

Burp Suite使用中的一些技巧

 pa55w0rd.online/burp

0x00 前言

Burp Suite是Web应用程序测试的最佳工具之一，其多种功能可以帮我们执行各种任务.请求的拦截和修改,扫描web应用程序漏洞,以暴力破解登陆表单,执行会话令牌等多种的随机性检查。

大家都很熟悉工具的使用了，这里介绍几个技巧，**欢迎大家补充**

详细教程参考：

Burp Suite 说明书（开车版） 链接: <https://pan.baidu.com/s/1tNTzSO1eKDtRg7-T0cdjVQ>
提取码: 8un3 复制这段内容后打开百度网盘手机App，操作更方便哦

Burp Suite 实战指南

0x01 专业版激活问题

使用burp-loader-keygen.jar注册机激活

支持1.6-目前最新的都能使用该注册机激活

注意有人利用这个破解补丁添加后门传播病毒，后门分析帖子》》

原始注册机文件的MD5

burp-loader-keygen.jar MD5: A4A02E374695234412E2C66B0649B757

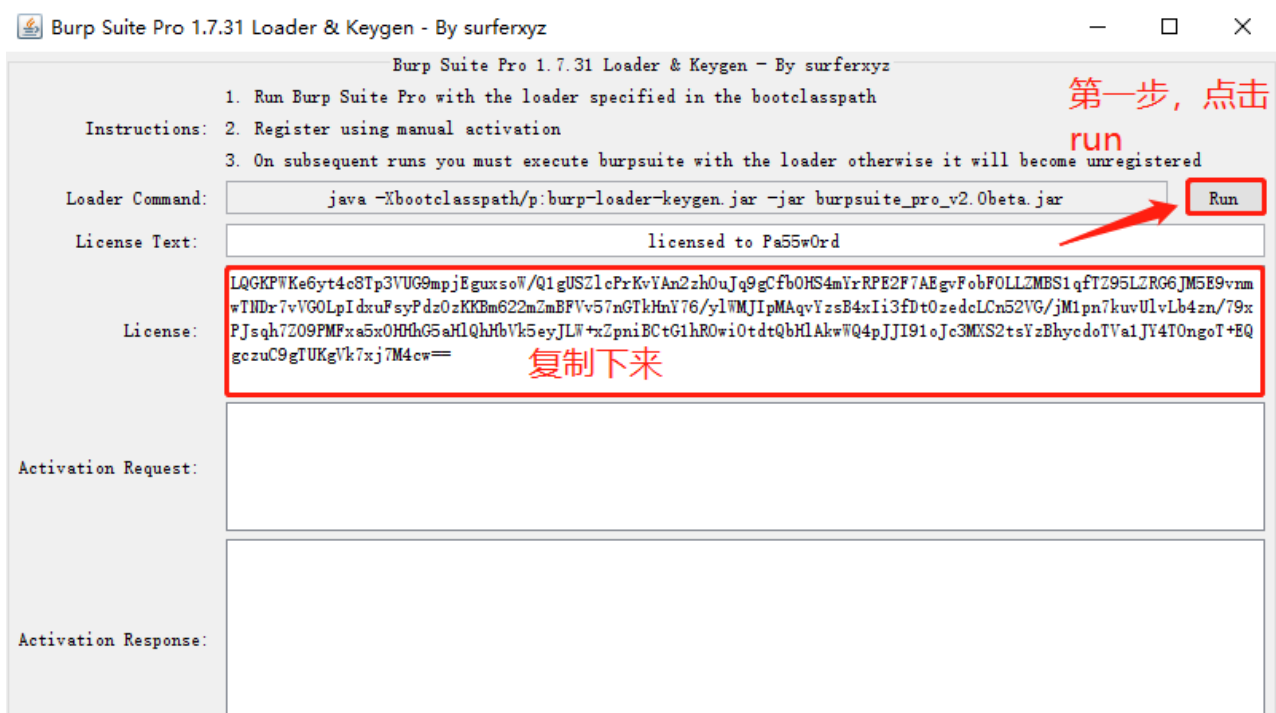
源文件MD5官网查看：

<http://releases.portswigger.net>

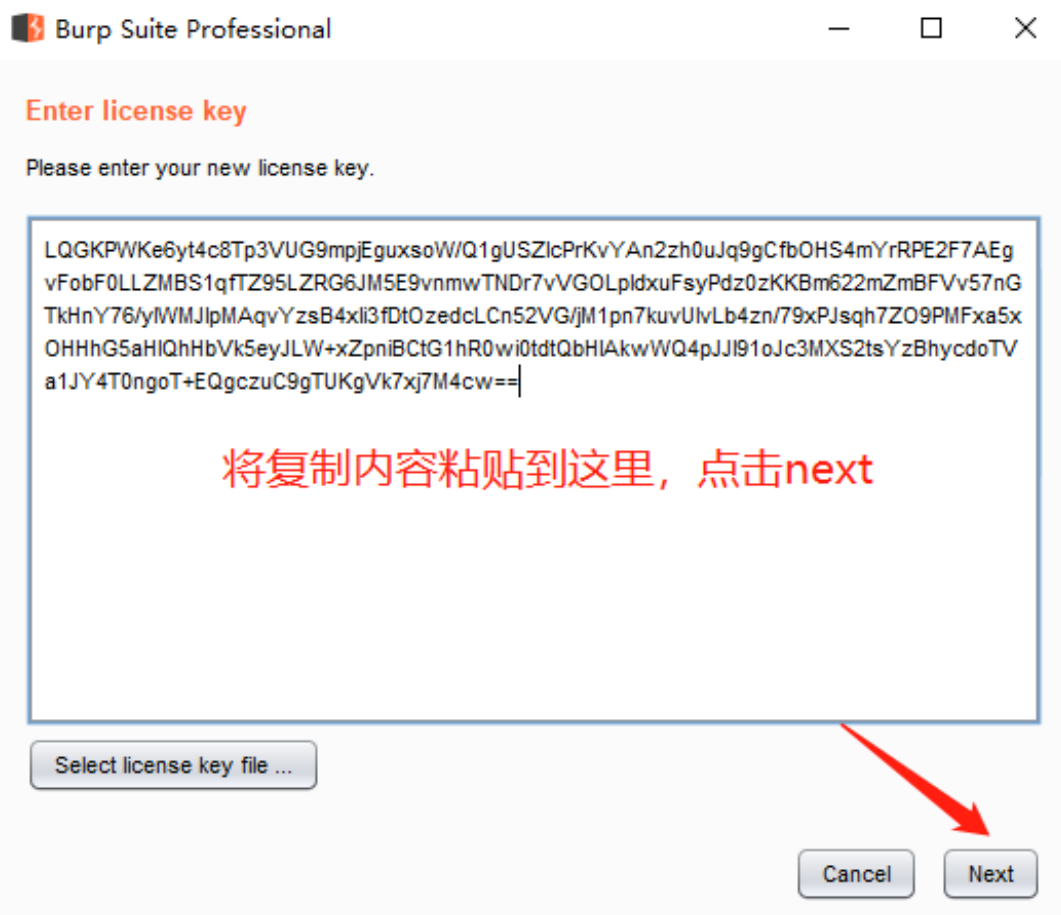
使用burp-loader-keygen.jar破解的，每次都需要在该工具上启动，或使用命令行启动 `java -Xbootclasspath/p:burp-loader-keygen.jar -jar burpsuite_pro.jar`

激活步骤

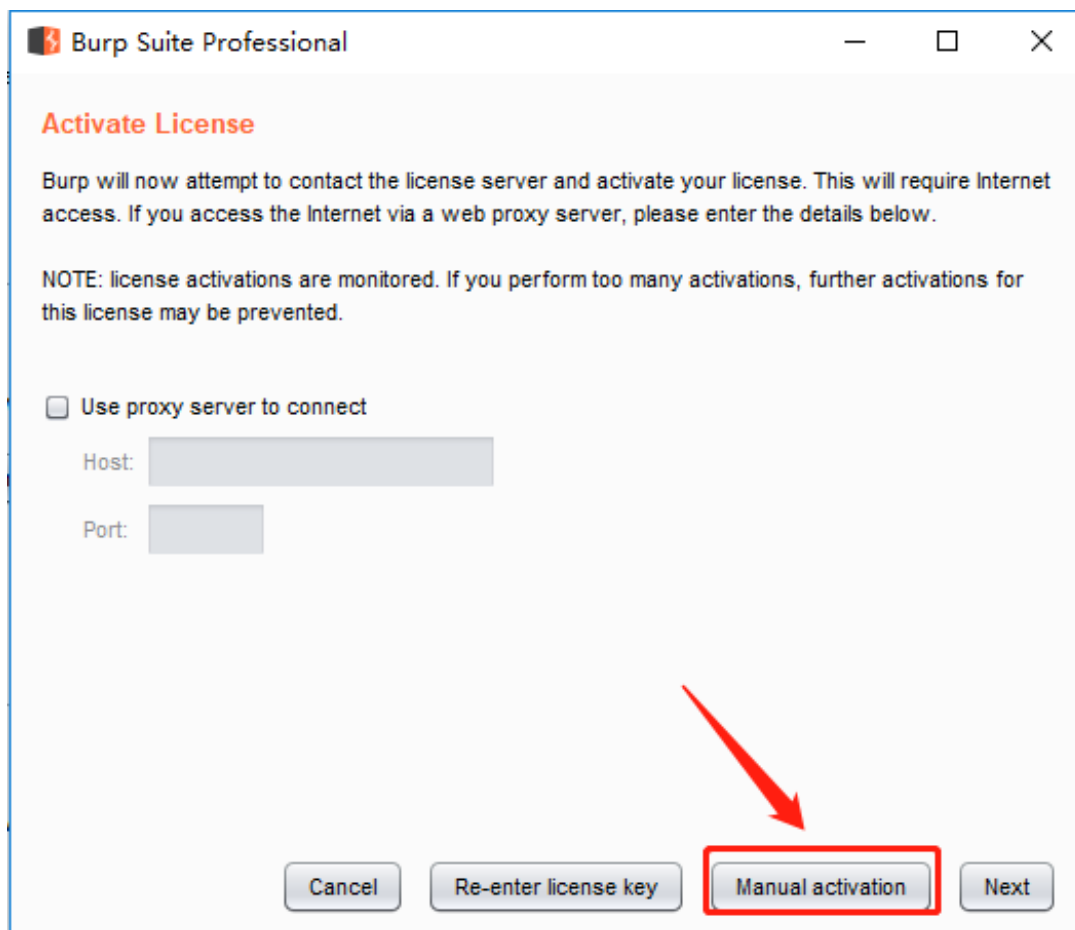
1. 首先打开burp-loader-keygen.jar



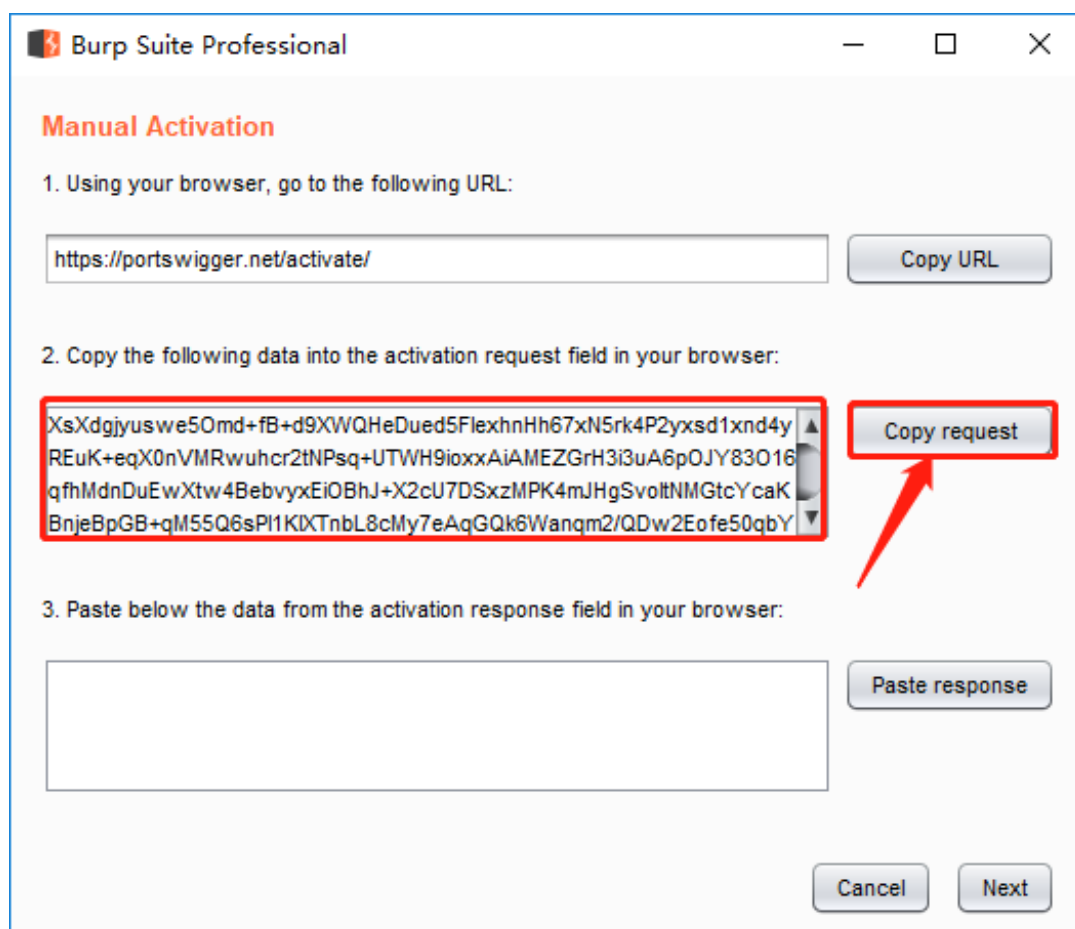
2. 点击burp, 将复制内容粘贴到Enter license key,点击next



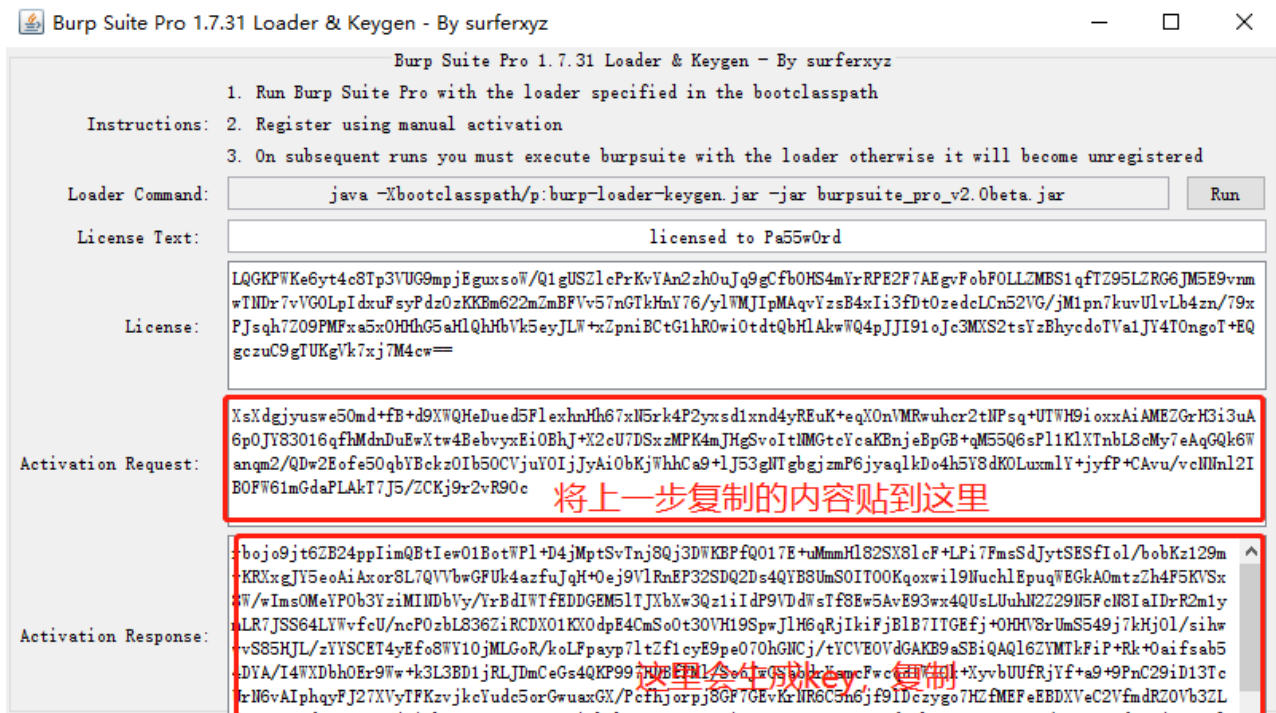
3. 点击Manual activation



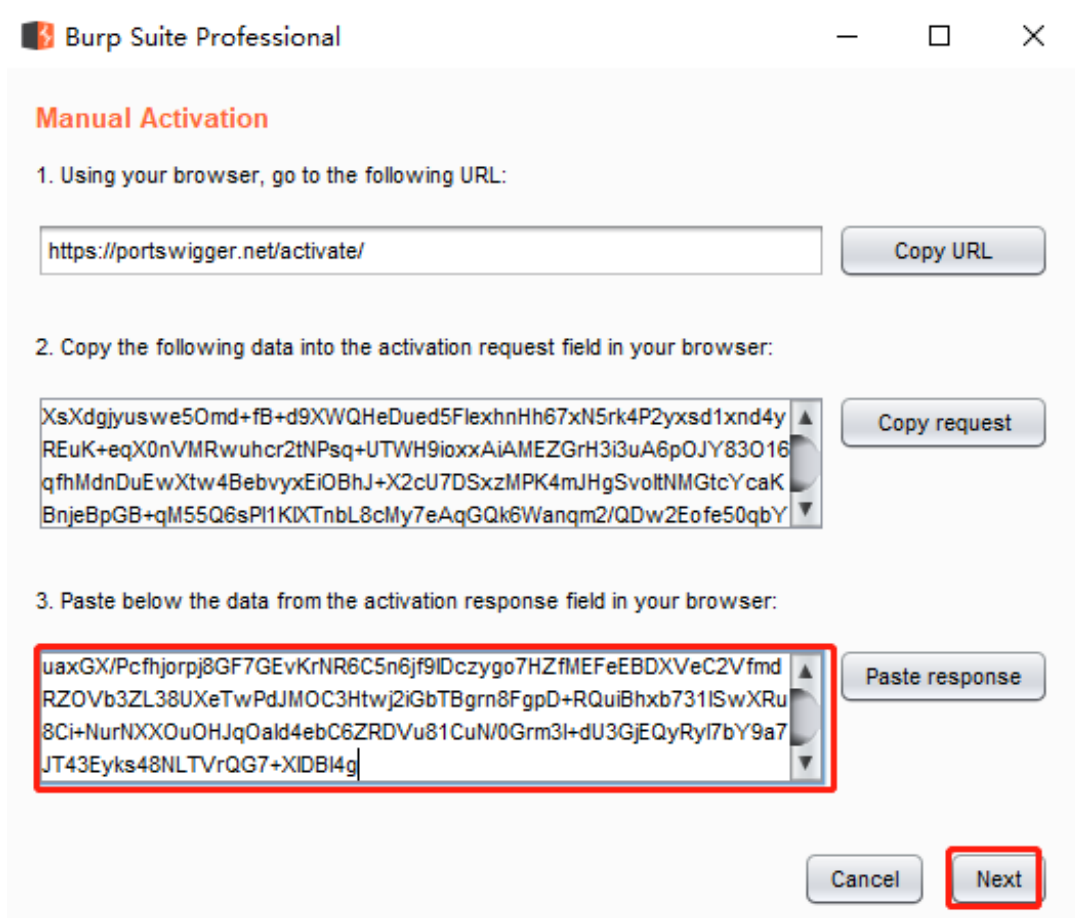
4. 复制2 中的内容

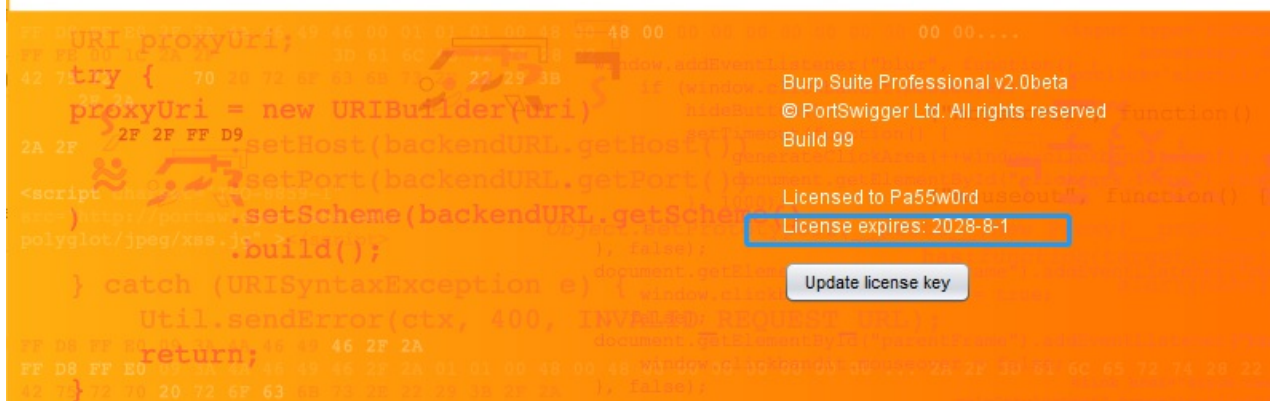


5. 将复制内容粘贴到burp-loader-keygen.jar中的Activation Request, 下方Activation Response会生成key, 复制



6. 回到burp, 将复制内容粘贴到3中, 点击next, 激活成功

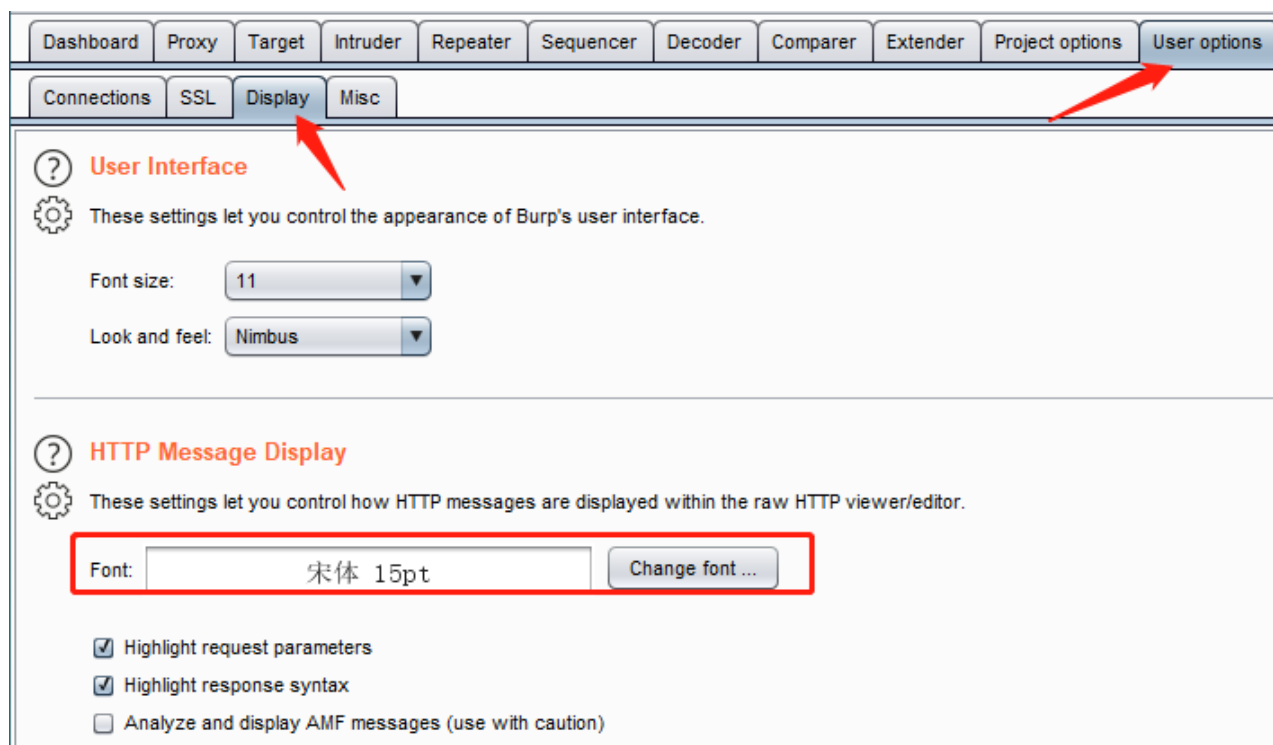




0x02 中文显示乱码问题

User options - Display - HTTP Message Display - Change font...

选择一个中文的字体格式，调整显示字体大小



如果是Unicode编码，可以使用插件转换成中文

0x03 https网站导入证书

访问burp, 127.0.0.1:8080，下载CA Certificate证书，将证书导入浏览器中

以Firefox 68为例 选项-隐私与安全-证书-查看证书-证书颁发机构-导入之前下载的 cacert.der证书文件

0x04 移动端安装证书

在使用Burp代理分析移动设备应用通信的时候，会遇到使用SSL/TLS的应用，这时候就会因为证书验证不通过而无法抓包分析，这时候就需要在移动设备上安装证书，使其信任Burp代理

0x03中，在浏览器中打开证书颁发机构，找到PortSwigger CA文件，导出PortSwiggerCA.crt，复制到手机中导入

安卓手机

设置-更多设置-系统安全-加密与凭证-从存储设备安装

PS: 不同收集安装路径可能不一样，我这个是小米8的

模拟器安装证书

安装过程和上面差不多，之前碰到过一个问题，导入证书的时候证书文件是灰色的，解决方法：安卓模拟器安装证书问题

IPhone 设备

直接传到手机端安装证书，安装成功后，在设置 - 通用 - 描述文件 里可以看到证书

在设置 - 通用 - 关于本机 - 证书信任设置中信息刚刚安装的证书

然后通过设置wifi代理，就可以进行抓包了

0x05 代理设置

BurpSuite的核心是代理Proxy，通常情况下使用BurpSuite的套路是：浏览器设置BurpSuite代理——>访问Web应用程序——>BurpSuite抓包分析

一般情况

一般情况下，直接设置浏览器的代理服务器为Burp即可，推荐浏览器插件**SwitchyOmega**，chrome和火狐中都有，设置好情景模式一键切换

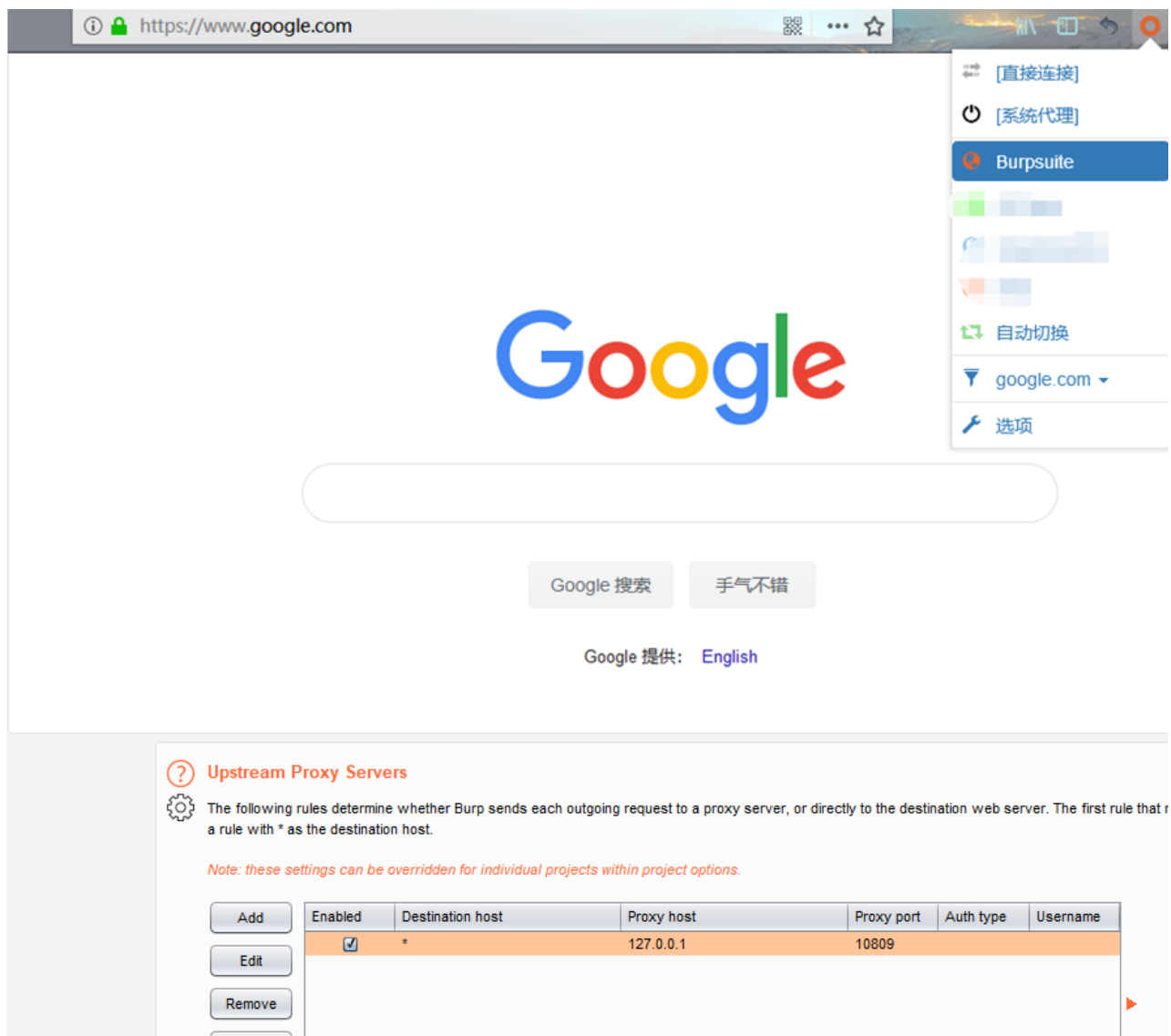
代理链

在某些网络环境中，访问目标网站需要走代理。浏览器设置代理之后就不能再设置Brup代理，这种情况可以借助代理链

1. upstream proxy servers

在该设置项中，可以设置多个上游代理服务器规则，满足规则的请求将被发送至相应的代理服务服务器。

User options – Connections – Upstream Proxy Servers



2. SOCKS Proxy

与Upstream Proxy Servers的作用类似，SOCKS Proxy的功能也是将请求内容发送至相应的代理服务器。不同之处在于，SOCKS Proxy作用于TCP协议层，因此如果设置了该项，那么所有的请求数据都会被发送至SOCKS代理服务器

User options – Connections – SOCKS Proxy

? SOCKS Proxy



These settings let you configure Burp to use a SOCKS proxy. This setting is applied at the TCP level, and all outbound requests will be sent via this proxy. If you have configured other proxies, they will be sent via the SOCKS proxy configured here.

Note: these settings can be overridden for individual projects within project options.

☒ Use SOCKS proxy

SOCKS proxy host: 127.0.0.1

SOCKS proxy port: 10808

Username:

Password:

☐ Do DNS lookups over SOCKS proxy

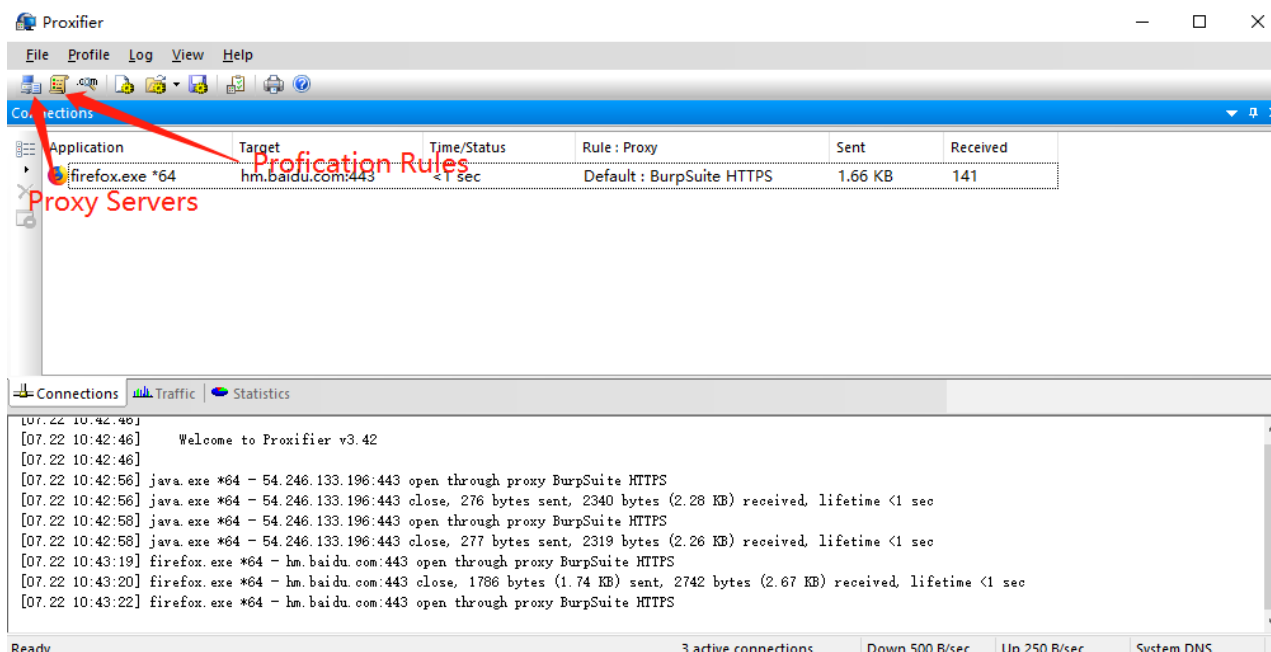
不支持代理的客户端

有时候，我们想对电脑上的某客户端进行抓包分析，然而这个客户端并没有代理设置的功能，怎么办？如果直接设置系统代理当然也是可以的，但是这样一来所有程序的流量都要经过BurpSuite，一方面有可能影响非测试软件的使用；另一方面，BurpSuite中非测试软件的请求记录过多也影响我们的分析测试。

Proxifier能够为那些本身不能设置代理的软件设置SOCKS或者HTTPS代理（链）

软件使用略过

Proxy Servers和Profication Rules



点击工具栏第一个图标，打开Proxy Servers对话框。Proxy Servers对话框分为上下两部分，上半部分用于设置代理服务器，下半部分用于设置代理链。

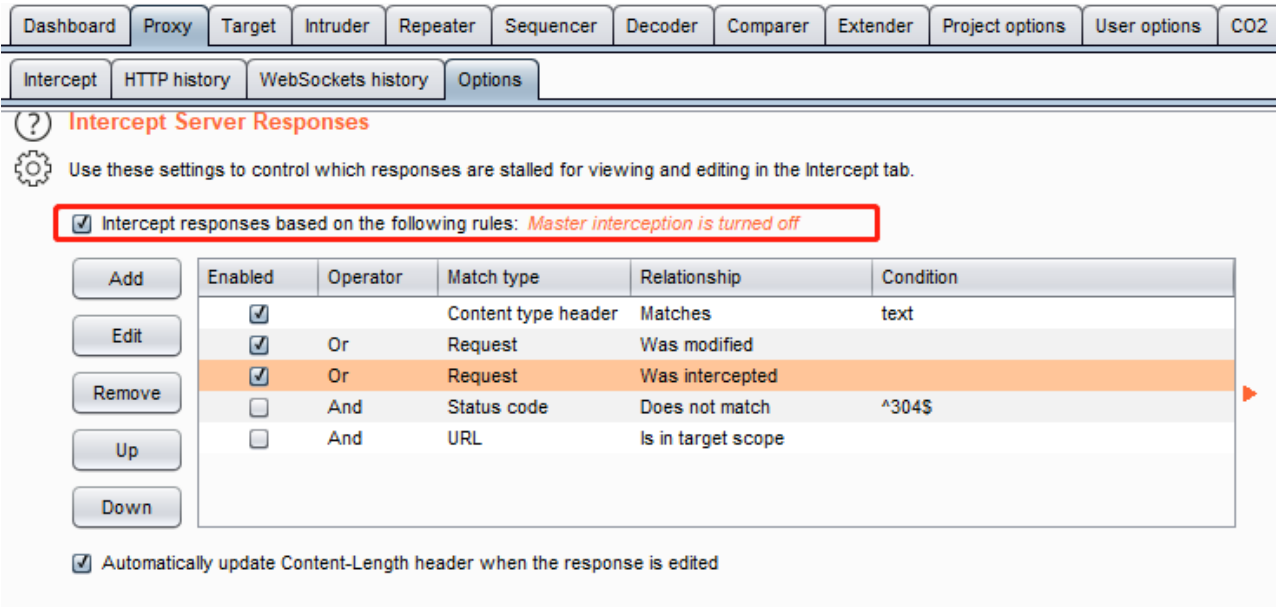
代理服务器设置完成之后开始进行规则设置，点击工具栏第二个图标进入Proxification Rules对话框

点击Add添加规则即可

0x06 拦截响应包

很多时候，在测试验证码等认证漏洞的时候，只在前端验证，可以通过修改响应包中的值绕过验证

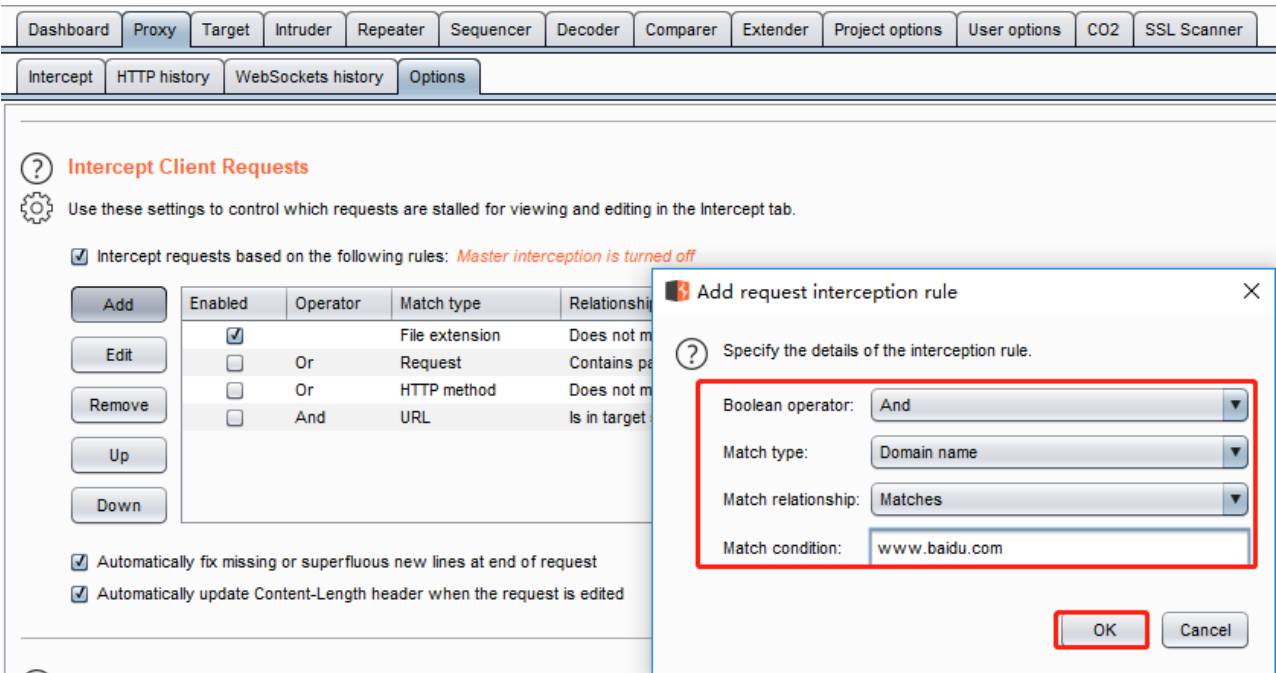
Proxy – Options – Intercept Server Responses 勾选启用



0x07 拦截指定url的请求响应包

请求包设置

Proxy – Options – Intercept Client Requests – Add



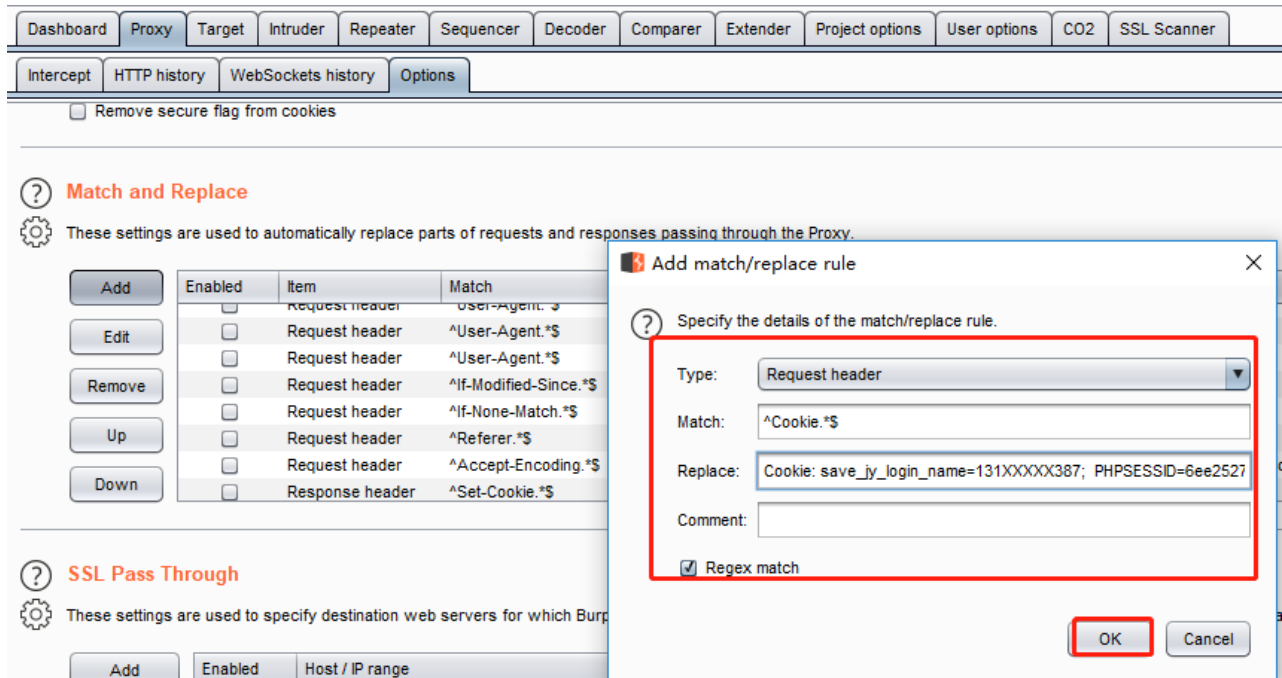
响应包设置

Proxy – Options – Intercept Server Responses – Add，同上图设置

0x08 匹配和自动替换cookie等

如盲打xss，接收到cookie后，需要在每个请求包中都替换cookie

Proxy - Options - Match and Replace - Add(增加其他自动替换同理，可以自动替换请求/响应内容，这里以cookie为例)



1. 挖掘CORS漏洞时，自动添加 Origin: foo.example.org ，从HTTP history中筛选返回内容 Access-Control-Allow-Origin: foo.example.org
2. 自动添加XFF头等伪造IP

0x09 Intruder模块匹配返回包内的中文

在使用Intruder模块进行爆破或fuzz的时候，一般都通过返回包的长度或者状态码来识别是否爆破成功，如果碰到状态码一样或者返回包长度都不同的情况，可以通过Intruder->Option->Grep-Match 进行返回包内容的匹配

匹配类型有简单字符串和正则匹配

如果要匹配中文，需要将中文转换成十六进制，使用正则匹配的方式

0x10 生成csrf poc

对拦截的请求，右键

The screenshot displays the Burp Suite Professional v2.0beta interface. The main window is titled 'Intruder attack 5' and shows the results of an attack. The 'Grep - Match' section is configured with a regex expression to match the success message in the response. The 'Intruder attack 5' window shows the results of the attack, with the success message highlighted.

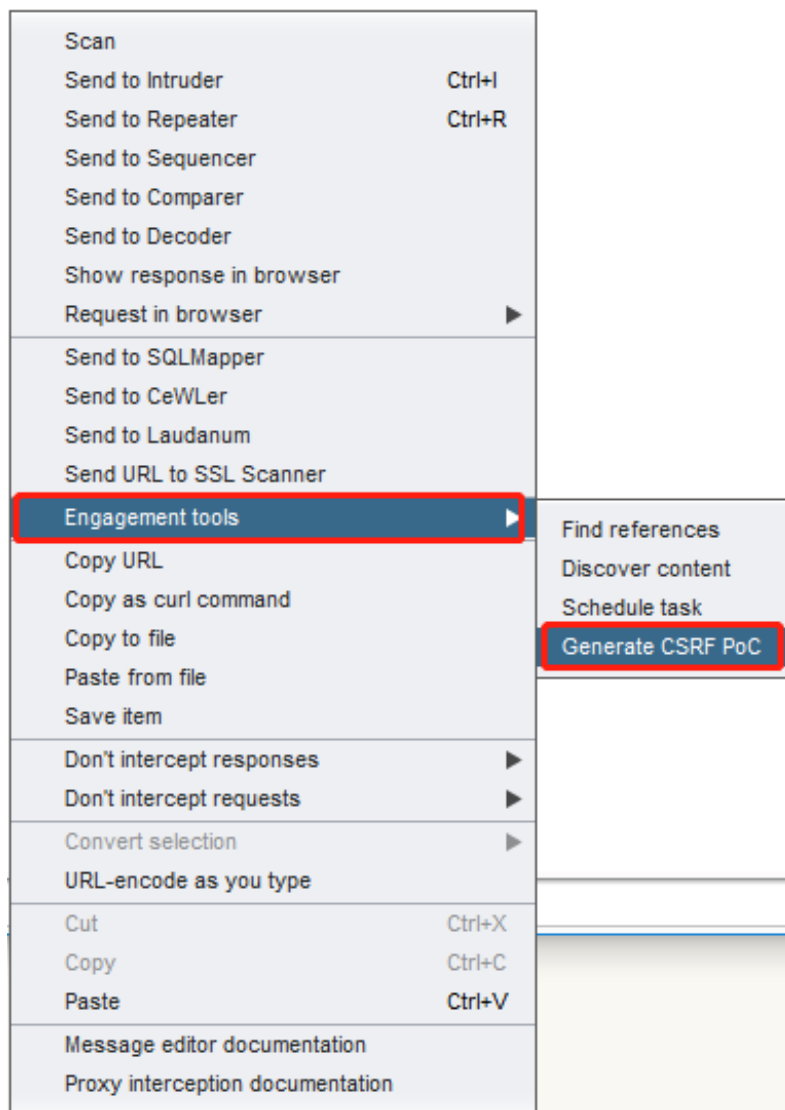
Grep - Match Configuration:

- Match type: ☒ Regex
- Exclude HTTP headers: ☒

Intruder attack 5 Results:

Request	Payload	Status	Error	Timeout	Length	Match	Comment
12	passvvd	200			5602	<input checked="" type="checkbox"/>	
0		200			5552	<input type="checkbox"/>	
1	123456	200			5553	<input type="checkbox"/>	
2	a123456	200			5554	<input type="checkbox"/>	
3	123456a	200			5554	<input type="checkbox"/>	
4	5201314	200			5554	<input type="checkbox"/>	
5	1111111	200			5553	<input type="checkbox"/>	
6	woaini1314	200			5557	<input type="checkbox"/>	
7	qq123456	200			5555	<input type="checkbox"/>	
8	123123	200			5553	<input type="checkbox"/>	
9	000000	200			5553	<input type="checkbox"/>	
10	1qaz2wsx	200			5555	<input type="checkbox"/>	

The response content shows the success message: `<p>登录成功Welcome to the password protected area admin</p></div>`. The success message is highlighted in the original image.



0x11 条件竞争漏洞测试

“竞争条件”发生在多个线程同时访问同一个共享代码、变量、文件等没有进行锁操作或者同步操作的场景中。

开发者在进行代码开发时常常倾向于认为代码会以线性的方式执行，而且他们忽视了并行服务器会并发执行多个线程，这就会导致意想不到的结果。

简单的说：本来你有100块钱，买一个商品要花100，你可以多开启多个线程去跑，有可能不止一个用户买成功

“竞争条件”漏洞有时很难通过黑盒/灰盒的方法来进行挖掘，因为这个漏洞很受环境因素的影响，比如网络延迟、服务器的处理能力等。一般都会通过对代码进行审计来发现此类问题

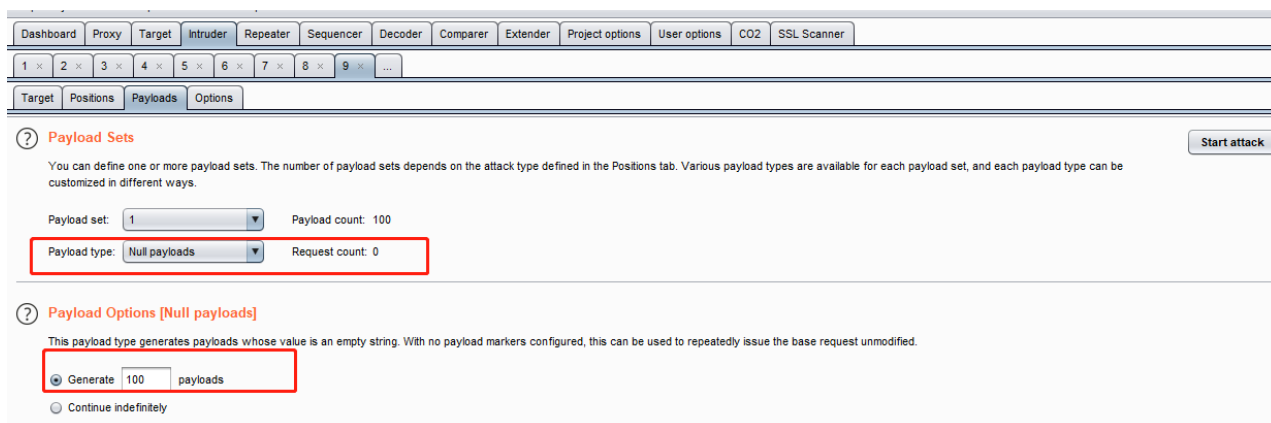
可以使用Burp的intruder功能来实现发送多个并发请求

将请求包发送至Intruder

Intruder – Payloads – Payload Stes

Payload type设置为NULL payloads

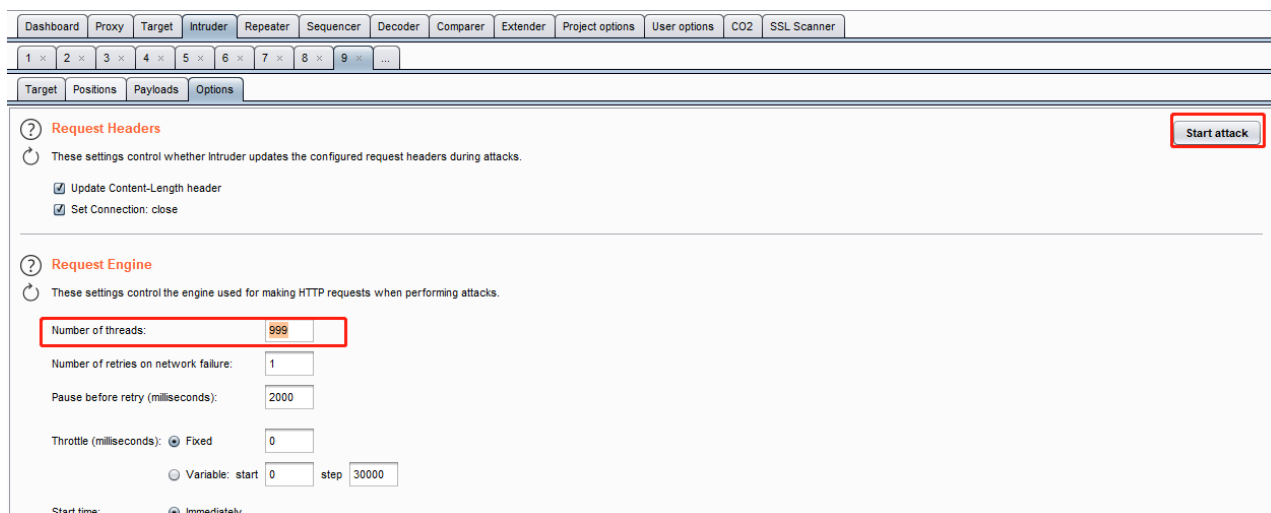
Payload Options 次数设置100次



The screenshot shows the 'Payload Options' tab in Burp Suite. The 'Payload Sets' section has 'Payload set' set to 1 and 'Payload count' set to 100. The 'Payload type' is set to 'Null payloads', and 'Request count' is 0. The 'Payload Options [Null payloads]' section has 'Generate' selected with a value of 100, and 'Continue indefinitely' is unselected. A 'Start attack' button is in the top right.

Intruder – Options – Request Engine

线程数设置最大999，点击Start attack



The screenshot shows the 'Request Engine' section in the 'Intruder - Options' tab. The 'Number of threads' is set to 999. Other settings include 'Number of retries on network failure' (1), 'Pause before retry (milliseconds)' (2000), 'Throttle (milliseconds)' (Fixed at 0), and 'Start time' (Immediately). A 'Start attack' button is in the top right.

0x12 交互式应用程序安全测试 IAST

主界面菜单项 burp - Burp Infiltrator

参考：<https://www.hackingarticles.in/advance-web-application-testing-using-burpsuite/>

0x13 点击劫持

主界面菜单项 burp - Burp Clickbandit

参考：<https://www.hackingarticles.in/advance-web-application-testing-using-burpsuite/>

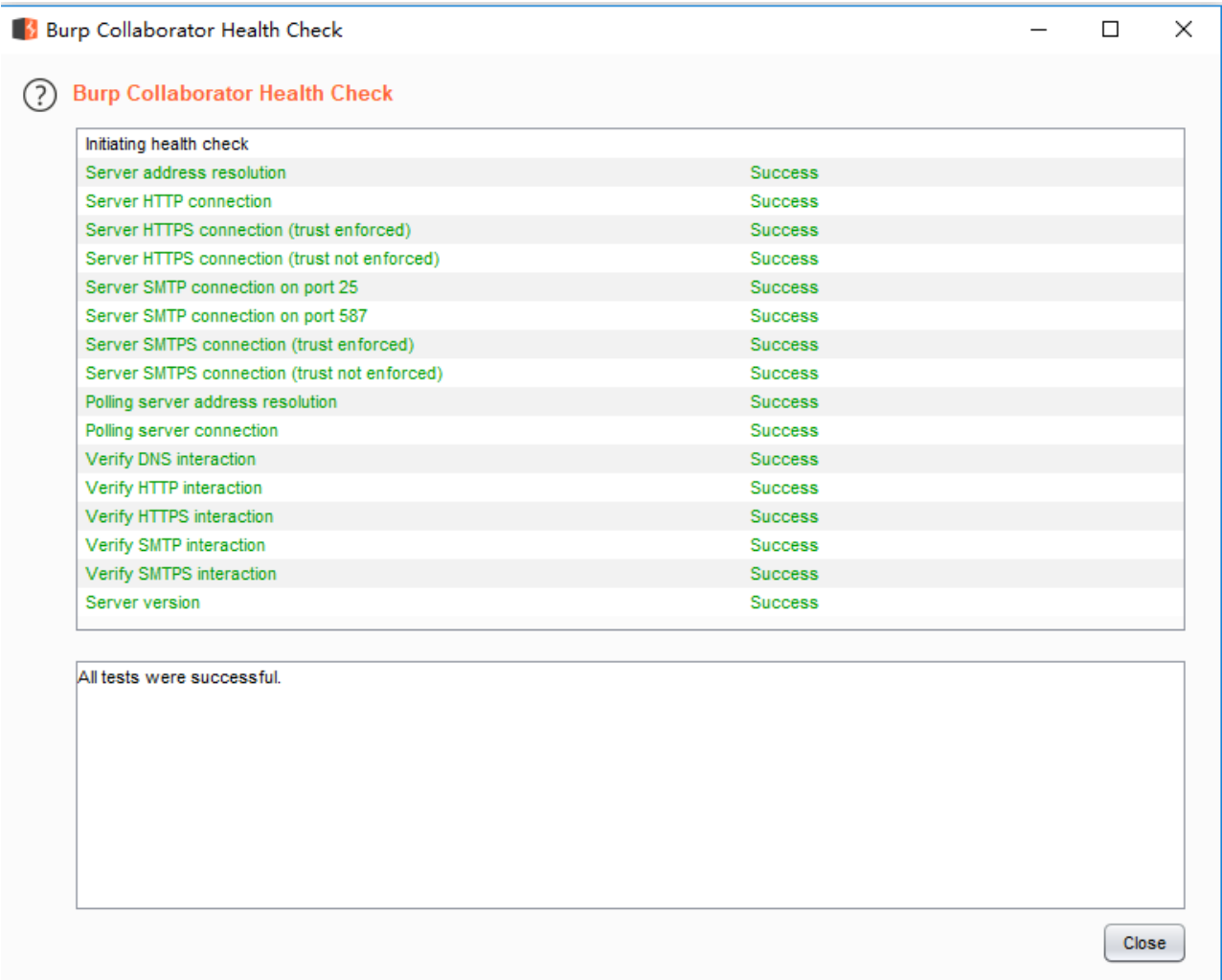
0x14 DNSlog功能

Burp Collaborator是从Burp suite v1.6.15版本添加的新功能，也就是DNSlog，监控DNS解析记录和HTTP访问记录，在检测盲注类漏洞很好用，也可以借助第三方服务

这里引出两个概念，**In-band attack与 out-band attack（带内与带外攻击）**，带内与带外的区别核心在于是否使用不同的通信通道。

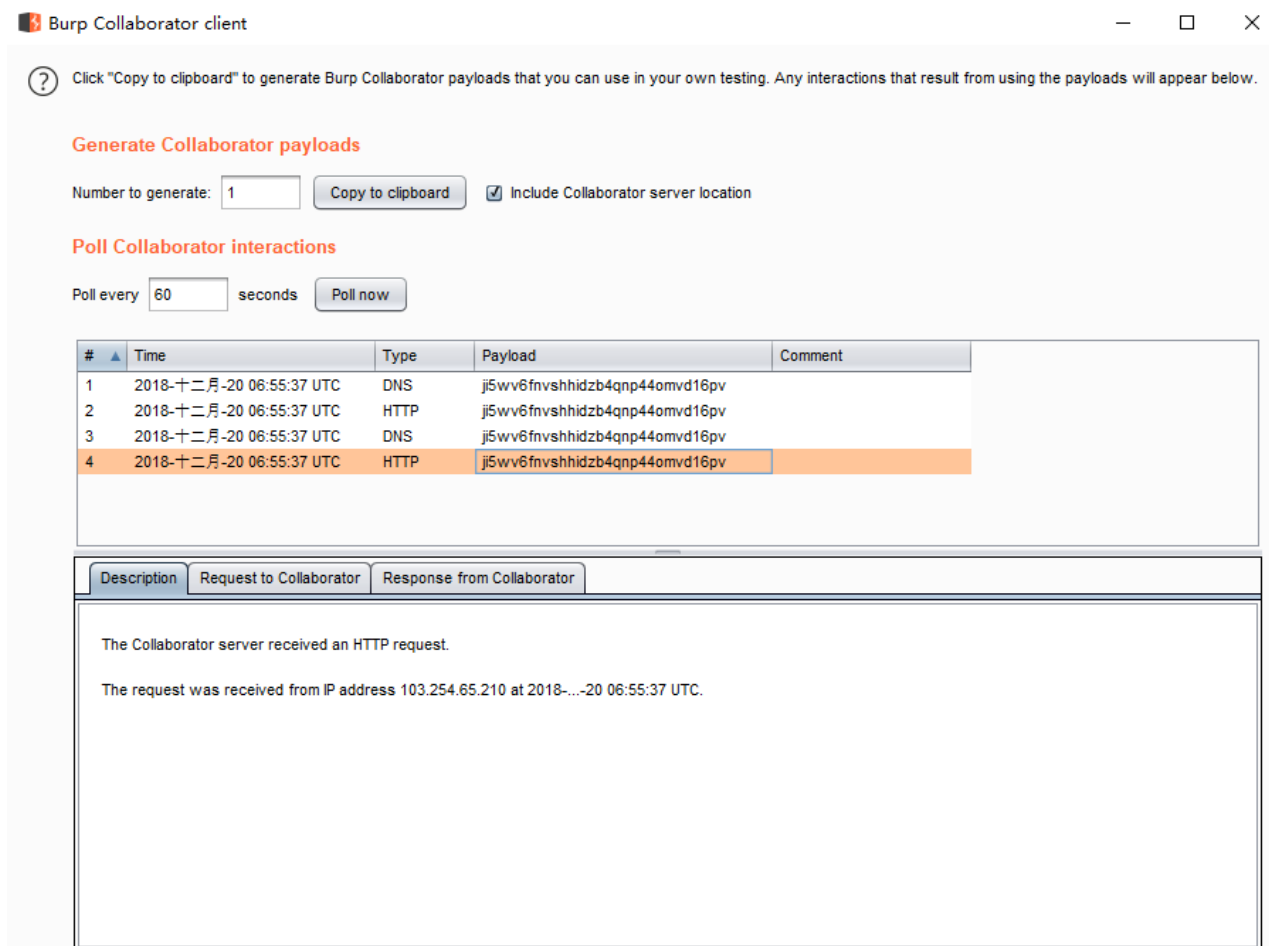
- **带内攻击**
在一次攻击当中,只有一条通道，属于in-band（带内）攻击。常规的web测试模型就是我们向目标发送payloads，然后分析目标返回的数据。
- **带外攻击**
现在同一次攻击下，不止一条信道，则属于out-band（带外）攻击。与外部服务交互行为发生在一个payload提交到目标应用上，导致目标通过某个网络协议和一个外部的域名进行信息交互。和ssrf攻击类似，让目标与Collaborator交互

首先通过project options - misc - burp collaborator server - run health check 检查信道是否通畅



启用 主界面菜单项 burp - burp collaborator client 即可启用

点击Copy 头clipboard，添加到payload中



带外信道根据不同场景一般用如下几类：

1. burp 自带的 collaborator。主界面菜单项 burp - burp collaborator client 即可启用，可以在 project options - misc - burp collaborator server - run health check 检查信道是否通畅。无需第三方服务、不用注册，即开即用。由于我 burp 不离手，所以，这种带外信道方便、集成度高是它的最大优点，另外，burp 进行各类盲注（XXE、SQLi、CMDi 等等）的主动扫描时，也会用到 collaborator；
2. 用脚本语言快速启用 web server。运行 `python3 -m http.server 8653` 或 `php -S 0.0.0.0:8088` 后，所有对它的 GET、HEAD 请求几类都能在日志中查看到。这种方式非常适用于攻击端与目标同在内网的场景，比如，无公网环境的 CTF 竞赛。不支持 POST 是它最大的短板；
3. 借助第三方服务（<http://ceye.io/>、<http://requestbin.net/>）。效果上，类似第二种方式，但支持 DNS 查询、HTTP POST 方法，此外，由于这是长期有效的服务，所以，很适用于那些非实时触发的带外访问请求（如，二次 SQLi）。你可以隔两三天再去看有无访问记录。

0x15 插件(非BApp Store中的插件)

- HackBar ，火狐的hackbar插件收费了，几个替代品都不怎么好用，这个插件功能比火狐的多且强大
- reCAPTCHA自动识别图形验证码并用于burp intruder爆破模块的插件
- knife增加了几个右键菜单，更新cookie、更新header，集成hackbar、sqlmap、u2c（将unicode转换成中文）

- domain_hunter 域名收集
- HTTPHeadModifer修改header头，伪造源IP、UA、添加Origin、更新cookie
- chunked-coding-converter 安全客有一篇文章《利用分块传输吊打所有WAF》
- jsEncrypter加密传输爆破