

**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 | 刘天瑞 | | 院系 | 未来技术学院 | | |
| 班级 | 20W0362 | | 学号 | 7203610121 | | |
| 任课教师 | 刘亚维 | | 指导教师 | 刘亚维 | | |
| 实验地点 | G001 | | 实验时间 | 2022.10.26 | | |
| 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： |
| （注：实验报告模板中的各项内容仅供参考，可依照实际实验情况进行修改。）  本次实验的主要目的。  答：  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： |
| 概述本次实验的主要内容，包含的实验项等。  答：  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；  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： |
| 以文字描述、实验结果截图等形式阐述实验过程，必要时可附相应的代码截图或以附件形式提交。  **主要是了解实验相关基础知识**  **(i)分组嗅探器的基本概念**  观察在正在运行协议实体间交换报文的基本工具被称为分组嗅探器（packet sniffer）。顾名思义，一个分组嗅探器俘获（嗅探）计算机发送和接收的报文。一般情况下，分组嗅探器将存储和显示出被俘获报文的各协议头部字段的内容。下图为一个分组嗅探器的结构。    图1 分组嗅探器抽象化结构说明  如上图所示，右边是计算机上正常运行的协议（在这里Internet协议）和应用程序（如：web浏览器和ftp客户端）。分组嗅探器（虚线框中的部分）是附加计算机普通软件上的，主要有两部分组成。分组俘获库（packetcapture library）接收计算机发送和接收的每一个链路层帧的拷贝。高层协议（如：HTTP、FTP、TCP、UDP、DNS、IP等）交换的报文都被封装在链路层帧中，并沿着物理媒体（如以太网的电缆）传输。并且图1假设所使用的物理媒体是以太网，上层协议的报文最终封装在以太网帧中。  分组嗅探器的第二个组成部分是分析器。分析器用来显示协议报文所有字段的内容。为此，分析器必须能够理解协议所交换的所有报文的结构。例如：我们要显示图中HTTP协议所交换的报文的各个字段。分组分析器理解以太网帧格式，能够识别包含在帧中的IP数据报。分组分析器也要理解IP数据报的格式，并能从IP数据报中提取出TCP报文段。然后，它需要理解TCP报文段，并能够从中提取出HTTP消息。最后，它需要理解HTTP消息。  **(ii)Wireshark**  基本介绍：Wireshark是一种可以运行在Windows, UNIX以及Linux等操作系统上的分组分析器。Wireshark（前称 Ethereal）是一个网络封包分析软件。网络封包分析软件的功能是撷取网络封包，并尽可能显示出最为详细的网络封包资料。Wireshark使用WinPCAP作为接口，直接与网卡进行数据报文交换。  运行Wireshark程序，再打开浏览器，选择捕获网络接口WLAN，浏览器输入网址，Wireshark抓包并筛选HTTP协议报文时，其显示的用户界面如下图所示：    图2 用户可视界面  详细用户界面：命令菜单、俘获分组列表、分组头部明细、分组内容窗口、筛选俘获分组等信息与实验指导书中给出的基本一致，不再展示。 |
| 实验结果： |
| 采用演示截图、文字说明等方式，给出本次实验的实验结果。   1. **HTTP分析 - HTTP GET/response交互**   由于实验指导书上的示例网站 <http://hitgs.hit.edu.cn/news> 无法打开，因此，在本报告中以 <http://jwts.hit.edu.cn/> (哈尔滨工业大学本科教学管理服务系统)代替。  按照实验指导书上的流程进行如下操作：  (1)启动 Web browser，然后启动 Wireshark 分组嗅探器。在窗口的显示过滤说明处输入“http”，分组列表子窗口中将只显示所俘获到的HTTP报文；  (2)开始 Wireshark 分组俘获；  (3)在打开的Web browser窗口中输入以下地址: <http://jwts.hit.edu.cn/> ；  (4)停止分组俘获。  其结果如下图所示：    图3 HTTP GET  回答以下问题：  1.你的浏览器运行的是HTTP1.0，还是HTTP1.1？你所访问的服务器所运行HTTP协议的版本号是多少？  答：都是HTTP 1.1。  2.你的浏览器向服务器指出它能接收何种语言版本的对象？  答：zh-CN, zh （即中文）。  3.你的计算机的IP地址是多少？服务器 <http://jwts.hit.edu.cn/> 的IP地址是多少？  答：本机IP：172.20.228.212；服务器IP：10.160.2.157。  4.从服务器向你的浏览器返回的状态代码是多少？  答：200 OK。     1. **HTTP分析 - HTTP条件GET/response交互**   按照实验指导书上流程进行操作：  (1)启动浏览器，清空浏览器的缓存（在浏览器中，选择“工具”菜单中的“Internet选项”命令，在出现的对话框中，选择“删除文件”）；  (2)启动Wireshark分组俘获器。开始Wireshark分组俘获；  (3)在浏览器的地址栏中输入以下URL: <http://jwts.hit.edu.cn/> ,在你的浏览器中重新输入相同的URL或单击浏览器中的“刷新”按钮；  (4)停止Wireshark分组俘获，在显示过滤筛选说明处输入“http”,分组列表子窗口中将只显示所俘获到的HTTP报文。  结果如下图所示：    图4 HTTP条件GET    回答以下问题：  1.分析你的浏览器向服务器发出的第一个“HTTP GET”请求的内容，在该请求报文中，是否有一行是：IF-MODIFIED-SINCE？  答：没有。  2.分析服务器响应报文的内容，服务器是否明确返回了文件的内容？如何获知？  答：服务器明确返回了文件内容，若返回的状态码是200，代表明确返回了文件；若返回状态码为404，则不返回文件。  3.分析你的浏览器向服务器发出的较晚的“HTTP GET”请求，在该请求报文中是否有一行是：IF-MODIFIED-SINCE？如果有，在该首部行后面跟着的信息是什么？  答：有。在该首部行后面跟着的信息是Sun, 01 Mar 2020 07:20:02 GMT\r\n，代表着缓存最后更新的时间。  4.服务器对较晚的“HTTP GET”请求的响应中的HTTP状态代码是多少？服务器是否明确返回了文件的内容？请解释。  答：服务器对较晚的“HTTP GET”请求的响应中的HTTP状态代码是304 Not Modified。服务器不会明确返回文件内容，因为服务器判断的结果为Not Modified，在此情况下，客户端可以使用本地仍为最新版本的缓存文件。     1. **TCP分析**   按照实验指导书上流程进行：  **（1）俘获大量的由本地主机到远程服务器的 TCP 分组**  (i)启动浏览器，打开 <http://gaia.cs.umass.edu/wireshark-labs/alice.txt> 网页，得到ALICE'S ADVENTURES IN WONDERLAND文本，将该文件保存到你的主机上；  (ii)打开<http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html> ，在Browse按钮旁的文本框中输入保存在你的主机上的文件ALICE'S ADVENTURES INWONDERLAND的全名（含路径），此时不要按“Upload alice.txt file”按钮；  (iii)启动Wireshark，开始分组俘获；  (iv)在浏览器中，单击“Upload alice.txt file”按钮，将文件上传到gaia.cs.umass.edu服务器，一旦文件上传完毕，一个简短的贺词信息将显示在你的浏览器窗口中；  (v)停止俘获。  结果如下图所示：    图5 TCP分析  **（2）浏览追踪信息**  在显示筛选规则中输入“tcp”,可以看到在本地主机和服务器之间传输的一系列tcp和http报文，你应该能看到包含 SYN 报文的三次握手。也可以看到有主机向服务器发送的一个HTTP POST报文和一系列的“http continuation”报文。  根据操作思考以下问题：  1.向gaia.cs.umass.edu服务器传送文件的客户端主机的IP地址和TCP端口号是多少？  答：客户端主机的IP地址：172.20.228.212，TCP端口号：60709。  2.Gaia.cs.umass.edu服务器的IP地址是多少？对这一连接，它用来发送和接收TCP报文的端口号是多少？  答：服务器的IP地址：110.249.194.67，用来发送和接收 TCP 报文的端口号：80。    **（3）TCP基础**  根据操作思考以下问题：  1.客户服务器之间用于初始化TCP连接的TCP SYN报文段的序号（sequence number）是多少？在该报文段中，是用什么来标示该报文段是SYN报文段的？  答：如图所示：    图6 SYN报文段标示  初始化TCP连接的TCP SYN报文段的序号是0。在该报文段中，通过设置Flags中的SYN位为1，来表示该报文段是SYN报文段。  2.服务器向客户端发送的SYNACK报文段序号是多少？该报文段中，Acknowledgement字段的值是多少？Gaia.cs.umass.edu服务器是如何决定此值的？在该报文段中，是用什么来标示该报文段是SYN ACK报文段的？  答：如图所示：    图7 SYN ACK报文段标示  SYN ACK报文段序号是0；  Acknowledgement字段的值是1；  Gaia.cs.umass.edu服务器根据上一次客户端发给服务器的seq+1得到该字段；  在该报文段中，通过Flags位中SYN与ACK均为1来标示该报文段是SYN ACK。  3.你能从捕获的数据包中分析出tcp三次握手过程吗？  答：如下图所示：    图8 RST上下段握手过程  客户端先向服务器发送一个seq = 0的建立连接请求，然后服务器向客户端返回 seq = 0, ack = 1的响应。  4.包含HTTP POST命令的TCP报文段的序号是多少？  答：152588，如下图所示：    图9 HTTP报文段中的HTTP POST命令的序号    5.如果将包含HTTP POST命令的TCP报文段看作是TCP连接上的第一个报文段，那么该TCP连接上的第六个报文段的序号是多少？是何时发送的？该报文段所对应的ACK是何时接收的？  答：第六个报文段为234，在HTTP POST发送之前，TCP连接建立之后发送。ACK即为服务器返回的第六个 ACK。  6.前六个TCP报文段的长度各是多少？  答：如图：    图10 前六个TCP报文段长度  长度分别为747,1460,1460,1460,1460,1460。  7.在整个跟踪过程中，接收端公示的最小的可用缓存空间是多少？限制发送端的传输以后，接收端的缓存是否仍然不够用？  答：接收端公示的最小的可用缓存空间是142，且窗口大小整体递增，并未出现不够用的情况。    图11 接收端公示的最小的可用缓存空间    8.在跟踪文件中是否有重传的报文段？进行判断的依据是什么？  答：没有出现重传，因为客户端发送的报文序列号没有出现重复的情况。  9.TCP连接的throughput (bytes transferred per unit time)是多少？请写出你的计算过程。  答： 发送数据总的长度为152741B + 108 x 54B = 158573，发送时间间隔约为1.72s  因此吞吐量为158573B / 1.72s = 92193.6 Bps。  **4．IP分析**  按照实验指导书上流程进行：  **(i)通过执行traceroute执行捕获数据包**  1.启动Wireshark并开始数据包捕获；  2.启动pingplotter并“Address to Trace Window”域中输入目的地址。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；  3.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；  4.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；  5.停止Wireshark 的分组捕获。    图12 pingplotter pro运行界面    图13 Wireshark执行traceroute捕获数据包运行界面  **(ii)对捕获的数据包进行分析**  在你的捕获窗口中，应该能看到由你的主机发出的一系列ICMPEcho Request包和中间路由器返回的一系列ICMP TTL-exceeded消息。选择第一个你的主机发出的ICMP Echo Request消息，在packet details窗口展开数据包的Internet Protocol部分。  思考下列问题：  1.你主机的IP地址是什么？  答：172.20.228.212。  2.在IP数据包头中，上层协议（upper layer）字段的值是什么？  答：01，如下图所示：    图14 IP数据包头中的上层协议字段    3.IP头有多少字节？该IP数据包的净载为多少字节？并解释你是怎样确定该IP数据包的净载大小的？  答：如下图所示，IP头有20字节，数据报净载Total Length-Header Length=56B-20B = 36B。    图15 IP头字节与数据包计算    4.该IP数据包分片了吗？解释你是如何确定该IP数据包是否进行了分片？  答：有分片，因为有分片的偏移量，如下图所示：    图16 分片的偏移量    单击Source列按钮，这样将对捕获的数据包按源IP地址排序。选择第一个你的主机发出的ICMP Echo Request消息，在packet details窗口展开数据包的Internet Protocol部分。在“listing of captured packets”窗口，你会看到许多后续的ICMP消息 （或许还有你主机上运行的其他协议的数据包）。  思考下列问题：  1.你主机发出的一系列ICMP消息中IP数据报中哪些字段总是发生改变？  答：Time to Live、Identification、Header Checksum。  2.哪些字段必须保持常量？哪些字段必须改变？为什么？  答：Identification ⽤于区分不同的数据包，必须改变；Time to Live用于区分经过几个路由器，必须改变；Header Checksum 由前面的部分计算而得，因此也必须改变。除此之外，其他字段保持常量。  3.描述你看到的IP数据包Identification字段值的形式。  答：16位，且加一递增。    找到由最近路由器（第一跳）返回给你主机的ICMP Time-to-live exceeded消息。  思考下列问题：  1.Identification字段和TTL字段的值是什么？  答：如下图所示：    图17 ICMP消息详情  Identification：17839  TTL：252  2.最近的路由器（第一跳）返回给你主机的ICMP Time-to-live exceeded消息中这些值是否保持不变？为什么？  答：不变，对于Identification标识来说，相同的标识是为了分段后组装成同一段，不代表序号；因为是第一跳路由器返回的数据报，所以TTL也不变。  单击Time列按钮，这样将对捕获的数据包按时间排序。找到在将包大小改为2000字节后你的主机发送的第一个ICMP Echo Request消息。  思考下列问题：  1.该消息是否被分解成不止一个IP数据报？  答：如下所示：    可以看出被分成了两个数据报。  2.观察第一个IP分片，IP头部的哪些信息表明数据包被进行了分片？IP头部的哪些信息表明数据包是第一个而不是最后一个分片？该分片的长度是多少？  答：如下图所示：    图18 IP分片头部  Flags标识位中，More fragments位被置为1，表示其被分片，但不是最后一个分片。    **(iii)找到在将包大小改为3500字节后你的主机发送的第一个ICMP Echo Request消息**  思考下列问题：  1.原始数据包被分成了多少片？  答：3片，如下所示：    2.这些分片中IP数据报头部哪些字段发生了变化？  答：前两片的More fragments位均为1，而最后一片为0；另外，第二片的分片的offset为1480，最后一片为2960。  **5. 抓取ARP数据包**  按照实验指导书上流程进行：  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。  思考下面问题：  1.利用MS-DOS命令：arp或c:\windows\system32\arp查看主机上ARP缓存的 内容。说明 ARP缓存中每一列的含义是什么?  答：如下图所示：    图19 在终端输入命令查看主机ARP上缓存内容  ARP缓存中的每一列表示IP地址所对应的物理地址和类型（动态配置或静态配置）。  清除主机上ARP缓存的内容,抓取ping命令时的数据包。  分析数据包,回答下面的问题：  1.ARP数据包的格式是怎样的？由几部分构成，各个部分所占的字节数是多少？  答：数据包格式如图所示：    图20 ARP数据包格式  其由9部分构成，分别是：硬件类型（2字节），协议类型（2字节），硬件地址长度（1字节），协议地址长度（1字节），OP（2字节)，发送端MAC地址（6字节），发送端IP地址（4字节），目的MAC地址（6字节）以及目的IP地址（4字节）。  2.如何判断一个ARP数据是请求包还是应答包？  答：通过OP字段查看。OP为0x0001时表明该ARP数据为请求包， OP为0x0002时表明该ARP数据是应答包。  3.为什么ARP查询要在广播帧中传送，而ARP响应要在一个有着明确目的局域网地址的帧中传送？  答：在进行ARP查询时，发送主机并不知道目的IP对应的MAC地址，所以需要进行广播查询。而ARP响应报文明确知道查询主机的MAC地址，且局域网中的其他主机不需要此次查询的结果，因此ARP响应要在一个有着明确目的局域网地址的帧中传送。    **6. 抓取UDP数据包**  按照实验指导书上流程进行：  1.启动Wireshark，开始分组捕获；  2.发送QQ消息给你的好友；  3.停止Wireshark组捕获；  4.在显示筛选规则中输入“udp”并展开数据包的细节。    图21 Wireshark抓取UDP数据包运行界面  分析QQ通讯中捕获到的UDP数据包。根据操作思考以下问题：  1.消息是基于UDP的还是TCP的？  答：如上图所示，是基于UDP的。  2.你的主机ip地址是什么？目的主机ip地址是什么？  答：主机IP：172.20.228.212，目的主机IP：120.232.131.250。  3.你的主机发送QQ消息的端口号和QQ服务器的端口号分别是多少？  答：主机发送QQ消息的端口号：37051，QQ服务器的端口号：18400。  4.数据报的格式是什么样的？都包含哪些字段，分别占多少字节？  答：UDP数据报格式如下图所示：    图22 UDP数据包格式  UDP数据报由5部分构成，分别是源端口号（4字节），目的端口号（4字节），长度（4字节），校验和（4字节）和其上附加的应用层数据。  5.为什么你发送一个ICQ数据包后，服务器又返回给你的主机一个ICQ数据包？这UDP的不可靠数据传输有什么联系？对比前面的TCP协议分析，你能看出UDP是无连接的吗？  答：服务器返回一个ICQ数据包，是因为服务器需要将接收到的结果返回给发送的客户端。  6.这和UDP的不可靠数据传输的联系是：  答：在UDP不可靠数据传输的机制下，服务器只提供一次返回的ACK，无法保证数据一定送达。  7.能否看出UDP无连接：  答：可以看出。因为UDP数据包没有序列号，因此不能像TCP协议那样先进行握手再进行数据发送。     1. **利用WireShark进行DNS协议分析**   按照实验指导书上流程进行：   1. 打开浏览器键入：www.baidu.com； 2. 打开Wireshark,启动抓包； 3. 在控制台回车执行完毕后停止抓包。   结果如下图所示：    图23 Wireshark进行DNS协议分析运行界面 |
| 问题讨论： |
| 对实验过程中的思考问题进行讨论或回答。  答：  实验过程中遇到了关于ICMP协议的相关内容，在此对ICMP协议进行一定的展示。ICMP（Internet Control Message Protocol）Internet控制报文协议。它是TCP/IP协议簇的一个子协议，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。  事实上ICMP是IP的一个组成部分，与 IP 协议、ARP 协议、RARP 协议及 IGMP 协议共同构成 TCP/IP 模型中的网络层。 |
| 心得体会： |
| 结合实验过程和结果给出实验的体会和收获。  答：   1. 对于计算机网络模型有了更加深入的认识：尤其是对应用层、传输层、网络层以及数据链路层了解更深入了，通过对以上协议报文的抓包分析，对于其结构与工作原理也更加熟悉； 2. 具体来说，通过Wireshark分析各种网络协议的执行，可以加深对于各种协议交互过程的理解。并且对于协议中各个字段的作用有了更深的了解。熟悉了Wireshark工具的使用，能够熟练地进行数据包的抓取与分析。而且我认为Wireshark是个十分强大的工具，不仅仅在本门实验中会利用到他的一些知识，在以后的学习工作生活中还有很大的作用； 3. 对计算机网络体系结构中不同的协议有了更清楚的认识，并且对数据包在网络传输中过程的认识更加深刻。 |