实验四

Ethernet and ARP

**实验目的**

此次实验室中我们将探究以太网协议和ARP协议。

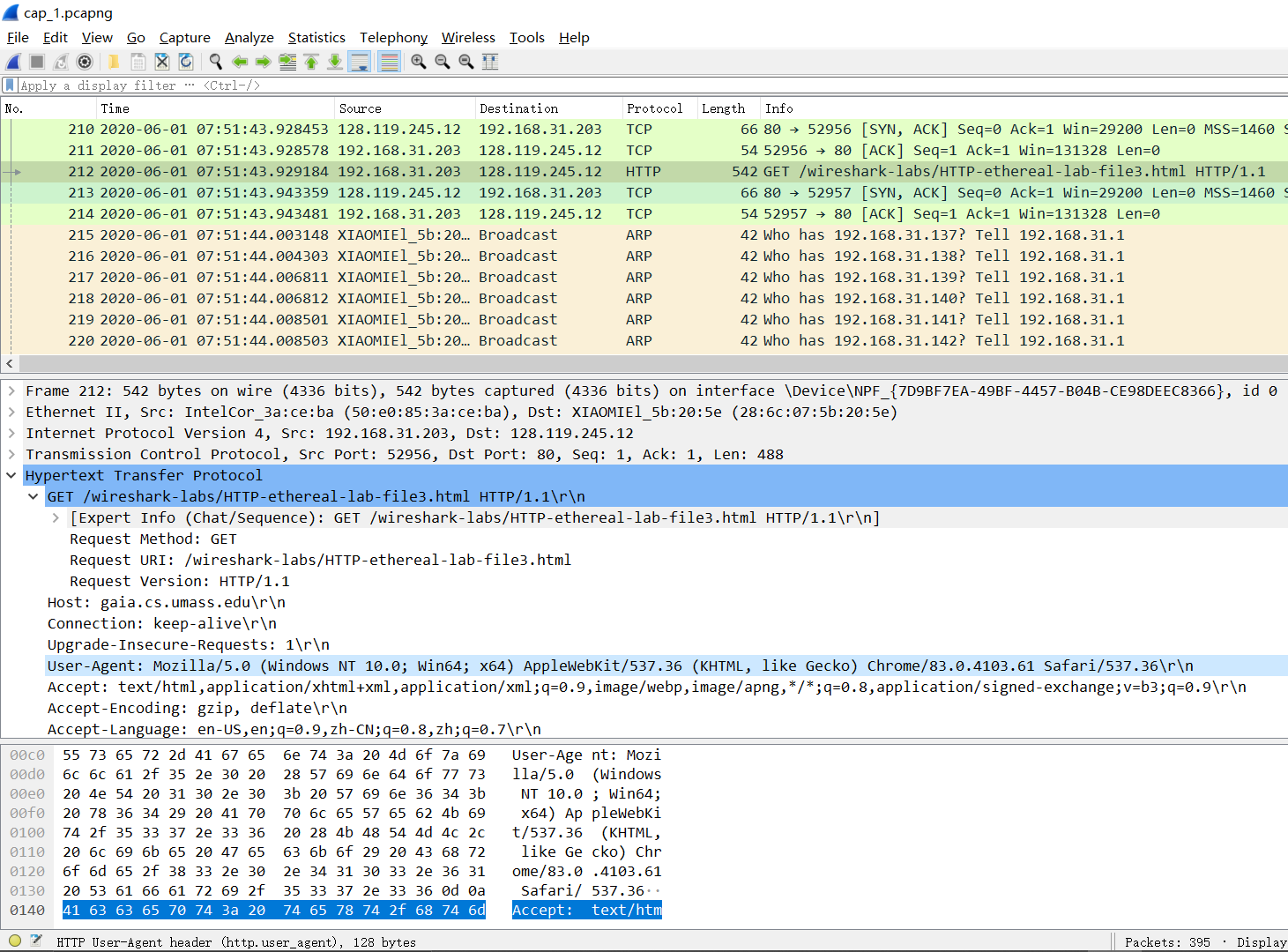
在开始实验之前，请先复习课本中的第6.4.1节(链路层寻址和ARP)和第6.4.2节(以太网)。 RFC 826(ftp://ftp.rfc-editor.org/in-notes/std/std37.txt)包含了ARP协议的详尽细节，IP设备使用ARP协议来确定已知以太网地址的远程接口的IP地址。

**实验步骤**

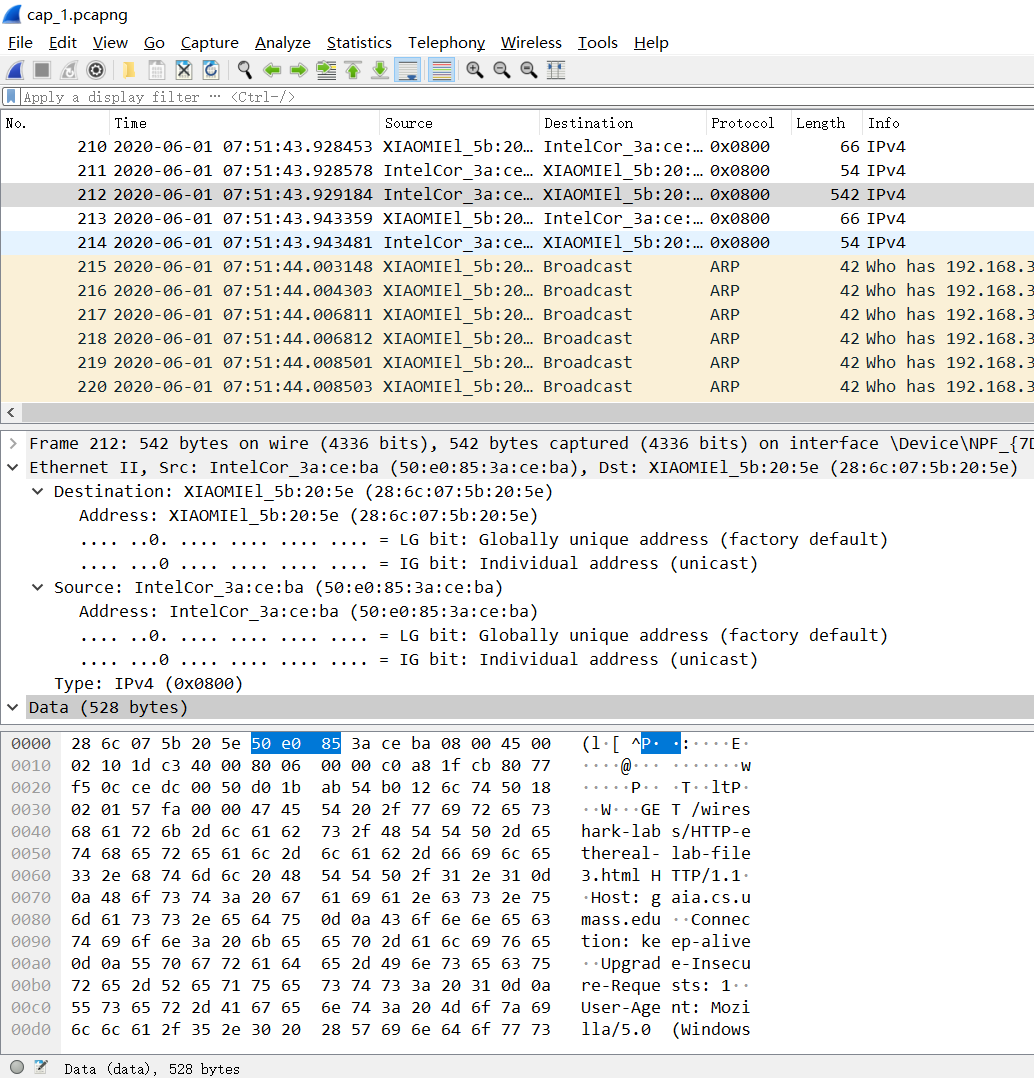
1. 捕获和分析以太网帧

让我们从捕捉一组以太网帧开始。

* 首先，请清空浏览器缓存。启动Wireshark分组嗅探器进行分组抓取
* 把URL输入到http://gaia.cs.umass.edu/wireshark-labs/HTTP-ethereal-lab-file3.html浏览器地址栏中，这将显示相当长的美国版权法案
* 停止Wireshark分组捕获。找到从你的计算机发送到gaia.cs.umass.edu的HTTP GET消息的分组编号，以及gaia.cs.umass.edu发送给你的计算机的HTTP response消息的开始处。此时应该观察到一个像下面这样的屏幕截图(下图截屏中的分组212包含HTTP GET消息）



* 由于这个实验是关于以太网和ARP的，我们对IP或更高层的协议不感兴趣。所以改变Wireshark的“listing of captured packets”窗口，使其只显示IP以下协议的信息。要让Wireshark执行此操作，请选择*Analyze->Enabled Protocols*。然后取消选中“IPv4”框并选择“确定”。现在应该会看到一个类似下图的Wireshark窗口：



为了回答后面的问题，需要查看分组的详细信息和分组内容窗口（Wireshark中的中间及下方的显示窗口）。

选择包含HTTP GET消息的以太网帧。（回想一下，HTTP GET消息是在TCP报文段中携带的，TCP报文段是在IP数据报中携带的，IP数据报是在以太网的帧之中携带的；如果感觉对此有点不清楚，请重新阅读教科书中的第1.5.2节）。展开“分组详细信息”窗口中的“Ethernet II”信息。请注意，以太网帧的内容（分组首部和负载）显示在“分组内容”窗口中。

根据包含HTTP GET消息的以太网帧的内容，回答以下问题。只要有可能，当你回答一个问题时，在实验报告中包含打印出来的用来回答问题的捕获跟踪到的分组。并注释打印出来的数据来解释你的答案。若要打印分组，请使用“文件”->“打印到文件”，选择“仅选定分组”，选择“分组摘要行”，然后选择回答问题所需的分组的最小详细信息量。

1. 你的计算机的48位以太网地址是什么？
2. 以太网帧的48位目标地址是什么？这是gaia.cs.umass.edu的以太网地址吗？（提示：答案是否定的）哪个设备拥有这个以太网地址？[注：这是一个很重要的问题，有些同学会弄错。请再阅读课本相关内容，确保理解这里的答案。]
3. 给出两字节帧类型字段的十六进制值。这对应于哪个上层协议？
4. 以太网帧显示GET的ASCII字符 "G" 到该帧的起始一共有多少字节？

接下来，根据包含HTTP响应消息的第一个字节的以太网帧的内容，回答以下问题。

1. 以太网源地址的值是多少？这是你的电脑的地址，还是gaia.cs.umass.edu的（提示：答案是否定的）？哪个设备的以太网地址是这个？
2. 以太网帧的目标地址是什么？这是你的电脑的以太网地址吗？
3. 给出双字节帧类型字段的十六进制值，其所对应的上层协议是什么？
4. 从以太网帧的一开始，到"OK"（即HTTP响应码）中的ASCII字符"O"共有多少字节？
5. 地址解析协议

在本节中，我们将观察实际运行中的ARP协议。强烈建议在此之前先复习阅读教材的第6.4.1节。

**ARP缓存**

回想一下，ARP协议通常在你的计算机上维护一个IP到以太网地址转换对的缓存。*arp*命令（无论MSDOS还是Linux/Unix中）用于查看和操作这个缓存的内容。由于*arp*命令和ARP协议具有相同的名称，因此很容易混淆它们。但请记住它们是不同的 —— arp命令用于查看和操作ARP缓存内容，而ARP协议定义发送和接收的消息格式和含义，并定义消息发送和接收时所采取的动作。

让我们来看看你计算机上的ARP缓存的内容：

* **MS-DOS系统：***arp*命令位于c:\ windows\system32中，因此在MS-DOS命令行中键入"arp -a"或c:\ windows\system32\arp -a"（不带引号）。
* **Linux/Unix/MacOS：***arp*命令的可执行文件可以在不同的地方执行。常见的位置是/sbin/arp（对于linux）和/usr/etc/arp（对于某些Unix变体）。

参数为-a的Windows *arp*命令将显示计算机上ARP缓存的内容。运行*arp -a*命令。

1. 记录下你的计算机ARP缓存的内容。每一列的值是什么含义？

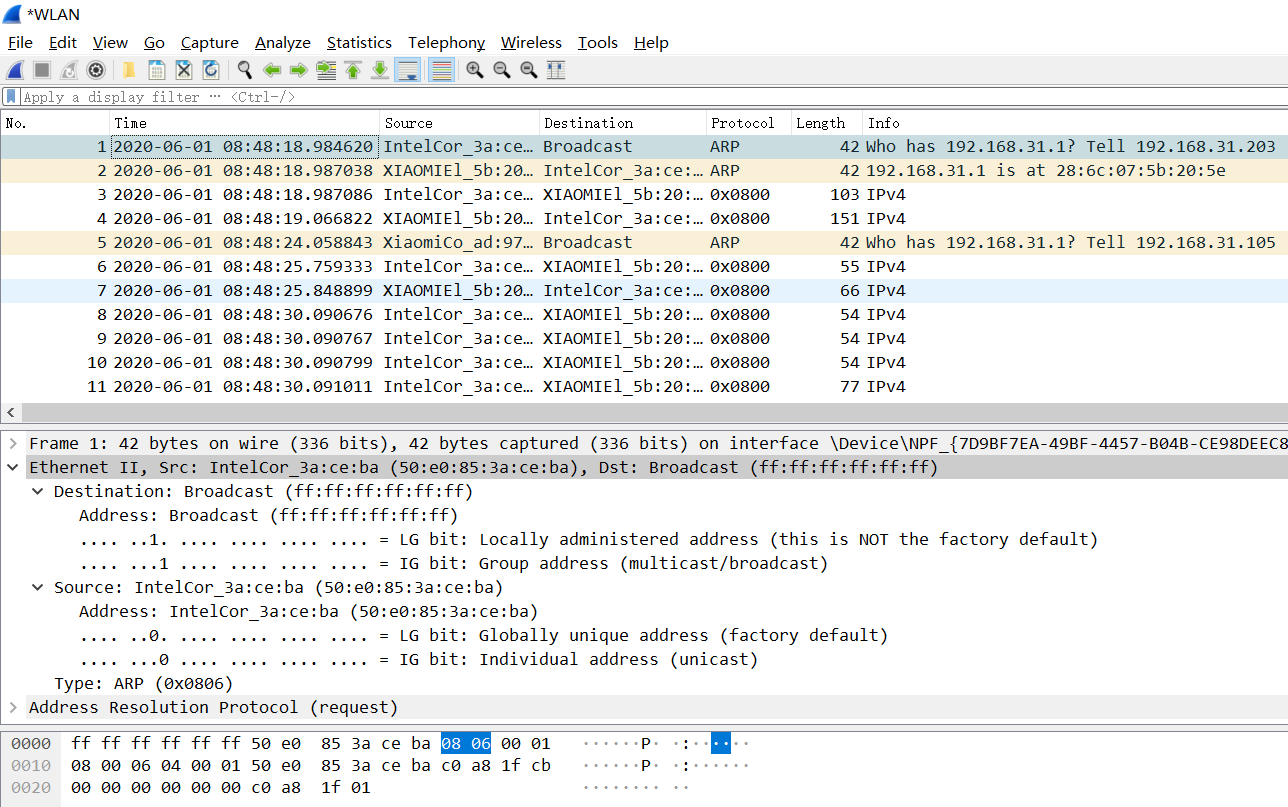
为了观察你的计算机发送和接收的ARP消息，我们需要清除ARP缓存，否则你的计算机可能会在其缓存中找到所需的IP以太网地址转换对而不需要发送ARP消息。

* **MS-DOS系统**：MS-DOS *arp –d \** 命令将清除你的ARP缓存。*–d*标志表示删除操作，\*是通配符，表示删除所有表项。
* **Linux/Unix/MacOS**：*arp –d \** 将清除你的ARP缓存。要运行此命令，你需要根权限。

**观察运行中的ARP**

执行以下操作：

* 如上所述，清除ARP缓存。
* 接下来，确保浏览器缓存为空。
* 启动Wireshark分组嗅探器
* 输入http://gaia.cs.umass.edu/wireshark-labs/http-wireshark-lab-file3.html这个URL到浏览器中，浏览器会再次显示相当长的美国版权法。
* 停止Wireshark分组捕获。同样，我们对IP或更高层的协议不感兴趣，因此更改Wireshark的“listing of captured packets”窗口，使其仅显示IP以下协议的信息。要让Wireshark执行此操作，请选择“*Analyze->Enabled Protocols*。然后取消选中“IPv4”框并选择“确定”。现在应该会看到一个类似下面的Wireshark窗口：



在上面的例子中，跟踪的前两个帧包含ARP消息（第五个消息也是）。

回答下列问题：

1. 包含ARP请求消息的以太网帧中源地址和目标地址的十六进制值是多少？
2. 给出两字节以太网帧类型字段的十六进制值。这对应于哪个上层协议？
3. 现在找到响应ARP请求而发送的ARP应答。
   1. 从以太网帧的最开始处到ARP *opcode*字段开始有多少字节?
   2. 在作出ARP响应的以太网帧的ARP有效负载中*opcode*字段的值是多少?
   3. 在ARP消息中，对先前的ARP请求的应答（具有该以太网地址的机器的IP地址，其对应的IP地址正在被查询）在哪里？

**额外加分：**

一个条目在被删除之前保留在你的ARP缓存中的默认时间是多少？你可以根据经验（通过监视缓存内容）或在操作系统文档中查找来确定这个值。指明你是如何/在哪里确定这个值的。

加1. *arp* 命令:

*arp -s InetAddr EtherAddr*

允许你手动向ARP缓存添加一个表项，该表项将IP地址*InetAddr*解析为物理地址*EtherAddr*。如果手动添加表项时，输入了正确的IP地址，但该远程接口的以太网地址不正确，会发生什么情况？

加2. 一个表项在被删除之前保留在你的ARP缓存中的默认时间是多少？你可以根据经验（通过监视缓存内容）或在操作系统文档中查找来确定这个值。在实验报告中指明你是如何/在哪里得到这个值的。