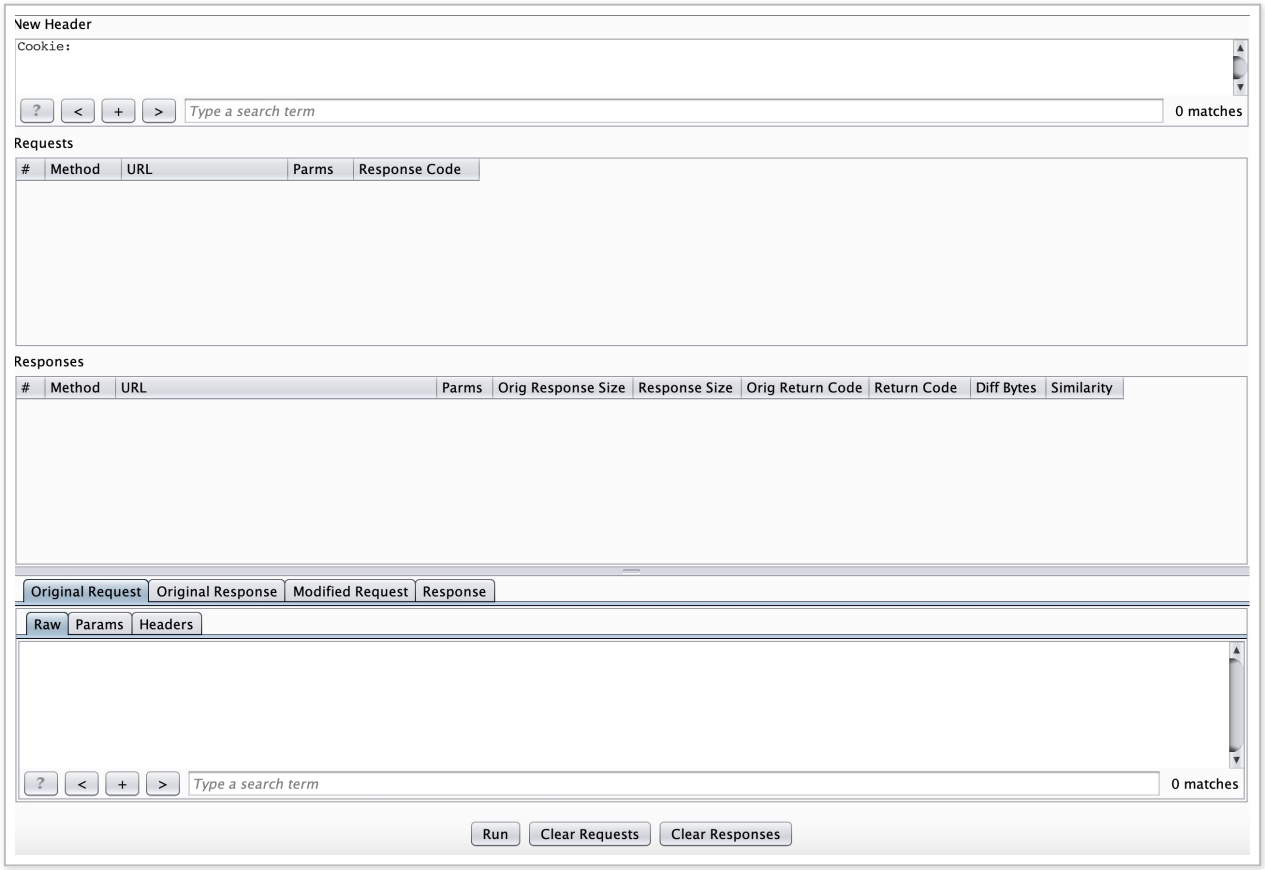
基于 BurpSuite 快速探测越权 - Authz 插件 · Chen's Blog

背景

在平时的测试中，会经常的碰到业务功能较多的站点，如果想全面又快速的完成逻辑越权漏洞的检测不得不借助 Authz 插件去辅助检测越权问题。

Authz 的工作原理

我们平时做测试的时候发现越权问题都是基于修改 ID 的方式：A 的 ID 改成 B 的 ID 然后进行请求查看是否可以越权获取到信息，或当 ID 的规律已知情况下基于 Burp Intruder 模块直接去遍历 ID。而基于 Authz 的检测是不一样的，其是将用户认证的 HTTP 请求头进行修改（Cookie 之类的），然后通过响应长度、响应状态码判断是否存在越权；从本质上来讲没有任何区别，只是换了一个角度，但这样的好处是一定程度上的减少了测试的时间（例如：一个商城的业务系统，你有 A、B 账户，A 账户买了个商品获得一个订单信息请求，当你想测试是否能越权获取 B 账户订单时就需要使用 B 账户去再购买，然后判断测试。）

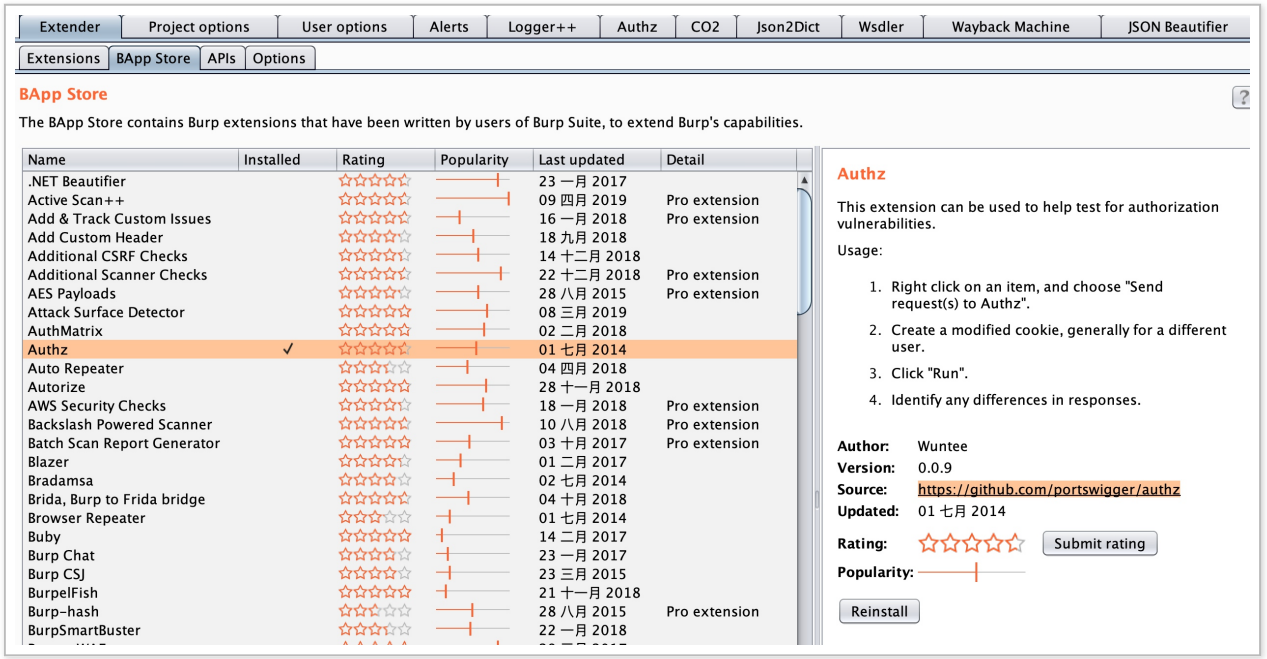


BurpSuite Authz 插件界面

安装 Authz 插件

Github 地址： https://github.com/portswigger/authz

快速安装 -> 在 BurpSuite 的 BApp Store 应用市场可以直接下载安装：



使用 Authz 插件检测

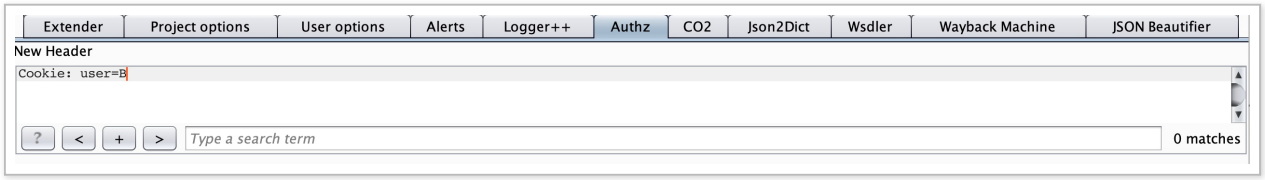
使用插件检测的前提条件：同个业务系统中两个测试账号

作用：A 账户用于功能的操作，B 账户用于提供凭证（Cookie 或者其他的用户身份凭证

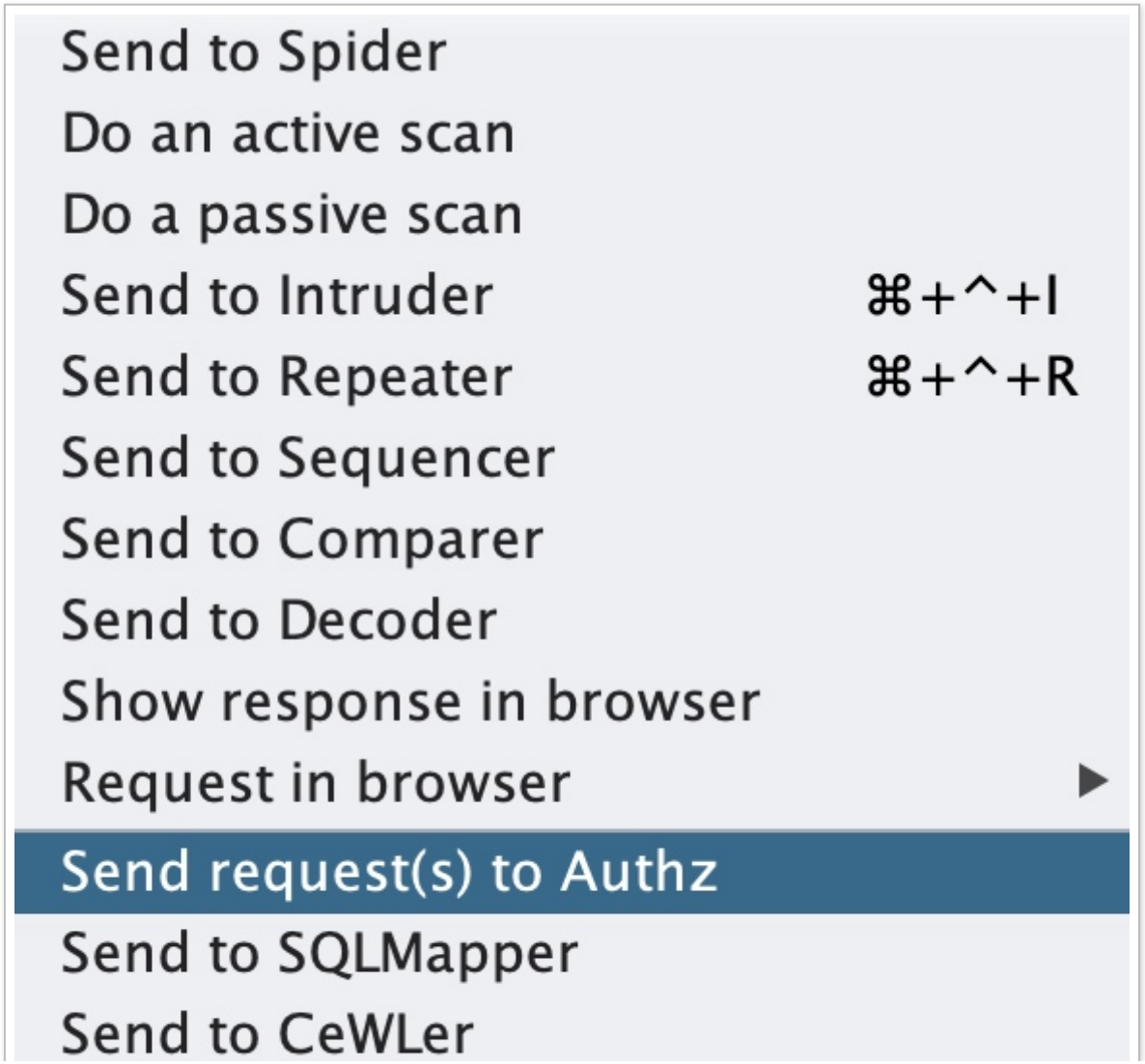
请求头）

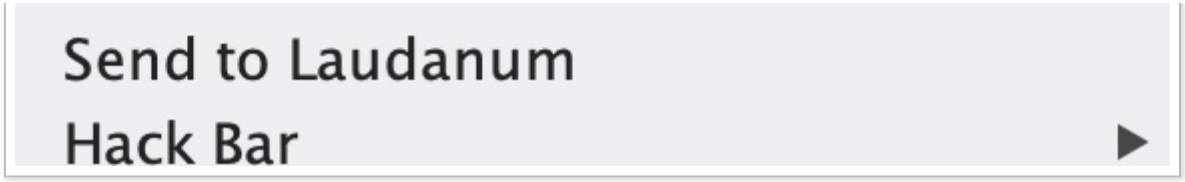
举例说明：

一个业务系统，将 A、B 账户登入，同时获取 B 账户的 Cookie 或者其他的用户身份凭证请求头，填入到 Authz 的 New Header 里：

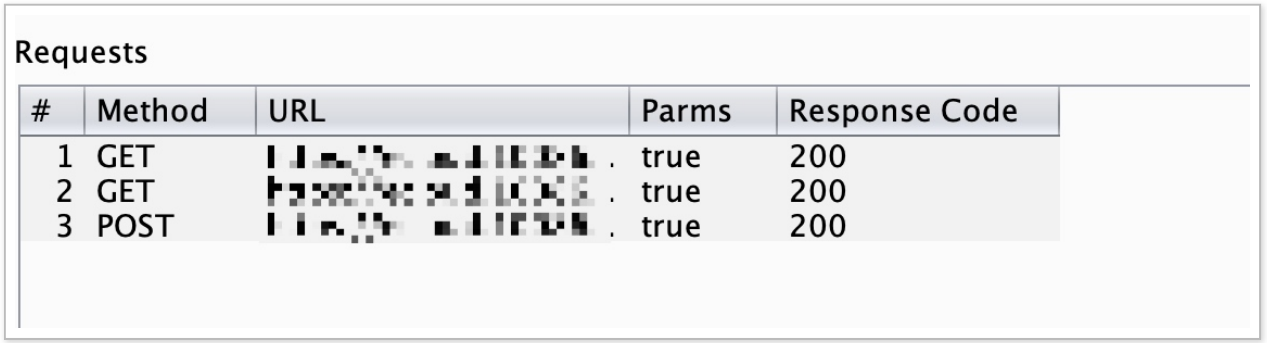


A 账户去请求（Burp 别忘了监听着），寻找读取类请求（该类请求要包含 ID 之类的特征）然后右键请求包将该请求发送到 Authz 插件内：

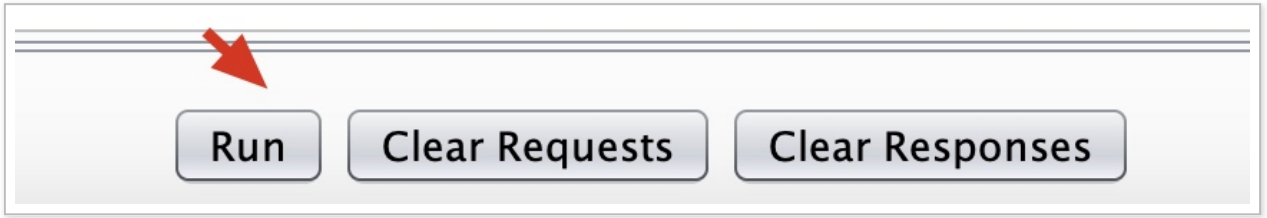

Send to Spider
Do an active scan
Do a passive scan
Send to Intruder ⌘+^+I
Send to Repeater ⌘+^+R
Send to Sequencer
Send to Comparer
Send to Decoder
Show response in browser
Request in browser ▶
Send request(s) to Authz
Send to SQLMapper
Send to CeWLer



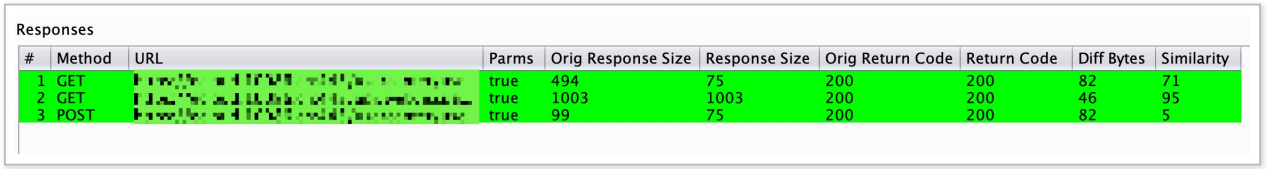
发送的请求会在 Burp 的 Authz 的 Tab 标签窗口内：



当收集的差不多了，点击 run 跑起来：

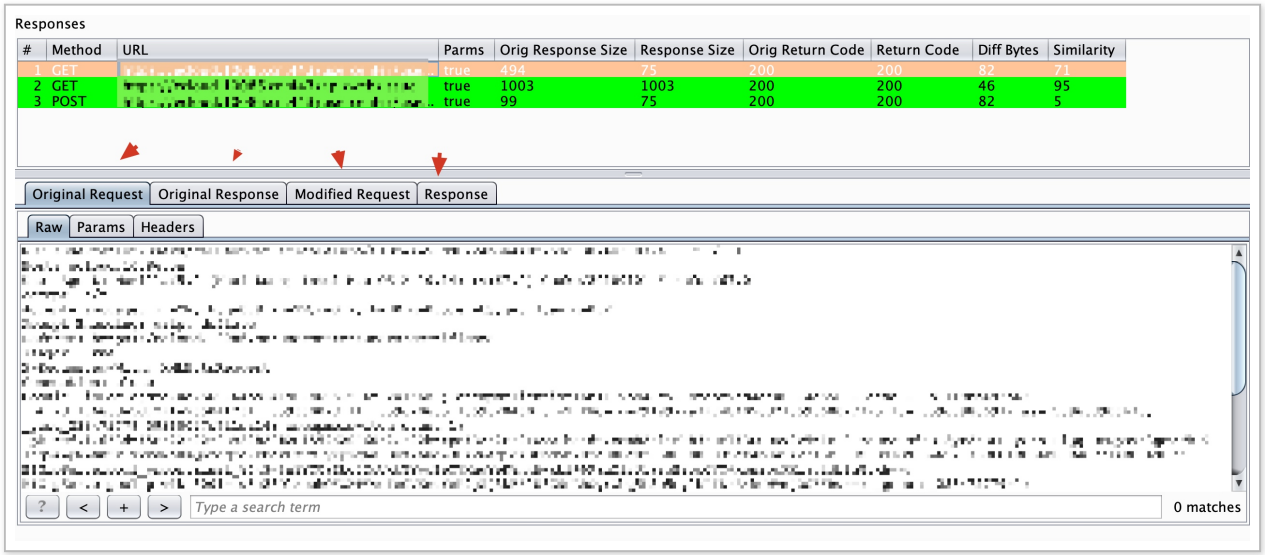


结果会在 Responses 处显示：



当原响应内容长度、响应状态码和被修改后请求的响应内容长度、响应状态码一致则会绿。

也就代表着存在越权，单击选择一行即可在下面展示出请求、响应的报文：



这里经过进一步检验（理论上不需要检验，但出于对测试的严谨态度还是检验一下比较好~）顺利的发现了三枚越权访问漏洞。

一个业务系统测完之后就 Clear 掉所有的东西，接着下一个业务系统咯：

Authz 的优点和缺点总结

优点：使用简单、省时省力

缺点：只是适用于检测越权读取类操作，删除编辑类操作还需人工判断。