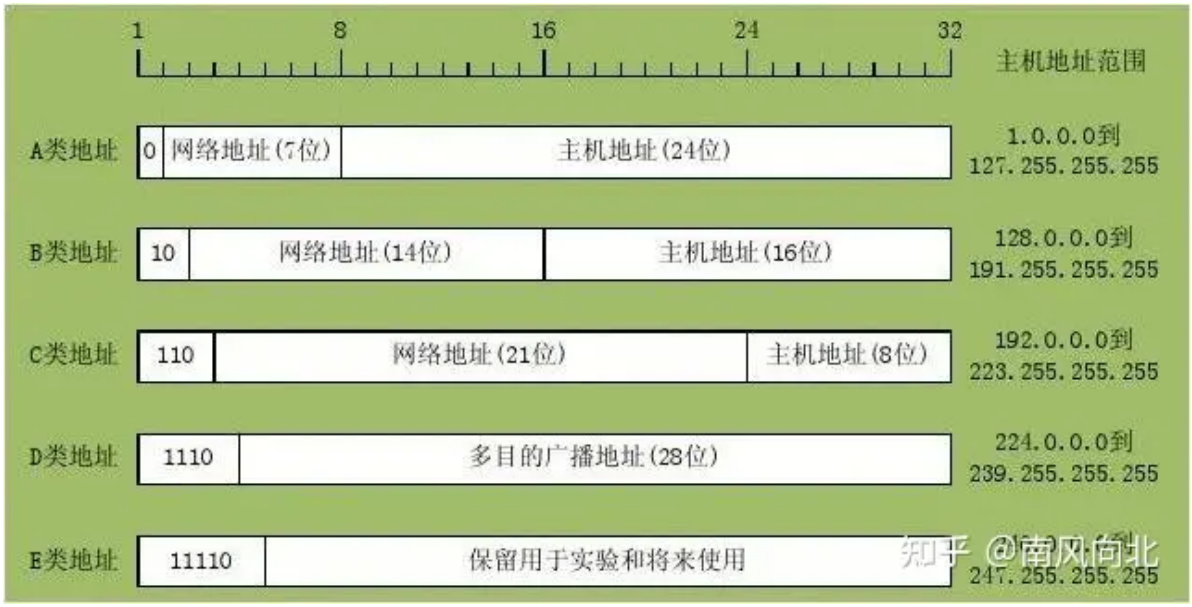
IP地址就像是我们的网购的收货地址（门牌号码），如果你要网购某样东西寄给朋友，那么你就要知道对方的收货地址，这样快递员才能把东西送到朋友手里。

2.2.2.IP地址介

IP地址由网络号与主机号两部分共32位组成，总共4段，每段用“.”隔开，被称为“点分十进制表示法”，如：192.168.1.1

2.2.3、IP地址分类

IP地址根据地址范围分为A到E五类，由下图可见其各类IP地址的主机地址范围：



注：A类地址子网号:0-127，其中0代表任何地址，127为回环测试地址，因此，A类ip地址的实际范围是1-126.

子网掩码的概念：它是就是用来分割子网和区分哪些是同一个网段的，哪些不是同一网段的，通过子网掩码可得知，IP地址的网络位。

```
ip地址：192.168.1.1 子网掩码：255.255.255.0
ip地址：192.168.1.2 子网掩码：255.255.0.0

这两个ip地址虽然在不看掩码的情况下，像是一个网段的，但他们并不是同一个网段内的。
这个可以从子网掩码来判断：

192.168.1.1：   11000000.10101000.00000001. 00000001   （32bit=4字节）
255.255.255.0：11111111.11111111.11111111. 00000000

192.168.1.2：   11000000.10101000.00000001. 00000010
255.255.0.0：   11111111.11111111.00000000. 00000000
```

IP地址还分为私网地址和公网地址，其中私网地址只能在局域网内部使用，不能访问Internet。

私有IP地址：

① A类：10.0.0.0-10.255.255.255

② B类：172.16.0.0-172.31.255.255

③ C类：192.168.0.0-192.168.255.255

④ 自动私有地址：169.254.0.0/16（当计算机无法获取IP地址时自动配置）

特殊地址：

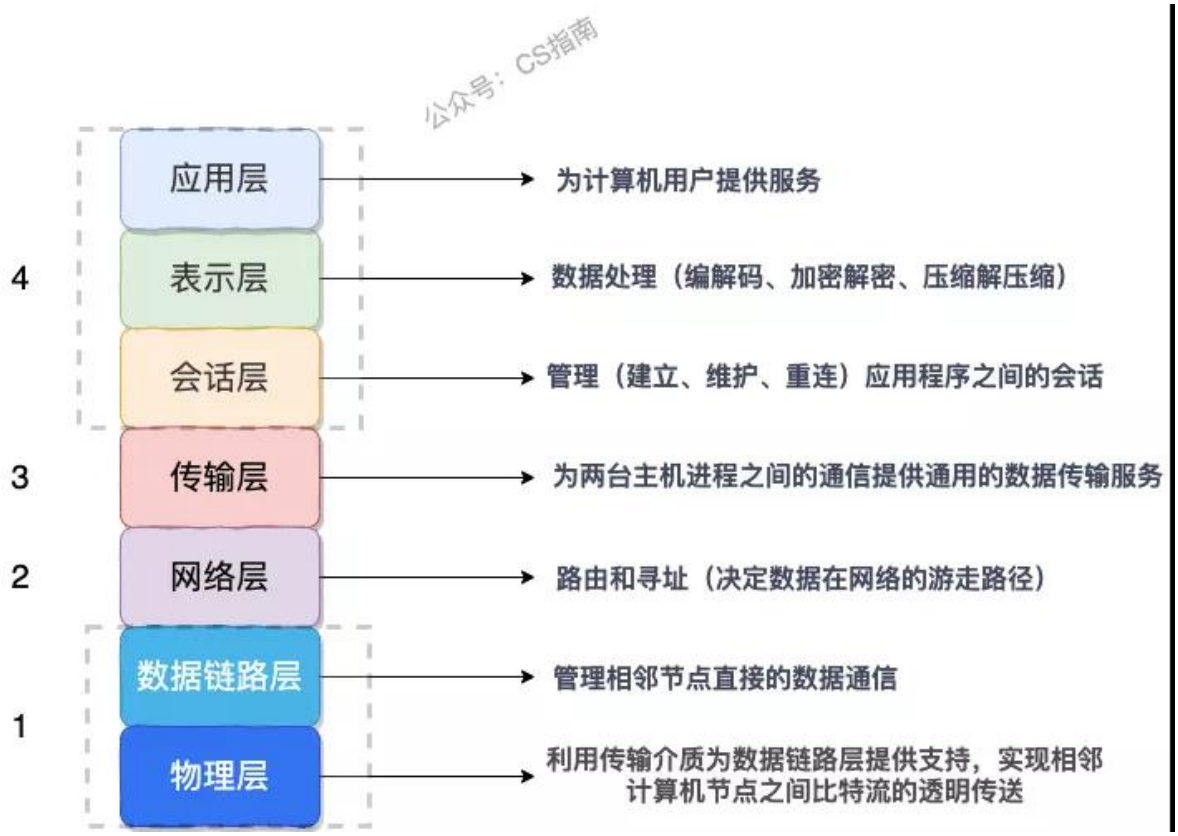
本地环回地址：127.0.0.1（测试本机的网络配置，能ping通127.0.0.1说明本机的网卡和IP协议安装都没有问题）

本地广播地址：255.255.255.255

2.3 OSI七层模型

2.3.1 为什么会有OSI七层模型?

网络创建之初，每家公司都有它的开发标准。这就导致了A公司开发的产品不能和B公司的产品一起使用，这个时候国际标准化组织提出了OSI七层模型



2.3.2 什么是OSI七层模型?

应用层

应用层位于 OSI 参考模型的第七层，其作用是通过应用程序间的交互来完成特定的网络应用

该层协议定义了应用进程之间的交互规则，通过不同的应用层协议为不同的网络应用提供服务。例如域名系统 DNS，支持万维网应用的 HTTP 协议，电子邮件系统采用的 SMTP协议等

在应用层交互的数据单元我们称之为**报文**

常见的应用层协议：HTTP、DNS、Telnet、SSH等

应用层是直接面向用户的

单位：报文

表示层

表示层的作用是使通信的应用程序能够解释交换数据的含义，其位于 OSI参考模型的第六层，向上为应用层提供服务，向下接收来自会话层的服务

该层提供的服务主要包括**数据压缩**，**数据加密**以及数据描述，使应用程序不必担心在各台计算机中表示和存储的内部格式差异

单位：数据格式

会话层

会话层就是负**责建立、管理和终止**表示层实体之间的**通信会话**

该层提供了数据交换的定界和同步功能，包括了建立检查点和恢复方案的方法

单位：会话

传输层

传输层的主要任务是**为两台主机进程之间的通信提供服务，处理数据包错误、数据包次序**，以及其他一些关键传输问题

传输层向高层屏蔽了下层数据通信的细节。因此，它是计算机通信体系结构中关键的一层

其中，主要的传输层协议是TCP和UDP

单位：段

例子：家具运输

网络层

两台计算机之间传送数据时其通信链路往往不止一条，所传输的信息甚至可能经过很多通信子网

网络层的主要任务就是选择合适的网间路由和交换节点，确保数据按时成功传送

在发送数据时，网络层把传输层产生的报文或用户数据报封装成分组和包，向下传输到数据链路层

在网络层使用的协议是无连接的网际协议（Internet Protocol）和许多路由协议，因此我们通常把该层简单地称为 IP 层

获取最短/最优路由

单位：包

数据链路层

数据链路层通常也叫做链路层，在物理层和网络层之间。两台主机之间的数据传输，总是在一段一段的链路上传送的，这就需要使用专门的链路层协议

在两个相邻节点之间传送数据时，数据链路层将网络层交下来的 **IP数据报组装成帧，在两个相邻节点间的链路上传送帧**

每一帧的数据可以分成：报头head和数据data两部分:

head 标明数据发送者、接受者、数据类型，如 MAC地址
data 存储了计算机之间交互的数据
通过控制信息我们可以知道一个帧的起止比特位置，此外，也能使接收端检测出所收到的帧有无差错，如果发现差错，数据链路层能够简单的丢弃掉这个帧，以避免继续占用网络资源

单位：帧

物理层

作为OSI 参考模型中最低的一层，物理层的作用是实现计算机节点之间比特流的透明传送

该层的主要任务是确定与传输媒体的接口的一些特性（机械特性、电气特性、功能特性，过程特性）

高低电频与1001之间的相互转换

单位：比特

2.4 TCP/IP 4层模型

> 由于OSI七层模型分的太细了，所以后面有推出了 TCP/IP 4层模型

2.4.1. 网络访问层

1.1 作用

(1) 实现网卡接口的网络驱动，以处理数据在以太网线等物理媒介上的传输

(2) 网络驱动程序隐藏了不同物理网络的不同电气特性，为上层协议提供一个统一的接口

1.2 协议应用

ARP和RARP(Reverse Address Resolve Protocol)即逆地址解析协议，该协议实现了IP地址和物理地址(MAC地址)之间的转换

2.4.2. 网络层

2.1 作用

网络有分局域网(LAN, Local Area Network)和广域网(WAN, Wide Area Network)。对于后者通常需要使用众多分级的路由器来连接分散的主机或者LAN，即通讯的两台主机一般不是直接连接，而是通过多个中间节点(路由器)连接的，从而形成网络拓扑连接。 **(寻找最优路径)**

(1) 网络层的任务之一就是选择这些中间节点，以确定两台主机间的通讯路径。

(2) 其次网络层对上层协议隐藏了网络拓扑连接的细节，在使得传输层看来通讯双方是直接连接的

2.2 协议应用

(1) IP协议: IP协议(Internet Protocol)是网络层最核心的协议，它根据数据包的目的IP地址来决定如何投递该数据包。若数据包不可直接发送给目标主机，那么IP协议就为它寻找一个合适的下一跳路由器，并将数据包交付给该路由器去转发，如此循环直至到达目标主机或者发送失败而丢弃该数据包。

(2) ICMP协议: ICMP协议(Internet Control Message Protocol，因特网控制报文协议)是IP协议的补充，用于检测网络的连接状态 **(PING)** ，如ping应用程序就是ICMP协议的使用。ICMP包发送是不可靠的，所以不能依靠接收ICMP包解决网络问题；ICMP与TCP/UDP不同，它们是传输层协议，虽然都具有类型域和代码域，但是前者和后者不同，ping用到的ICMP协议，不是端口。ICMP协议使用的是IP协议而非使用下层协议提供的的服务，所以严格来讲它并非网络层协议，而是网络层程序。

2.4.3. 传输层

3.1 作用

传输层的作用是为应用程序提供端对端通讯的"错觉"，即为应用程序隐藏了数据包跳转的细节，负责数据包的收发、链路超时重连等。

3.2 协议应用

(1) TCP协议: TCP协议(Transmission Control Protocol, 传输控制协议)为应用程序提供可靠的、面向连接的、基于流的服务，具有超时重传、数据确认等方式来确保数据包被正确发送到目的端。因此**TCP服务是可靠的**，使用TCP协议通讯的双方必须先建立起TCP连接，并在系统内核中为该连接维持一些必要的数据结构，比如连接的状态，读写缓冲区，多个定时器等。当通讯结束时双方必须关闭连接以释放这些内核数据。基于流发送意思是数据是没有长度限制，它可源源不断地从通讯的一段流入另一端。

(2) UDP协议: UDP协议(User Datagram Protocol, 用户数据报协议)与TCP协议相反，它为应用程序提供的是**不可靠的**、无连接的基于数据报的服务。

无连接: 通讯双方不保持一个长久的联系，因此应用程序每次发送数据都要明确指定接收方的地址;

基于数据报的服务: 这是相对于数据流而言的，每个UDP数据报都有一个长度，接收端必须以该长度为最小单位将其内容一次性读出，否则数据将被截断。

UDP不具有发送时是被重发功能，所以UDP协议在内核实现中无需为应用程序的数据保存副本，当UDP数据报被成功发送之后，UDP内核缓冲区中该数据报就被丢弃了。

(3) SCTP协议: SCTP(Stream Control Transmission Protocol, 流控制传输协议)是为了在因特网上传输电话信号而设计的。

2.4.4. 应用层

4.1 作用

前面所述的三层负责处理网络通讯的相关细节，这部分需要稳定高效，因此它们是在操作系统的内核空间中，而应用层是在用户空间实现的，负责处理众多业务逻辑，如文件传输、网络管理。

4.2 协议应用

应用层的协议很多，如:

(1) telne协议: 远程登录协议，它使我们能在本地完成远程任务

(2) OSPF协议: OSPF协议(Open Shorttest Path First, 开放最短路径优先)是一种动态路由更新协议，用于路由器之间的通讯，以告知对方自身的路由信息

(3) DNS协议: DNS协议(Domain Name Service, 域名服务)提供机器域名到IP地址的转换。如百度的机器域名是www.baidu.com，对应的IP地址是http://119.75.217.109/。

2.5 数据包的封装

```
封装
|data:emall|
|src:port|dst:port|data:email|
|src:ip|dst:ip||src:port|dst:port|data:email|
|src:mac|dst:mac|src:ip|dst:ip||src:port|dst:port|data:email|
1001

解封装
1001
|src:mac|dst:mac|src:ip|dst:ip||src:port|dst:port|data:email|    判断MAC地址
|src:ip|dst:ip||src:port|dst:port|data:email|                    判断IP地址
|src:port|dst:port|data:email|                                   判断端口号来决定交给哪个服务
|data:email|
```

ICMP 数据包分析

2.6 举例

`192.168.0.1 ping 192.168.0.2`会发生什么?

```
封装ICMP报文：
网络层：源IP地址：192.168.0.1，目标IP地址：192.168.0.2
数据链路层：源Mac地址：11-11-11-11-11-11，目标Mac地址：？？？
```

2.7 ARP协议

2.7.1 概述

如果要给TCP/IP协议栈选择一个**"最不安全的协议"**，那么我会毫不犹豫把票投给ARP协议。我们经常听到的这些术语，包括"网络扫描"、"内网渗透"、"中间人拦截"、"局域网流控"、"流量欺骗"，基本都跟ARP脱不了干系。大量的安全工具，例如大名鼎鼎的Cain、功能完备的Ettercap、操作傻瓜式的P2P终结者，底层都要基于ARP实现。

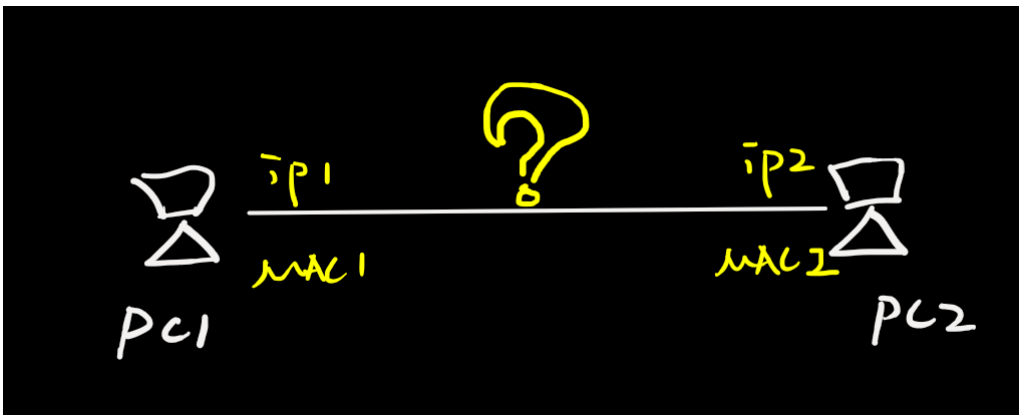
①ARP（Address Resolution Protocol）即地址解析协议， 用于实现从 IP 地址到 MAC 地址的映射，即询问目标IP对应的MAC地址。

②在网络通信中，主机和主机通信的数据包需要依据OSI模型从上到下进行数据封装，当数据封装完整后，再向外发出。所以在局域网的通信中，**不仅需要源目IP地址的封装，也需要源目MAC的封装。**

③一般情况下，上层应用程序更多关心IP地址而不关心MAC地址，所以需要通过ARP协议来获知目的主机的MAC地址，完成数据封装。

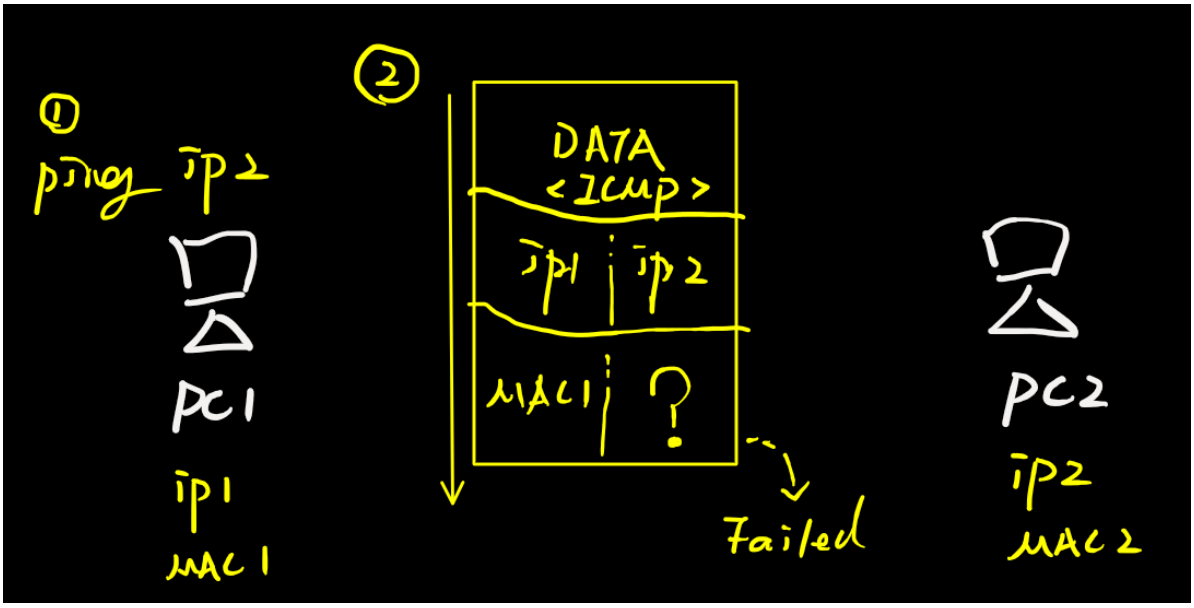
2.7.2 原理

同一个局域网里面，当PC1需要跟PC2进行通信时，此时PC1是如何处理的?



根据OSI数据封装顺序，发送方会自顶向下（从应用层到物理层）封装数据，然后发送出去

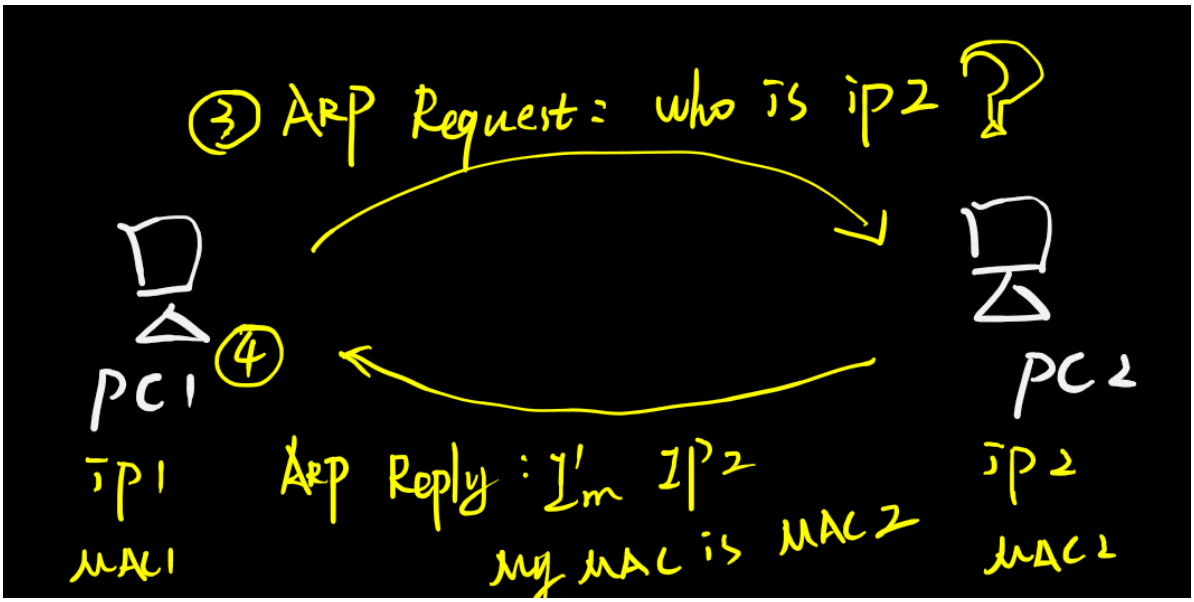
这里以PC1 ping PC2的过程举例==>



PC1封装数据并且对外发送数据时，上图中出现了"failed"，即数据封装失败了，为什么?

我们给PC1指令-"ping ip2"，这就告知了目的IP，此时PC1便有了通信需要的源目IP地址，但是PC1仍然没有通信需要的目的MAC地址。这就好比我们要寄一个快递，如果在快递单上仅仅写了收件人的姓名（IP），却没有写收件人的地址（MAC），那么这个快递就没法寄出，因为信息不完整。

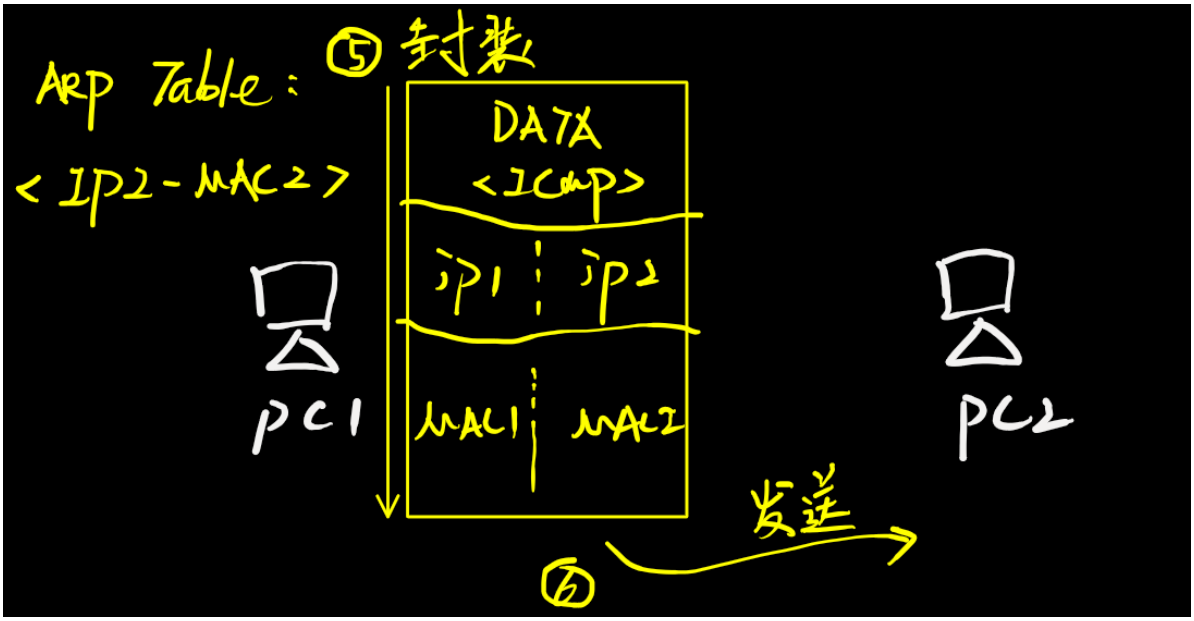
那么，现在PC1已经有了PC2的IP地址信息，如何获取到PC2的MAC地址呢？此时，ARP协议就派上用场了。我们接着上面这张图，继续==>



通过第三和第四步骤，我们看到PC1和PC2进行了一次ARP请求和回复过程，通过这个交互工程，PC1具备了PC2的MAC地址信息。接下来PC1会怎么做呢？在真正进行通信之前，

PC1还会将PC2的MAC信息放入本地的【ARP缓存表】，表里面放置了IP和MAC地址的

映射信息，例如 IP2<->MAC2。接下来，PC1再次进行数据封装，正式进入PING通信，如下==>



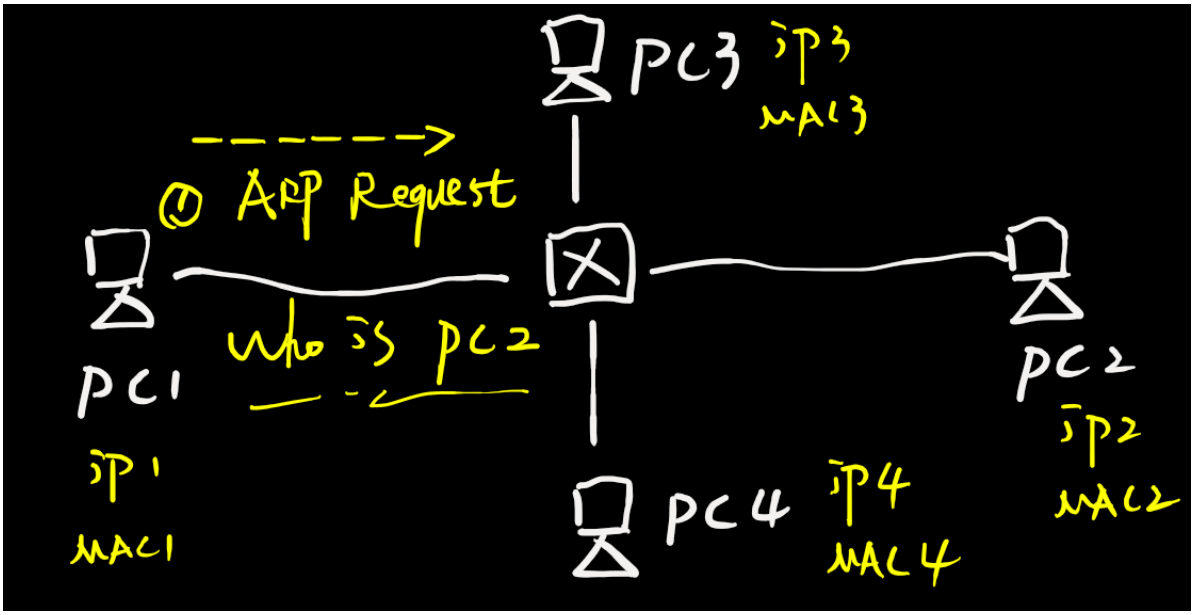
小结：经过上面6个步骤的处理，PC1终于把数据包发送出去了，之后便可以进行正常的通信了。看到了吧，ARP的功能和实现过程是如此的简单：它在发送方需要目标MAC地址的时及时出手，通过"一问一答"的方式获取到特定IP对应的MAC地址，然后存储到本地【ARP缓存表】

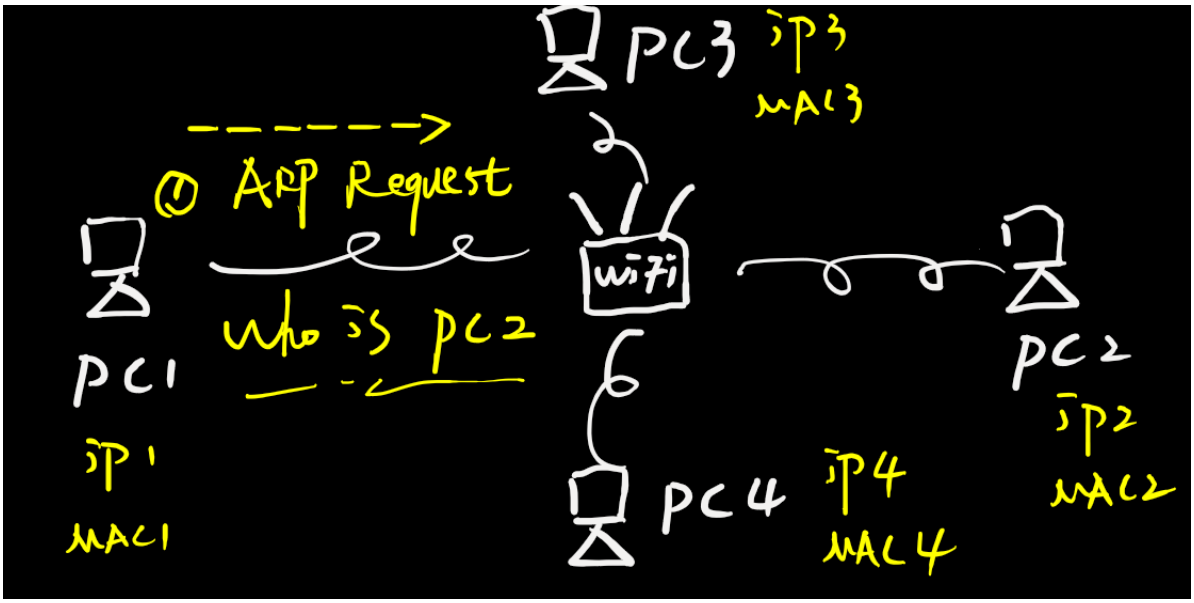
```
C:\Users\19374>arp -a

Interface: 192.168.56.1 --- 0x3
  Internet Address      Physical Address      Type
  192.168.56.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.2             01-00-5e-00-00-02     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
```

2.7.3 ARP原理之广播请求单播回应

上面的图解过程看上去简单又纯粹，好像我们就已经把这个协议的精髓全部get到，但其实，我们只是刚揭开了它的面纱，接下来我们才真正进入正题。例如，上面的图解过程中，整个局域网（LAN）只有PC1和PC2两个主机，所以这个一问一答过程非常的顺畅。而实际网络中，这个LAN可能有几十上百的主机，那么请问，PC1如何将这个【ARP请求包】顺利的交给PC2，而PC2又如何顺利的把【ARP回应包】返回给PC1？我们看下面的图：

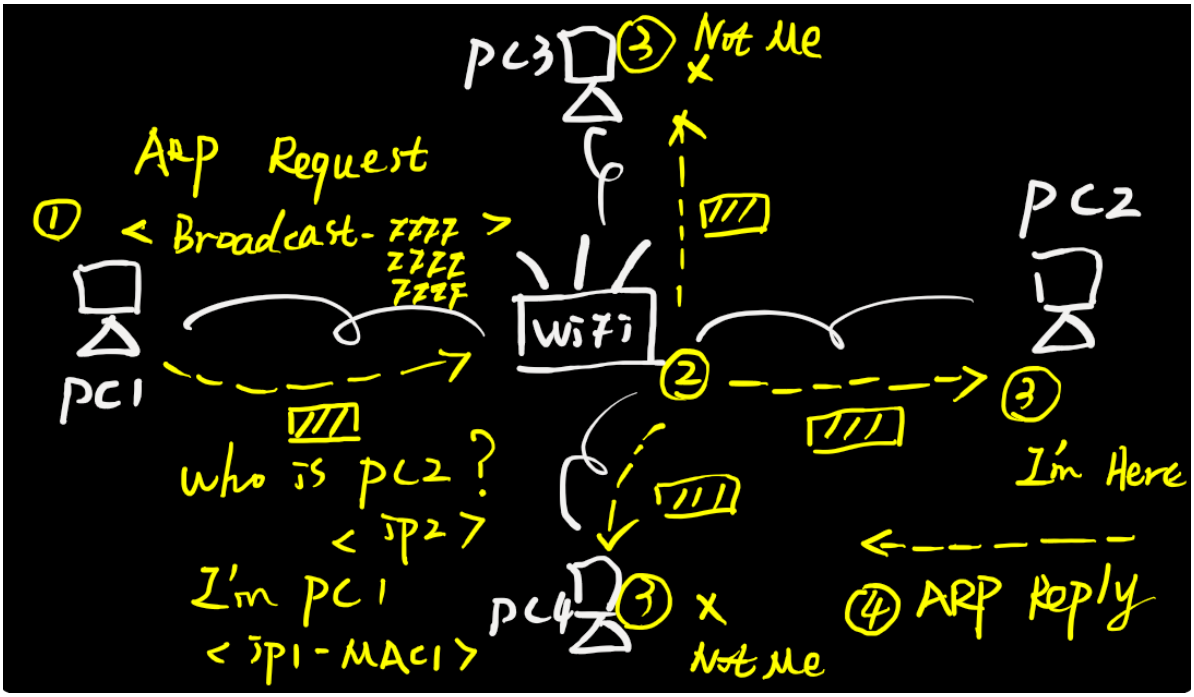




为了营造出"几十上百"的效果，这里多添加了2个主机进来ﾍ(ﾟ∀ﾟ)ﾉ，并且增加了有线和无线的环境。那么，在多主机环境下，PC1现在发出的ARP请求包，怎么交到PC2手里？

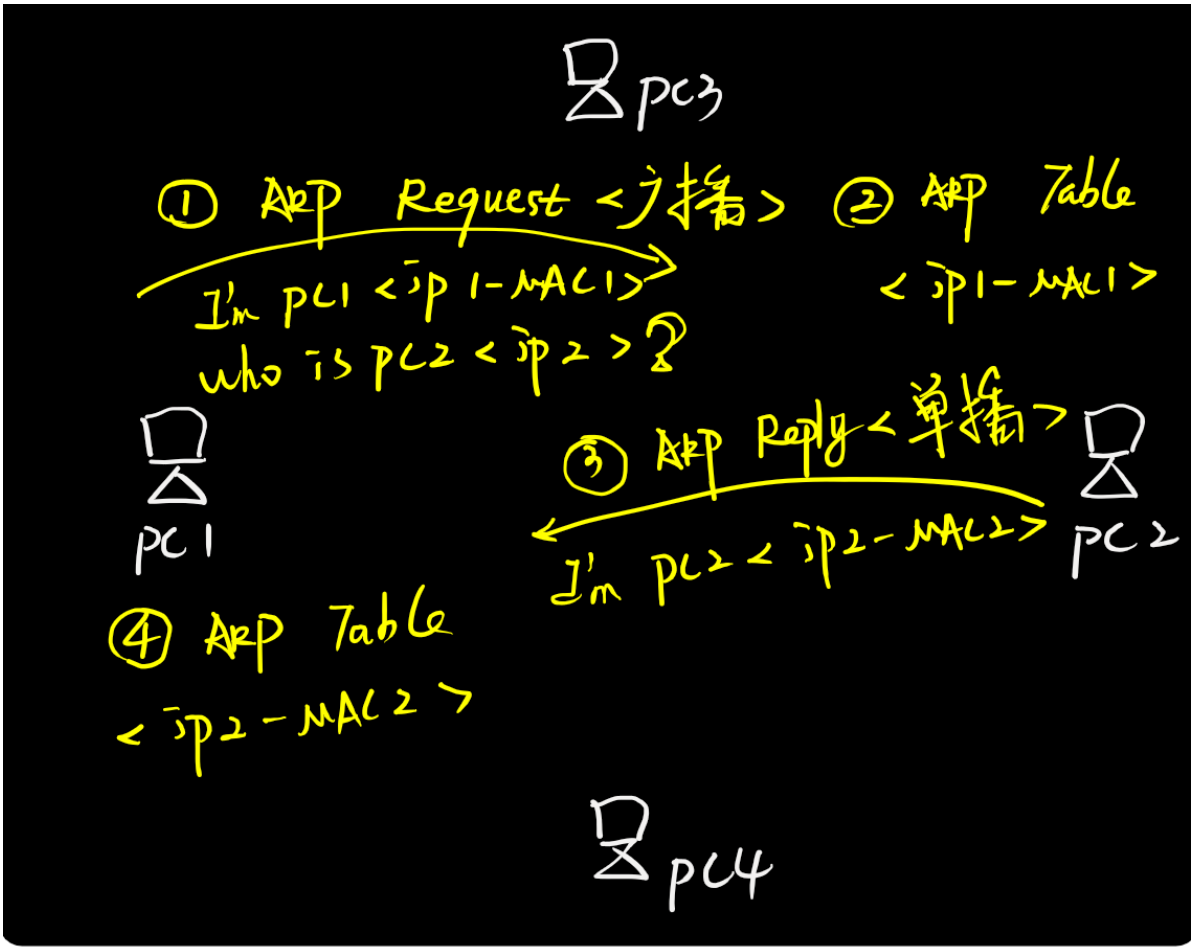
这时，ARP协议就需要采用以太网的"广播"功能：将请求包以广播的形式发送，交换机或WiFi设备（无线路由器）收到广播包时，会将此数据发给同一局域网的其他所有主机。

注明：什么是广播？对于初学者而言，我们只需要知道，大部分的广播包，它们有一个共同特征：二层封装时目的MAC是全f（ffff.ffff.ffff）或三层封装时目的IP是全1（255.255.255.255）。可以这样更方便的记住：目的地址最大的，就是广播。



根据上图我们看到，PC1发送的请求广播包同时被其他主机收到，然后PC3和PC4收到之后（发现不是问自己）则丢弃。而PC2收到之后，根据请求包里面的信息（有自己的IP地址），判断是给自己的，所以不会做丢弃动作，而是返回ARP回应包。

ARP请求是通过广播方式来实现的，那么，PC2返回ARP回应包，是否也需要通过广播来实现呢？答案是否定的。**大部分网络协议在设计的时候，都需要保持极度克制，不需要的交互就砍掉，能合并的信息就合并，能不用广播就用单播，以此让带宽变得更多让网络变得更快。**那么，ARP回应包是如何处理的？这里需要特别关注ARP请求包的内容，在上面的图解里面，ARP请求包的完整信息是：我的IP地址是IP1，MAC地址是MAC1，请问谁是PC2，你的IP2对应的MAC地址是多少？简单来说，**ARP请求首先有"自我介绍"**，然后才是询问**。这样的话，PC2在收到请求之后，就可以将PC1的IP和MAC映射信息存储在本地的【ARP缓存表】，既然知道PC1在哪里，就可以返回ARP单播回应包。

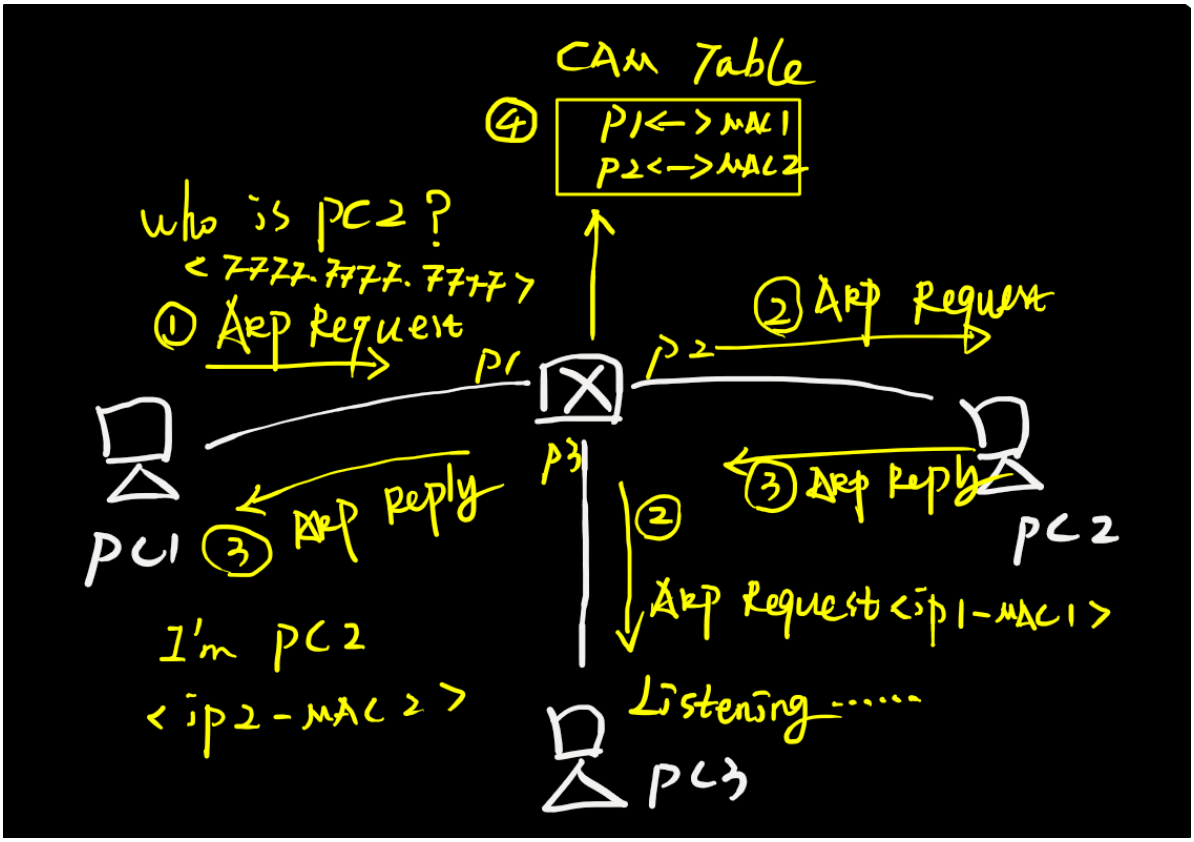


这张图我们需要得到两个信息：①被询问者PC2先生成了ARP映射信息，然后才是询问者PC1；②PC3和PC4等其他主机，无法收到这个ARP回应包，因为是单播形式。

小结：ARP协议通过"一问一答"实现交互，但是"问"和"答"都有讲究，"问"是通过广播形式实现，"答"是通过单播形式。

3. ARP欺骗

正常ARP协议过程



①PC1需要跟PC2通信，通过ARP请求包询问PC2的MAC地址，由于采用广播形式，所以交换机将ARP请求包从接口P1广播到P2和PC3。（注：**交换机收到广播/组播/未知帧都会其他接口泛洪**）

②PC2根据询问信息，返回ARP单播回应包；此时PC3作为攻击者，没有返回ARP包，但是处于"**监听**"状态，为后续攻击做准备。

③PC1和PC2根据ARP问答，将各自的ARP映射信息（IP-MAC）存储在本地ARP缓存表。

④交换机根据其学习机制，记录MAC地址对应的接口信息，存储在**CAM缓存表**（也称为MAC地址表）。交换机收到数据包时，会解封装数据包，根据**目标MAC**字段进行转发。

关于上面的图解，我们要记住这些关键知识（敲黑板！）：

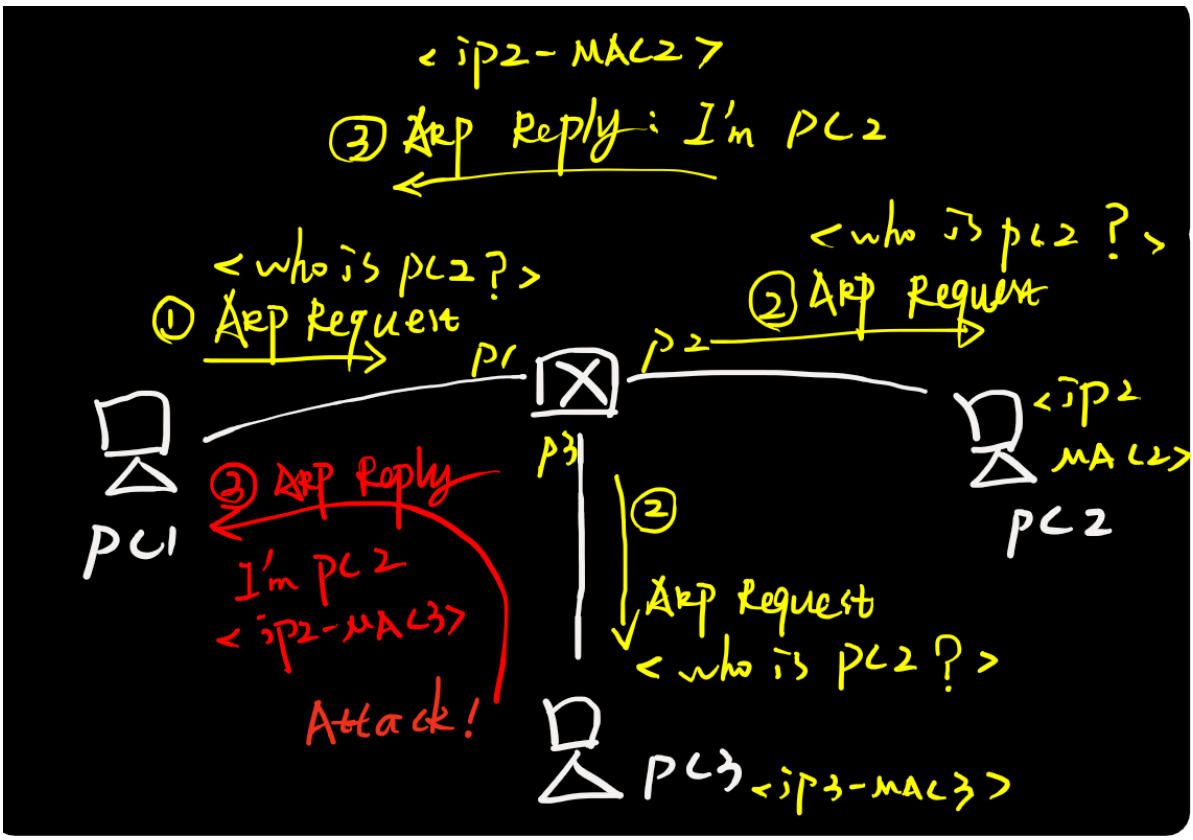
①主机通信需要查找ARP表，而交换机通信需要查找CAM表（路由器则查找Route表）。

注：ARP表：ip<->mac CAM表：mac<->port （Route表：route<->port）

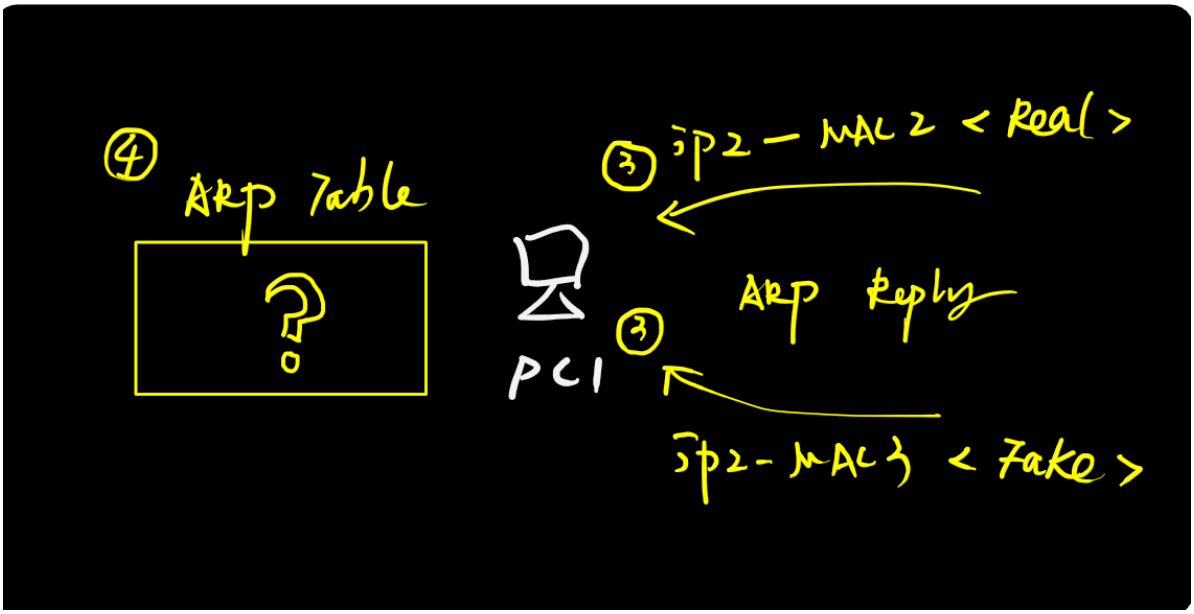
②交换机基于源MAC地址学习，基于目的MAC地址转发。

③同一局域网内，攻击者可以根据主机的ARP广播请求监听其IP和MAC信息。

ARP协议攻击过程



正常情况下，若收到的ARP请求不是给自己的，则直接丢弃；而这里PC3（Hacker）在监听之后，发起了ARP回应包：**我就是PC2**（IP2-MAC3）**。从拓扑可以出现，PC3明明是IP3对应MAC3，很显然这就是一个ARP欺骗行为。于此同时，PC2正常的ARP回应包也交到了PC1手中，我们来看PC1接下来如何处理的：

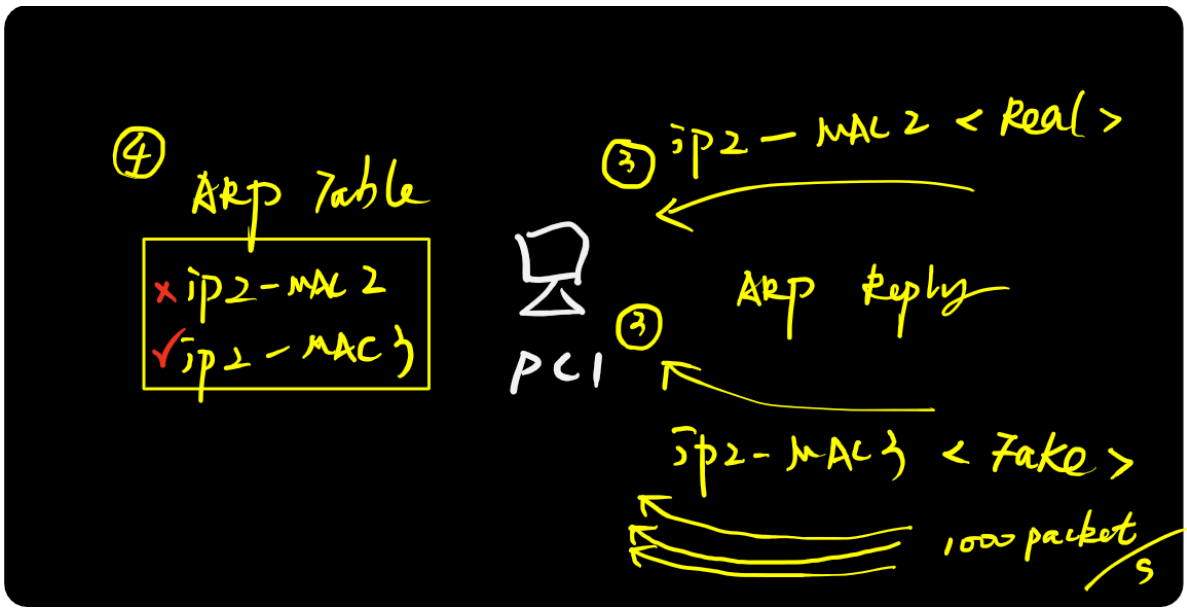


PC1收到两个ARP回应包，内容分别如下：

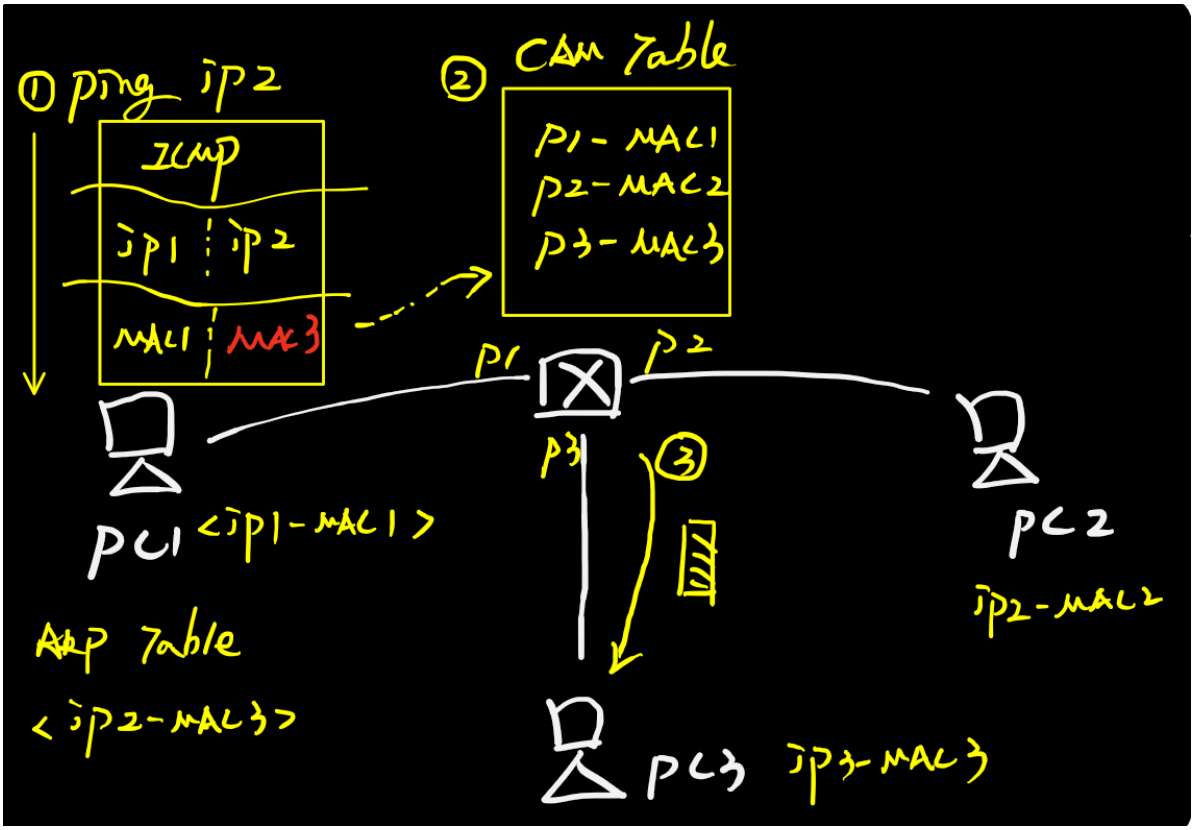
③我是PC2，我的IP地址是**IP2**，我的MAC地址是**MAC2**；

③我是PC2，我的IP地址是**IP2**，我的MAC地址是**MAC3**；

PC1一脸懵：咋回事？还有这操作？**不管了，我选最新的！（后到优先）**



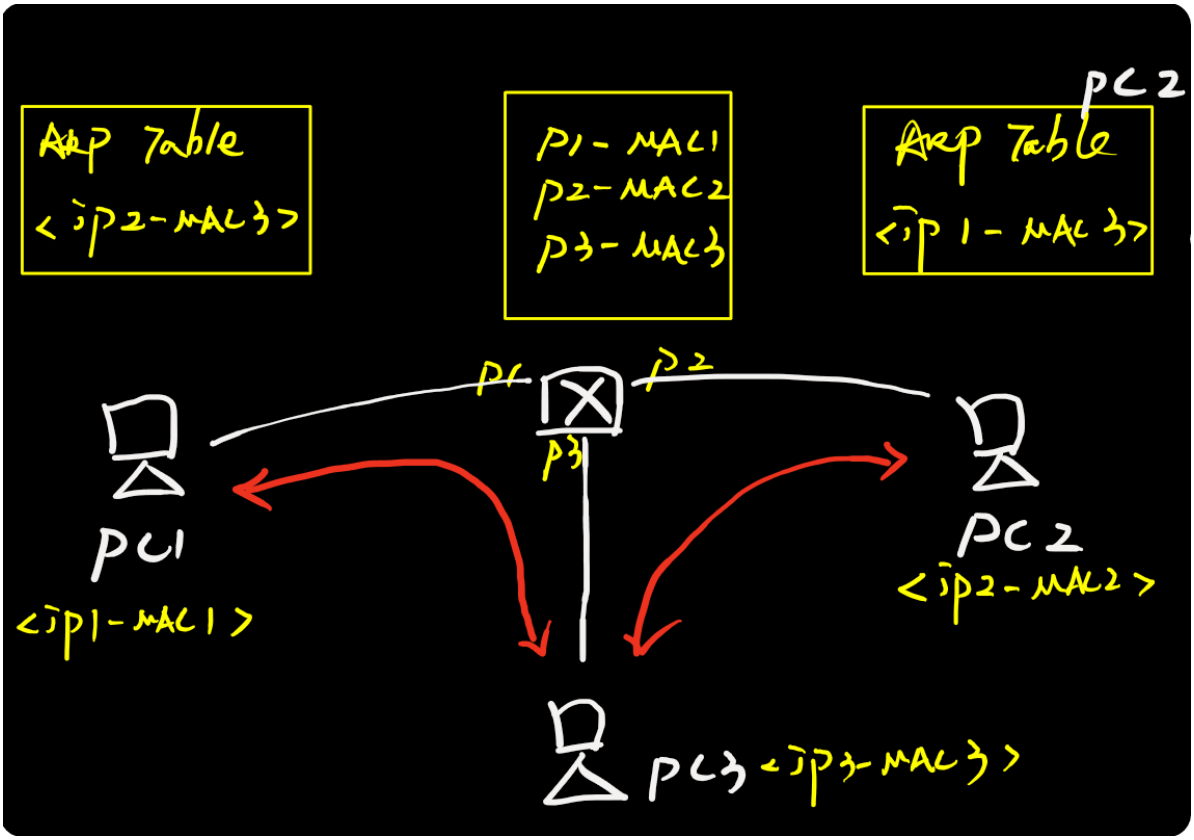
小白 vs 黑客，很明显的较量，PC1最终记录的是虚假的ARP映射：IP2<->MAC3，得到错误信息的PC1，接下来会发生什么情况呢？（我们以PC1 ping PC2为例）



根据数据封装规则，当PC1要跟PC2进行通信时，无论是发生PING包还是发送其他数据，

首先要查找ARP表，然后在网络层打上源目IP，在链路层打上源目MAC，然后将数据包发送给交换机。交换机收到之后对数据进行解封装，并且查看CAM表（基于目的MAC转发），由于目标MAC3对应Port3，所以交换机自然而然将其转发给PC3。

就这样，PC1本来要发给PC2的数据包，落到了PC3（Hacker）手里，这就完成了一次完整的ARP攻击。反过来，**如果PC2要将数据包发送给PC1**，PC3仍然可以以同样的ARP欺骗实现攻击**，这就有了下面这张图（PC3既欺骗了PC1，也欺骗了PC2）。



此时，PC1和PC2的通信数据流被PC3拦截，形成了典型的"**中间人攻击**"。那么，一旦被攻击并拦截，攻击者能做什么，普通用户又会遭受什么损失？这里给大家举几个常见的例子=>

①攻击者既然操控了数据流，那么直接断开通信是轻而易举的，即**"断网攻击"**，例如，PC1发给PC2的数据在PC3这里可以直接丢弃，而如果这里的PC2是一台出口路由器（无线路由器），那就意味着PC1直接无法连上互联网。

②"断网攻击"显然容易被发现，而且比较"残忍"，所以就有了更加常见的应用-**"限速"**。例如，在宿舍上网突然很慢，在网吧上网突然打不开网页，如果这个网络没有安全防御，那么很有可能有"内鬼"。

③其实无论是"断网攻击"还是"限速"，整体还是比较"善良"，因为这里流量里面的核心数据还没有被"提取"出来。如果攻击者是一名真正的黑客，他的目的一定不会这么无聊，因为内网流量对于黑客是没有太大价值的，而只有**"用户隐私"**，例如常见网站的登录账号密码，这些才是最有价值的。

4. ARP欺骗攻击

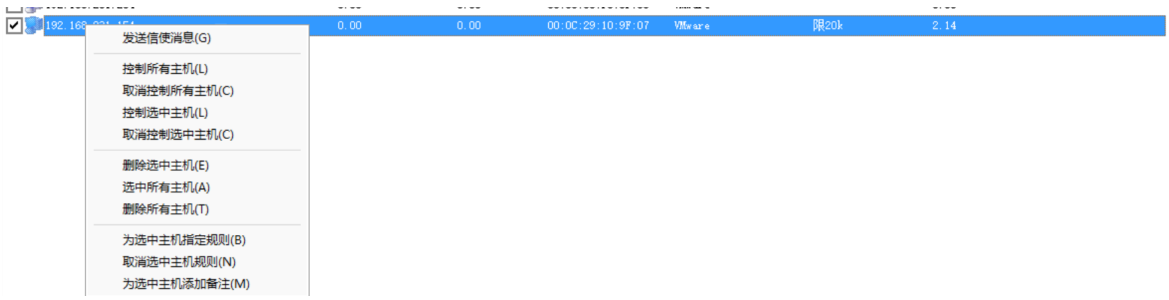
1. 启动P2P终结者，选择“智能选择网卡”


P2P终结者4.34[当前网络管理优先级:最高]
系统(F) 软件设置(C) 其他功能(O)
启动提速 停止提速 扫描网络
局域网管理 本机网速使用
IP地址
系统设置
网络设置 控制设置 常规设置 界面设置
说明：请首先指定上网方式，再使用“智能选择控制网卡”功能！！
请您指定本机上网方式及网络环境：
智能检测网络环境
我接入了局域网，并通过路由器上网
请选择连接到待控制网段的网卡：
智能选择控制网卡
Intel(R) PRO/1000 MT Network Connection - VirtualBox Bri
IP地址： 172.16.70.130
MAC地址： 00:0C:29:F5:2D:96
子网掩码： 255.255.255.0
网关地址： 172.16.70.2
网关MAC地址： 00:50:56:FC:66:62
确定
取消
恢复配置

2. 扫描网络


P2P终结者4.34[当前网络管理优先级:最高]
系统(F) 软件设置(C) 其他功能(O) 帮助(H)
启动提速 停止提速 扫描网络 高级选项 规则设置 时间计划设置 提升优先级 流量查询 网速测试 备份配置
局域网管理 本机网速使用 控制日志列表
IP地址 机器名 上传速度(KB/S) 下载速度(KB/S) MAC地址
172.16.70.130 控制机 0.00 0.00 00:0C:29:F5:2I
172.16.70.1 -- 0.00 0.00 00:50:56:C0:0C
172.16.70.131 -- 0.00 0.00 00:0C:29:43:01
172.16.70.254 -- 0.00 0.00 00:50:56:E6:1

3. 指定规则


发送信使消息(G)
控制所有主机(L)
取消控制所有主机(C)
控制选中主机(L)
取消控制选中主机(C)
删除选中主机(E)
选中所有主机(A)
删除所有主机(T)
为选中主机指定规则(B)
取消选中主机规则(N)
为选中主机添加备注(M)