4.2 使用wireshark抓包及分析网络

wireshark是一款非常优秀的网络抓包工具，可在多种平台上运行。下面介绍wireshark相关的知识和抓包的操作步骤。

4.2.1 认识wireshark

wireshark是一款可运行在UNIX和Windows系统中的开源网络协议分析器。它可以实时检测网络通信数据，也可以检测其抓取的网络通信数据快照文件。可以通过图形界面浏览这些数据，也可以查看网络通信数据包中每一层的详细内容。wireshark拥有许多强大的特性，包括强显示过滤器语言和查看TCP会话重构流的能力，支持上百种协议和媒体类型。

wireshark不会入侵侦测系统，对于网络上的异常流量行为，wireshark也不会产生警示或是任何提示。仔细分析wireshark截取的数据包能够帮助使用者对于网络行为有更清楚的了解。wireshark不会对数据包产生内容的修改，也不会发送出数据包到网络上。

wireshark的应用非常广泛，网络管理员使用wireshark来检测网络问题，网络安全工程师使用wireshark来检查资讯安全的相关问题，开发者使用wireshark来为新的通信协议排除错误，普通使用者使用wireshark来学习网络协议的相关知识。当然，有的人也会“居心叵测”地用它来寻找一些敏感信息。

4.2.2 使用wireshark抓包

wireshark的工作流程包括：选择捕获接口、使用捕获过滤器、使用显示过滤器、使用着色规则、构建图标以及重组数据。在Kali中自带了wireshark。

1. 启动侦听

首先介绍软件的启动和侦听的启动操作。

Step 01 在“嗅探/欺骗”组中找到并选择wireshark选项，如图4-5所示。

Step 02 在列表中显示了系统中的所有网络接口，这里双击eth0接口选项，如图4-6所示。

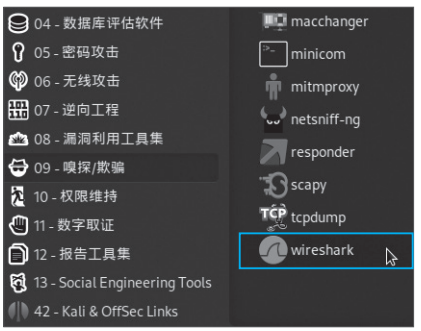

图 4-5

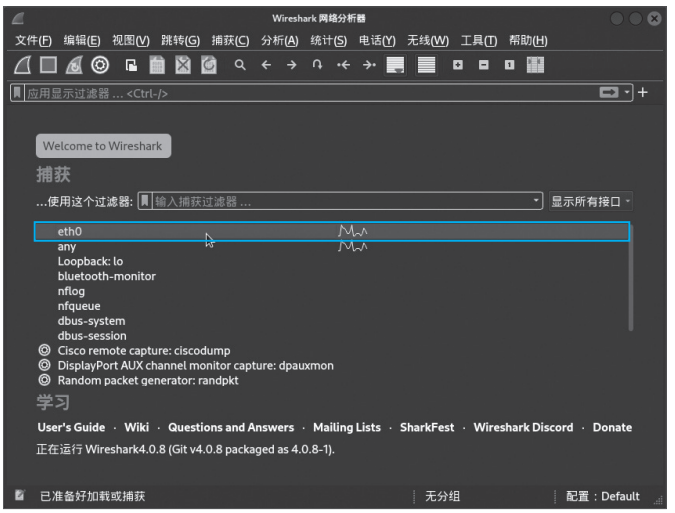

图 4-6

Step 03 接下来wireshark会自动进行数据包的抓取，抓包信息会不断滚动，等待一段时间后，单击“停止捕获分组”按钮，如图4-7所示。可以显示包的编号、时间、源地址、目标地址、协议、包长度和信息。

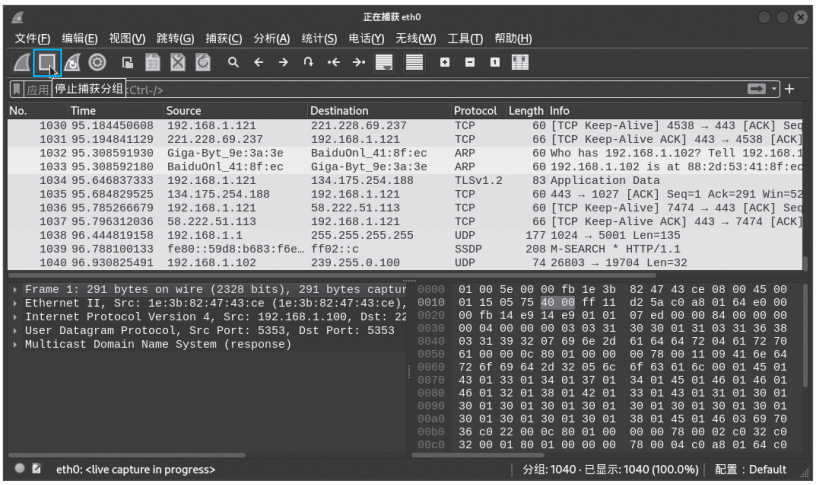

图 4-7

2. 数据面板

停止捕获后，就可以选择数据包进行分析了。wireshark可以分成3个面板部分：正上方是数据包列表，左下方是数据包的详细信息，右下方是数据包的原始信息。这3个面板相互关联，在数据包列表中选中一个数据包之后，在数据包信息面板处就可以查看这个数据包的详细信息。

一般而言，数据包详细信息中包含的内容是用户最关心的。一个数据包通常需要使用多个协议，这些协议一层层地将要传输的数据包封装起来，选择不同的数据包会显示不同的控制层，本例中的数据包依次为Frame（物理层的帧信息）、Ethernet Ⅱ（数据链路层的MAC信息）、Internet Protocol Version4（网络层的IPv4信息）、Transmission Control Protocol（传输层的TCP信息）、Data（数据信息），如图4-8所示。

每一层的前面有一个黑色的三角形图标，单击图标可以展开数据包这一层的详细信息，例如，查看这个数据包中传输层的详细信息就可以单击前面的三角形图标，如图4-9所示，对应着TCP报文的格式。

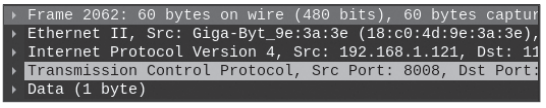

图 4-8

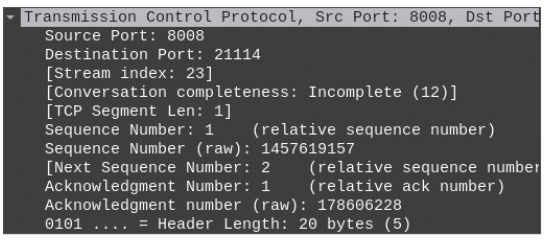

图 4-9

知识拓展

其他层

除了以上的层之外，其他的数据包还可能显示Address Resolution Protocol（request）（地址解析协议）、User Datagram Protocol（传输层UDP信息）、Simple Service Discovery Protocol（应用层，简单服务发现协议）、Transport Layer Security（传输层安全）、Domain Name System（域名信息）等。

3. 筛选数据

很多用户面对这么多的包会无所适从。其实抓包后需要对包进行筛选，找到需要的数据包。在wireshark中叫作应用显示过滤器，在主界面快捷按钮下方。常用的筛选数据包的语言格式如下。

- **ip.addr==1.2.3.4**：筛选源地址或目标地址为1.2.3.4的数据包。
- **ip.src_host==1.2.3.4**：筛选源地址是1.2.3.4的数据包。
- **ip.dst_host==1.2.3.4**：筛选目标地址是1.2.3.4的数据包。
- 如果需要筛选协议，直接使用数据协议的名称，如TCP、UDP、HTTP等。
- **tcp.srcport==80**：筛选出TCP协议源端口号是80的包（筛选目标端口为80的TCP包，则使用tcp.dstport==80。筛选所有使用80端口的TCP包，则使用tcp.port==80，UDP类似。）
- **筛选协议**，则直接使用协议名，如TCP、UDP、DNS、IP、SSL、HTTP、FTP、ARP、ICMP、SMTP、POP、TELNET、SSH、RDP、SIP等。

如筛选目标IP是192.168.1.121的数据包，可以输入ip.dst==192.168.1.121，按回车键后显示结果如图4-10所示。

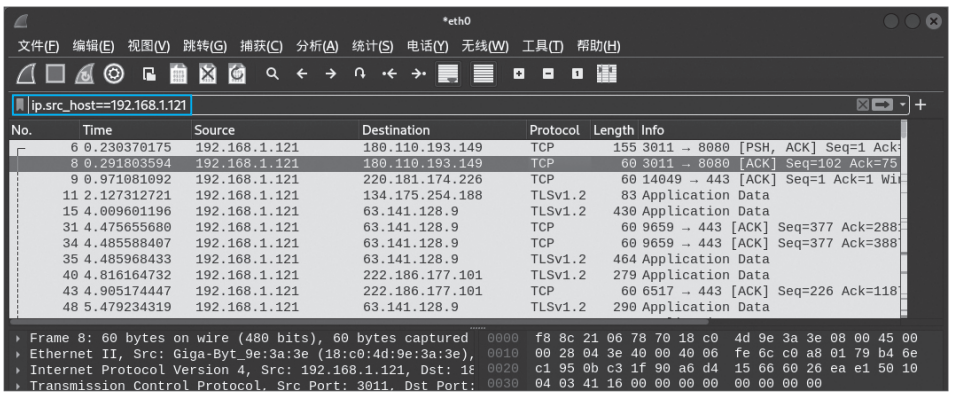

图 4-10

如果要通过多个条件组合筛选，则条件之间用比较运算符连接，如“&&”（与）、“||”（或）、“！”（非）。如筛选源IP地址为192.168.1.121，且目标端口为80的TCP数据包，则使用“ip.src_host==192.168.1.121 && tcp.dstport==80”命令，如图4-11所示。

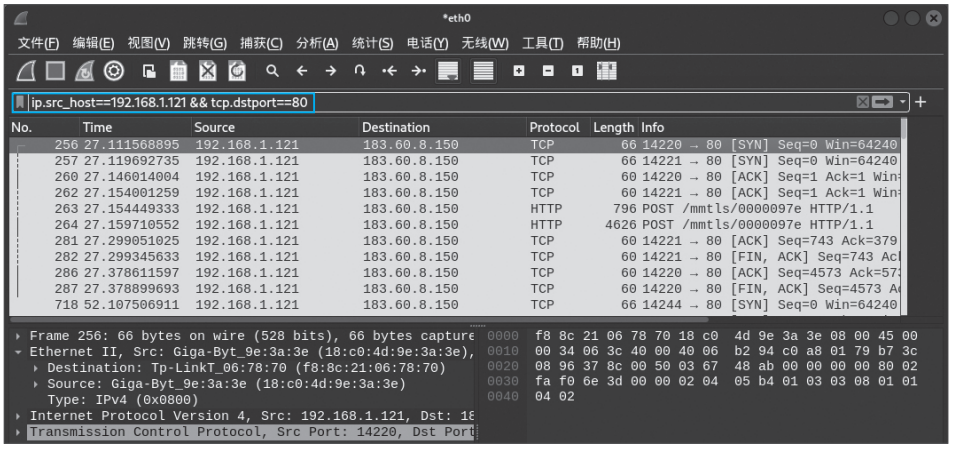

图 4-11

知识拓展

直接搜索协议

如搜索协议为ARP请求的数据包，则输入arp即可，如图4-12所示。

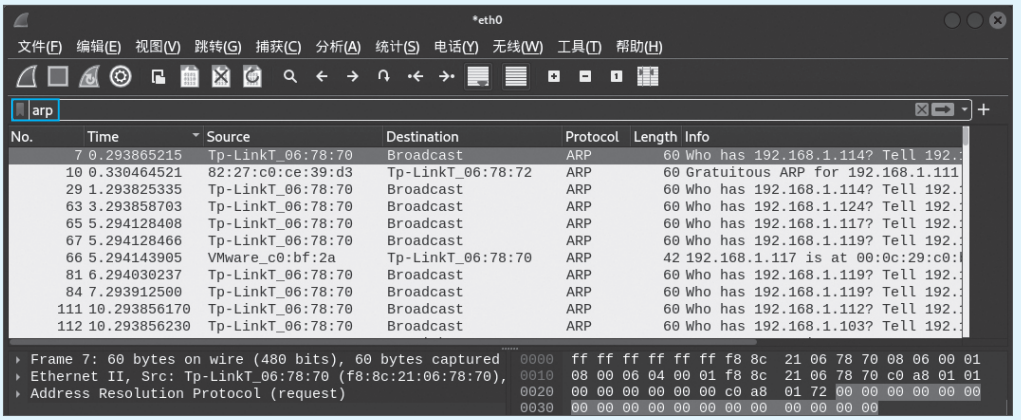

图 4-12

4. 捕获前过滤

捕获全部数据并在捕获后筛选是比较推荐的，这也是网络管理员的日常必备的操作项目。如果发现有异常数据需要尝试监测，则需要在捕获前设置过滤条件。捕获前过滤和捕获后筛选的命令不同，下面介绍一些捕获前过滤的常用命令及用法。

- **host 1.2.3.4**：只捕获IP为1.2.3.4的数据包。
- **net 1.2.3.0/24**：只捕获某个IP地址范围内的数据包。
- **src 1.2.3.4**：只捕获源地址为1.2.3.4的数据包。
- **dst 1.2.3.4**：只捕获目的地址为1.2.3.4的数据包。
- **port 80**：只捕获HTTP（端口80）通信数据包。

在命令行中，也可以使用逻辑运算符号来执行更复杂的过滤，实现精准捕获。如捕获源地址为本机的HTTP包，可以在过滤窗口中输入过滤条件，如捕获源地址为192.168.1.121、端口号为80的数据包，可以使用“src 192.168.1.121&&port 80”命令，选择监控的网络接口，单击左上角的“开始捕获分组”按钮，如图4-13所示。

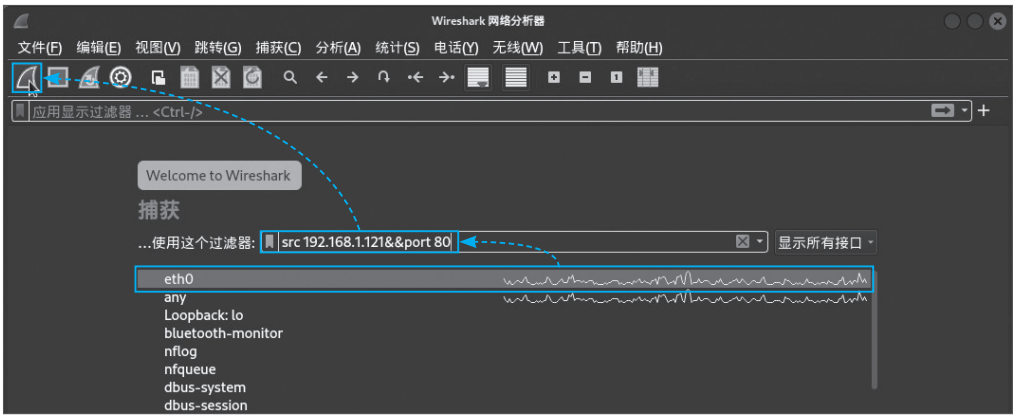

图 4-13

过滤完成后，可以查看捕获的效果，如图4-14所示。只有满足条件的数据包会被显示出来。

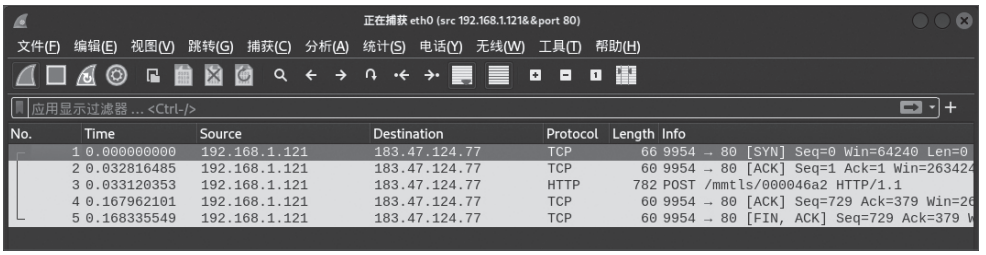

图 4-14

5. 追踪数据流

一个完整的数据流传输一般由很多包组成，可以使用追踪数据流的方法查看并分析一组数据包。下面介绍追踪的方法。

Step 01 在需要追踪数据流的某个数据包上右击，在弹出的快捷菜单中选择“追踪流”|“TCP流”选项，如图4-15所示。

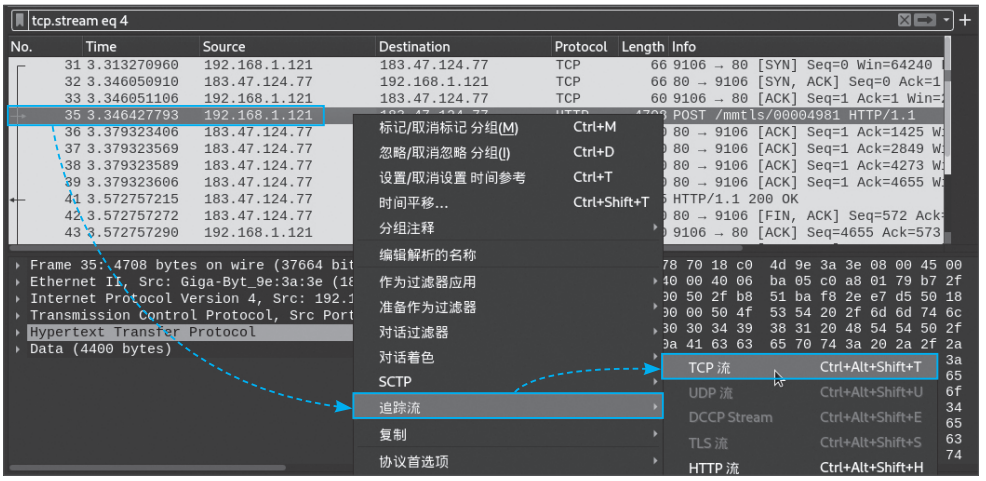

图 4-15

Step 02 软件会筛选出该数据流的所有数据包，如图4-16所示。

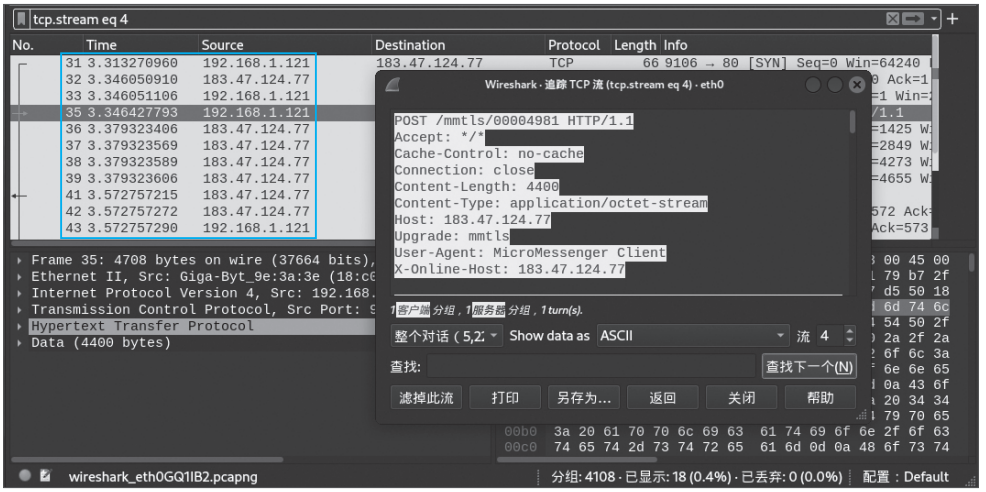

图 4-16