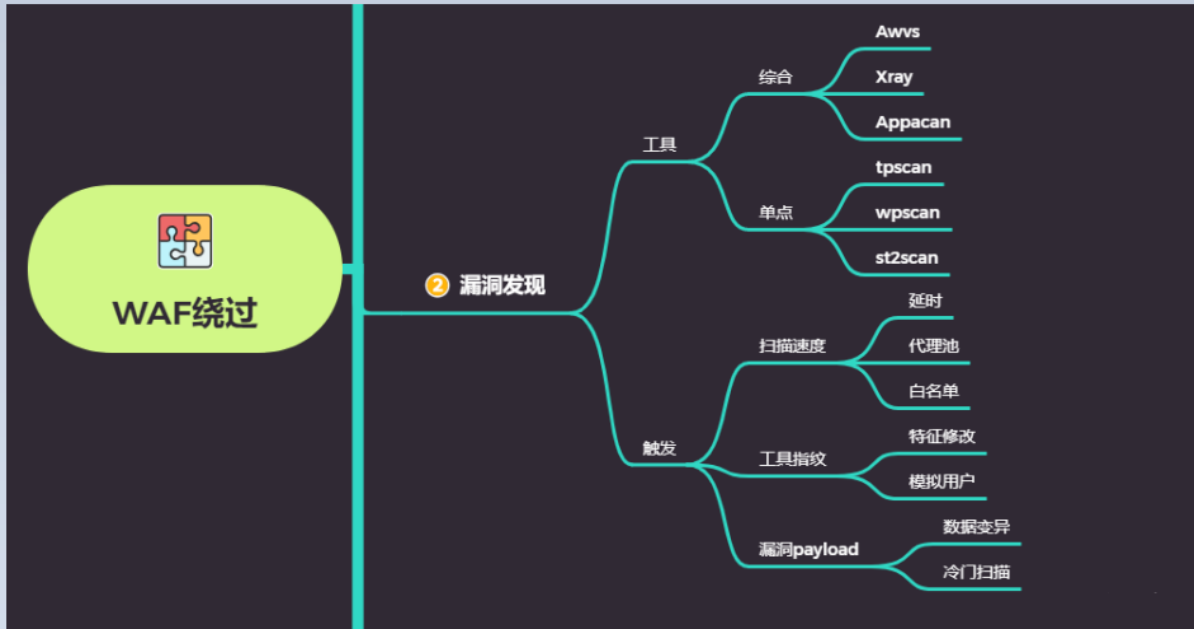
Day47　WAF绕过–漏洞发现之代理池指纹被动探针



47.1 漏洞发现触发WAF点-针对xray，awvs等

```
1.扫描速度（绕过方法：代理池，延迟，白名单）
2.工具指纹（绕过方法：特征指纹，伪造模拟真实是用户）
3.漏洞payload（绕过方法：数据变异，数据加密，白名单）
```

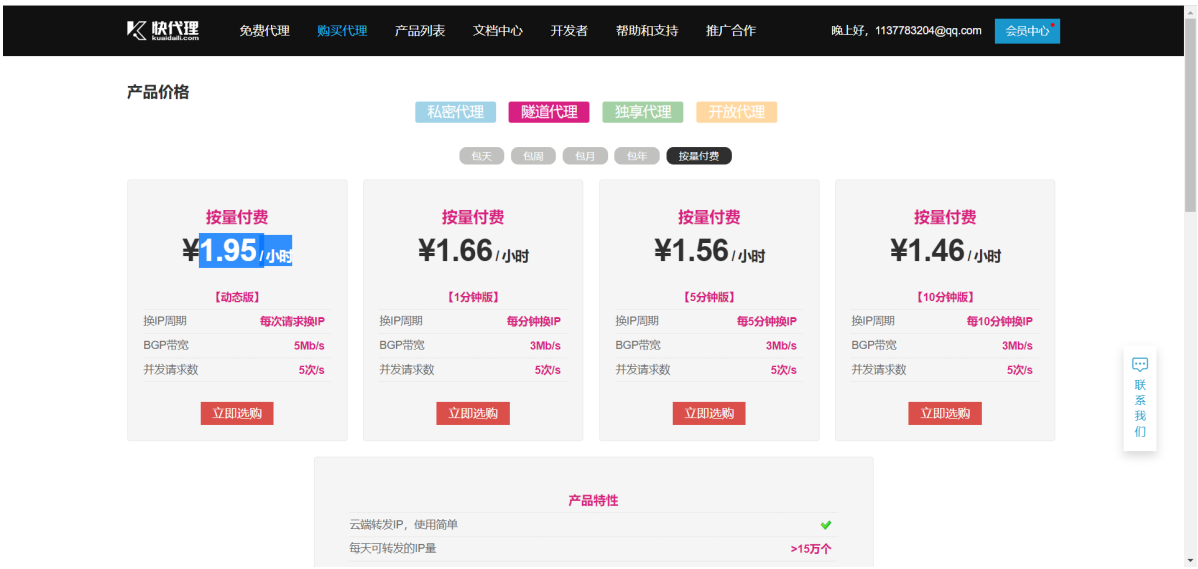
47.2 案例1：代理池Proxy_pool项目搭建及使用解释

- 地址：**https://github.com/jhao104/proxy_pool**
- 需要安装Redis数据库
- 安装依赖： pip install -r requirements.txt
- 因为太麻烦了，这里就不弄了，可以去b站上搜索教程

47.3 案例2：充钱代理池直接干safedog+BT+aliyun探针

免费代理不好用，那我们就买代理，建议选择隧道代理，每次请求都会换IP。

```
1  快代理：https://www.kuaidaili.com/
```

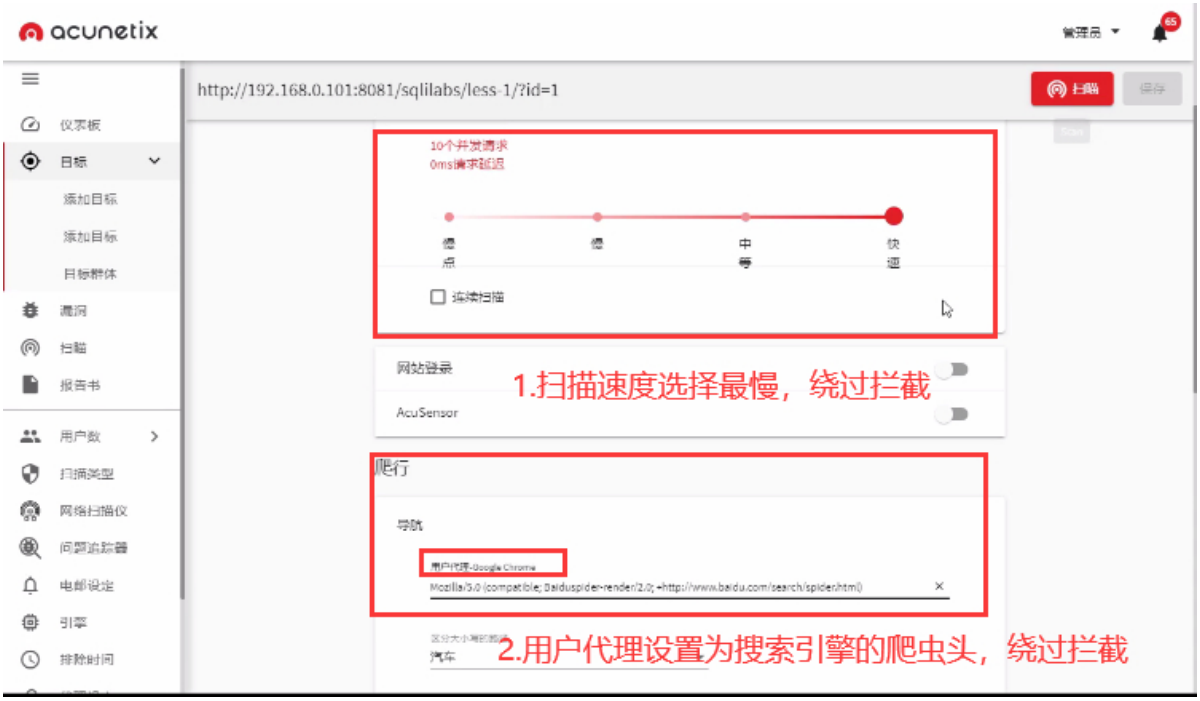


使用上次的脚本，配置收费代理：


```
1  使用网站：【http://httpbin.org/ip】检测访问其他网站的原始ip是哪个
```

47.4 案例3：safedog-awvs漏扫注入测试绕过-延时,白名单

1. awvs扫描sqli-labs（绕过安全狗），设置最低速或者修改AWVS指纹头
2. 控制速度，可以转发给burp suite 然后控制BP放包速度，（手动，或者按键精灵）

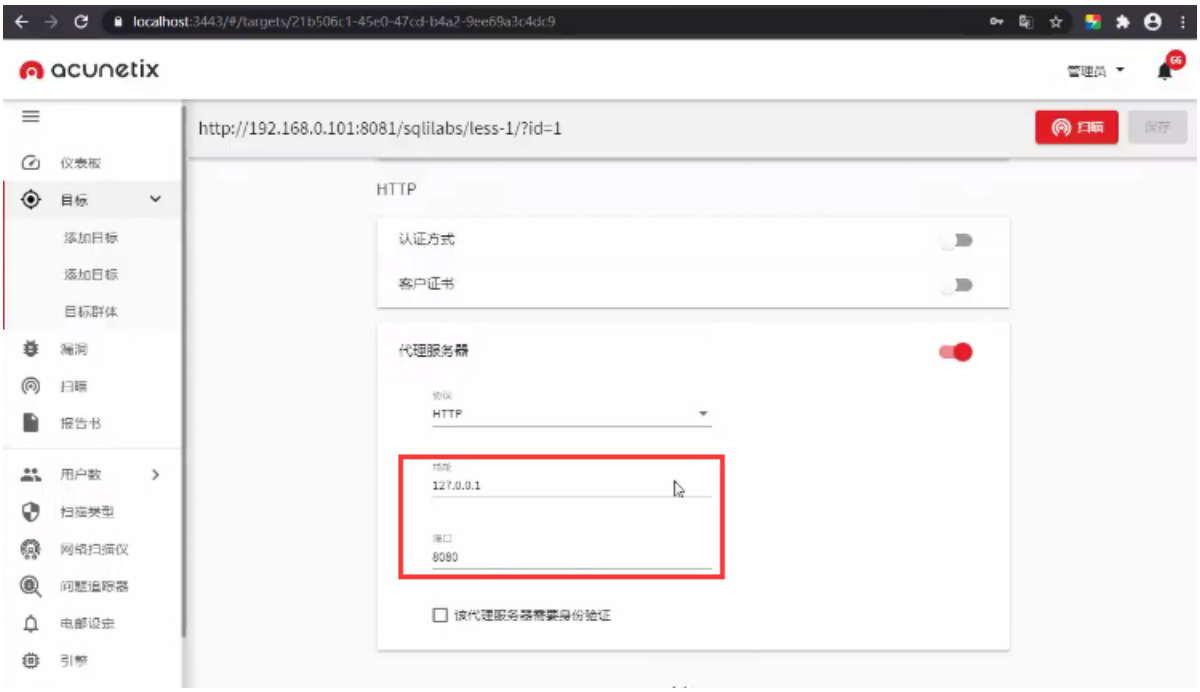


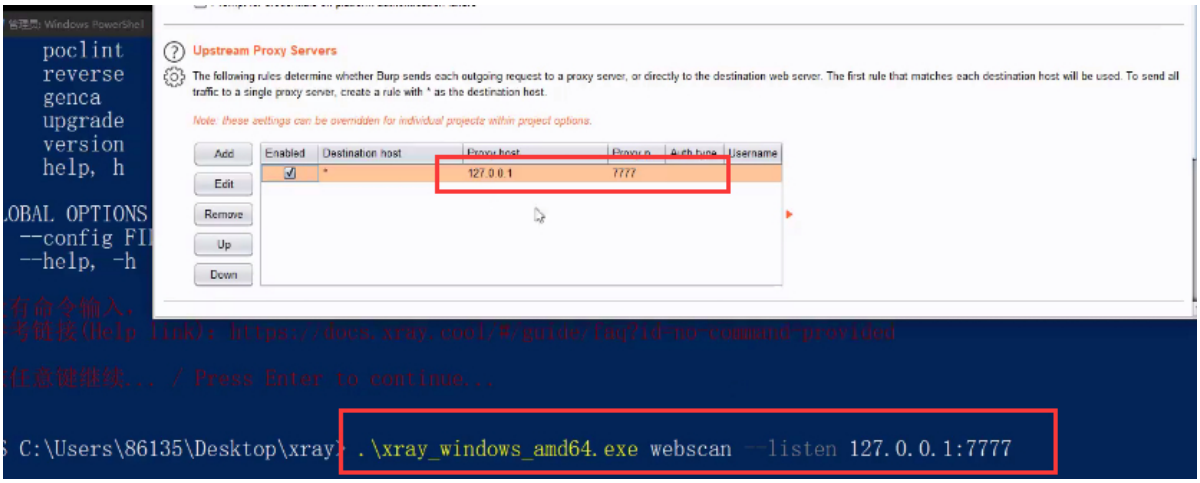
47.5 案例4：Aliyun_os-awvs漏扫注入测试绕过-延时白名单

```
注意：不是每个工具都可以控制扫描速度和改变指纹头，比如xray貌似就不行。

此时，可以对工具使用代理，将数据包发送到burpsuite上，人为的对每个数据包点击放行（太麻烦了吧），以控制速度。也可以自己写个鼠标点击器，代替人为点击。

对于漏洞payload触发WAF这种情况，我们怎么绕过呢？

我们举个例子，由于每个工具判断注入点的方式不同，假设awvs通过and 1=1判断注入，xray通过or 1=1判断注入，那么当waf对and 1=1进行拦截时，我们就不能使用awvs来扫描了，因为扫描不出结果，但是我们可以换xray扫描，这样就绕过了waf拦截，所以实际操作时，我们可以多换几个工具进行扫描。（方法1）

此外还可以使用冷门扫描工具，因为工具冷门，所以漏洞验证和指纹都不会被WAF采集到，可以绕过。（方法2）

使用开源的或者自己编写的工具扫描时，可以针对触发WAF点进行数据变异，从而绕过拦截。（方法3）
```

47.6 案例5：BT（baota）-awvs+xray漏扫payload绕过-延时被动

```
awvs扫描，配置代理，将数据包发送到burp，burp配置代理，将数据包发送给xray，实现三者联动。
此时如果awvs控制了扫描速度，那么xray也会被动延时。
```





47.7 案例6：充钱代理池干safedog+BT+AliyunOS漏洞发现

资源：

```
http://httpbin.org/ip
https://www.kuaidaili.com/
https://github.com/jhao104/proxy_pool
```