WebScarab 入门教程

WebScarab 具有大量的功能，因而可能会让新用户有一种无从下手之感。为求简单起见，拦截和修改浏览器和 HTTP/S 服务器的请求和响应可以作为初学者很好的入门课，因为这无需学习太多的内容就可以完成。

首先，我们假定您能够自由访问因特网，也就是说，您并非位于一个代理之后。为简单起见，我们还假定您使用的浏览器是 Internet Explorer。

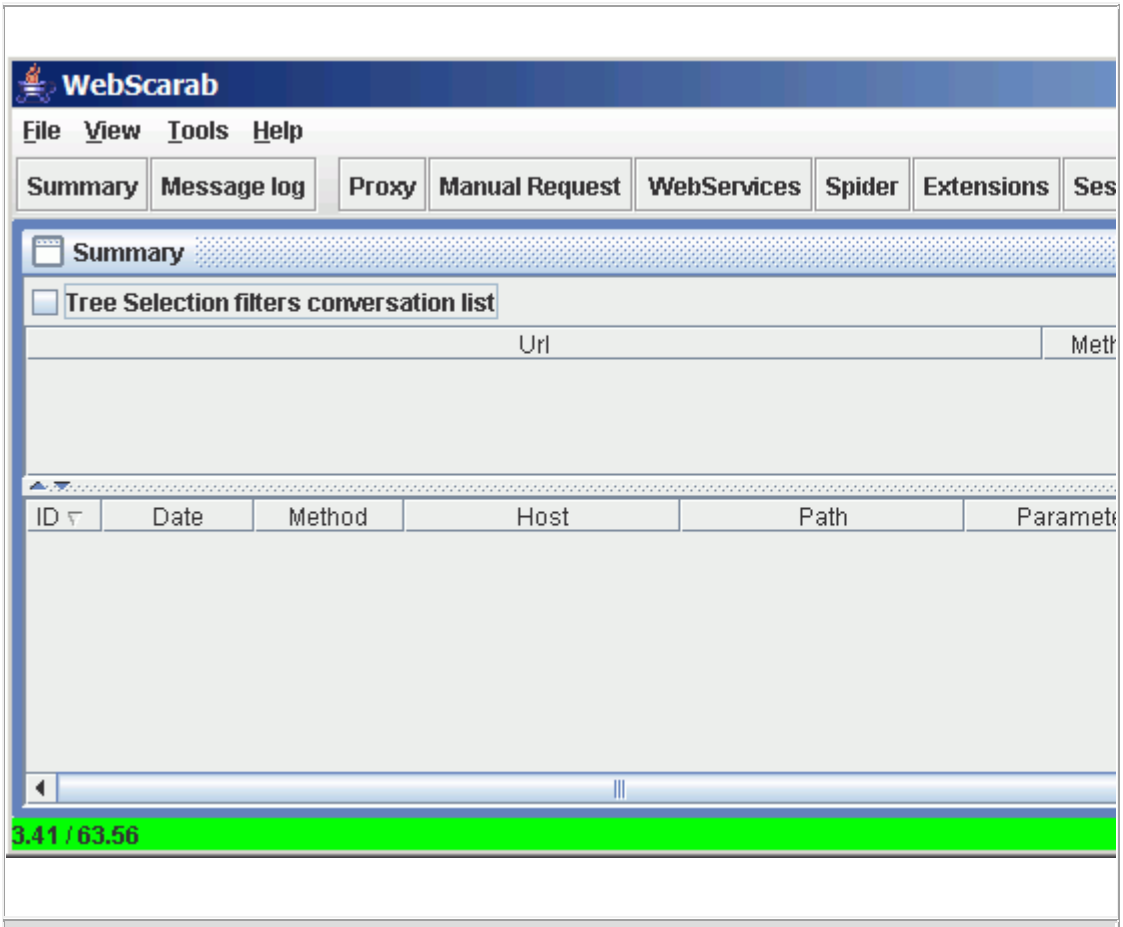

图 7

上面是 WebScarab 启动后的截图，其中有几个主要的区域需要介绍一下。首先要介绍的是工具栏，从这里可以访问各个插件，摘要窗口(主视图）和消息窗口。

摘要窗口分成两个部分，上面部分是一个树表，显示我们访问的站点的布局，以及各个 URL 的属性。下面部分是一个表格，显示通过 WebScarab 可以看到的所有会话，正常情况下它们以 ID 逆序排列，所以靠近表顶部的是最近的会话。当然，会话的排列次序是可以更改的，如果需要的话，只需通过单击列标头即可。

为了将 WebScarab 作为代理使用，需要配置浏览器，让浏览器将 WebScarab 作为其代理。我们可以通过 IE 的工具菜单完成配置工作。通过菜单栏，依次选择选择“工具”菜单、“Internet 选项”、“连接”、“局域网设置”来打开代理配置对话框。

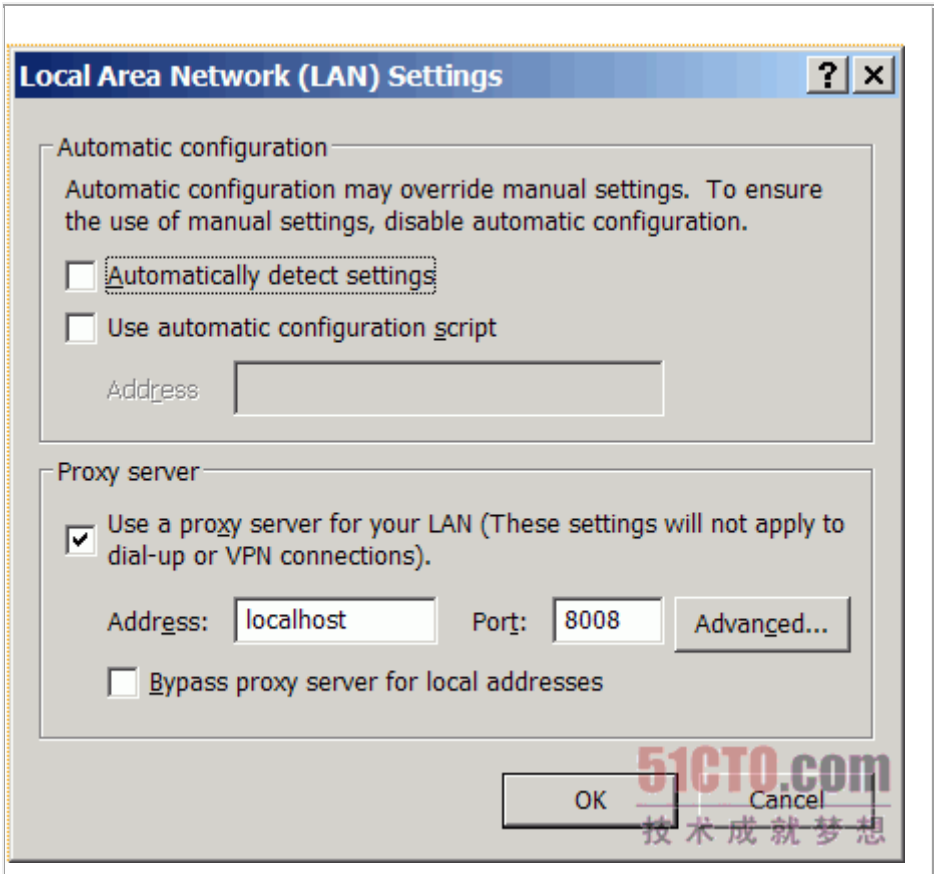

图 8

WebScarab 默认时使用 localhost 的 8008 端口作为其代理。需要对 IE 进行配置，让 IE 把各种请求转发给 WebScarab，而不是让 IE 读取这些请求，如上图所示。确保除“为 LAN 使用代理服务器”之外的所有复选框都处于未选中状态。为 IE 配置好这个代理后，在其它对话框中单击确定按钮，并重新回到浏览器。浏览一个非 SSL 的网站，于是转向 WebScarab。

这时，您应该看到如下图所示的画面；否则的话，或者是在浏览时遇到错误的话，您应当回到上面的步骤，检查你的 Internet Explorer 中的代理设置是否如上所述。如果代理设置是正确的，还有一种可能原因是端口 8008 已经被其他程序占用，这样的话 WebScarab 就无法正常使用该端口了。如果是这样的话，您应当停用那个程序。后面我们会介绍如何让 WebScarab 使用不同的端口。

注意：如果您正在使用 WebScarab 测试的站点与浏览器位于同一个主机之上(即 localhost 或者 127.0.0.1)，并且浏览器为 IE7 的话，则需要在主机名的后面添加一个点号“.”，从而强迫 IE7 使用您配置的代理。这可不是 WebScarab 的一个 bug，而是 IE 开发人员所做的一个令人遗憾的设计决策。 如果 IE 觉得您试图访问的服务器位于本地计算机上，它就会忽略所有的代理设置，欺骗它的一个方法是在主机名后面加一个点，例如 http://localhost./WebGoat/attack。这将强迫 IE 使用我们配置的代理。

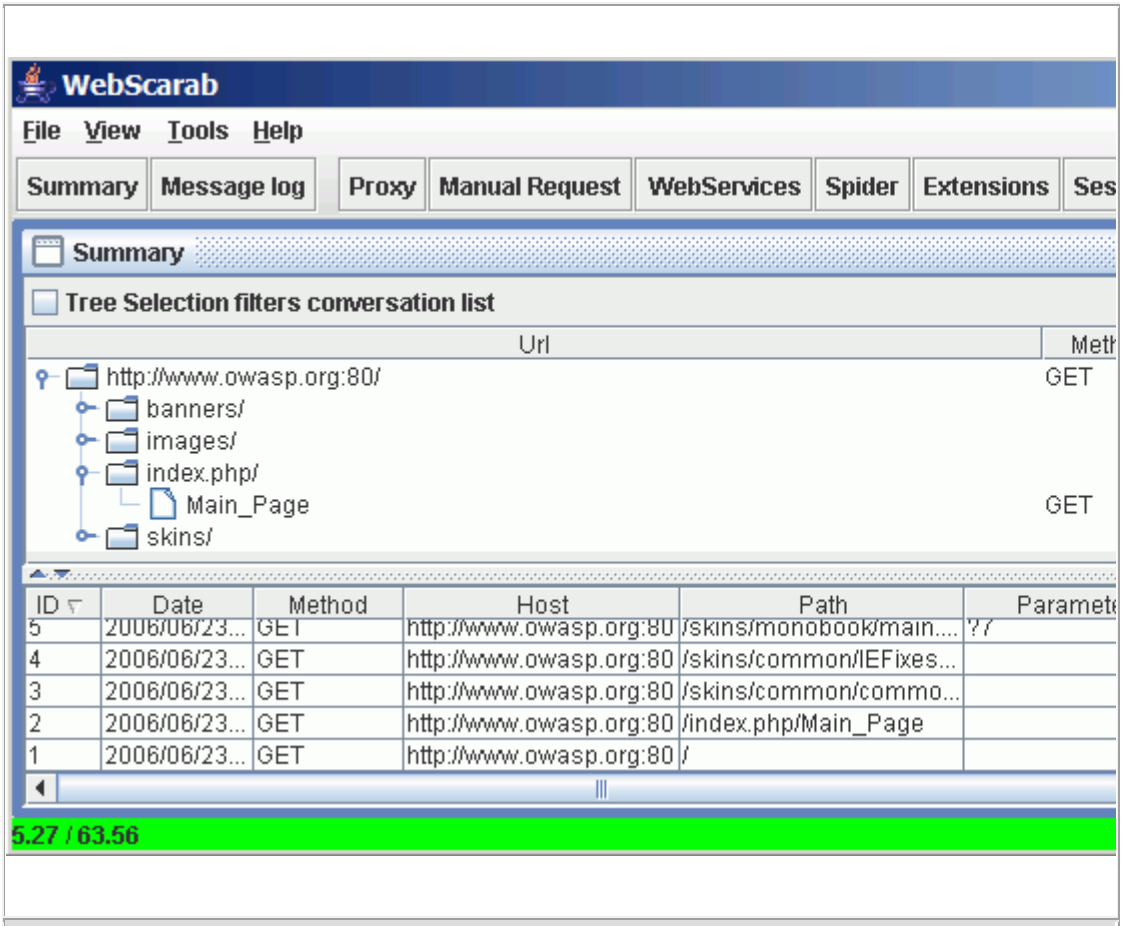

图 9

这里您可以看到一个 URL 树，用来表示站点布局，以及经过 WebScarab 的各个会话。要想查看一个特定会话的详细信息，您可以双击表中的一行，这时会弹出一个显示请求和响应的详细信息的窗口。您可以通过多种形式来查看请求和响应，这里显示的是一个 Parsed 视图，在这里，报头被分解成一个表，并且请求或者响应的内容按照 Content-Type 报头进行显示。您还可以选择 Raw 格式，这样的话，请求或者响应就会严格按照它们的原始形态进行展示。

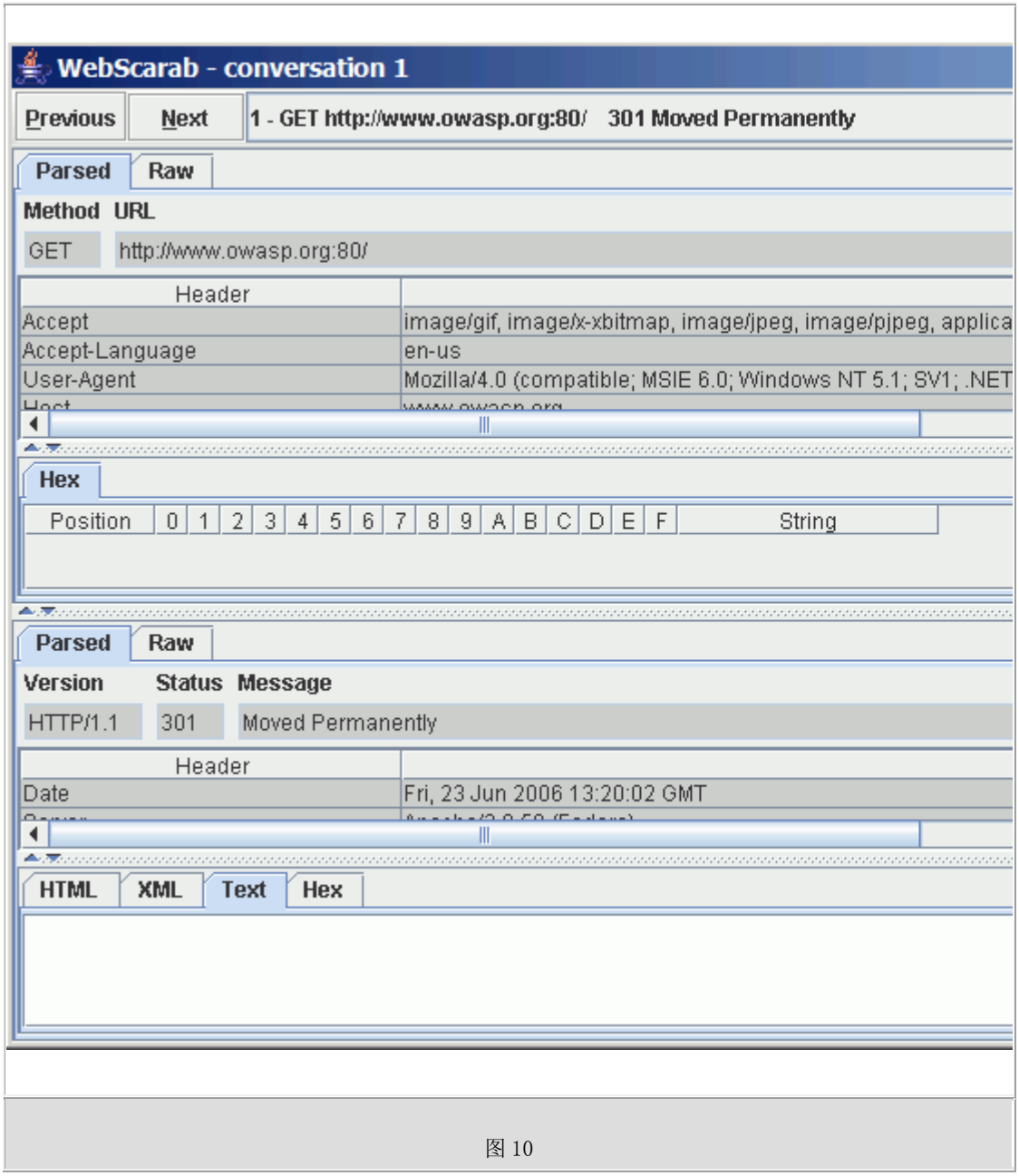

图 10

在会话窗口中，您可以通过“previous”按钮和“next”按钮从一个会话切换到另一个会话，也可通过下拉式组合框直接跳到特定的会话。

现在，您已经熟悉了 WebScarab 的基本界面，并且正确地配置了浏览器，接下来要做的就是拦截一些请求，并且在它们被发送给服务器之前对其进行修改。

我们可以启用代理插件的拦截功能，方法是通过工具栏上的“proxy”按钮。然后，选择“Manual Edit”选项卡。一旦选中“Intercept Requests”复选框，我们就可以选择希望拦截的请求方法（大部分情况下是 GET 或者 POST），甚至可以使用 Ctrl+单击的方式选择多个方法。 目前，我们只选择“GET”。

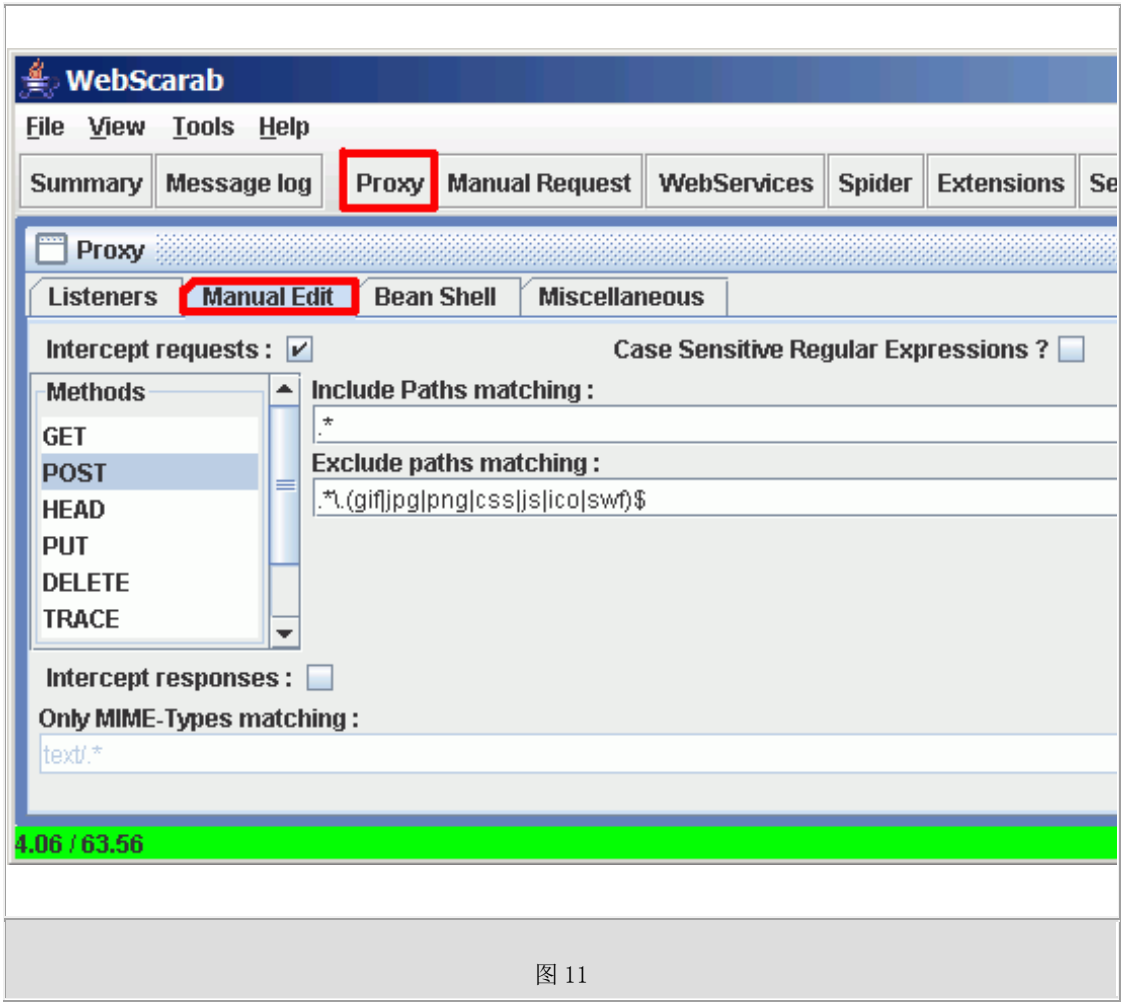

图 11

现在，返回到你的浏览器，并单击一个链接。这时，将会看到如下所示的一个窗口。最初，它只是在任务拦闪烁，只要点选它，就能正确显示了。

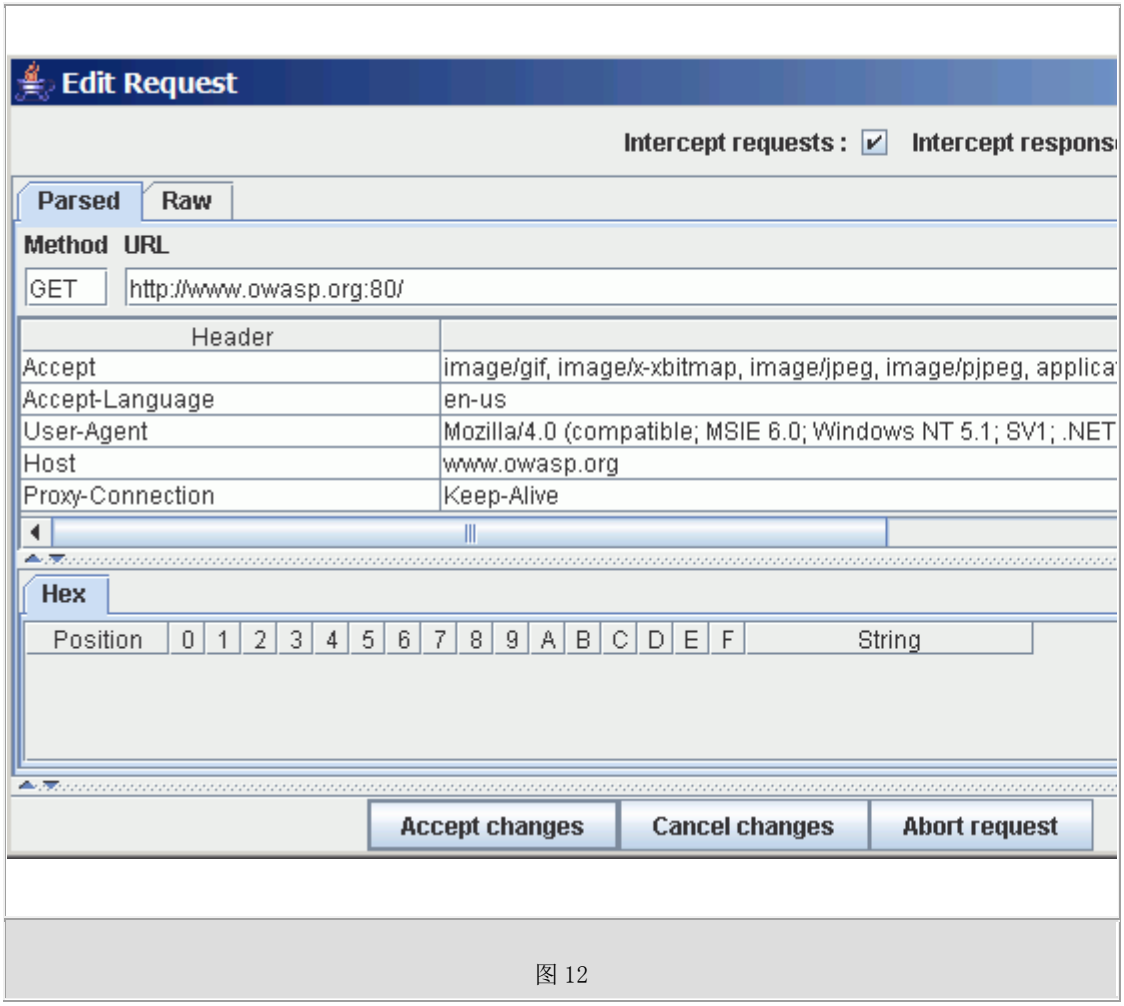

图 12

现在，我们就可以编辑选择的请求的任何部分了。需要注意的是，报头是以 URL 译码形式显示的，而输入的一切都会自动地 URL 编码。如果您不想这样的话，则可以使用 Raw 模式。在某些情况下，使用 Raw 模式可能是最简单的形式，尤其是您希望粘贴某些东西的时候。

作出修改后，单击“Accept changes”按钮就会将修改后的请求发送到服务器。如果您希望取消所在的修改，可以单击“Cancel changes”按钮，这样就会发送原始的请求。 您还可以单击“Abort request”按钮，如果您根本不想给服务器发送一个请求的话，这会向浏览器返一个错误。最后，如果打开了多个拦截窗口(也就是说浏览器同时使用了若干线程)，您可以使用“Cancel ALL intercepts”按钮来释放所有的请求。

WebScarab 将一直拦截所有的匹配我们指定的方法的请求，直到您在拦截会话窗口或者 Proxy 插件的“Manual Edit”选项卡取消选中“intercept requests”复选框为止。但是，您可能会奇怪：为什么 WebScarab 不会拦截对图像、样式表、javascript 等内容的请求。如果您返回到“Manual Edit”选项卡，将会看到一个标识为“Exclude paths matching :”的字段。 这个字段包含一个正则表达式，用于匹配请求的 URL，如果匹配，则该请求就不会被拦截。

如果您想改变页面某些行为的话，您还可以通过配置 WebScarab 使其拦截有关响应，举例来说，您可以禁用 javascript 验证，修改 SELECT 字段可选项，等等。

九、小结

WebGoat 是由著名的 OWASP 负责维护的一个漏洞百出的 J2EE Web 应用程序，这些漏洞并非程序中的 bug，而是故意设计用来讲授 Web 应用程序安全课程的。这个应用程序提供了一个逼真的教学环境，为用户完成课程提供了有关的线索。本文对该工具的安装和使用做了详细的介绍，希望本文能够对读者有所帮助。