****BurpSuite有以下这些功能：****

截获代理– 让你审查修改浏览器和目标应用间的流量。

爬虫 – 抓取内容和功能

Web应用扫描器\* –自动化检测多种类型的漏洞

Intruder – 提供强大的定制化攻击发掘漏洞

Repeater – 篡改并且重发请求

Sequencer –测试token的随机性

能够保存工作进度，以后再恢复

插件\*– 你可以自己写插件或者使用写好的插件，插件可以执行复杂的，高度定制化的任务

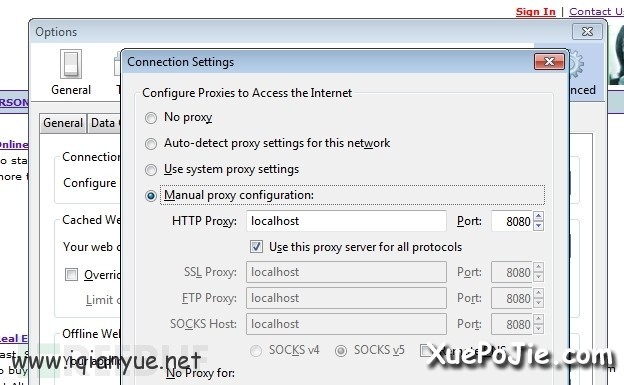
\*表示需要Burp Suite Pro授权。

****Intercepting Proxy（截取代理）****

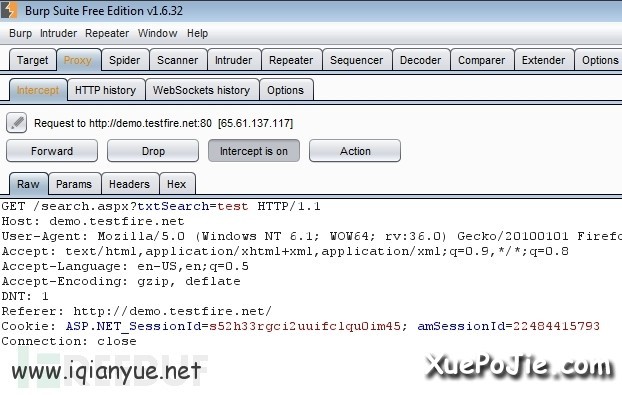
Intercepting proxy是针对web应用渗透测试工具的功能。Burp Suite的代理工具非常容易使用，并且能和其他工具紧密配合。要使用这个功能，第一步就是建立代理监听(Proxy–> Options功能下)。我的设置为了默认值localhost (127.0.0.1)，端口为8080。



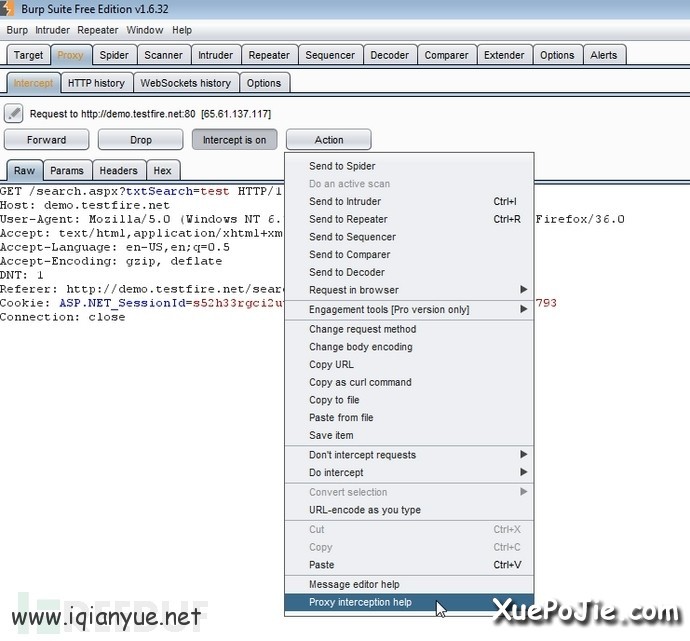
你可以点击编辑(“Edit”)进行修改，或者添加新的监听端口。一旦建立好，你就要到浏览器的网络连接设置处手动配置代理设置：



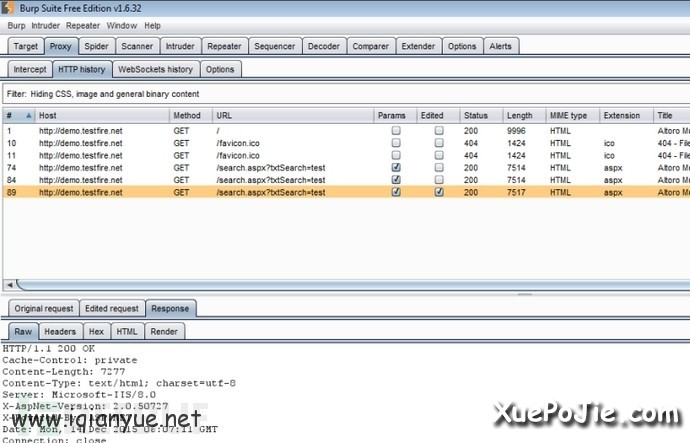
[我们](https://www.xuepojie.com/" \t "https://www.xuepojie.com/_blank)现在可以访问我们要测试的应用，然后看到发送的所有请求了。到Proxy –> Intercept标签页，然后确保截获功能开启(“Intercept is on”)，然后就能看到所有的请求了。



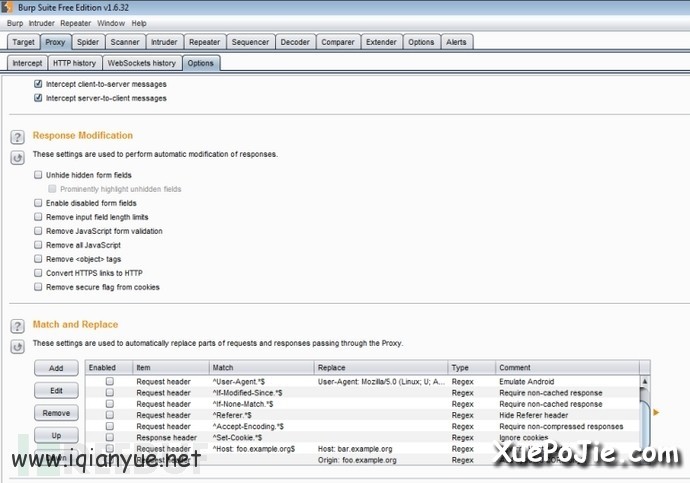
你可以修改请求，然后点击“Forward”发送修改后的请求，如果不想发送某些请求你也可以点击“Drop”按钮。“Actions”按钮下还有很多其他的功能。



如果你想回过头看下前面发送的请求，你可以切换到Proxy –> HTTP History标签页，这里有所有的请求列表，还有些详情如响应的长度，MIME类型和状态码。如果你修改过请求，你会看到两个标签，分别是修改前和修改后的请求：

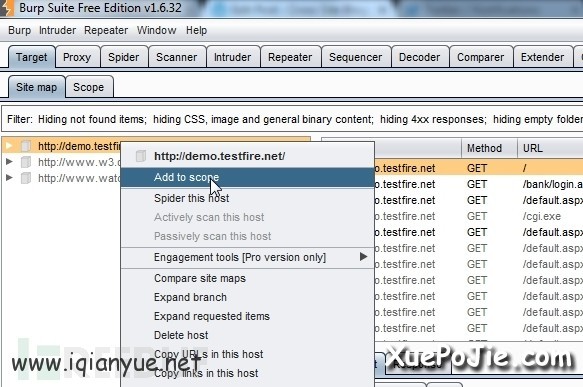


另一个有用的功能是自动修改请求/响应，功能位于Proxy –> Options。通过这个功能可以去除JavaScript的表单验证。你也可以用正则表达式匹配替换请求/响应：

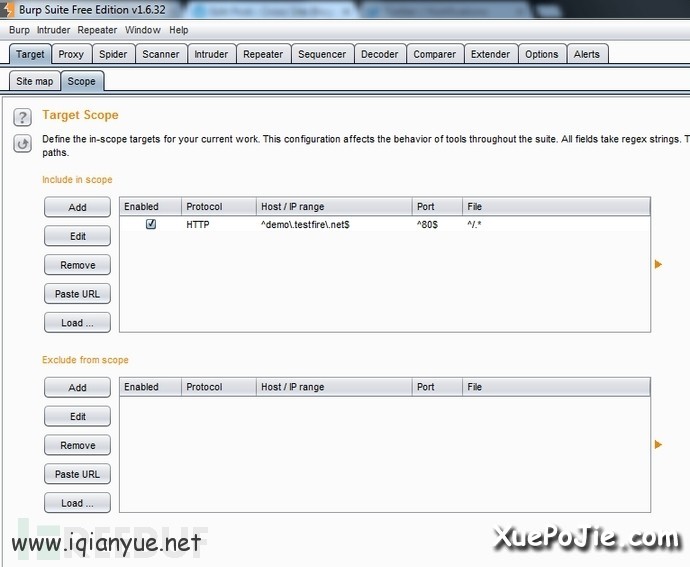


****Spider（爬虫）****

当你在对web应用进行初步检查的时候，Burp Suite的spider工具非常有用。当你浏览Web应用时，它会从HTML响应内容中主动生成一份URL列表，然后尝试连接URL。要使用爬虫功能，我们首先要切换到Target–> Site Map标签，然后右键域名，选择“Add To Scope”：



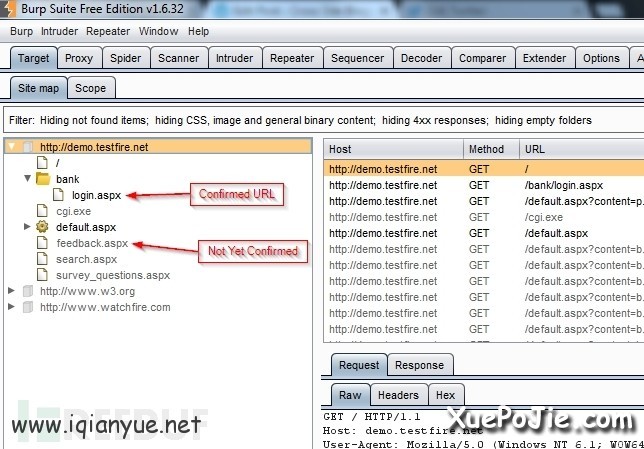
所有加入的域名都在Target –> Scope标签页里。你可以手动添加域名，修改，或者添加需要配出的URL（比如如果你不希望对“联系我们”的表单进行自动化测试，就可以把它排除掉）：



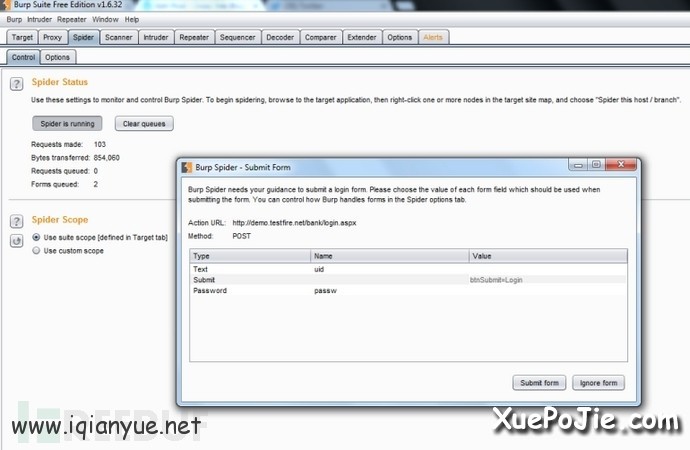
如果我们现在进入Spider –> Control标签，就能看到有些URL正在排队中，注意看下面，爬虫只会对scope中的域名进行测试：



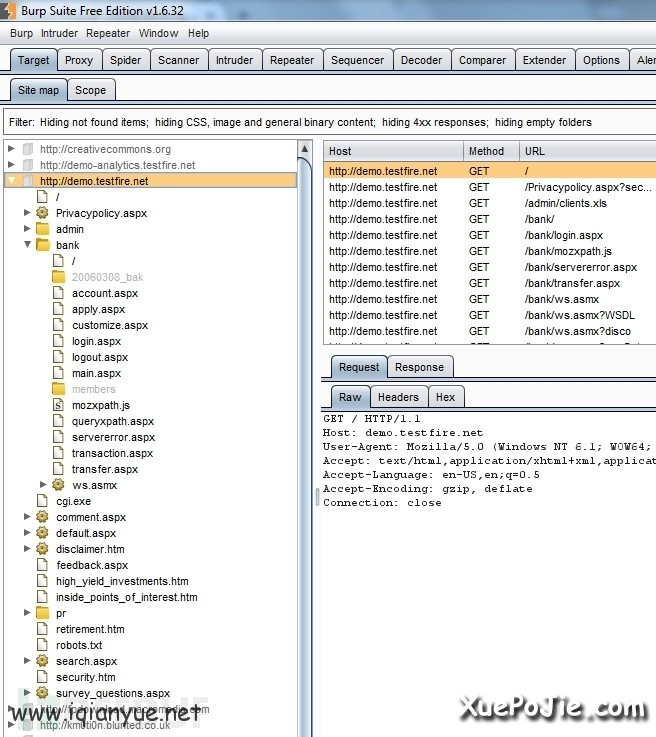
回到Site Map我们可以看到URL的列表，黑色代表我们已经成功访问过那个页面，爬虫确认过是有效的。灰色代表爬虫在HTML响应中找到了这个URL但是还没有确认是否有效：



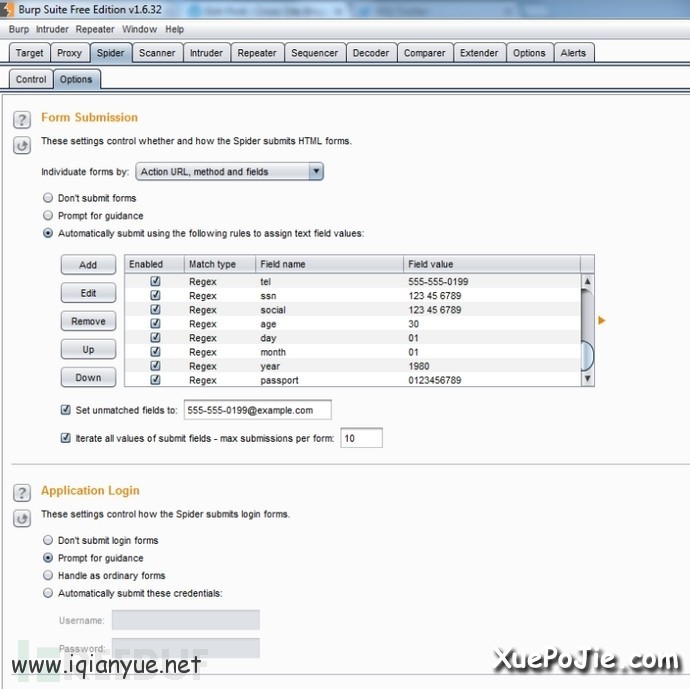
基本的设置后，我们返回到Spider –> Control标签，点击“Spider Is Paused”按钮运行工具，它会尝试连接所有之前找到的URL，包括在运行过程中找到的新的。如果过程中有表单需要填写，它会弹出表单供你填写，确保能收到有效的响应：



现在Site Map中就有整理整齐的URL了：



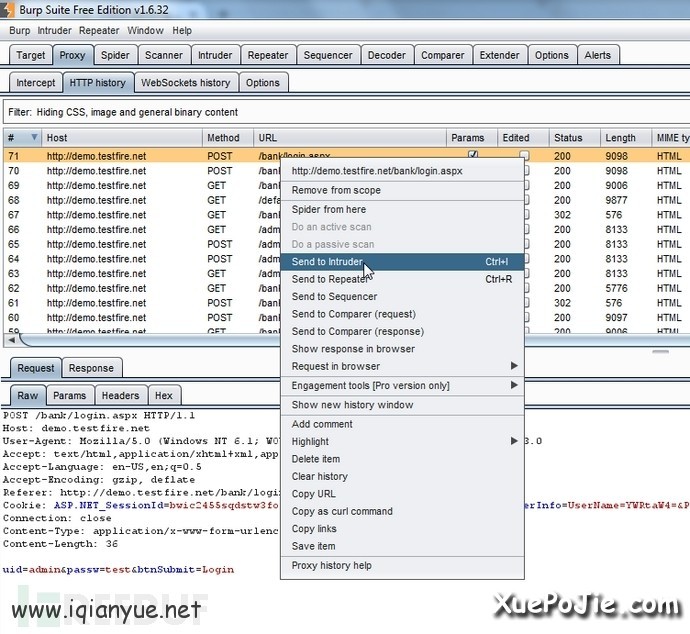
Spider –> Options标签下有些你可以调整的选项，如user-agent ，或者爬虫应该爬多深，两个重要的设置是表单提交和应用登录，设置好之后爬虫可以自动为你填写表单：



****Intruder****

Intruder是Burp Suite中最受欢迎的工具。Intruder是获取Web应用信息的工具。它可以用来爆破，枚举，漏洞测试等任何你想要用的测试手段，然后从结果中获取数据。

我举个例子来演示Intruder的使用方法。即爆破登录页面的管理员密码（假设没有帐号锁定）。首先，我们切换到Proxy-> HTTP History，右键要测试的请求，点击“Send To Intruder”：



接下来我们切换到Intruder标签，准备攻击。程序会在Target标签里自动填上请求中的host和端口。在Position（位置）标签出哦我们可以看到我们选择的请求并设置我们要攻击的位置。用鼠标高亮想要攻击的位置， 然后点击右边的“Add”，如果需要的话可以选择多个位置：

最上面的地方有多种攻击类型，本例中我们使用默认的Sniper，但实际上每种攻击类型都有特定用途：

Sniper – 这个模式使用单一的payload组。它会针对每个位置设置payload。这种攻击类型适合对常见漏洞中的请求参数单独地进行fuzzing测试。攻击中的请求总数应该是position数量和payload数量的乘积。

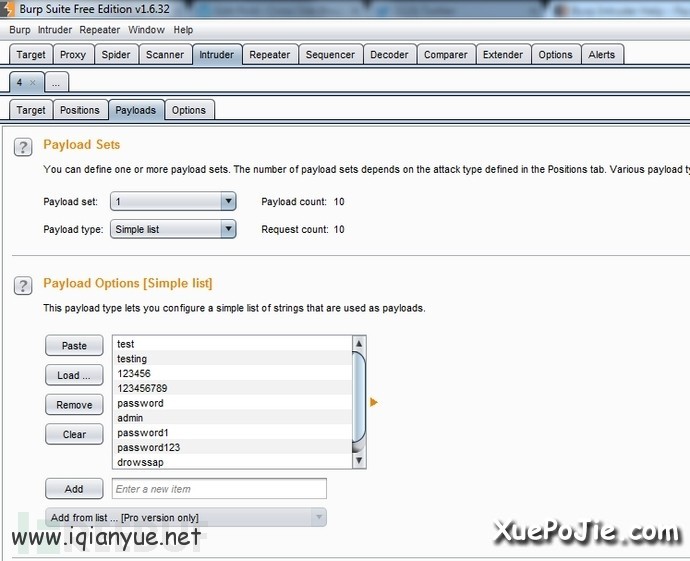
Battering ram – 这一模式使用单一的payload组。它会重复payload并且一次把所有相同的payload放入指定的位置中。这种攻击适合那种需要在请求中把相同的输入放到多个位置的情况。请求的总数是payload组中payload的总数。

Pitchfork – 这一模式使用多个payload组。对于定义的位置可以使用不同的payload组。攻击会同步迭代所有的payload组，把payload放入每个定义的位置中。这种攻击类型非常适合那种不同位置中需要插入不同但相关的输入的情况。请求的数量应该是最小的payload组中的payload数量。

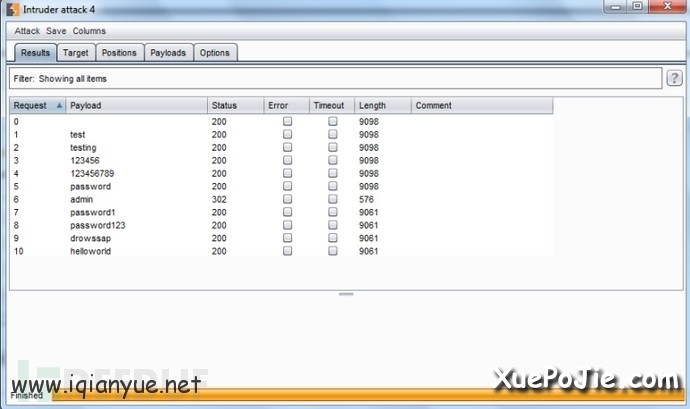
Cluster bomb – 这种模式会使用多个payload组。每个定义的位置中有不同的payload组。攻击会迭代每个payload组，每种payload组合都会被测试一遍。这种攻击适用于那种位置中需要不同且不相关或者未知的输入的攻击。攻击请求的总数是各payload组中payload数量的乘积。

位置设定好之后我们切换到Payloads标签，选择攻击时使用的数据。顶部的地方你可以看到payload组。各个组都对应设置的各个位置。我们可以选择payload类型，如简易列表(Simple List)。

在那下面有一些payload选项。每个payload类型都有不同的选项，供用户为你的测试进行修改。我经常使用的是数字(Numbers)，你可以设置范围，选择是连续的数字还是随机数字，还有每次攻击时的步长等。不过对于我们要搞的爆破攻击，我们只需要添加一个密码字典就行：



接下来就是点击右上角的开始攻击(Start Attack)按钮。程序就会弹出一个新的窗口，显示的是尝试的每个payload和响应的详情。我们的例子中，第六个请求获取到了正确的密码：



我们返回主窗口，然后切换到Intruder –> Options标签页，可以发现还有些别配置。其中一个很重要的是“Grep– Match”功能，这个功能可以让你基于HTML中的字符串或者正则表达式标记出结果。这些标记会在新增的栏里出现。