

**计算机网络**

**课程实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 实验名称 | 利用Wireshark进行协议分析 | | | | | |
| 姓名 | 许坤彬 | | 院系 | 计算学部 | | |
| 班级 | 2203103 | | 学号 | 2022114586 | | |
| 任课教师 | 聂兰顺 | | 指导教师 | 聂兰顺 | | |
| 实验地点 | G001 | | 实验时间 | 2024/10/24 | | |
| 实验课表现 | 出勤、表现得分(10) |  | 实验报告  得分(40) |  | 实验总分 |  |
| 操作结果得分(50) |  |
| 教师评语 | | | | | | |
|  | | | | | | |

****

|  |
| --- |
| 实验目的： |
| 熟悉并掌握 Wireshark 的基本操作，了解网络协议实体间进行交互以及报文交换的情况。 |
| 实验内容： |
| 1) 学习 Wireshark 的使用  2) 利用 Wireshark 分析 HTTP 协议  3) 利用 Wireshark 分析 TCP 协议  4) 利用 Wireshark 分析 IP 协议  5) 利用 Wireshark 分析 Ethernet 数据帧  6）选做内容：  a) 利用 Wireshark 分析 DNS 协议  b) 利用 Wireshark 分析 UDP 协议  c) 利用 Wireshark 分析 ARP 协议 |
| 实验过程： |
| 1. 学习使用Wireshark 2. 首先启动Wireshark软件，选择捕获菜单下的选项。      1. 在这里选择无线网卡WLAN，随后点击开始。 2. 打开浏览器，输入一个网页。在这里我们选择<http://www.arnc2024.cn/hello.html>，这个网页是我个人使用nginx部署在阿里云服务器上的一个用于实验测试的HTML。在浏览器得到全部网页内容之后，停止捕获，在内容过滤器中输入HTTP，可以过滤得到HTTP协议相关的捕获信息。     点击第一条信息，可以在下方依次看到以太网帧、 IP 数据报、 TCP 报文段、以及 HTTP 报文首部信息。   1. HTTP分析 2. HTTP GET/response交互   按照上面的方法，现在wireshark中打开捕获信息，随后在浏览器中打开<http://hitgs.hit.ediu.cn/news>，出现内容后停止信息捕获，并通过过滤器筛选HTTP相关信息。通过查看HTTP报文和IP报文中的相关内容可以得到指导书中问题的答案。    2． HTTP 条件 GET/response交互  首先清空浏览器的缓存，随后输入URL：<http://hitgs.hit.edu.cn/news>，访问一次，在得到服务器返回的内容后，点击浏览器的刷新按钮，再次进行访问，使用wireshark进行分组捕获。随后停止wireshark的分组俘获，在过滤筛选处输入http，在列表中寻找对应的HTTP报文。  三、TCP分析  首先访问<http://gaia.cs.umass.edu/Wireshark-labs/alice.txt>网页，得到文本，并且右键将文本保存到电脑上    随后打开https://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html，先输入文件路径，随后打开Wireshark开始分组俘获，再点击上传，上传完毕，贺词信息出现在浏览器中之后，再停止俘获。  四、IP分析  1. 通过执行traceroute捕获数据包  利用traceroute执行捕获数据包，利用traceroute发送具有不同大小的数据包给目的主机，等待接受ICMP的警告信息。随后使用Windows自带的tracet命令不允许用户更改tracet命令发送的ICMP echo请求信息的大小。在这里为了方便使用我们用pingplootter来完成实验。首先下载PingPllotter，随后用Wireshark开始数据包的捕获。之后启动pingplotter并在Address to trace window中输入目的域，在“# of times to Trace”域中输入“3”，这样就不过采集过多的数据。Edit->Options->Packet，将 Packet Size(in bytes,default=56)域设为 56，这样将发送一系列大小为 56 字节的包。 然后按下“Trace”按钮。  随后开始实验，首先在Edit->Options->Packet，然后将 Packet Size(in bytes,default=56)域改为 2000，这样将发送一系列大小为 2000 字节的包。然后按下“Resume”按钮。最后，将 Packet Size(in bytes,default=56)域改为 3500，发送一系列大小为 3500 字节的包。然后按下“Resume”按钮。之后停止 Wireshark 的分组捕获。  五、抓取ARP数据包  （1）利用MS-DOS命令：arp或c:\windows\system32\arp查看主机上ARP缓存的内容。  （2）在命令行模式下输入：ping 192.168.1.82（或其他IP地址）  （3）启动Wireshark，开始分组俘获。  六、抓取UDP数据包  首先启动Wireshark，并且开始捕获分组，随后发送QQ消息给好友，之后停止wireshark的分组捕获，最后在显示筛选规则中输入udp查找对应信息。  七、利用wireshark进行DNS协议分析  首先打开浏览器输入[www.baidu.com](http://www.baidu.com)，随后打开wireshark启动抓包，回到浏览器回车检索网页，待返回网页内容后回到wireshark停止抓包。 |
| 实验结果： |
| 一、wireshark使用  二、HTTP分析  1. HTTP GET/response交互  （1）浏览器运行的是HTTP 1.1    （2）浏览器向服务器指出能够接受语言版本为zh\_CN，即简体中文。    （3）我的计算机IP地址为172.20.233.126，服务器的IP地址为219.217.2226.25    （4）服务器为我返回的状态代码为200    2. HTTP 条件 GET/response交互  在这一部分中，我使用网站<http://hitgs.hit.edu.cn/news>的时候，无论如何发送、重复几次都无法发送出带有If-modified-since的请求头，并且无法接收到304，我猜测这个网站本身就没有提供last-modified字段，所以无法完成这个实验的验证。因此，在这里我们将使用的网站切换为<http://www.arnc2024.cn/hello.html>，这个网站是我自己使用nginx部署到阿里云服务器上的，会在返回的HTTP报文中增加last-modified字段。能够完成本实验的验证。   1. 第一次发送给服务器的请求中没有IF-MODIFIED-SINCE 2. 第一次服务器响应的报文返回了文件内容，因为返回的状态码为200     （3）较晚的HTTP GET请求中包含了IF-MODIFIED-SINCE字段，字段后面跟着的内容是本地缓存中所存储的上次HTTP报文返回的Last-Modified字段  （4）较晚的HTTP GET请求的响应HTTP状态编码为304，没有返回文件内容。     1. TCP分析   1. 上传文件    2. 浏览追踪信息  （1）向gaia.cs.umass.edu服务器发送文件的客户端主机的IP地址是172.20.233.126，端口号为61184。  （2） gaia.cs.umass.edu服务器的IP地址是183.240.139.243，用来发送和接受TCP报文的端口号是20058。    3. TCP基础  （1）SYN报文段的序号是403855647，在该报文段中使用flag的低两位来标注SYN报文段（第二位若为1则为SYN报文段）。      （2）SYNACK的报文段序号是2710228023，通过使用Syn和Ack两位来用来标注SYN ACK。      （3）tcp三次握手的过程    （4）包含了HTTP POST命令的TCP报文段序号是    （5）第六个报文段的序号是    它的发送时间是  （6）前6个TCP报文段的长度    （7）window最小为65536，在限制发送端的传输之后，window的大小会越来越大，直到最后恢复到241664。限制传输之后，不会再出现接收端缓存不够用的情况。  （8）跟踪文件是有重传的，在检测的分组信息中出现了像下面的的信息。    （9）传输的总长度为:  总长度为：7124 bytes  起止时间为1.923552，3.910840  则传输速度为7124/（3.910840-1.923552）= 3597.9798 Bps  四、IP协议分析  1. 查看第一个主机包的IP协议情况  （1）我主机的IP地址是172.20.207.157    （2）上层协议字段的值是1，表示是ICMP协议    （3）IP头一共是20个字节，静载是36个字节，头部可以从捕获的信息中直接读出，随后再读出总长，之后使用总长-头部得到静载    （4）该IP数据包并没有分片，从标志位就可以看出。    2. 单击source列按钮，对捕获的数据包按照源地址进行排序。    （1）ID、TTL、首部校验和还有数据域会经常改变，剩下的数据都会经常保持不变；  （2）变量改变：  由于标识ID代表着每个不同的数据包，因此必须互不相同；  由于ICMP的ping探测导致了TTL在不断增大；  由于ID和TTL一直改变，因此首部校验和也会改变；  由于数据域中含有ICMP报文，ICMP报文首部又一直改变因此数据域也在一直改变。  （3）identification字段值为0，TTL为254，这些值中，TTL保持不变，而ID改变。TTL再第一个路由器中设定为RFC所指定的值，所以是一样的，而ID字段对于每一个IP字段来说却是唯一的，所以会改变。    3. 观察包大小为2000字节的情况：  （1）被分为了不止1个IP数据报，数据报中标志位的MF=1，Fragment offset不为0可以看出      （2）第一个数据分片的MF=1，offset为0说明是首片而且不是最后一篇，由第二个数据分片的MF=0，offset=1480可以得到不是首片而且是最后一篇。  4. 查看包大小为3500字节的情况  （1）数据报被分为了三片        （2）主要是MF和offset的组合情况发生了变化，前两个MF为1，第三个为0；offset依次为0，1480，2960。  五、ARP协议分析  （1）使用arp在命令行中查看缓存的内容    第一列为IP地址，第二列为MAC地址（即物理地址），第三列为其类型（分为动态和静态）。在这里为了进一步实验，我们接下来ping一个ip地址前24位为172.20.207.X的一个地址，从而观察arp协议的广播查询和响应。    （2）ARP数据包的格式由９部分组成，一共28个字节，分别为：硬件类型2个字节；协议类型2个字节；硬件地址长度1个字节；协议地址长度1个字节；OP 2个字节；源MAC地址6个字节；源IP地址4个字节；目的MAC地址6个字节；目的IP地址4个字节。    （3）通过opcode字段可以进行判断，当opcode字段为1说明是请求包，当opcode字段为2说明是应答包。      （4）这是因为ARP在查询的时候不知道IP对应的MAC地址，所以需要先广播进行查询。而在响应之后，ARP通过帧知道了对方的源MAC地址，所以只需要直接明确的发送即可。  六、抓取UDP包  （1）消息是基于udp的  （2）我的主机ip地址是172.20.207.157，对方的IP地址是119.157.166.89    （3）发送QQ消息的端口号是50578，QQ服务器的端口号是8000  （4）数据报的格式是    包含源端口号2个字节，目的端口号2个字节，UDP段长度2个字节，校验和2个字节。  （5）服务器返回ICQ用于确认服务器是否收到消息。这是由于UDP提供的是不可靠的无连接的传输服务，客户端并不知道服务器是否已收到信息，因此需要一个ICQ报文表示收到信息；而这个过程就可看出UDP是无连接的，并未采用TCP的三次握手过程。  七、利用wireshark进行DNS协议分析 |
| 问题讨论： |
| 在实验过程中，在ARP协议的抓包过程中，卡住了一下。后上互联网检索资料，需要ping一个本网段的IP。之后可以观察到对应的数据包，具体的可以参考博客：https://blog.csdn.net/wangyuxiang946/article/details/124504072 |
| 心得体会： |
| 对计算机网络的层次结构模型有了更加深刻的认识，对应用层、传输层、网络层、数据链路层的一些协议有了更多认识，对协议的报文结构有了更多了解。 |