使用WireShark简单分析ICMP、ARP报文

**ICMP协议介绍**

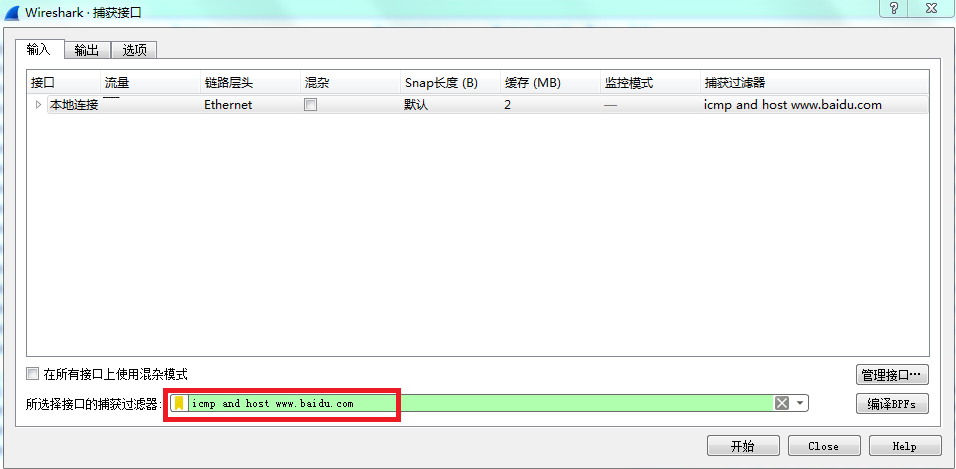
1、ICMP是“Internet Control Message Protocol”（Internet控制消息协议）的缩写。它是TCP/IP协议族的一个子协议，用于在IP主机、路由器之间传递控制消息。控制消息是指网络通不通、主机是否可达、路由是否可用等网络本身的消息。这些控制消息虽然并不传输用户数据，但是对于用户数据的传递起着重要的作用。

2、ICMP报文作为IP层数据报的数据，加上数据报的首部，组成数据报发送出去。

3、ICMP报文的种类有两种，即ICMP差错报告报文和ICMP询问报文。

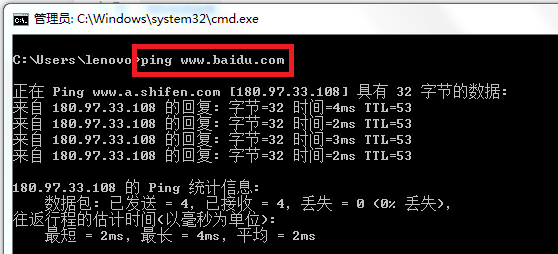
利用ping 程序分析一个ICMP协议交互过程:

首先打开Wireshark，设置**过滤条件：icmp and host www.baidu.com**

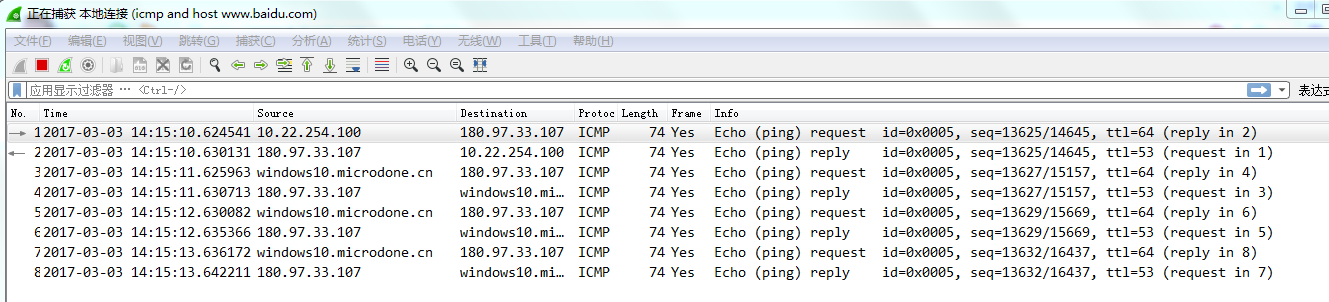


点击开始，发现现在抓取不到任何包：

**打开命令行窗口，执行命令：ping www.baidu.com**

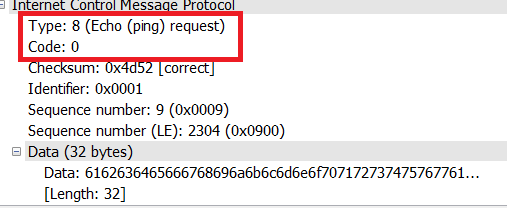


这时可以看到数据包抓取页面抓取到了8包，与命令行显示的已发送和已接受的包的数量是一致的：

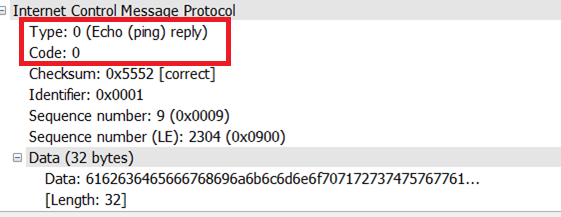


在这个试验中，可以发现，icmp的报文就只有两种，请求和应答：

请求：



应答：

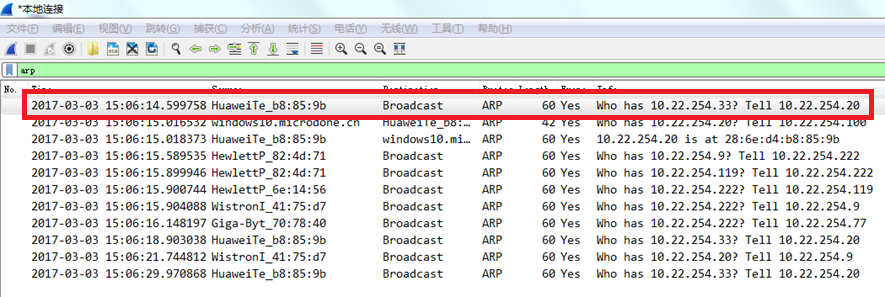


这两个报文的type不一样，8代表请求，0代表应答；code都为0，表示为回显应答；标示符和序列号都是一样的，表示这两个报文是配对的。

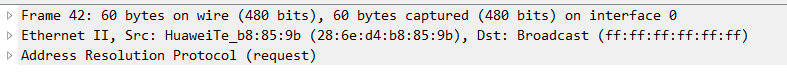
**ARP协议：**

地址解析协议，即ARP（Address Resolution Protocol），是根据[IP地址](http://baike.baidu.com/view/3930.htm" \t "_blank)获取[物理地址](http://baike.baidu.com/view/883168.htm)的一个[TCP/IP协议](http://baike.baidu.com/view/7649.htm)。[主机](http://baike.baidu.com/view/23880.htm)发送信息时将包含目标IP地址的ARP请求**广播**到网络上的所有主机，并接收返回消息，以此确定目标的物理地址；收到返回消息后将该IP地址和物理地址存入本机ARP缓存中并保留一定时间，下次请求时直接查询ARP缓存以节约资源。地址解析协议是建立在网络中各个主机互相信任的基础上的，网络上的主机可以自主发送ARP应答消息，其他主机收到应答报文时不会检测该报文的真实性就会将其记入本机ARP缓存；由此攻击者就可以向某一主机发送伪ARP应答报文，使其发送的信息无法到达预期的主机或到达错误的主机，这就构成了一个[ARP欺骗](http://baike.baidu.com/view/155386.htm)。[ARP命令](http://baike.baidu.com/view/935158.htm)可用于查询本机ARP缓存中IP地址和[MAC地址](http://baike.baidu.com/view/69334.htm" \t "_blank)的对应关系、添加或删除静态对应关系等。相关协议有[RARP](http://baike.baidu.com/view/32772.htm)、[代理ARP](http://baike.baidu.com/view/547486.htm" \t "_blank)。[NDP](http://baike.baidu.com/view/1420887.htm" \t "_blank)用于在[IPv6](http://baike.baidu.com/view/5228.htm" \t "_blank)中代替地址解析协议。

在下图中Filter后面的编辑框中输入：arp（注意是小写），然后回车或者点击“Apply”按钮



现在只有ARP协议了，其他的协议数据包都被过滤掉了。注意到中间部分的三行前面都有一个“+”，点击它，这一行就会被展开。如下图所示：



现在展开第一行。看到的结果如下：



在上图中我们看到这个帧的一些基本信息：

帧的编号：42（捕获时的编号）

帧的大小：60字节。

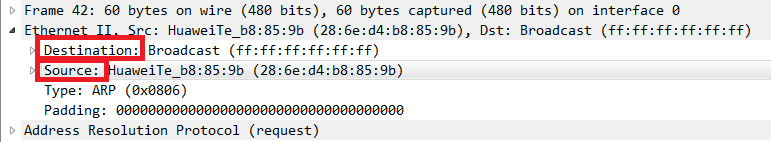
帧被捕获的日期和时间：Mar 3,2017……

帧距离前一个帧的捕获时间差：0.22……

帧距离第一个帧的显示时间差：0.00……

帧装载的协议：ARP

现在展开第二行：



我们可以看到：

帧中封装的协议类型：0x0806，这个是ARP协议的类型编号。

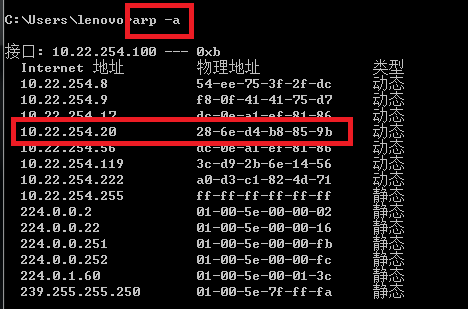
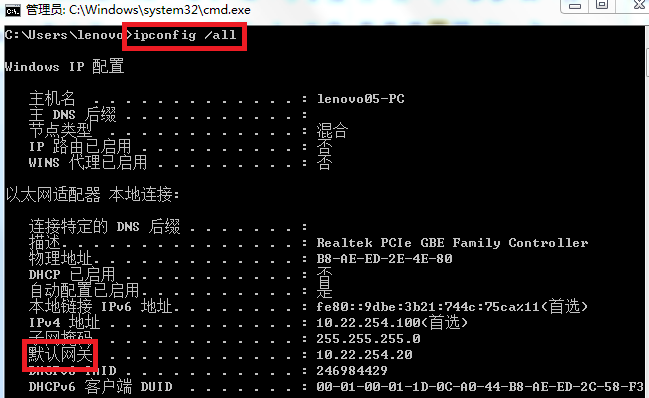
目的地址（Destination）：ff:ff:ff:ff:ff:ff（这是个MAC地址，这个MAC地址是一个广播地址，就是局域网中的所有计算机都会接收这个数据帧）

源地址（Source）：Elitegro\_2d:e7:db（00:0d:87:2d:e7:db）

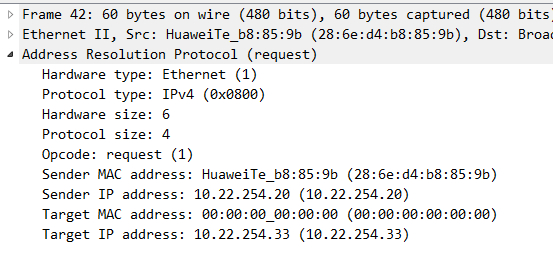


**这里的mac是我的网关的mac，这是如何确认的呢：**

通过**ipconfig/all**，找到网关，然后**arp –a**（显示和本机通信了的一些机器的IP和MAC）。PS：**arp –d**（清除缓存中的数据）。



展开第三行：

地址解析协议；硬件类型：以太网；协议类型：IP

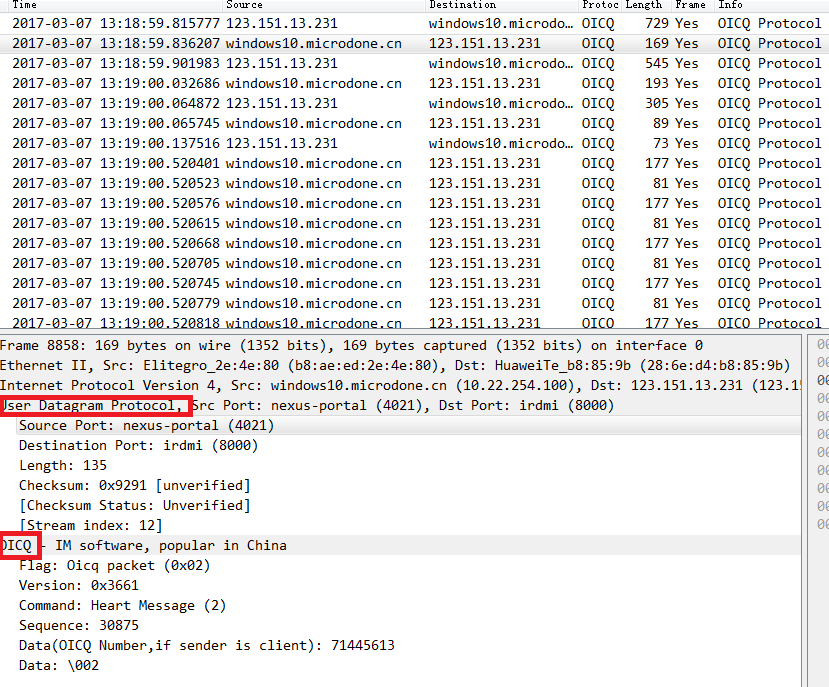
硬件大小：6；协议大小：4；发送方MAC地址

发送方IP地址；目的MAC地址；目的IP地址

思考题：

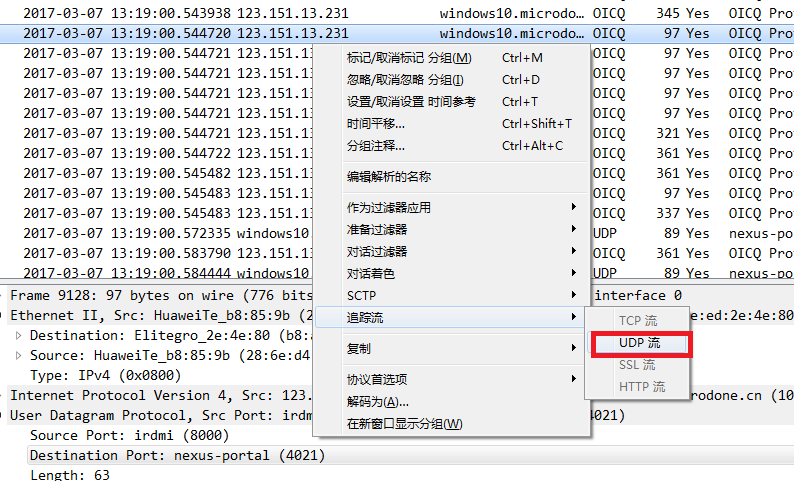
为什么课本将qq等软件是安全的，而支持pop3协议的foxmail是不安全性？

**提示：**



这些协议我们都已经非常熟悉了，而且wireshark把qq所用到的相关协议都称作oicp protocol。显示过滤器输入oicq。

可以看出目标端口是irdmi (8000),这在国内主要是QQ使用的端口号（irdmi表示为QQ聊天软件）；源端口是4021；数据包字节长度为135字节；检验和显示为验证禁用，不能验证。

查看传输数据 点鼠标右键，点“follow UDP stream（根据UDP码流）”可查看传输数据。



说明qq聊天内容是加密传送的，需要知道加密算法才能破解。

类似利用foxmail 去截获pop3协议，与oicq对比去分析。