<https://cloud.tencent.com/developer/article/1763983>

# <https://github.com/gh0stkey/HaE> burp插件

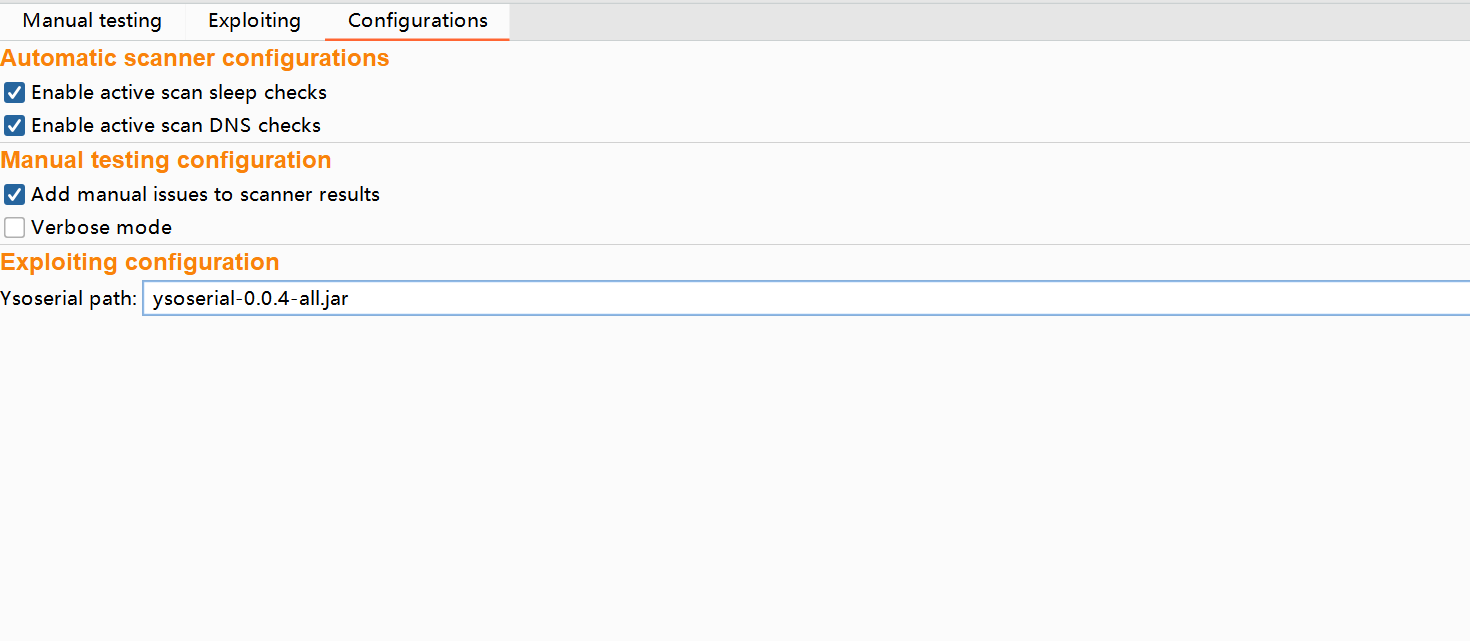
# Burpsuit 扩展

Java反序列化测试：

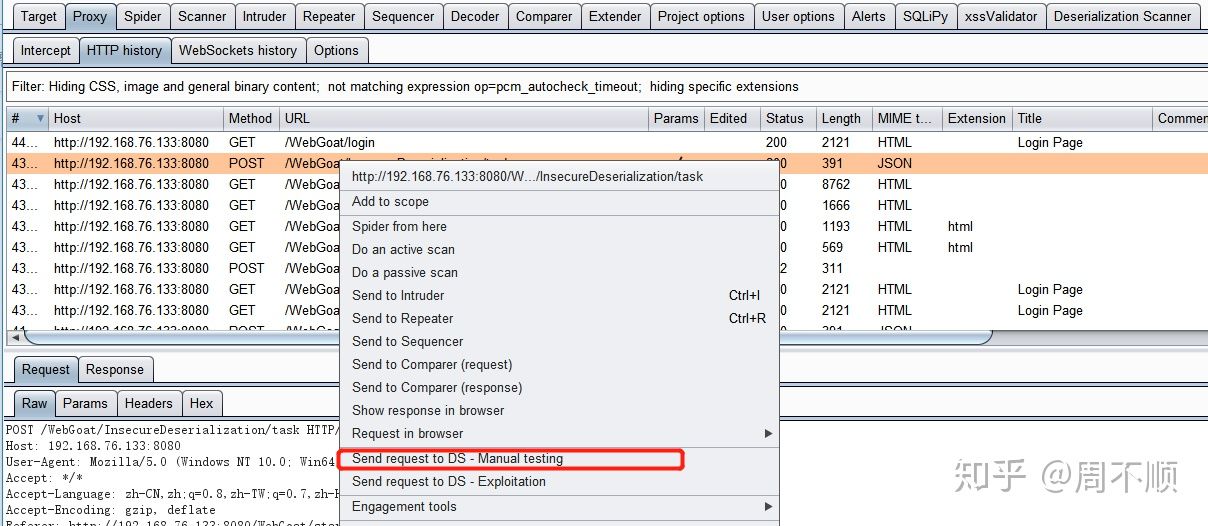
<https://github.com/federicodotta/Java-Deserialization-Scanner>

可在插件商店中直接安装

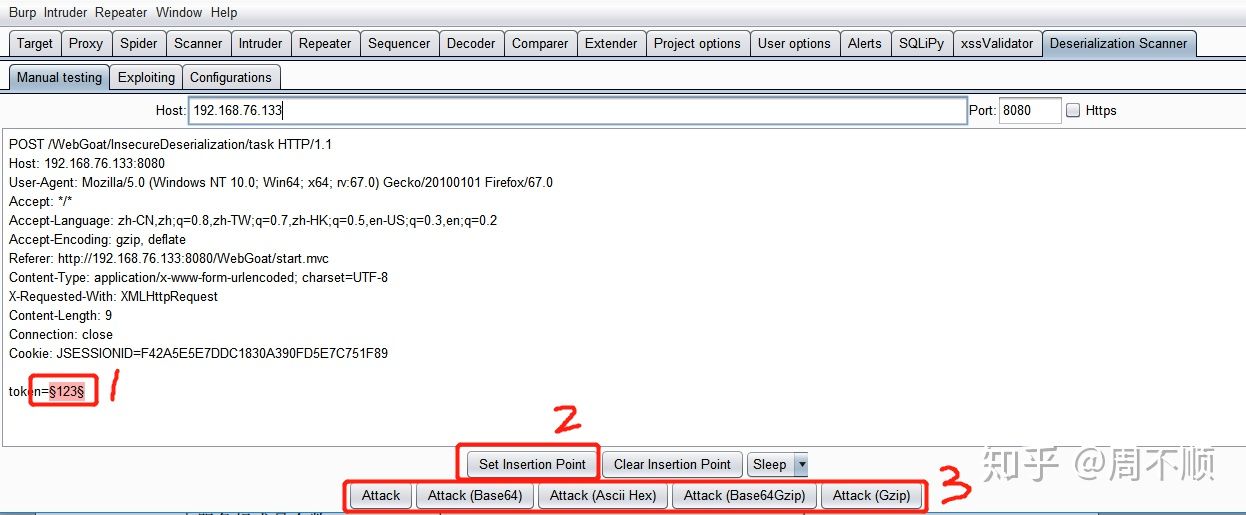
安装后需要配置ysoserial、https://github.com/frohoff/ysoserial 路径



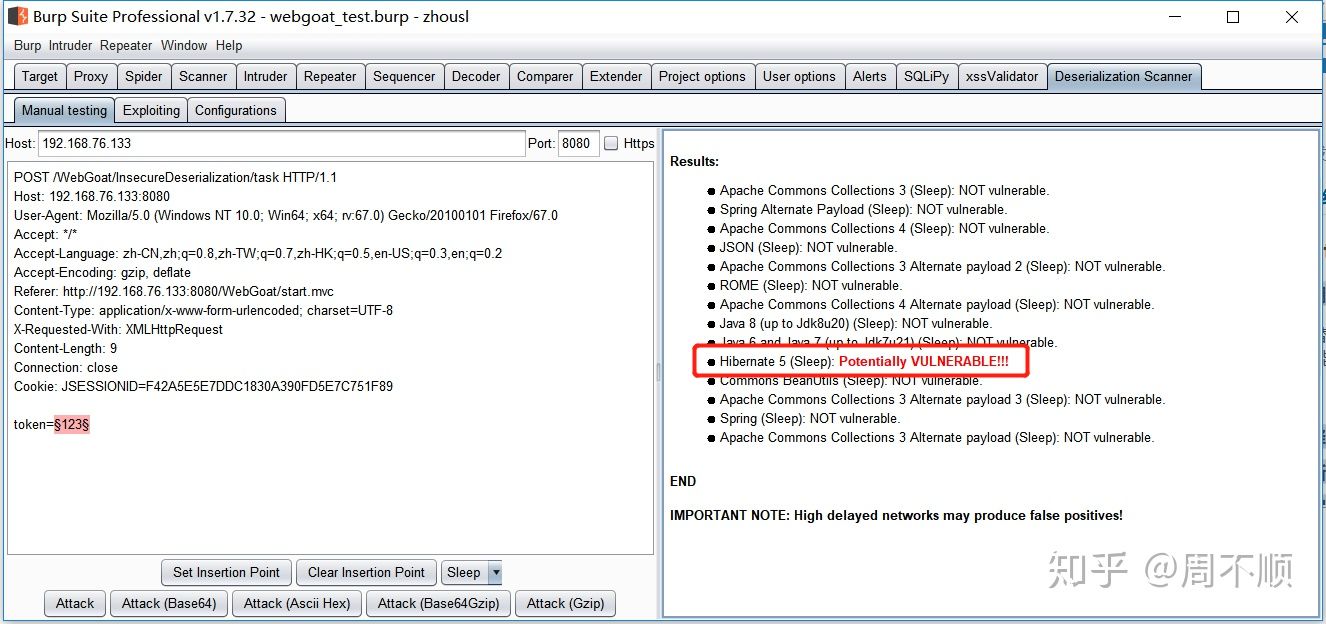
***学着使用burpsuite的Java Deserialization Scanner插件。***



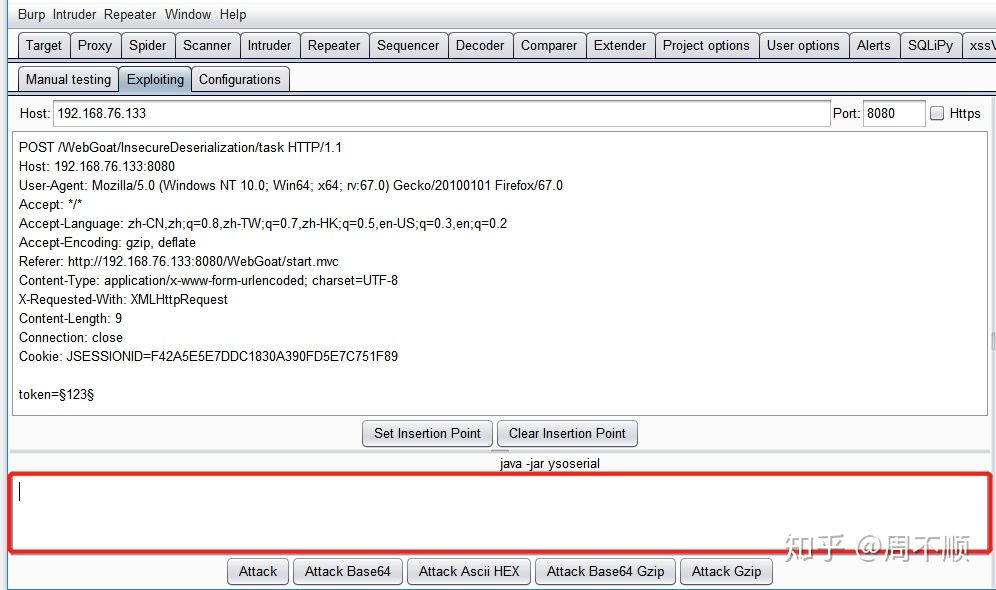
将需要测试的请求发送到manual testing，然后配置需要测试的参数，并根据参数的处理进行攻击。



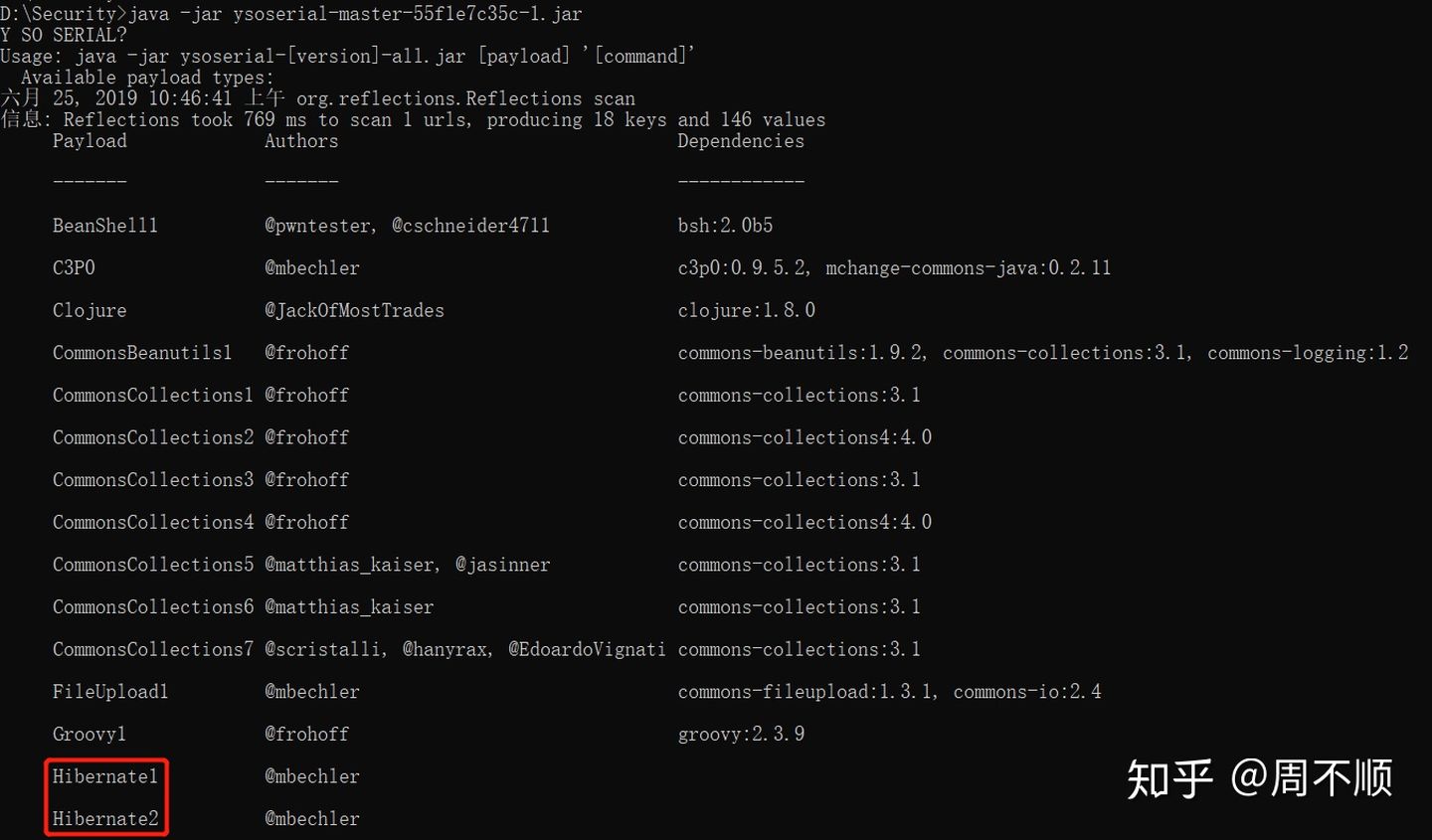
选中测试，点击set point，选择选择判断漏洞是否存在使用的方式，有 DNS（依赖于burpsuite的Collaborator功能来获取响应）, Sleep （如果有漏洞让服务器sleep 几秒钟）， 还有 cpu模式。这里选择 Sleep模式，如果只是裸rawpayload就是直接attack，如果是经过base64以后的就选择attack(base64)方式。可以在又侧看到结果。



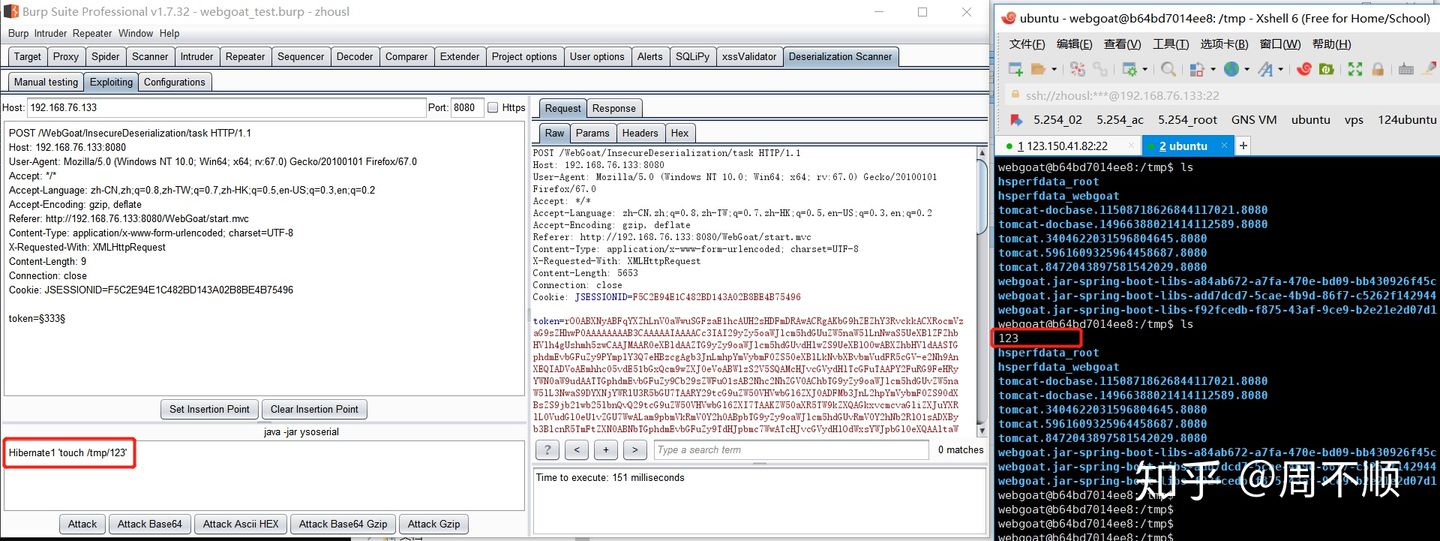
然后右键发送这个请求 Send to Exploitation tab：



在红框里面手写反序列化利用的payload，能看到生成payload用的就是命令 jar -jar ysoserial xxxx 这个命令，能支持的参数需要手动到命令行确认一下：



构造的payload指定为Hibernate1，指定command为touch '/tmp/123' 创建一个文件，这样好确认是否利用成功：



能看到webgoat容器里面的tmp目录下成功的生成了一个123文件，说明反序列化漏洞利用成功。