实战安全工程师训练佳品之 WebGoat 入门篇

发布:木木 | 发布时间: 2011 年 9 月 21 日

在发表文章前，我先把下载地址分享给大家：

下载地址：http://code.google.com/p/webgoat/downloads/list

【51CTO.com 独家特稿】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。

对于每堂课，都对应于 WebGoat 应用程序中的一个实际的安全漏洞，为了能亲身实践如何利用这个漏洞，您首先需要具备该漏洞的有关知识，虽然 WebGoat 应用程序本身提供了有关的简介，但是很可能需要查找更多的资料才能搞定这个漏洞，所以，它对于激发安全测试人员和开发人员来的学习兴趣和提高安全知识的理解及动手能力方面，都是非常有帮助的。举个例子，在其中一个课程中，用户必须使用 SQL 注入来窃取（杜撰的）信用卡号。——51CTO 王文文：看到这个，由衷的感叹老外对网络安全教育的认真和开放的程度。

一、为什么要设计 WebGoat

在学习和实践 Web 应用程序安全知识时，我们所面临的一大难点是：到哪里去找可以练手的 web 应用程序呢？显然，明目张胆地扫描在线书店或者网络银行可不是个好主意，小心警察叔叔会找上门来。此外，安全专业人员经常需要测试某些安全工具，以检查它们的功能是否如厂商所鼓吹的那般，这时他们就需要一个具有确定漏洞的平台作为活靶子。但是，无论学习 web 测试，还是检查工具性能，都要求在一个安全、合法的环境下进行。即使你的意图是好的，但是在未经许可的情况下企图查找安全漏洞也是绝不允许的。这时，WebGoat 项目便应运而生了。

WebGoat 项目的主要目标很简单，就是为 Web 应用程序安全学习创建一个生动的交互式教学环境。将来，项目研究小组希望将 WebGoat 发展成为一个安全性基准测试程序平台和一个基于 Java 的蜜罐网站。如果您有兴趣，也可以查阅这个项目的路线图，其中能够找到一些可以立即参与的任务。——51CTO 王文文：是不是挺像一个黑客游戏？既能过瘾又能练习网络安全技术，最重要的是不用去危害真实的网站。

二、WebGoat 概要

WebGoat 是一个用来演示 Web 应用程序中的典型安全漏洞的应用程序，旨在在应用程序安全审计的上下文中系统、条理地讲解如何测试和利用这些安全漏洞。WebGoat 是用 Java 语言写成的，因此可以安装到所有带有 Java 虚拟机的平台之上。此外，它还分别为 Linux、OS X Tiger 和 Windows 系统提供了安装

程序。部署该程序后，用户就可以进入课程了，该程序会自动通过记分卡来跟踪用户的进展。当前提供的训练课程有 30 多个，其中包括：跨站点脚本攻击（XSS）、访问控制、线程安全、操作隐藏字段、操纵参数、弱会话 cookie、SQL 盲注、数字型 SQL 注入、字符串型 SQL 注入、web 服务、Open Authentication 失效危险的 HTML 注释.....等等！

我们希望通过 WebGoat 帮助测试人员掌握以下技能：

◆理解 web 应用程序中的各种高级交互过程

◆确定出有助于发动攻击的客户端可见数据

◆识别和理解能将应用程序暴露在攻击之下的数据和用户交互

◆对这些交互进行测试，并暴露出它们的漏洞

◆攻击应用程序以演示和利用服务器的弱点

对于 WebGoat 来说，它的安装过程就是下载和解压缩，然后就可以使用了。然而，一些用户可能更喜欢下载 war 文件。下面就所有的安装方式分别做详细的说明。

三、WebGoat 标准版安装方法

WebGoat 是一个平台无关的 Web 安全漏洞实验环境，该环境需要 Apache Tomcat 和 JAVA 开发环境的支持。它分别为 Microsoft Windows 和 UN*X 环境提供了相应的安装程序，下面我们将根据操作系统分别加以介绍。

安装 Java 和 Tomcat

需要注意，从版本 5 开始，这一步可以省略，因为它们自身带有 Java Development Kit 和 Tomcat 5.5。首先安装 Java，您可以从 http://java.sun.com/downloads/安装和部署合适的版本，最低版本要求为 1.4.1，然后安装 Tomcat，您可以从 http://tomcat.apache.org/download-55.cgi 安装和部署 Tomcat。

安装到 Windows 系统

1.将 WebGoat-OWASP_Standard-5.2.zip 解压至合适的目录中。

2.若要启动 Tomcat，切换至前面存放解压后的 WebGoat 的目录，然后双击 webgoat.bat 即可。

3.启动浏览器，在地址栏输入 http://localhost/WebGoat/attack 。注意，这个链接地址是区分大小写的，务必确保其中使用的是大写字母 W 和 G。

安装到 Linux 系统

1.将 WebGoat-OWASP_Standard-x.x.zip 解压至您的工作目录。

2.将 webgoat.sh 文件中的第 17、19 和 23 行中的“1.5”改为“1.6”。

3.因为最新版本运行在一个特权端口上，所以您需要使用下列命令来启/停 WebGoat Tomcat：

(1). 当作为 root 用户运行在 80 端口时，使用：

```
sudo sh webgoat.sh start80

sudo sh webgoat.sh stop
```

(2). 当运行在 8080 端口时，使用：

```
sh webgoat.sh start8080

sh webgoat.sh stop
```

安装至 OS X（Tiger 10.4+）系统

1.将 WebGoat-OWASP_Standard-x.x.zip 解压至您的工作目录。

2.将 webgoat.sh 文件中的第 10 行中的“1.5”改为“1.6”。

3.因为最新版本运行在一个特权端口上，所以您需要使用下列命令来启/停 WebGoat Tomcat：

(1).当作为 root 用户运行在 80 端口时，使用：

```
sudo sh webgoat.sh start80

sudo sh webgoat.sh stop
```

(2). 当运行在 8080 端口时，使用：

```
sh webgoat.sh start8080

sh webgoat.sh stop
```

安装至 FreeBSD 系统

1.使用下面的命令来安装 Tomcat 和 Java ：

```
cd /usr/ports/www/tomcat55

sudo make install
```

2. 安装 Java JDK 的时候，可能需要手工方式进行下载，届时系统会给出详细的提示。

3. 将 WebGoat-OWASP_Standard-x.x.zip 解压至您的工作目录。

4. 将 webgoat.sh 文件中的第 17、19 和 23 行中的“1.5”改为“1.6”。

5. 因为最新版本运行在一个特权端口上，所以您需要使用下列命令来启/停 WebGoat Tomcat：

(1).当作为 root 用户运行在 80 端口时，使用：

```
sudo sh webgoat.sh start80

sudo sh webgoat.sh stop
```

(2). 当运行在 8080 端口时，使用：

```
sh webgoat.sh start8080

sh webgoat.sh stop
```

运行方法

1. 启动浏览器，并在地址栏输入 http://localhost/WebGoat/attack，注意这里使用的大写的字母 W 和 G。

2. 登录时，用户帐号使用 guest，密码为 guest。

四、WebGoat Developer 版安装方法

WebGoat 5.2 Developer 版（位于 SourceForge 网站），注意：这个版本旨在提供一个 WebGoat 实验室环境。如果您想开发自己的教学课程，请与 Google code 站点上的基线同步。

这个开发人员版本除了包含标准版本外，还多了一个已配置的 Eclipse 环境。这个开发人员版本使用也会简单，下载、解压缩然后单击脚本即可。如果您仅仅希望研究有关课程的话，它用起来跟标准版本没有什么区别。然而，如果希望组建实验室，或者在课堂上使用 WebGoat 的话，可以使用 eclipse.bat 脚本来启动一个预配置的 WebGoat 环境。具体的使用说明，请参见自带的_HOW TO create the WebGoat workspace.txt_文件。

1.将 Eclipse-Workspace.zip 抽取至工作目录

2.双击 eclipse.bat 文件

3.在 Eclipse 右上角的包资源管理器中，右键单击 WebGoat 项目，并刷新

4.在 Eclipse 右上角的包资源管理器中，右键单击 Servers 项目，并刷新

5.在 Eclipse 底部的服务器视图中，右键单击 localhost 服务器，并启动它

6.在浏览器中导航至 http://localhost/WebGoat/attack。

7.源代码发生的任何变化，都会自动地引起编译操作，保存后会自动重新部署。

五、WebGoat War 文件版安装方法

这个版本将假定已经预先安装了 WebGoat Standard 版本，或者主机已经安装了 java 1.5（或更高版本）和 tomcat 5.5。如果您尚未安装 Standard 版本，那么就需要修改 tomcat/conf/tomcat-users.xml 文件来添加 WebGoat 用户，具体请参阅 http://code.google.com/p/webgoat/wiki/FAQ。

1.从 WebGoat Downloads 链接下载 WebGoat-OWASP_WAR-X.X.zip。

2.如果 Tomcat 正在运行的话，请先将其关闭——只需关闭 Tomcat 窗口即可。

3.将 war 文件拷贝至 WebGoat-X.X\tomcat\webapps\webgoat.war

4.删除现有的 WebGoat-X.X\tomcat\webapps\webgoat 目录

(1).这会导致所有的课程状态被丢失

(2).若要保存课程状态，请保留 webapps\webgoat\users 文件夹的副本

(3).重新启动 WebGoat 之后恢复这个用户目录

5.切换至 WebGoat-X.X 目录

6.双击 webgoat.bat 文件

这时 Tomcat 窗口就会启动。

7.在浏览器中导航至 http://localhost/WebGoat/attack。

六、所需其他工具

对于老道的应用程序安全审计人员来说，可用的辅助工具有很多。就我们这种类型的安全审计来说，最常用的工具就是本地代理和 web/应用程序爬虫。为了完成全套 WebGoat 课程，web 代理程序是必不可少的。

应用程序审计代理

一般的 web 代理通常都能接收、处理和转发客户和服务器之间的 HTTP 和 HTTPS 数据，这样就能让所有的 web 通信流量都流经某个点，以便通过高速缓存或者应用安全策略来监视利用率、提高性能，等等。

应用程序代理工具可用来拦截本地客户端的浏览器和服务器端之间所有的 HTTP 和 HTTPS 通信，它实际上充当了一个可以监视、检查和(最重要地)修改所有的交互的中间人角色。

通过这种工具，审计人员可以准确确定出在客户和服务器之间传递的到底是什么样的数据。此外，它们还可以对这些数据进行分析和修改，从而测试对应用程序的影响。

在 WebGoat 的许多课程中，应用程序审计代理或者具备同等功能的软件都是必不可少的。下列是我们推荐的工具：

◆WebScarab：WebScarab Project

◆BurpProxy- http://portswigger.net/

◆ParosProxy - http://parosproxy.org

应用程序爬虫

所谓爬行一个站点，实际上就是识别和访问网站应用程序内所有预定的页面和链接，并建立本地副本；当然建立副本这一点通常是可选的。然后，我们就可以分析爬行结果，得到应用程序内目标脚本、表单、页面和字段等组成的明细表供后面的测试之用。镜像下来的内容也可以用来分析有关信息，这样做要比人工或者在线分析要快得多了。

下列是我们推荐的工具：

◆WebScarab：WebScarab Project

◆BurpSpider - http://portswigger.net

◆ParosProxy - http://parosproxy.org

七、WebGoat 操作指南

开始使用 WebGoat 之前，必须首先启动 Tomcat，这可以通过 Tomcat 的 bin 目录中的脚本/批处理程序 startup 来完成。此外，要想正常使用 WebGoat，它必须具备作为服务器运行所需的权限，并允许一些不常见的 web 行为。当主机运行 WebGoat 时，WebGoat 的安全漏洞会牵连到主机，从而使主机很容易遭到攻击。如果机器连接到了互联网，那么就应该将其断开。运行的个人防火墙可能会阻止 WebGoat 的正常使用。所以，运行 WebGoat 时需要禁用所有的个人防火墙。

我们可以通过浏览器浏览 localhost 的 80 端口访问 Tomcat 服务器，如 http://127.0.0.1。

WebGoat 位于 WebGoat 目录，其中的课程包含在 http://127.0.0.1/WebGoat/attack 中。

WebGoat 应用程序施行基于角色的安全机制。登录对话请求会要求输入身份凭证，登录时，可以将 guest 作为用户标识和密码使用。

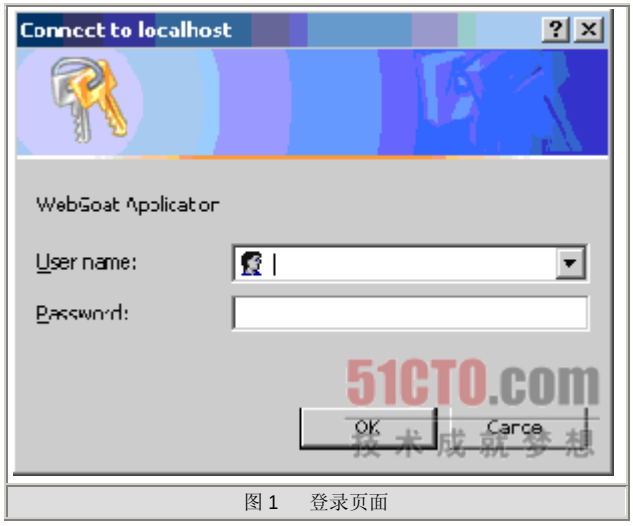

图 1　登录页面

成功登录之后，Tomcat 服务器将显示 WebGoat 的欢迎页面。

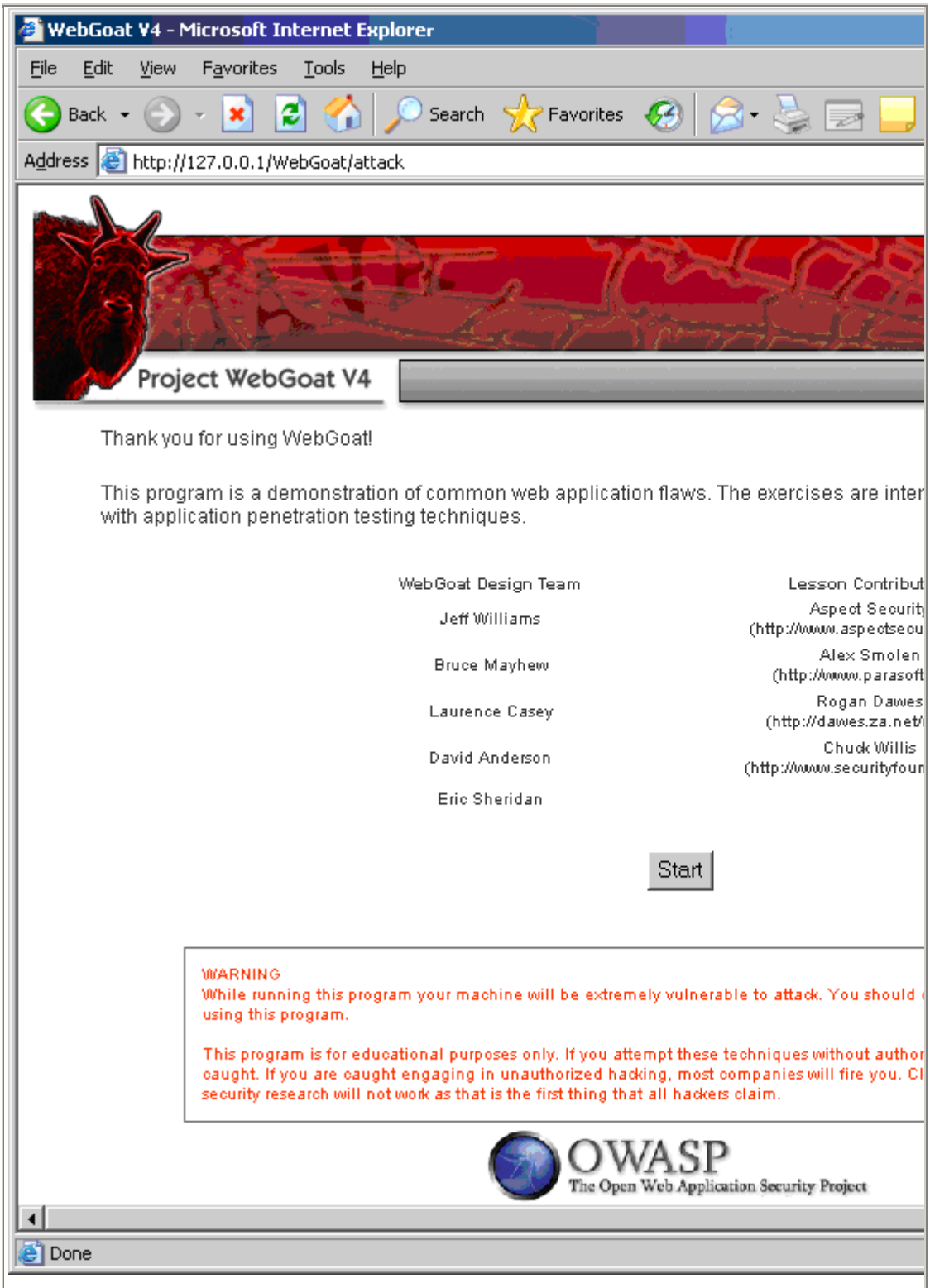

图 2 欢迎页面

下面介绍 WebGoat 的基本操作。我们知道，无论应用程序安全审计的哪个阶段，都需要对目标的运作机制有深入的了解。这通常包括：

◆考察客户端内容，诸如 HTML 和脚本

◆分析客户和服务器之间的通讯

◆检查 cookie 及其他本地数据

浏览器已经使得查看 HTML 源代码变得非常轻松，而 WebGoat 又增加了多种操作，包括显示参数、显示 HTML、显示 Cookies 和显示 Java 等。

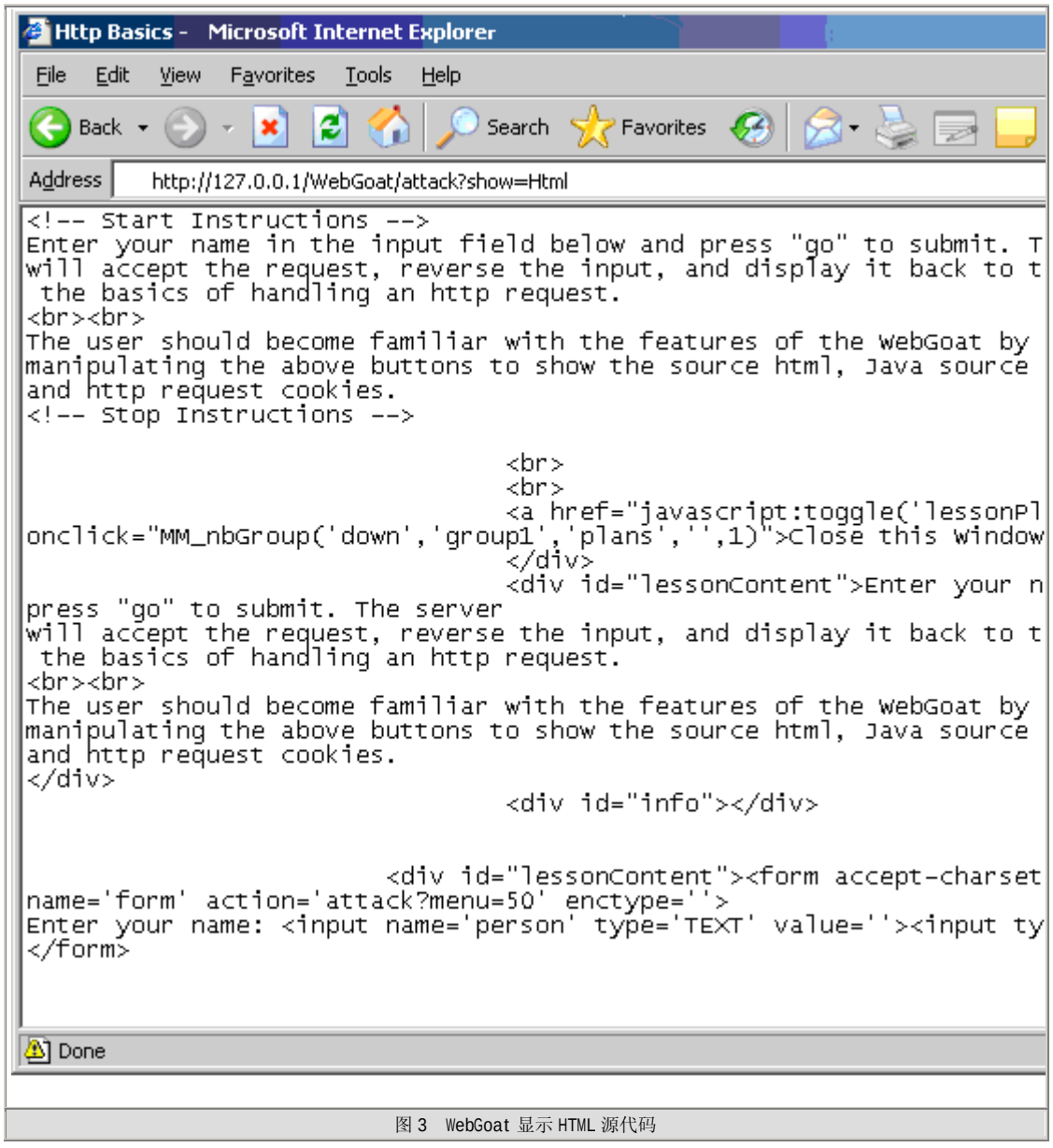

图 3　WebGoat 显示 HTML 源代码

在普通环境之下，浏览器只提供查看 HTML 源代码的功能部件，对于微软公司的 Internet Explorer 浏览器，可以通过“查看”菜单下的“源文件”选项来查看 HTML 源代码。对于 Firefox 浏览器来说，查看页面源码的功能同样位于“查看”菜单下的“页面源代码”下。 WebGoat 的显示 HTML 功能仅仅展示当前课程相应的 HTML 代码，而不包括侧栏和上栏对应的 HTML 代码。

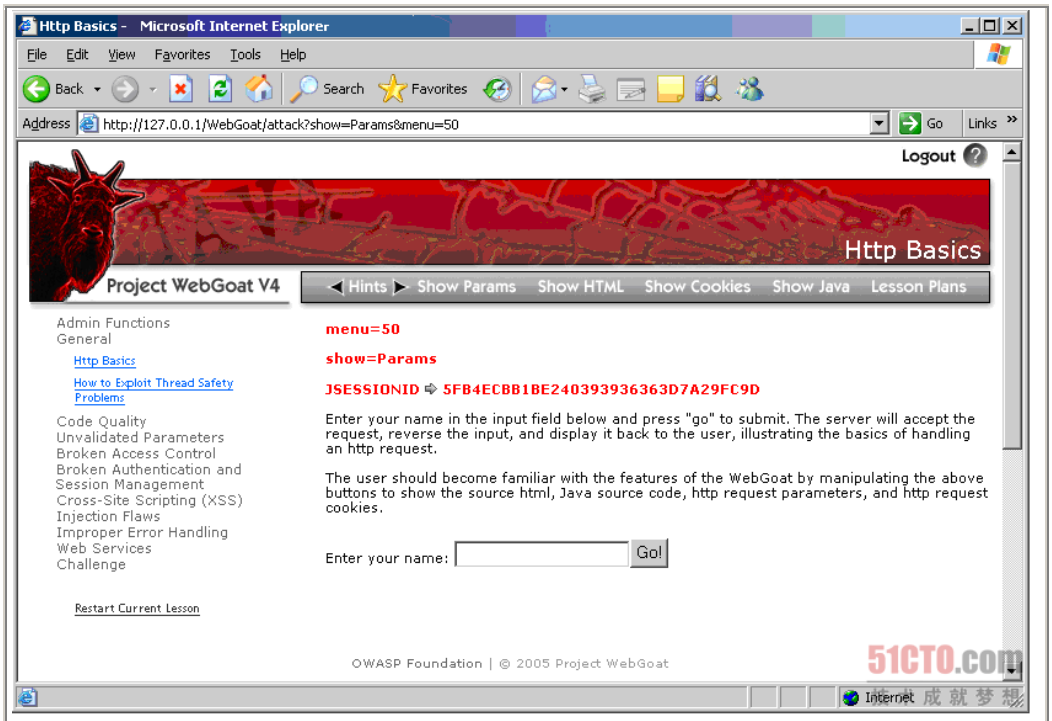

图 4　显示 HTML 源代码

这里，参数和 cookie 显示为红色。

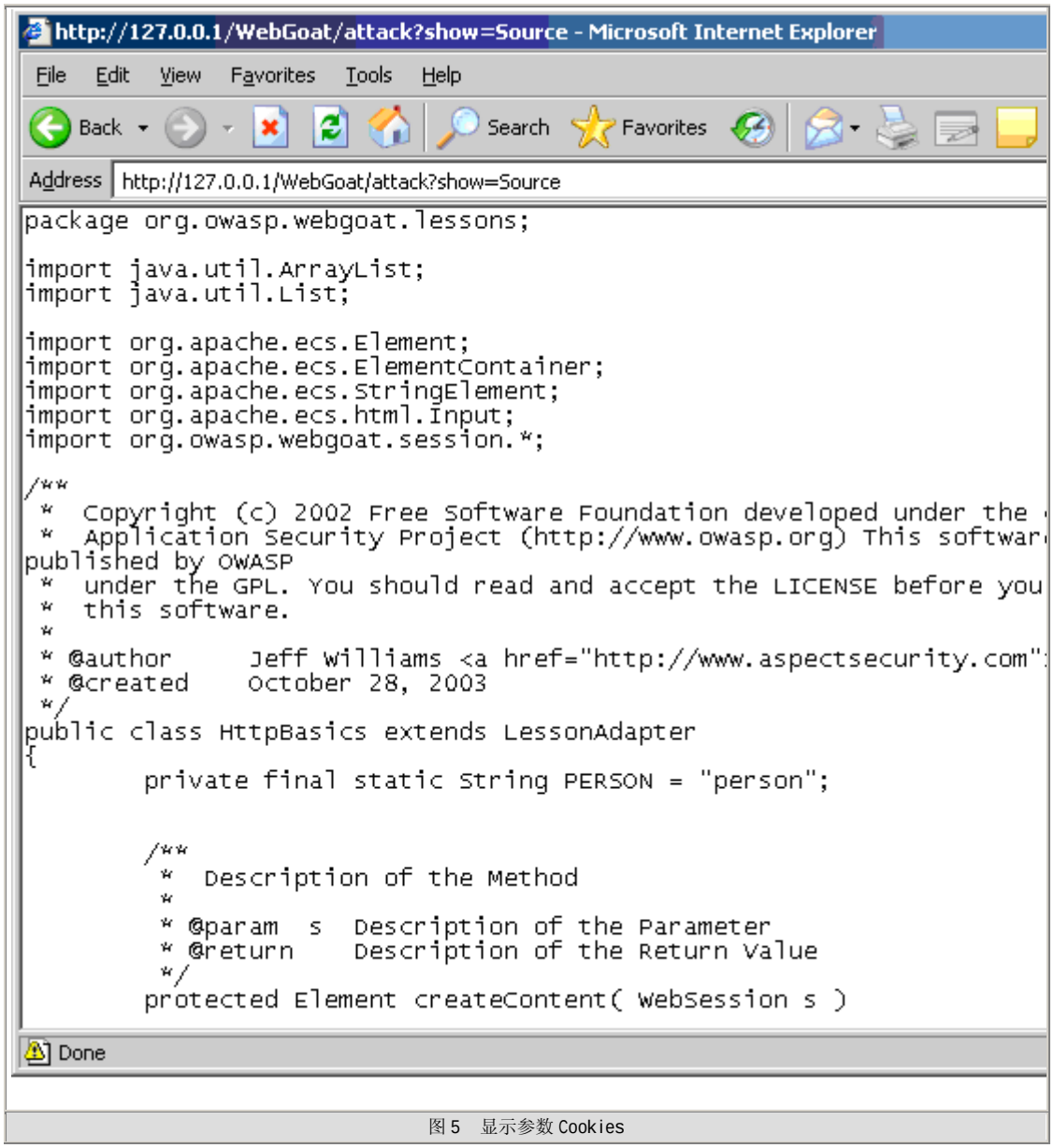

图 5　显示参数 Cookies

这里显示 Java 操作会弹出一个包含源代码的新窗口。

下面介绍如何使用代理，要想充分挖掘 WebGoat 的各种功能，我们需要借助以审计人员常用的应用程序审计代理程序。这有助于进行更深入的分析，并能修改客户端-服务器的交互和传输过程中的数据。由于不同的工具，其使用和配置方法也不相同，但基本概念是一致的：

◆应用程序审计代理必须位于客户端的浏览器和远程服务器之间。

◆它应该允许显示和修改传输中的所有 HTTP 数据。

该工具通常会直接插入浏览器，或者在另一个本机端口进行侦听。当代理程序直接插入浏览器的时候，需要在浏览器中键入一个特殊的 URL。当该工具侦听端口时，则需要对浏览器进行相应的配置，方可正常使用该工具。在微软公司的 Internet Explorer 中，可以通过工具菜单完成配置工作，如下所示：

1. 选择工具菜单中的“Internet 选项”菜单项。

2．选择“连接”选项卡。

3．单击选项卡下方的“局域网设置...”按钮。

4．在局域网设置对话框中，选中为 LAN 使用代理服务器的复选框。

5．不选“对本地地址不使用代理服务器”框。

6．输入代理工具将要侦听的地址和端口。对于 WebScarab 而言，其默认侦听端口是 8008。

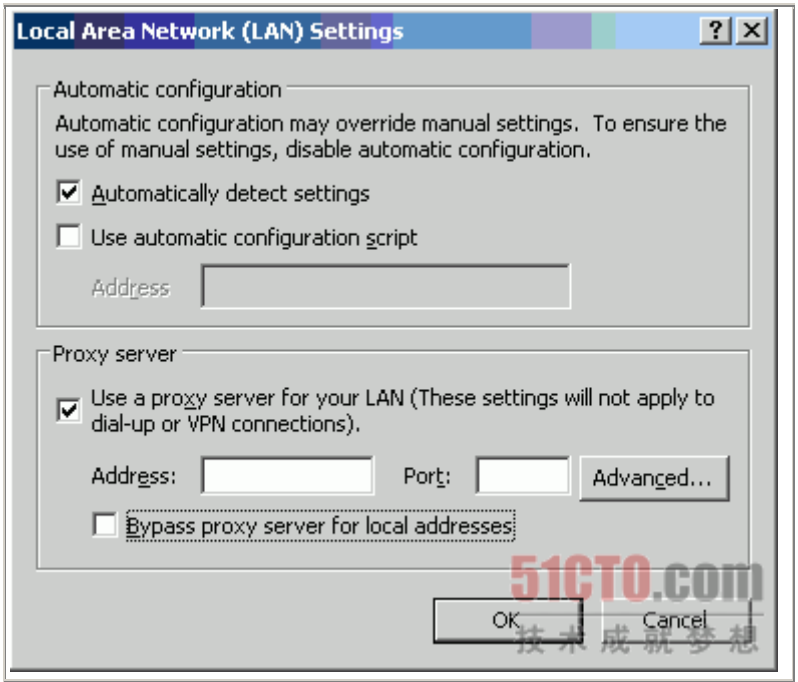

图 6　局域网设置

现在，每当从客户端的浏览器接收或者发送数据时，我们都能通过拦截、分析和修改这些 HTTP 请求来测试应用程序，从而安全性缺陷。审计人员可以借助这类代理获得多种能力，包括：

◆不管 GET/POST 参数的隐藏状态如何，都可以对其进行修改。

◆无论是持久性还是非持久性的 Cookie，当它们进入和离开浏览器时，我们都可以对其进行修改。

◆因为参数可以在发送给服务器之前进行即时修改，所以我们可以绕过所有的客户端数据验证。

◆能够暴露高速缓存的数据，以便于分析。

◆能够暴露出 Server:及其他报头，这对于调查远程 web 服务器类型和所用的应用程序-服务器技术非常有利。

八、WebScarab 入门指南

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。

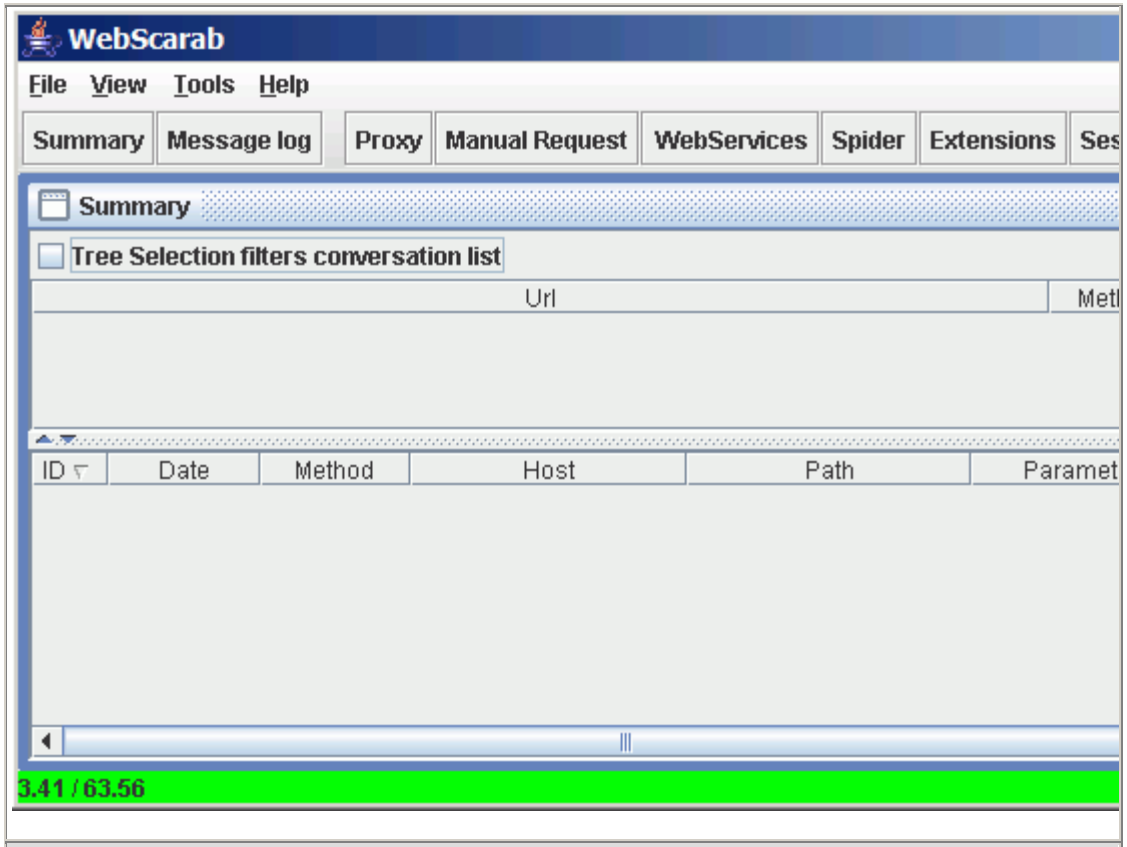

图 7

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。

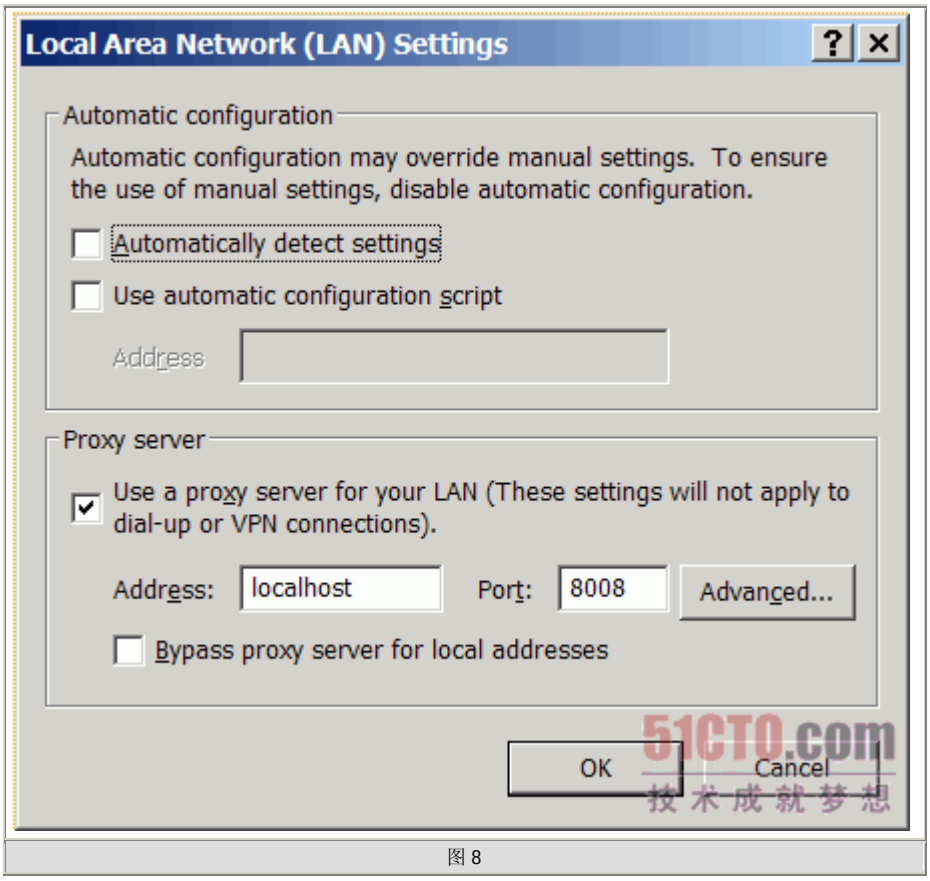

图 8

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。

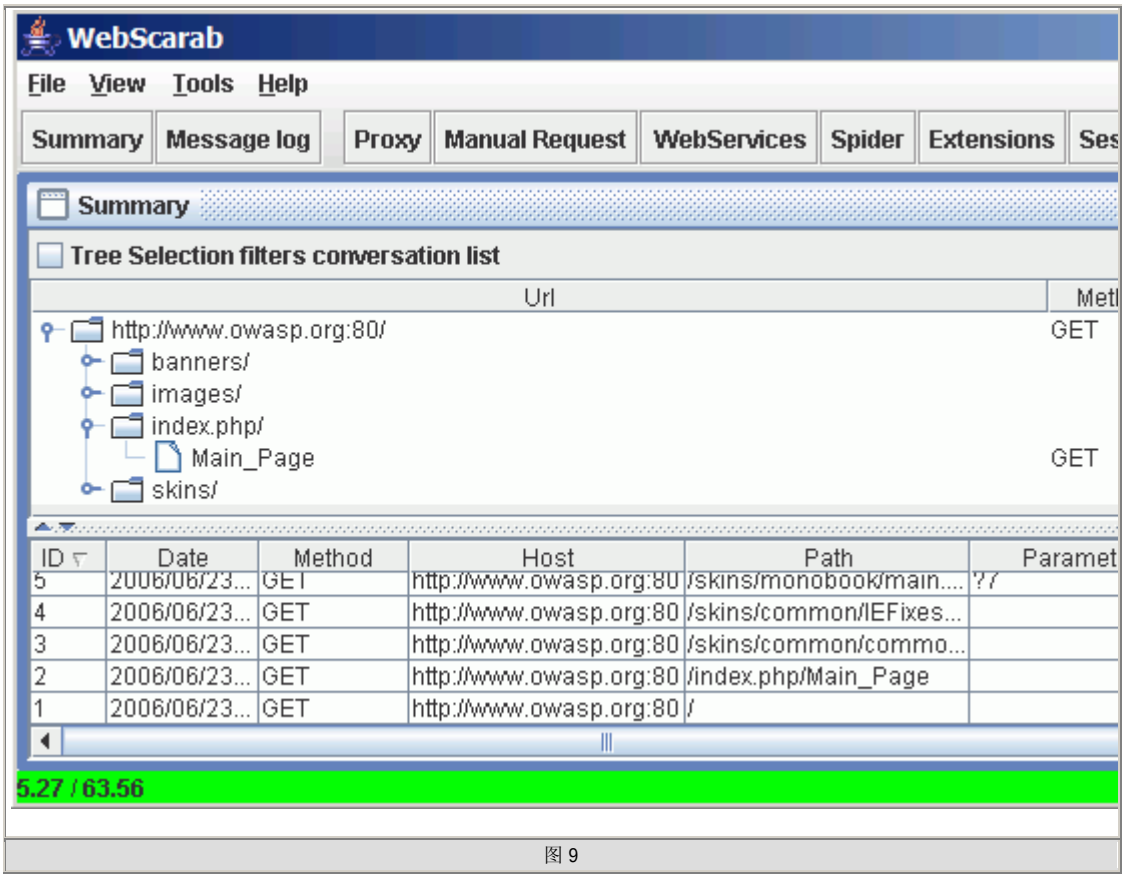

图 9

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。

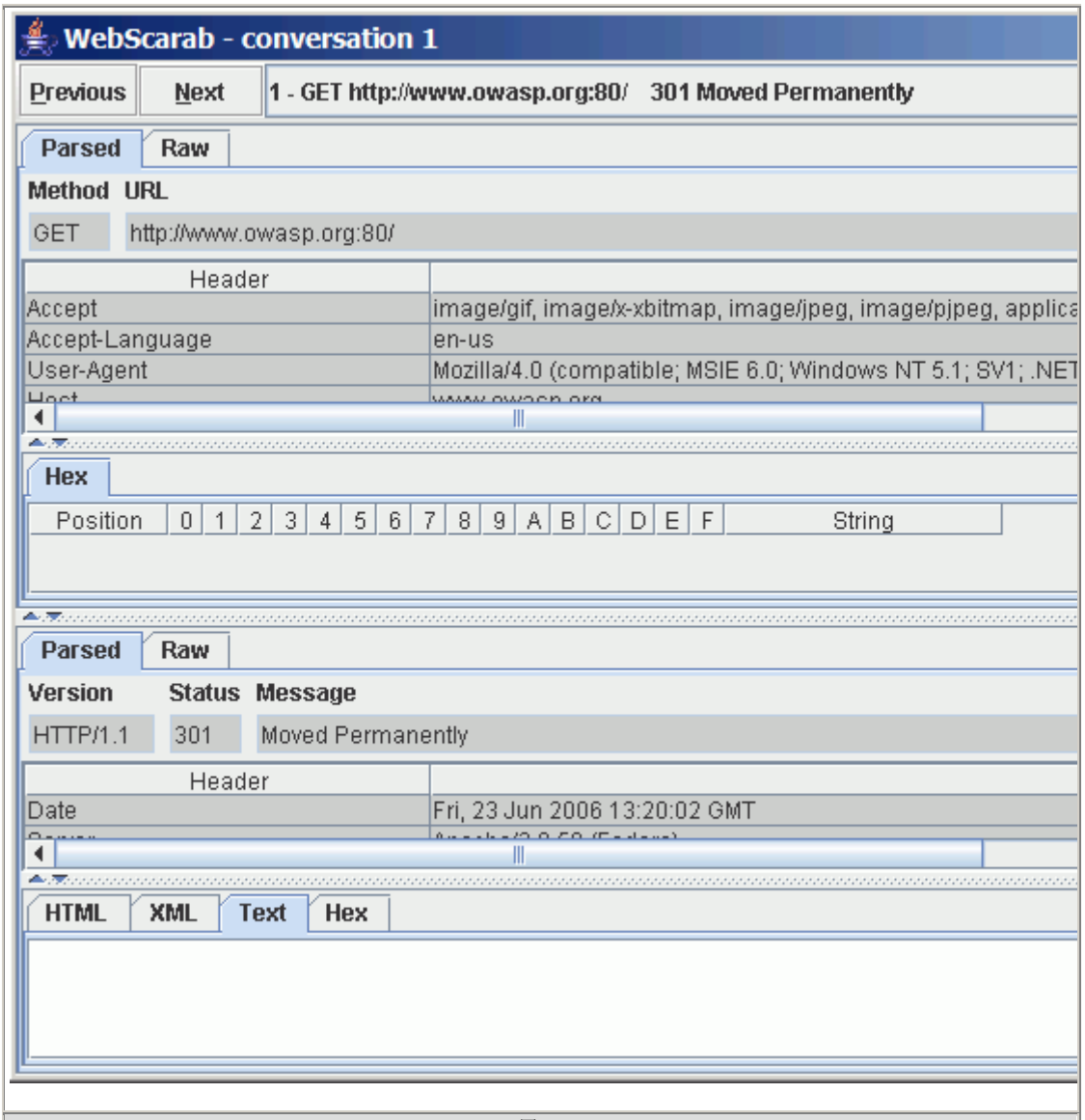

图 10

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”。

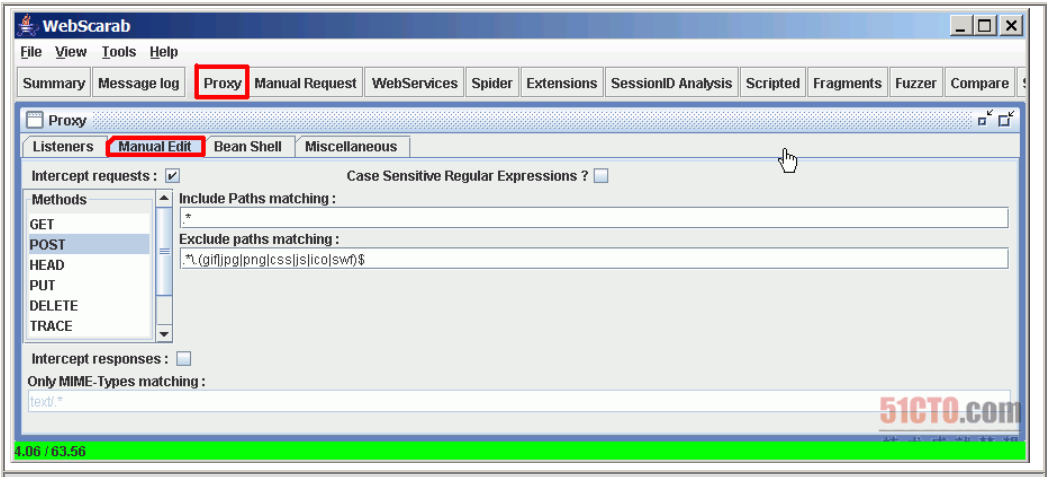

图 11

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。

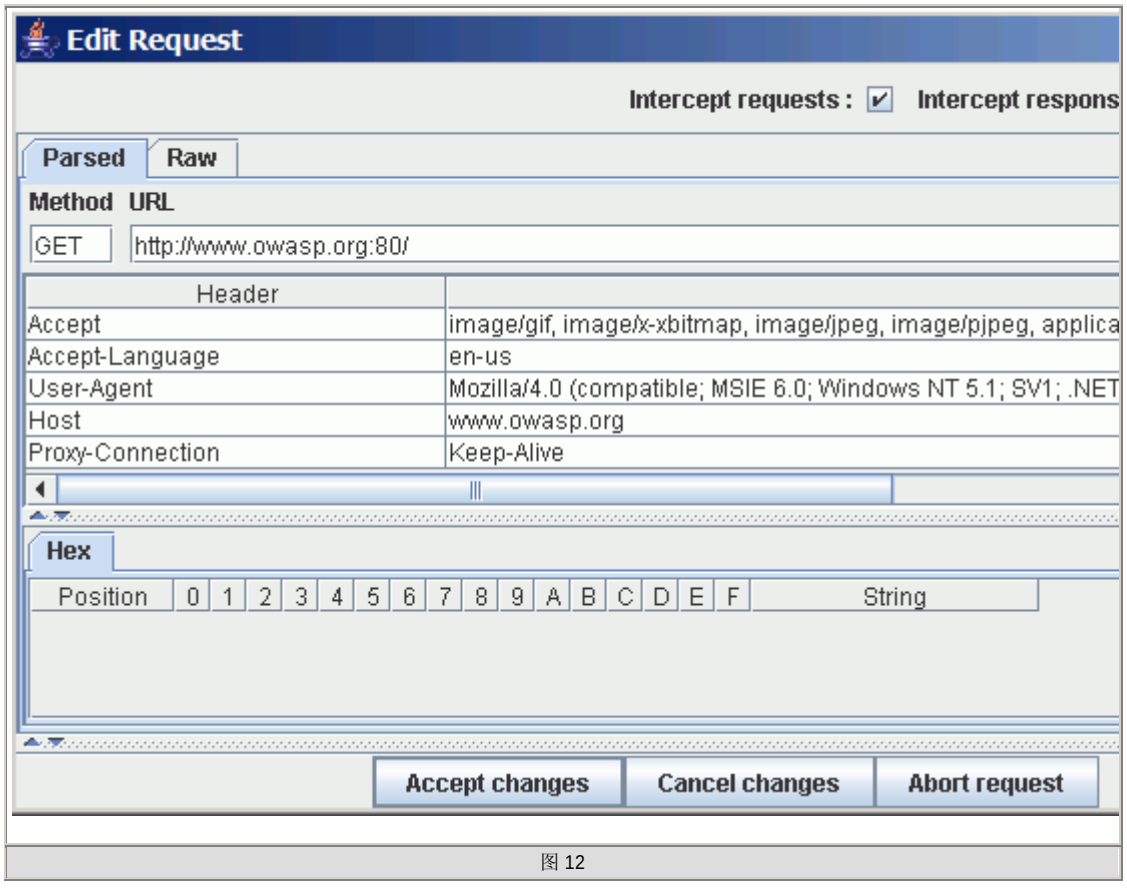

图 12

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。