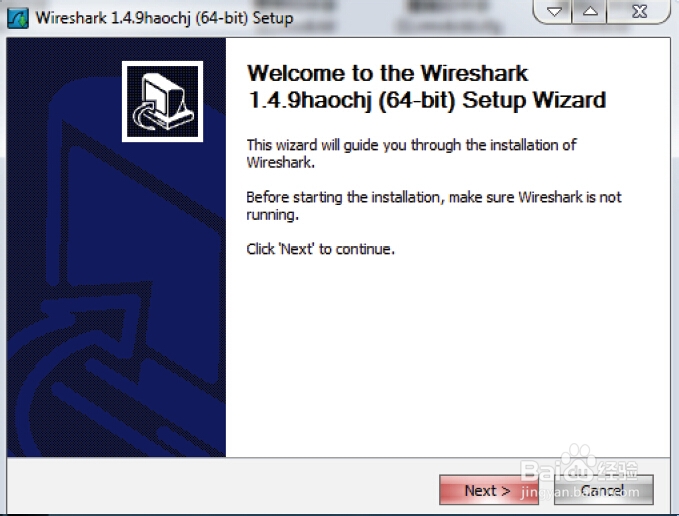
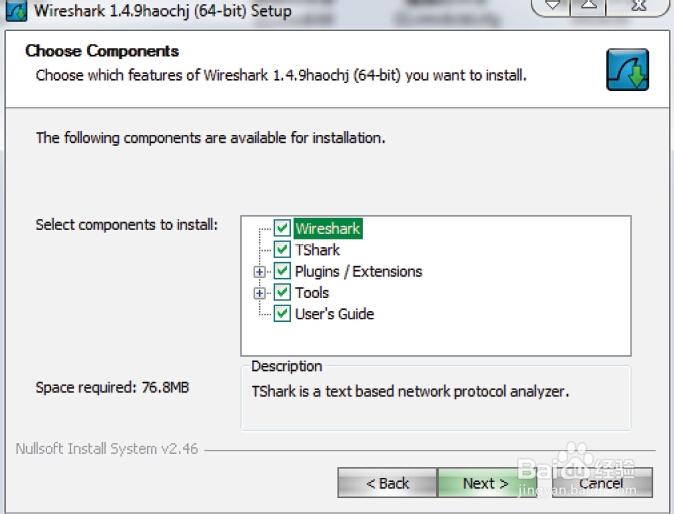
如何在windows上安装wireshark

* 浏览：5558
* |
* 更新：2014-10-23 15:47

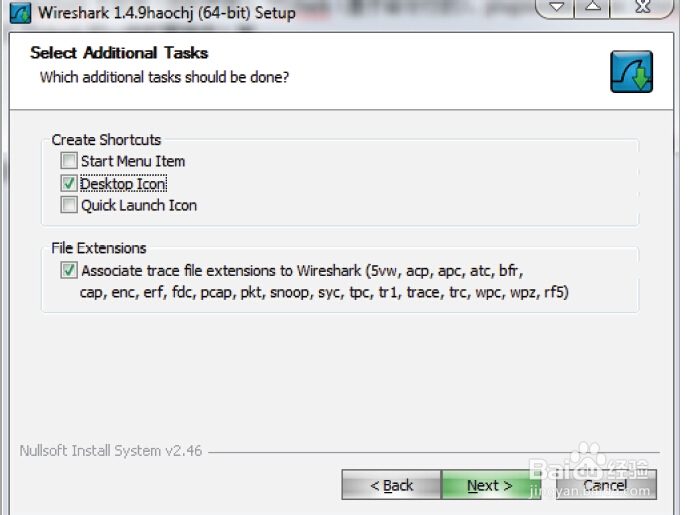
在互联网搜索到我们想要安装的软件版本，执行安装包后弹出如下欢迎界面：

点击下一步进行安装！

如图2-1-2这一步wireshark会询问我们需要安装哪些程序组件。我们可以选择性的安装我们必须的一些wireshark附带的工具； 

通常使用到的有：wireshark主程序（图形界面）、TSshark（基于命令行的）、plugins/extensions（wireshark以及TSshark的一些扩展插件）等...在此我们选择安装全部组件程序后点击“Next”进入到下一步安装； 

如图2-1-3所示在此界面wireshark安装程序会询问我们是否要在开始菜单（Start Menu Item）、桌面快捷方式（Desktop Icon）、快速启动栏（Quick Launch Icon）以及是否关联所列出的后缀名文件。在此我们选择建立桌面快捷方式并且关联到列出后缀名文件后点击“Next”进入到下一步安装；



如图2-1-4所示wireshark询问我们的安装位置。我们选择F盘点击“Next”；

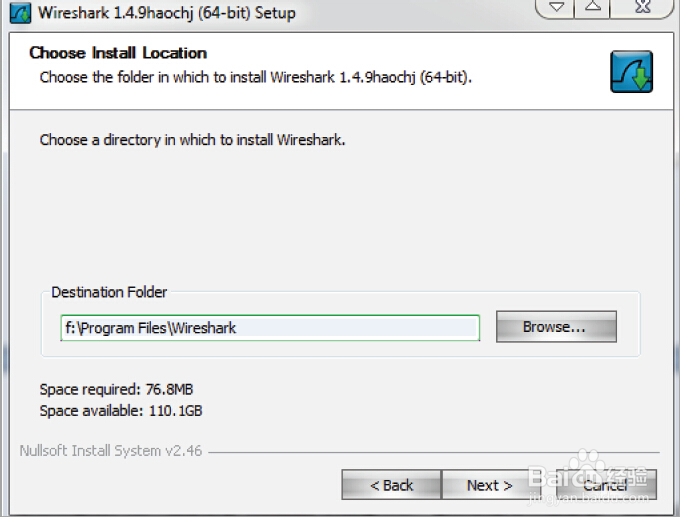
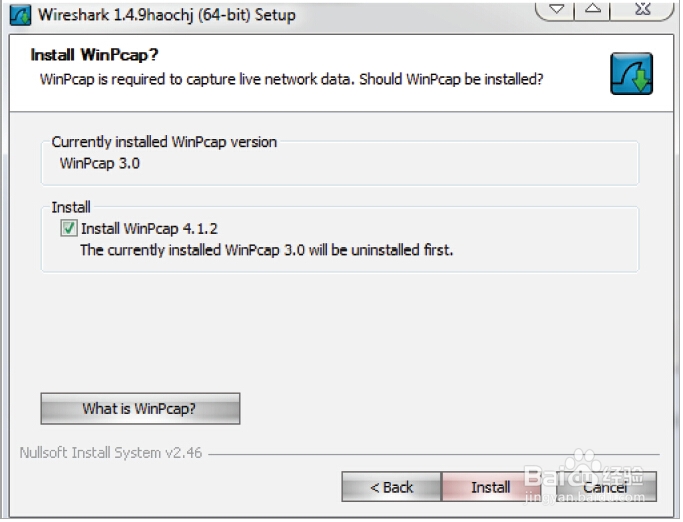
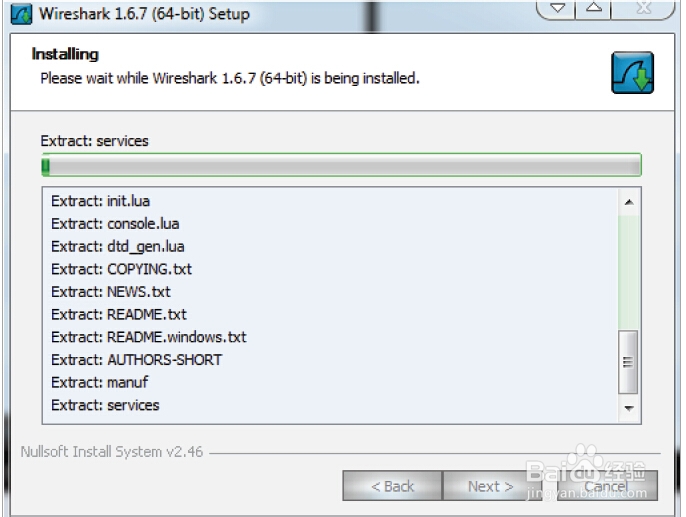


图2-1-5所示，wireshark安装程序询问我们是否安装依赖软件WinPcap 3.0（该软件主要用于捕获网络底层数据包），假如您的电脑上已经安装了该软件，那么我们可以取消掉该软甲的安装。在此处由于我们没有安装过该软件我们勾选该复选框后点击“Install”执行安装；



如图2-1-7所示我们进入到WinPcap的安装界面点击“Next”进入到下一步安装；



如图2-1-8点击“Next”进行到WinPcap的下一步安装操作；

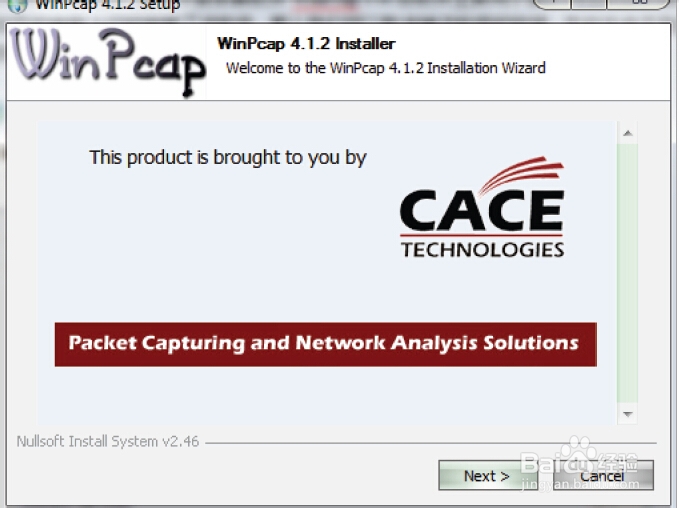
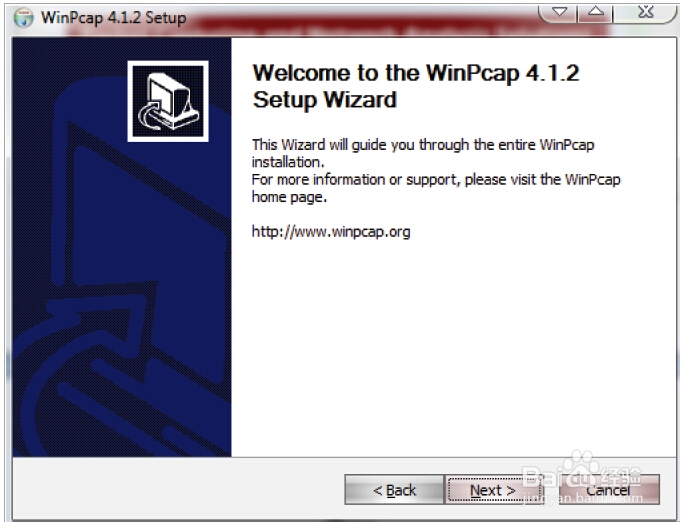
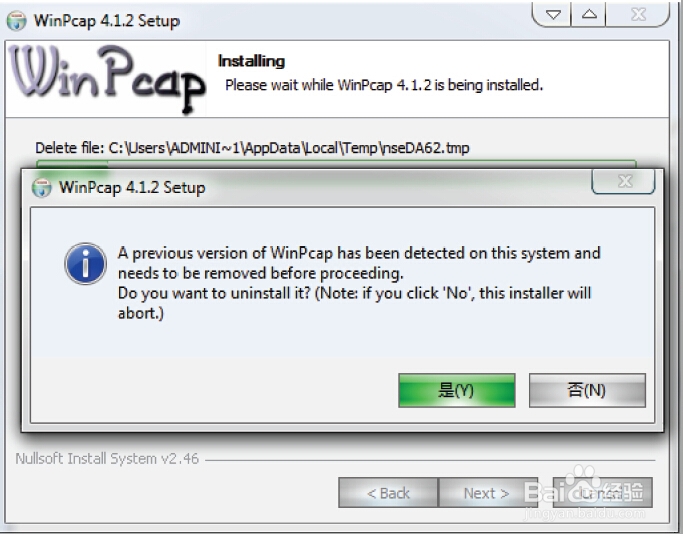


图2-1-9所示，WinPcap询问我们是否开机自动启动WinPcap驱动程序。我们采取默认选项后点击“Install”；



假如出现图2-1-10的提示，那么就是您的电脑已经安装了该类软件，WinPcap询问我们是否卸载以前所安装的WinPcap。在此我们选择“是”后WinPcap开始自动安装；



到了如图2-1-11所示的界面后说名wireshark已经安装成功了，点击“Finish”后在桌面我们可以看到wireshark的快捷方式；

## 基本使用

以下的介绍都是以mac下的wireshark 1.12.2版本为基础。

### 认识界面



说明：

常用按钮从左到右的功能依次是：

1、列出可用接口。

2、抓包时需要设置的一些选项。一般会保留最后一次的设置结果。

3、开始新的一次抓包。

4、暂停抓包。

5、继续进行本次抓包。

6、打开抓包文件。可以打开之前抓包保存后的文件。不仅可以打开wireshark软件保存的文件，也可以打开tcpdump使用-w参数保存的文件。

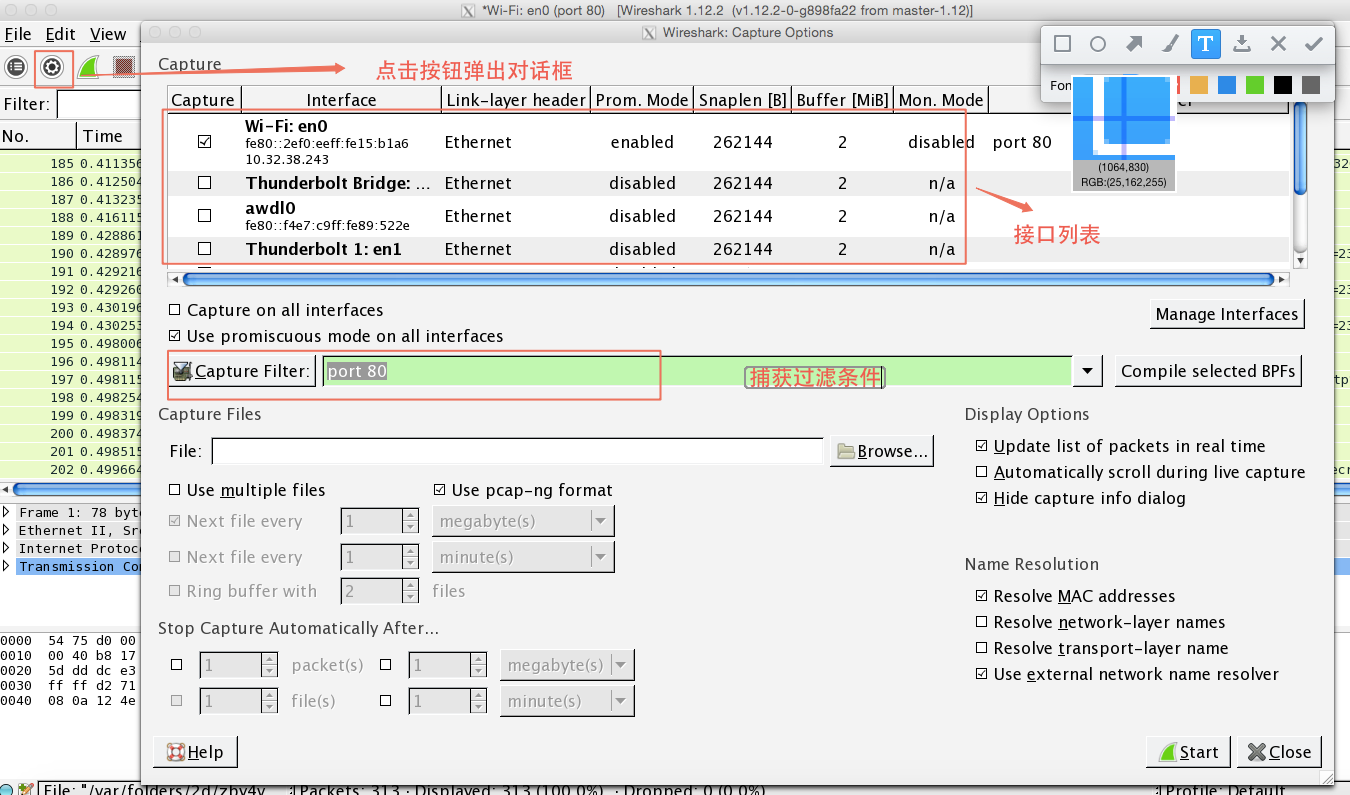
7、保存文件。把本次抓包或者分析的结果进行保存。

8、关闭打开的文件。文件被关闭后，就会切换到初始界面。

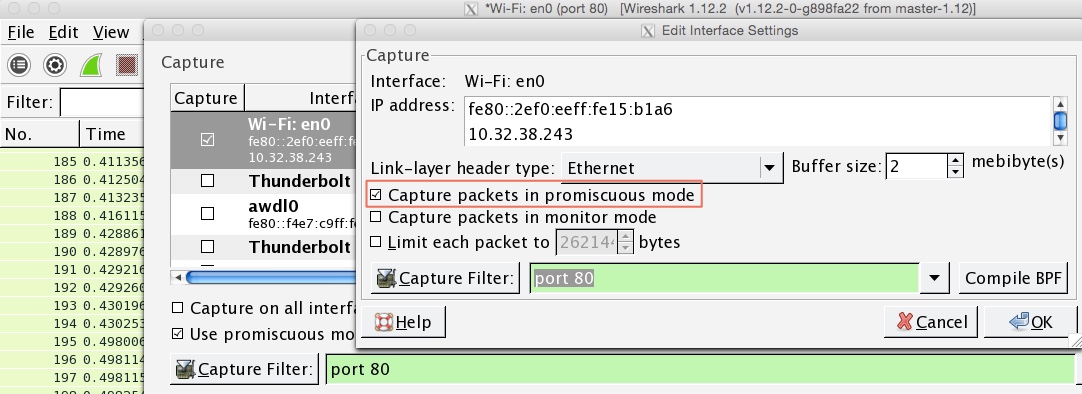
9、重载抓包文件。

### 设置数据抓取选项

点击常用按钮中的设置按钮，就会弹出设置选项对话框。在这个对话框中我们可以选中需要监听的接口，设置混杂模式，设置抓取数据包的过滤条件。如下图：



首先，选中需要监听获取数据包的接口。接口列表区列出了所有可以使用的接口。如果接口前面的复选框被选中，说明对这个接口监听捕获数据包。

其次，设置混杂模式。设置混杂模式的作用是将网卡设置到混杂模式。如果不设置混杂模式，你的计算机只能获取数据包发往的目标是你计算机和从你计算机出去的数据包。如果设置了混杂模式，你就可以捕获局域网中所有的数据包。如果窗口中的 "Use promiscuous mode on all interfaces"前面的复选框被选中，说明对所有的接口使用混杂模式。如果想单独设置，可以双击接口列表中的接口，会弹出如下的对话框。然后选中或者去掉“Capture packets in promiscuous mode”前面复选框。然后点ok按钮。  


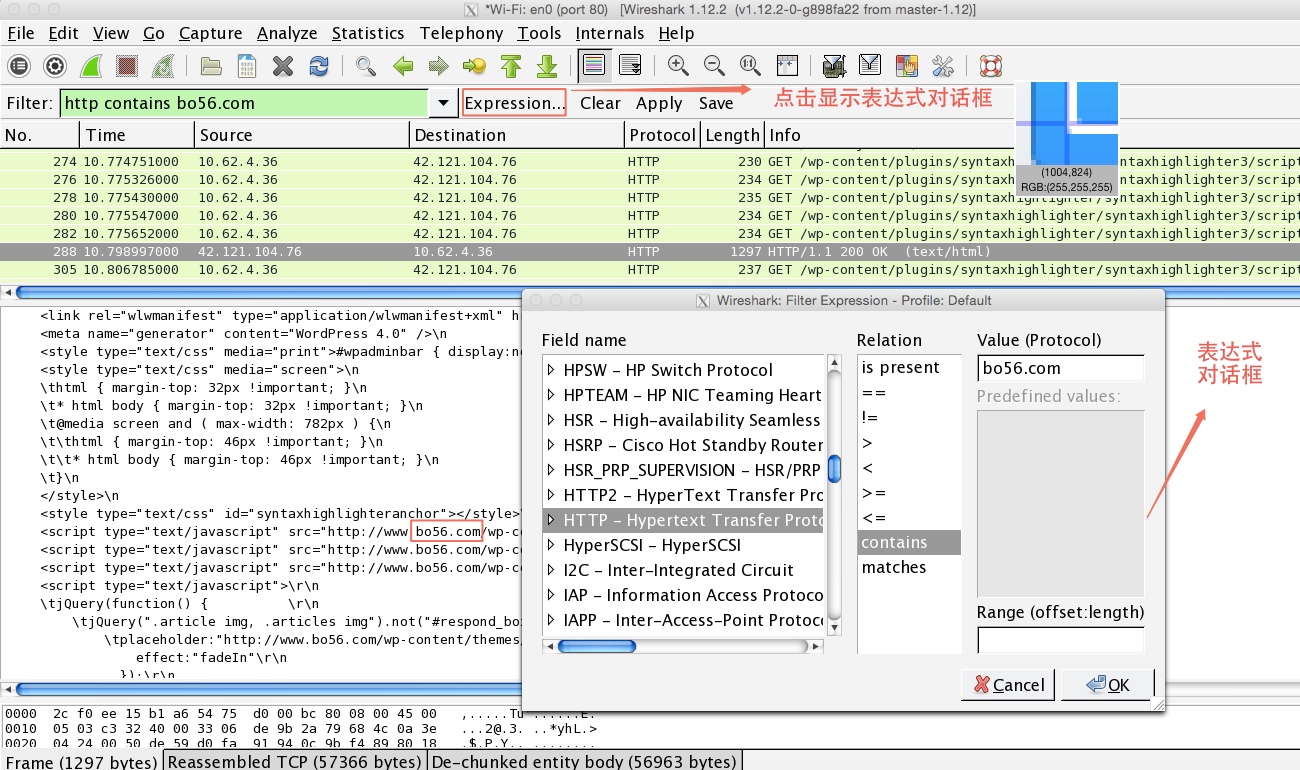
再次，设置捕获过滤条件。在点设置按钮弹出的主设置对话框中和双击接口列表弹出的对话框中都会有“Capture Filter”项。在文本框中我们可以设置捕获过滤条件。如，我们只捕获http相关的数据包，我们就可以设置捕获条件为“port 80”。因为http使用的是80端口。

最后，所有的设置完毕后，点击设置主窗口中的“Start”按钮，开始捕获数据。数据捕获完后，可以点常用按钮中的“保存”按钮保存数据。

### 使用显示过滤器

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。显示过滤器比捕获过滤器更常用。他可以用来过滤不想看到的数据包，但是不会把数据删除。如果想恢复原状，只要把过滤条件删除即可。

过滤器表达式对话框，是的wireshark的可以很简单的设置过滤表达式。点击“Expression”按钮就可以打开这个对话框。如下图：



对话框分左中右三部分。左边为可以使用的所有协议域。右边为和协议域相关的条件值。中间为协议域与条件值之间的关系。过滤器表达式对于初学者很有用。如上图，我们创建的表达式的作用是，只显示http协议包中包含关键词“bo56.com”的所有数据包。

Field name说明：  
这个列表中展示了所有支持的协议。点击前面的三角标志后，可以列出本协议的可过滤字段。当选中“Field name”列表中的任何一项，只需要输入你想要的协议域，就会自动定位到相应的协议域选项。

Relation说明：  
is present    如果选择的协议域存在，则显示相关数据包。  
contains     判断一个协议，字段或者分片包含一个值  
matches             判断一个协议或者字符串匹配一个给定的Perl表达式。

Value（Protocol）说明：  
此处输入合适的值。如果选择的协议域和这个值满足Relation中指定的关系，则显示相关数据包。

Predefined values说明：  
有些协议域包含了预先定义的值，有点类似于c语言中的枚举类型。如果你选择的协议域包含这样的值，你可以在这个列表中选择。

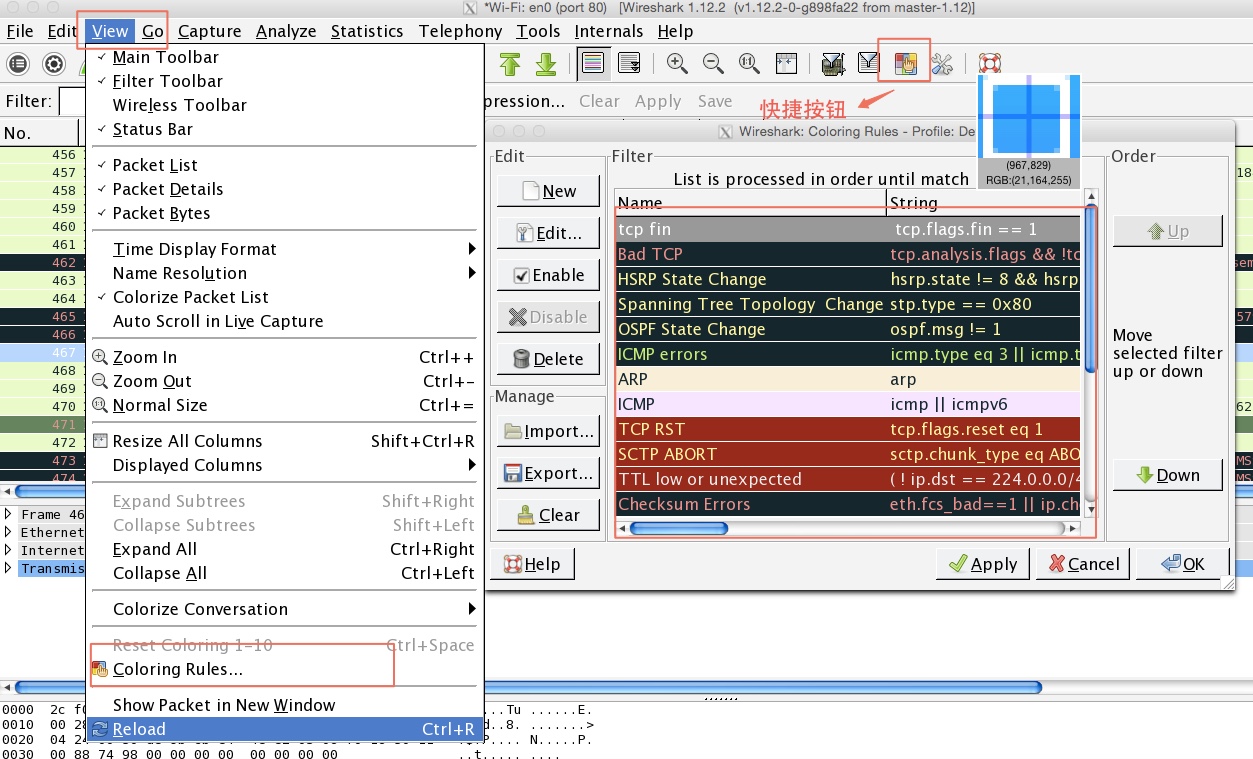
Function函数说明：  
过滤器的语言还有下面几个函数：  
upper(string－field)－把字符串转换成大写  
lower(string－field)－把字符串转换成小写  
upper((和lower((在处理大小写敏感的字符串比较时很有用。例如：  
upper(ncp.nds\_stream\_name) contains "BO56.COM"  
lower(mount.dump.hostname) =="BO56.COM"

如果你熟悉了这个规则之后你就会发现手动输入表达式更有效率。当时手动在flter文本框中输入表达时，如果输入的语法有问题，文本框的背景色会变成红色。这时候，你可以继续输入或者修改，知道文本框中的表达式正确后，文本框的背景色又会变成绿色。

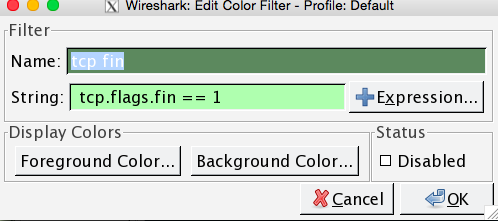
### 使用着色规则

你经常会在数据包列表区域中看到不同的颜色。这就是wireshark做的很人性化的一方面。它可以让你指定条件，把符合条件的数据包按指定的颜色显示。这样你查找数据包会更方便些。下面我们说一下如何设置颜色规则。

点击“view”菜单，然后选择“Coloring Rules”选项就会弹出设置颜色规则设置对话框。你点击颜色规则设置的快捷按钮也可以打开颜色设置对话框。如下图：



打开的对话框中默认已经有一些规则。我们抓取的数据包中经常会看到一些不同的颜色，就是应用的这些默认的规则。点击“New”按钮可以添加规则。如下图：



name字段中填写规则的名称，方便记忆。

string字段中填写过滤规则。这里的语法和显示规则表达式一致。点击 上图中的“Expression”按钮，你就会看到熟悉的规则表达式对话。

Foreground Color按钮用于选择前景色。

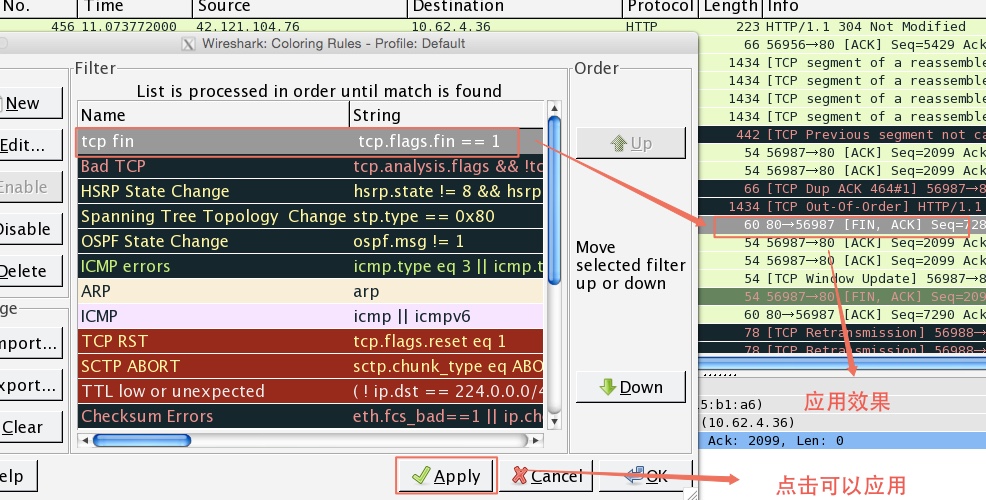
Background Color按钮用于选择背景色。

Disabled按钮用于指示是否禁用这条规则。

点击ok按钮后，规则自动会添加到规则列表中的最前端。

注意：wireshark在应用规则的时候，是按自上而下的顺序去应用规则。因此刚添加的规则会优先应用。如果你想调整顺序，可以选中要调整顺序的规则，然后点击右边的“UP” 或则 “Down” 按钮。

颜色规则设置好后，只需要点apply按钮就可以应用规则了。规则效果应用如下图：



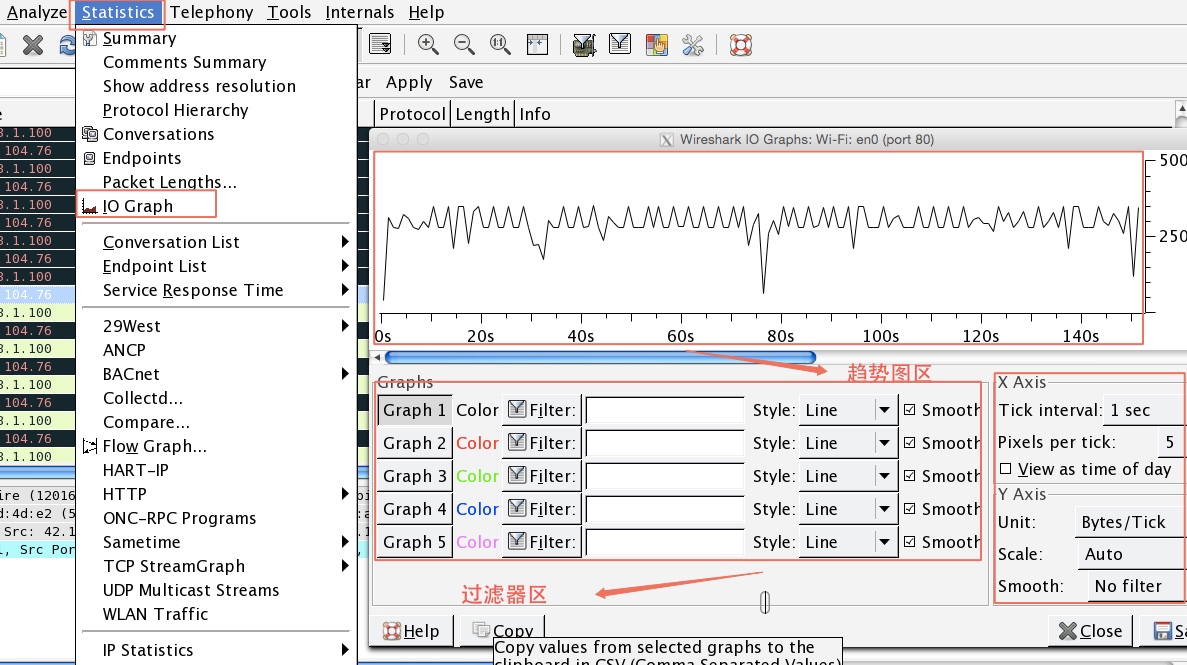
### 

### 使用图表

图形分析是数据分析中必不可少的一部分。也是wireshark的一大亮点。wireshark有不同的图形展现功能，以帮助你了解捕获的数据包。下面我们对经常使用的IO图，双向时间图做下介绍。

#### IO图

wireshark的IO图让你可以对网络上的吞吐量绘图。让你了解网络数据传输过程中的峰值和波动情况。通过“Statistics”菜单中的“IO Graphs”选项可以打开这个IO图对话框。如下图：



可以看到IO图表对话框中会分为三个区。

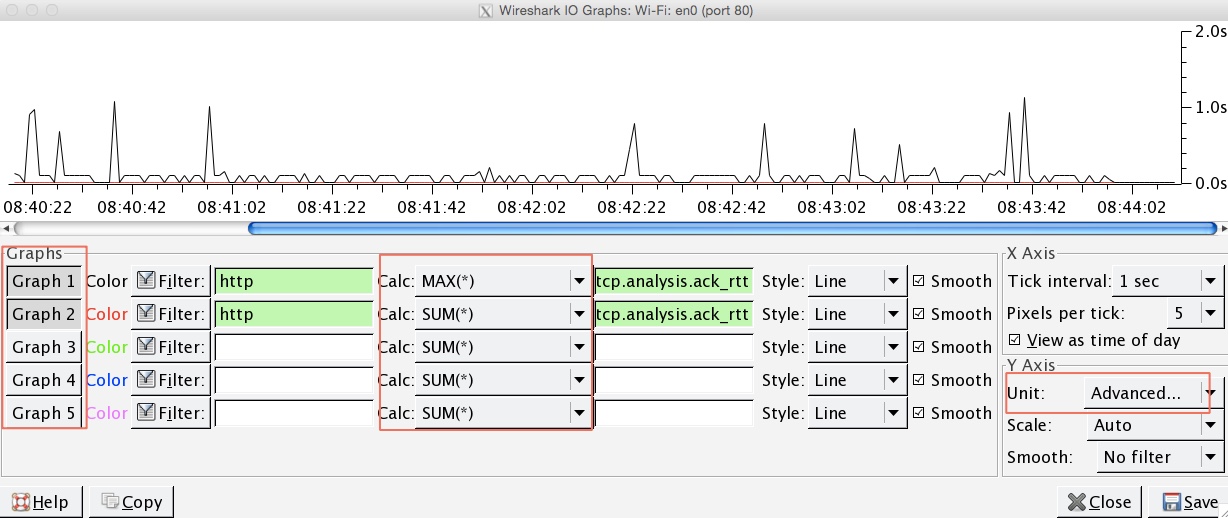
过滤器区：设置过滤条件，用于图形化展示过滤条件相关数据包的变化情况。而且可以为每个不同的条件指定不同的颜色。过滤条件的语法和之前介绍的显示过滤器的语法一致。过滤条件为空，此图形显示所有流量。

坐标区：在这里可以设置图表的x轴和y轴。x轴为时间，y轴为包的数量。如图，我们设置Y轴的单位是Bytes/Tick。

趋势图区：根据过滤器设置的条件和坐标区设置，数据分析后回在这个区域以图形化方式展示。点击图形中的点，会自动定位到相应的数据包。点击趋势图中的低谷点，你会发现大量的数据包重传。

IO图表还可以通过函数对数据进行聚合处理。

点击Y轴中Unit选项中的Advanced后，就会再过滤器区就会增加Calc选项。如下图：



相关函数说明：

MIN( ), AVG( ), MAX( ) 分别是统计协议域中数值的最小，平均和最大值。注意，这三个聚合函数只对协议域的值为数字的才有效。

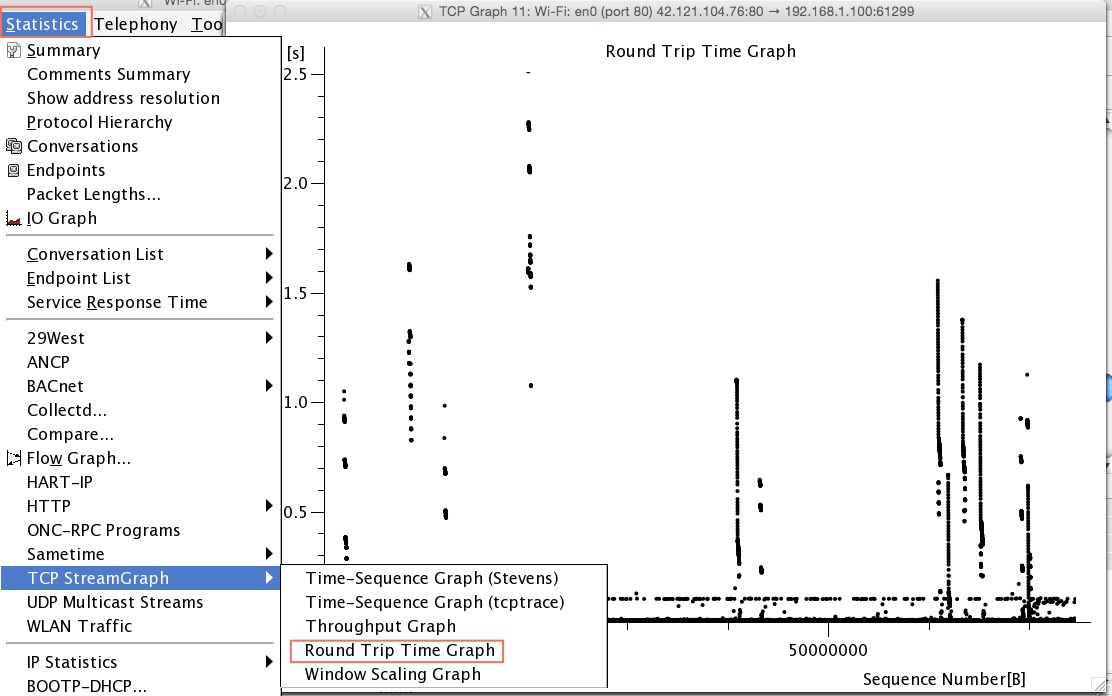
Count( ) 此函数计算时间间隔内事件发生的次数，在查看TCP分析标识符时很有用，例如重传。

Sum( ) 该函数统计事件的累加值。和MIN()函数一样，这个也只有协议域的值为数字的情况下才有效。

#### 双向时间图

wireshark还有一个功能就是可以对网络传输中的双向时间进行绘图。双向时间（round-trip time, RTT）,就是一个数据包被确认正常接收所花费的时间。以tcp协议为例，就是你push一个数据到一台主机，主机回应一个ack给你的主机，你的主机并成功接收ack回应。这两个过程花费的时间总和就是双向时间。双向时间通常用来寻找网络传输过程中的慢点和瓶颈，用以判断网络传输是否有延迟。

通过“Statistics”菜单中的“Tcp  StreamGraph”中的“Round Trip Time Graph”选项可以打开这个双向时间图对话框。如下图：

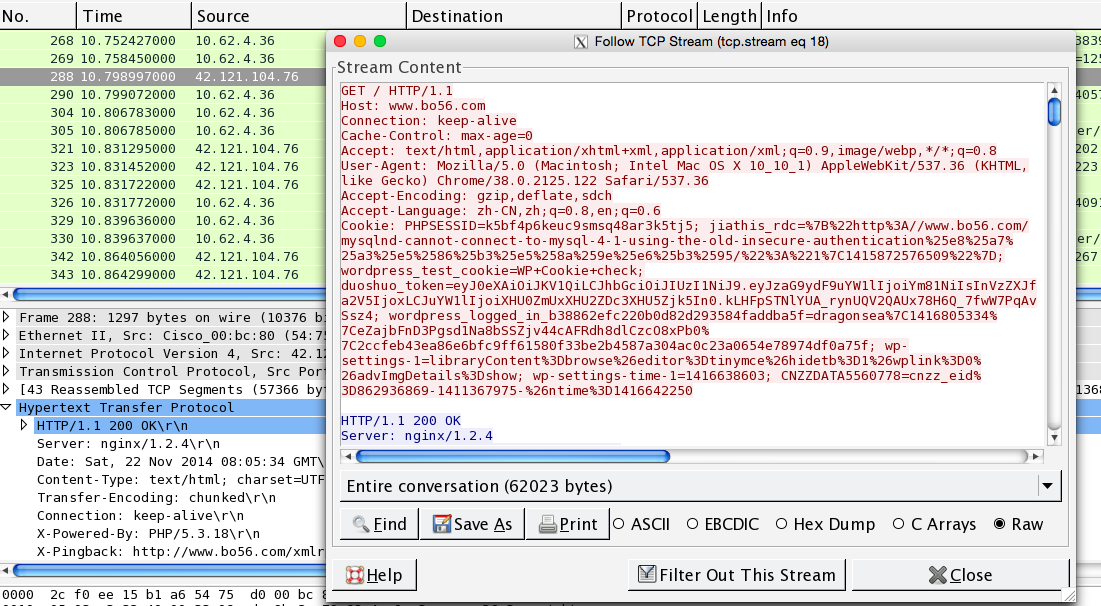


这个图表中的每个点代表一个数据包的双向时间。你可以单机图表中的任何一点，然后在数据包列表区就会自动定位到相应的数据包。从数据表来看，我们下载压缩包还是比较稳定的。数据包的rtt时间大多数在0.05s以下，其他大多数在0.1s左右，少数超过了1.5s。

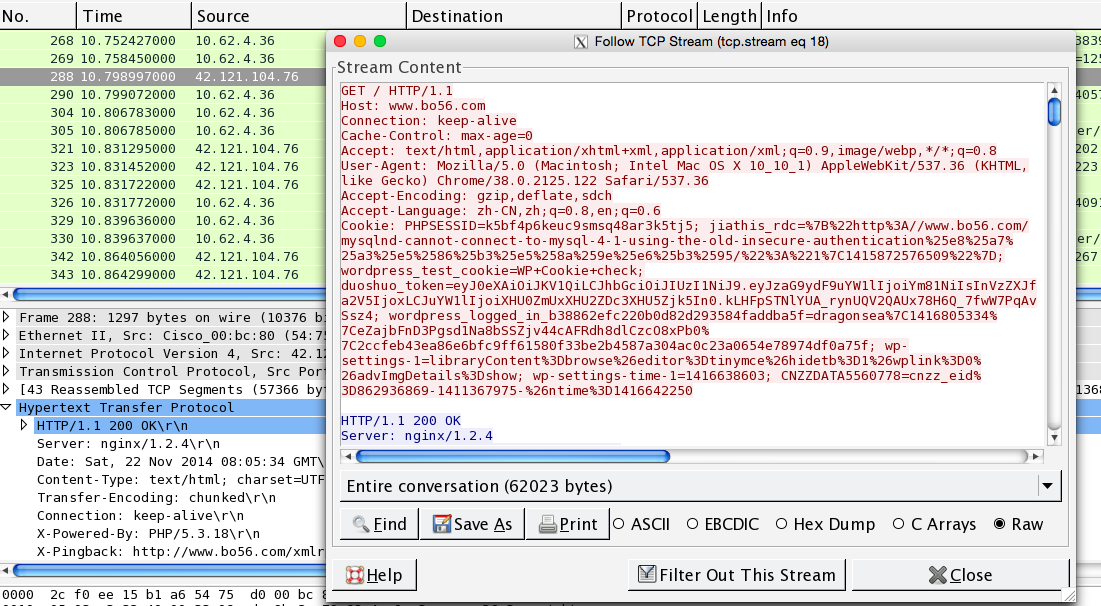
### 跟踪tcp流

Wireshark分析功能中最不错的一个功能是它能够将TCP流重组。重组后的数据格式更容易阅读。跟踪TCP流这个功能可以将接收到的数据排好顺序使之容易查看，而不需要一小块一小块地看。这在查看HTTP、FTP等纯文本应用层协议时非常有用。

我们以一个简单的HTTP请求举例来说明一下。打开wireshark\_bo56\_pcap.pcapng，并在显示过滤器中输入“http contains wireshark”，点击“apply”按钮后，在数据包列表框中就会只剩下一条记录。如下图。



右键单击这条记录并选择Follow TCP Stream。这时TCP流就会在一个单独的窗口中显示出来。如下图：

[](http://www.bo56.com/wp-content/uploads/2014/11/wireshark_tcp_follow_dialog.png)

我们看到这个窗口中的文字会有两种颜色。其中红色用于表示从源地址到目标地址的流量。在我们的例子里面就是从我们本机到web服务器的流量。你可以看到最开始的红色部分是一个GET请求。蓝色部分是和红色部分相反的方向，也就是从目标地址到源地址的流量。在我们的例子中，蓝色部分的第一行是“HTTP/1.1 200 OK”，是来自服务器的一个http成功响应。

在这个窗口中除了能够看到这些原始数据，你还可以在文本间进行搜索，将其保存成一个文件、打印，或者以ASCII码、EBCDIC、十六进制或者C数组的格式去查看。这些选项都可以在跟踪TCP流窗口的下面找到。