[**Fiddler 教程**](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html)

Fiddler是最强大最好用的Web调试工具之一，它能记录所有客户端和服务器的http和https请求，允许你监视，设置断点，甚至修改输入输出数据. 使用Fiddler无论对开发还是测试来说，都有很大的帮助。

阅读目录

1. [Fiddler的基本介绍](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#introduce)
2. [Fiddler的工作原理](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#yuanli)
3. [同类的其它工具](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#qita)
4. [Fiddler如何捕获Firefox的会话](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#firefox)
5. [Firefox 中使用Fiddler插件](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#fiddlerhook)
6. [Fiddler如何捕获HTTPS会话](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#https)
7. [Fiddler的基本界面](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#basic)
8. [Fiddler的统计视图](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#summaryview)
9. [QuickExec命令行的使用](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#quickexec)
10. [Fiddler中设置断点修改Request](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#request)
11. [Fiddler中设置断点修改Response](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#response)
12. [Fiddler中创建AutoResponder规则](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#autoresponder)
13. [Fiddler中如何过滤会话](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#filter)
14. [Fiddler中会话比较功能](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#compare)
15. [Fiddler中提供的编码小工具](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#encodingtool)
16. [Fiddler中查询会话](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#findsession)
17. [Fiddler中保存会话](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#savesession)
18. [Fiddler的script系统](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#script)
19. [如何在VS调试网站的时候使用Fiddler](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#vs)
20. [Response 是乱码的](http://www.cnblogs.com/TankXiao/archive/2012/02/06/2337728.html#decodeResponse)

**Fiddler的基本介绍**

Fiddler的官方网站:  [www.fiddler2.com](http://www.fiddler2.com/)

Fiddler官方网站提供了大量的帮助文档和视频教程， 这是学习Fiddler的最好资料。

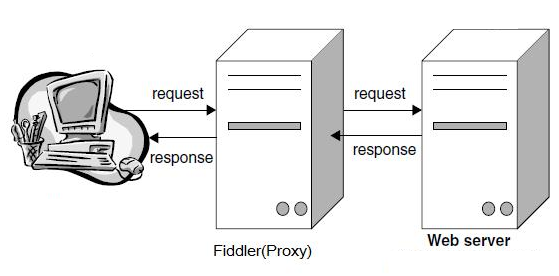
Fiddler是最强大最好用的Web调试工具之一，它能记录所有客户端和服务器的http和https请求，允许你监视，设置断点，甚至修改输入输出数据，Fiddler包含了一个强大的基于事件脚本的子系统，并且能使用.net语言进行扩展

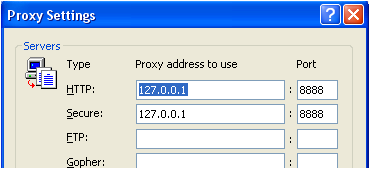
你对HTTP 协议越了解， 你就能越掌握Fiddler的使用方法. 你越使用Fiddler,就越能帮助你了解HTTP协议.

Fiddler无论对开发人员或者测试人员来说，都是非常有用的工具

**Fiddler的工作原理**

Fiddler 是以代理web服务器的形式工作的,它使用代理地址:127.0.0.1, 端口:8888. 当Fiddler会自动设置代理， 退出的时候它会自动注销代理，这样就不会影响别的程序。不过如果Fiddler非正常退出，这时候因为Fiddler没有自动注销，会造成网页无法访问。解决的办法是重新启动下Fiddler.





**同类的其它工具**

同类的工具有: httpwatch, firebug, wireshark

**Fiddler 如何捕获Firefox的会话**

能支持HTTP代理的任意程序的数据包都能被Fiddler嗅探到，Fiddler的运行机制其实就是本机上监听8888端口的HTTP代理。 Fiddler2启动的时候默认IE的代理设为了127.0.0.1:8888，而其他浏览器是需要手动设置的，所以将Firefox的代理改为127.0.0.1:8888就可以监听数据了。

Firefox 上通过如下步骤设置代理

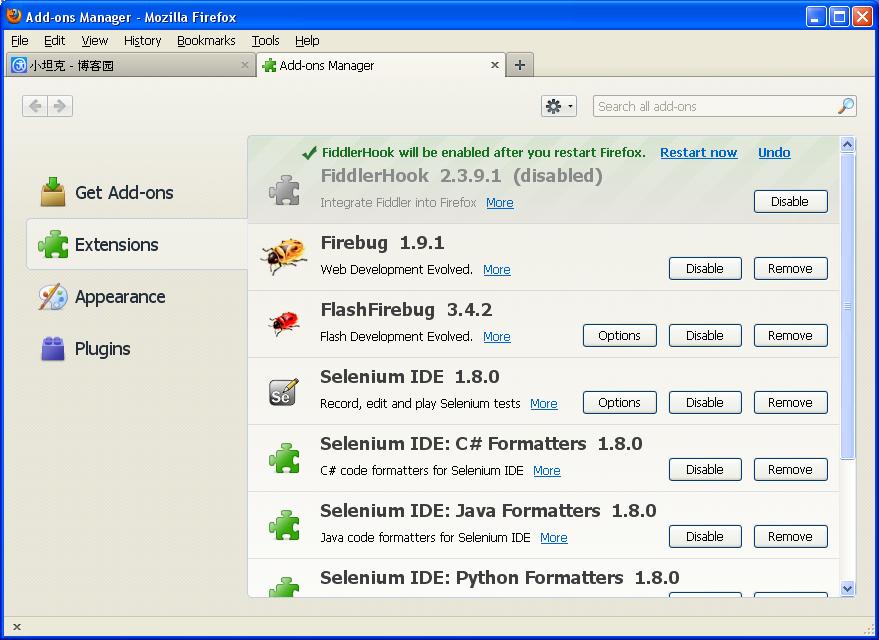
点击: Tools -> Options,  在Options 对话框上点击Advanced tab - > network tab -> setting.

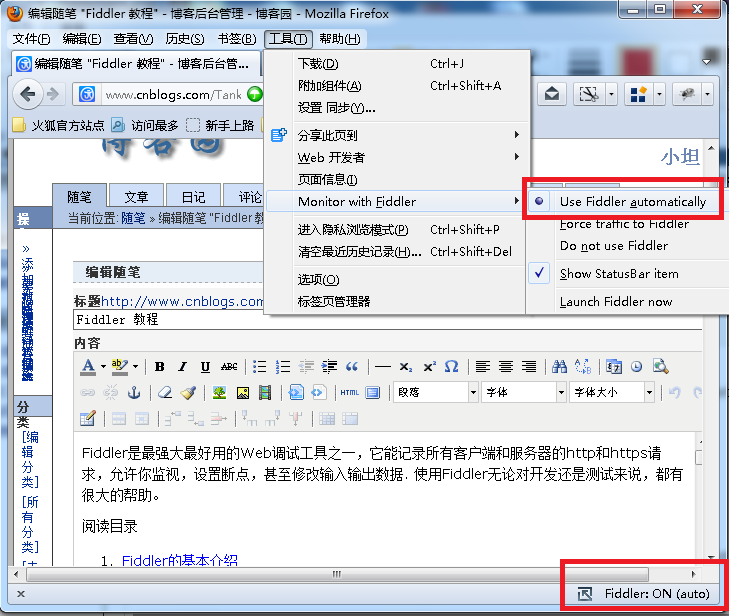
**Firefox 中安装Fiddler插件**

修改Firefox 中的代理比较麻烦， 不用fiddler的时候还要去掉代理。 麻烦

推荐你在firefox中使用fiddler hook 插件, 这样你非常方便的使用Fiddler获取firefox中的request 和response

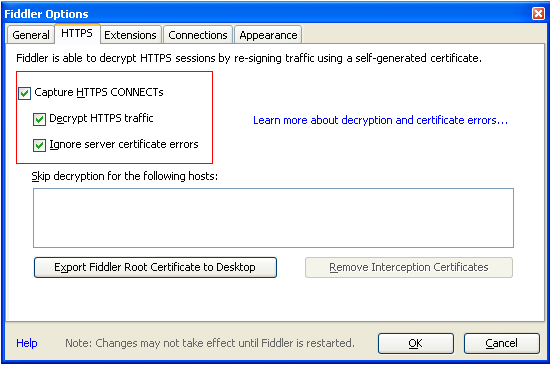
当你安装fiddler后， 就已经装好了Fiddler hook插件， 你需要到firefox中去启用这个插件  
打开firefox   tools->Add ons -> Extensions 启动 FiddlerHook



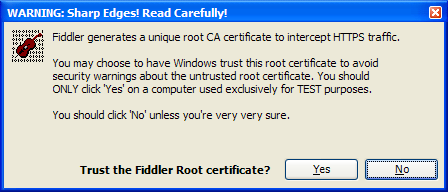


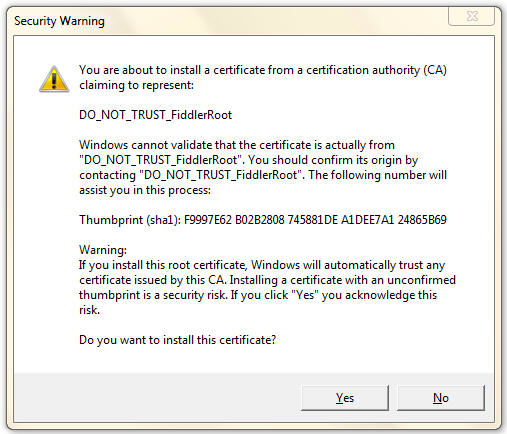
**Fiddler如何捕获HTTPS会话**

默认下，Fiddler不会捕获HTTPS会话，需要你设置下, 打开Fiddler  Tool->Fiddler Options->HTTPS tab



 选中checkbox, 弹出如下的对话框，点击"YES"

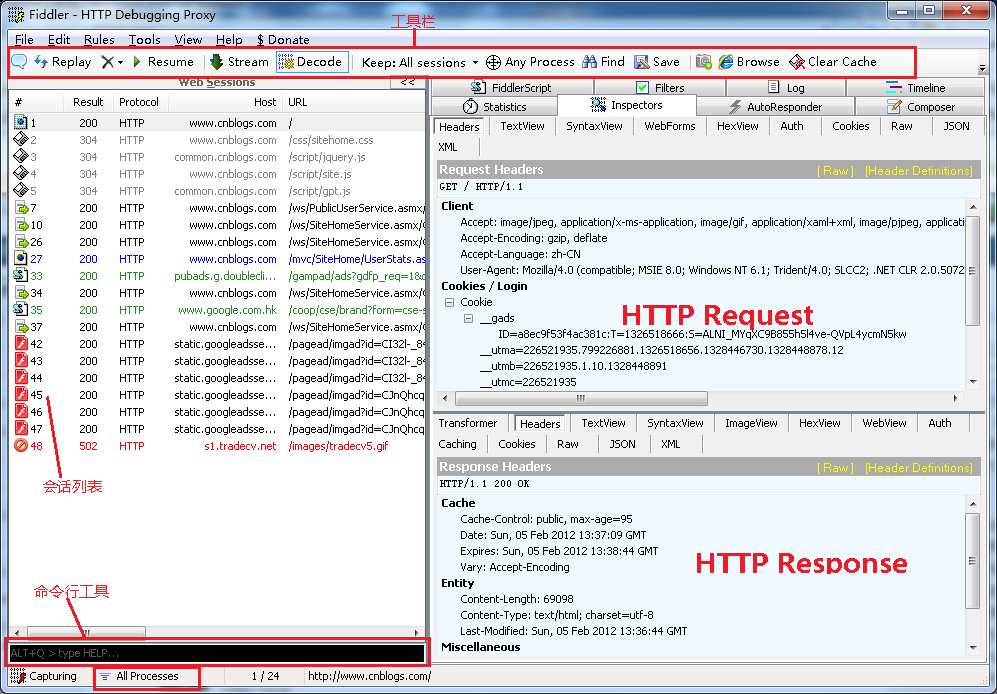




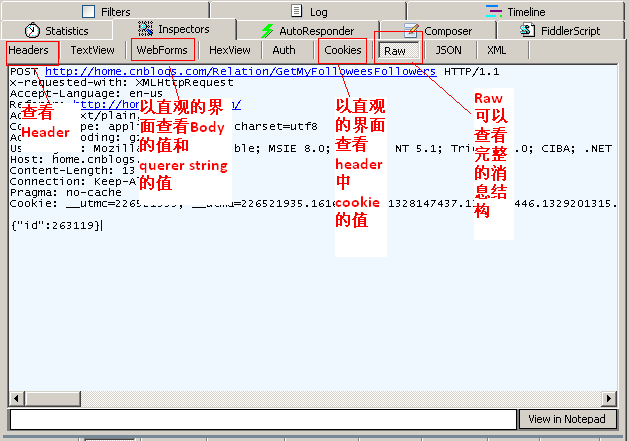
点击"Yes" 后，就设置好了

**Fiddler的基本界面**

 看看Fiddler的基本界面



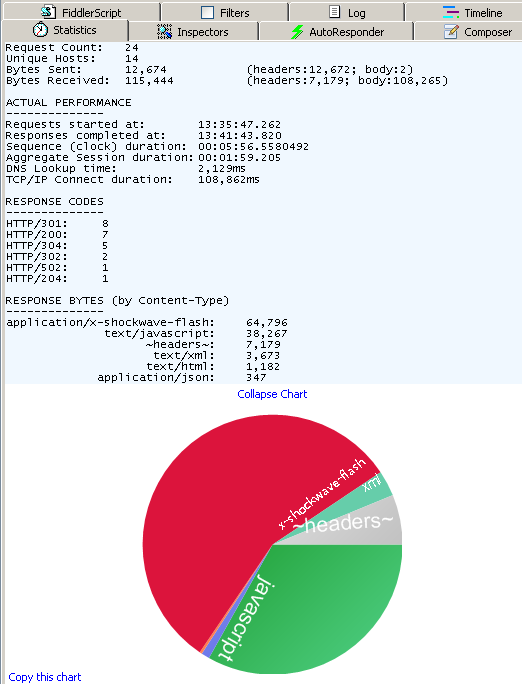
Inspectors tab下有很多查看Request或者Response的消息。 其中Raw Tab可以查看完整的消息，Headers tab 只查看消息中的header. 如下图



**Fiddler的HTTP统计视图**

通过陈列出所有的HTTP通信量，Fiddler可以很容易的向您展示哪些文件生成了您当前请求的页面。使用Statistics页签，用户可以通过选择多个会话来得来这几个会话的总的信息统计，比如多个请求和传输的字节数。

选择第一个请求和最后一个请求，可获得整个页面加载所消耗的总体时间。从条形图表中还可以分别出哪些请求耗时最多，从而对页面的访问进行访问速度优化



**QuickExec命令行的使用**

Fiddler的左下角有一个命令行工具叫做QuickExec,允许你直接输入命令。

常见得命令有

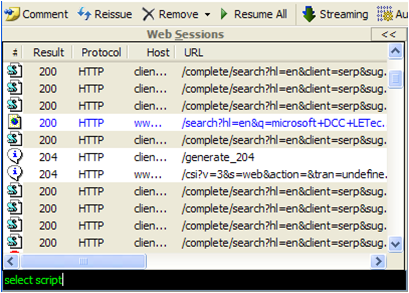
help  打开官方的使用页面介绍，所有的命令都会列出来

cls    清屏  (Ctrl+x 也可以清屏)

select  选择会话的命令

?.png  用来选择png后缀的图片

bpu  截获request



**Fiddler中设置断点修改Request**

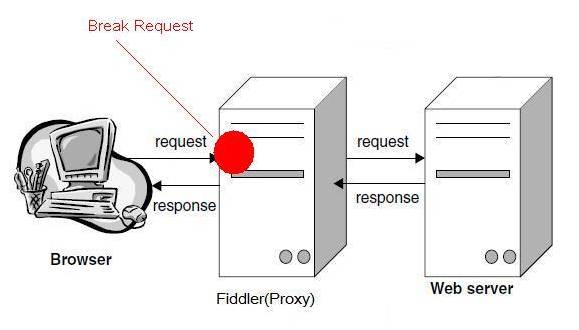
[[作者：小坦克]](http://www.cnblogs.com/TankXiao/admin/%5b%E4%BD%9C%E8%80%85%EF%BC%9A%E5%B0%8F%E5%9D%A6%E5%85%8B%5d%20http:/www.cnblogs.com/TankXiao/)  Fiddler最强大的功能莫过于设置断点了，设置好断点后，你可以修改httpRequest 的任何信息包括host, cookie或者表单中的数据。设置断点有两种方法

第一种：打开Fiddler 点击Rules-> Automatic Breakpoint  ->Before Requests(这种方法会中断所有的会话)

如何消除命令呢？  点击Rules-> Automatic Breakpoint  ->Disabled

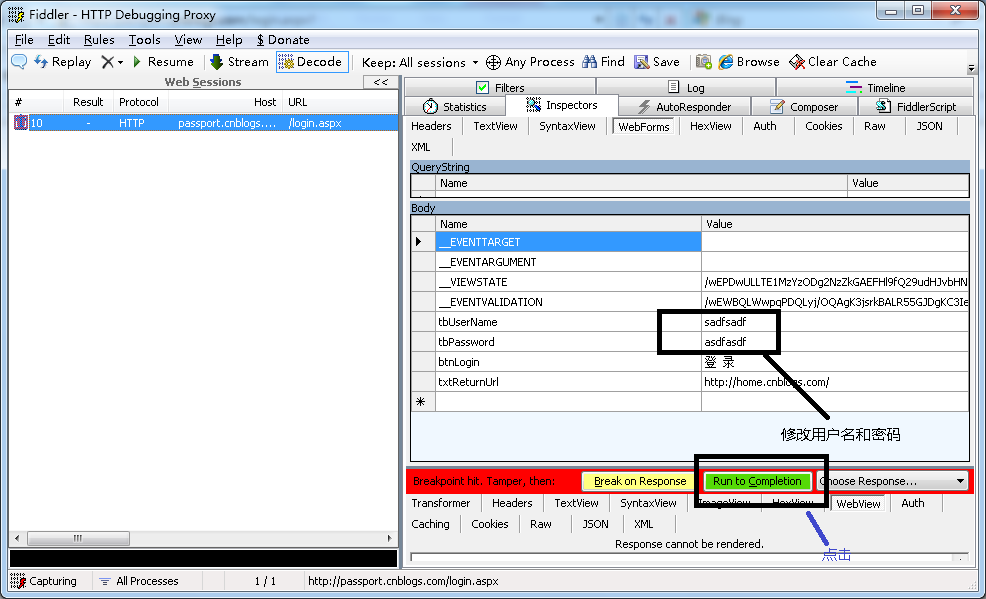
第二种:  在命令行中输入命令:  bpu www.baidu.com   (这种方法只会中断www.baidu.com)

如何消除命令呢？  在命令行中输入命令 bpu



看个实例，模拟博客园的登录， 在IE中打开博客园的登录页面，输入错误的用户名和密码，用Fiddler中断会话，修改成正确的用户名密码。这样就能成功登录

1. 用IE 打开博客园的登录界面  http://passport.cnblogs.com/login.aspx  
2. 打开Fiddler,  在命令行中输入bpu http://passport.cnblogs.com/login.aspx  
3. 输入错误的用户名和密码 点击登录  
4. Fiddler 能中断这次会话，选择被中断的会话，点击Inspectors tab下的WebForms tab 修改用户名密码，然后点击Run to Completion 如下图所示。  
5. 结果是正确地登录了博客园



**Fiddler中设置断点修改Response**

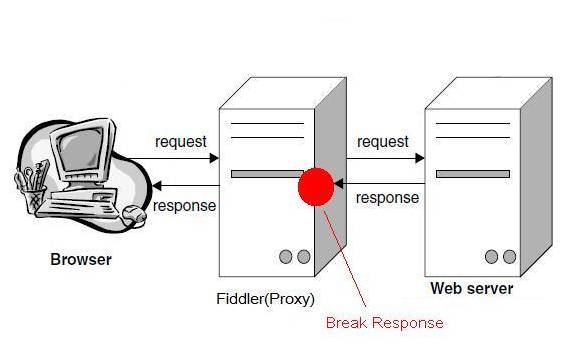
当然Fiddler中也能修改Response

第一种：打开Fiddler 点击Rules-> Automatic Breakpoint  ->After Response  (这种方法会中断所有的会话)

如何消除命令呢？  点击Rules-> Automatic Breakpoint  ->Disabled

第二种:  在命令行中输入命令:  bpafter www.baidu.com   (这种方法只会中断www.baidu.com)

如何消除命令呢？  在命令行中输入命令 bpafter,



具体用法和上节差不多，就不多说了。