4.5.2 显示过滤器

显示过滤器应用于捕获文件，用来告诉 Wireshark 只显示那些符合过滤条件的数据包。你可以在 Packet List 面板上方的 Filter 文本框中，输入一个显示过滤器。

显示过滤器比捕获过滤器更加常用，因为它可以让你对数据包进行过滤，却并不省略掉捕获文件中的其他数据。也就是说如果你想回到原先的捕获文件，则仅仅需要清空显示过滤表达式。

在有些时候，你可能会需要使用显示过滤器，来清理过滤文件中不相关的广播流量，比如清理掉 Packet List 面板中与当前的分析问题并没有什么联系的 ARP 广播，但是那些 ARP 广播之后可能会有用，所以最好是把他们暂时过滤掉，而不是删除它们。

如果想要过滤掉捕获窗口中所有的 ARP 数据包，那么将你的鼠标放到 Packet List 面板上方的 Filter 文本框中，然后输入！ arp，就可以从 Packet List 面板中去掉所有的 ARP 数据包了，如图 4-16 所示。如果想要删除过滤器，则可单击 X 按钮；如果想要保存过滤器的话，则可单击（+）按钮。

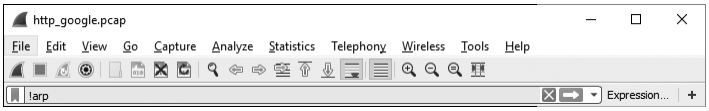

图 4-16　使用 Packet List 面板上方的 Filter 文本框创建一个显示过滤器

应用显示过滤器有两种方法，一种是就像刚才的例子一样，直接键入合适的语法表达式；另一种是使用显示过滤器对话框来选择构建，这也是初学过滤器的一个简单方法。让我们来看一看这两种方法吧，首先从简单的开始。

1．过滤器表达式对话框（简单方法）

过滤器表达式对话框，如图 4-17 所示，使得 Wireshark 的初学者也能很简单地创建捕获和显示过滤器。如果想要打开这个对话框，则可以先在 Capture Option 对话框中单击 Caputre Filter 按钮，然后单击 Expression 按钮。

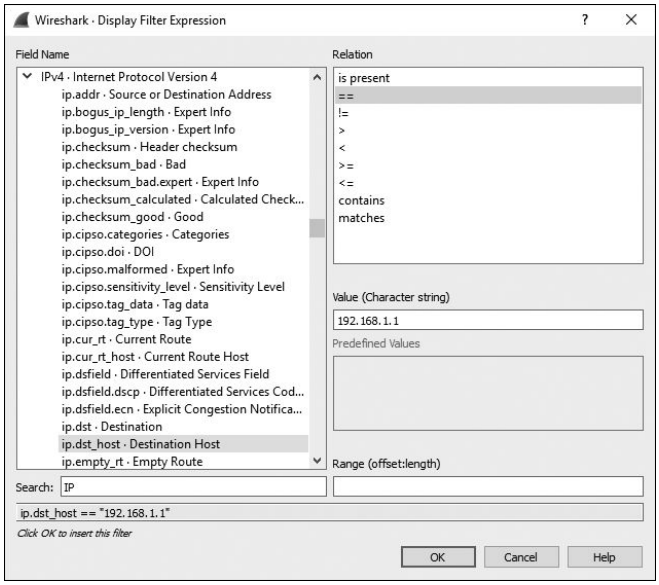

图 4-17　Filter Expression 对话框可以让你很容易地在 Wireshark 中创建过滤器

对话框左边列出了所有可用的协议域，这些域指明了所有可能的过滤条件。如果想创建一个过滤器，则可以按照如下步骤操作。

（1）单击一个协议旁边的加号（+），以展开所有与这个协议相关可作为条件的域，找到你要在过滤器中使用的那一项，然后单击选中它。

（2）选择一个你想要在选中条件域和条件值之间建立的关系，比如等于、大于和小于等。

（3）通过输入一个和你选中条件域相关的条件值来创建过滤器表达式。你可以自己定义这个值，也可以从 Wireshark 预定义的值中选择一个。

（4）当你完成所有上述步骤时，单击 OK 就可以看到你的过滤器表达式的文本表示。

Filter Expression 对话框对于初学者来说很好用，但在你熟悉了这一套规则之后，就会发现手动输入过滤器表达式更有效率。显示过滤器表达式的语法结构非常简单，但功能十分强大。

2．过滤器表达式语法结构（高级方法）

在使用一段时间的 Wireshark 后，为了节约时间你希望在主窗口下直接使用显示过滤器的语法。幸运的是，显示过滤器的语法遵从一个标准的模式并且是易于导航。在大多数情况下，这个语法模式以具体协议为中心并且遵从 protocol.feature.subfeature 的格式，就像你在显示过滤器表达式对话框看到的一样。现在让我们来看一看具体的几个例子。

你会经常用到捕获或者显示过滤器来对某一个协议进行过滤。举例来说，如果你要解决一个 TCP 问题，那么你就只希望看到捕获文件中的 TCP

流量。一个简单的 TCP 过滤器就可以解决这个问题。

现在让我们看一看另外一些情况。假如为了解决你的 TCP 问题，你使用了很多 ping 功能，因此产生了很多 ICMP 流量。你可以通过！ icmp 这个过滤器表达式，将你捕获文件中的 ICMP 流量屏蔽掉。

比较操作符允许你进行值的比较。举例来说，当检查一个 TCP/IP 网络中的问题时，你可能经常需要检查和某一个 IP 地址相关的数据包。等于操作符可以让你创建一个只显示 192.168.0.1 这个 IP 地址相关数据包的过滤器：

```
ip.addr==192.168.0.1
```

现在假设你只需要查看那些长度小于 128 字节的数据包，那么你可以使用「小于或等于」操作符来完成这个要求，其过滤器表达式如下：

```
frame.len<=128
```

表 4-4 给出了 Wireshark 过滤器表达式的比较操作符。

表 4-4　Wireshark 过滤器表达式的比较操作符

操作符	说明
	等于
!=	不等于
>	大于
<	小于
>=	大于或等于
<=	小于或等于

逻辑运算符可以让你将多个过滤器表达式合并到一个语句中，从而极大地提高过滤器的效率。举例来说，如果只想显示两个 IP 地址上的数据包，

那么我们可以使用 or 操作符来创建一个表达式，只显示这两个 IP 地址的数据包，如下：

```
ip.addr==192.168.0.1 or ip.addr==192.168.0.2
```

表 4-5 列出了 Wireshark 的逻辑操作符。

表 4-5　Wireshark 过滤器表达式的逻辑操作符

操作符	说明
and	两个条件需同时满足
or	其中一个条件被满足
xor	有且仅有一个条件被满足
not	没有条件被满足

3．显示过滤器表达式实例

虽然编写过滤器表达式在概念上很简单，但是在解决不同问题时创建的过滤器，仍然需要许多特定的关键词与操作符。表 4-6 给出了我经常使用的显示过滤器。

表 4-6　常用显示过滤器

过滤器	说明
!tcp.port==3389	排除 RDP 流量
tcp.flags.syn==1	具有 SYN 标志位的 TCP 数据包
tcp.flags.rst==1	具有 RST 标志位的 TCP 数据包
!arp	排除 ARP 流量

过滤器	说明
http	所有 HTTP 流量
tcp.port==23 || tcp.port==21	文本管理流量（Telnet 或 FTP）
smtp || pop || imap	文本 email 流量（SMTP、POP 或 IMAP）