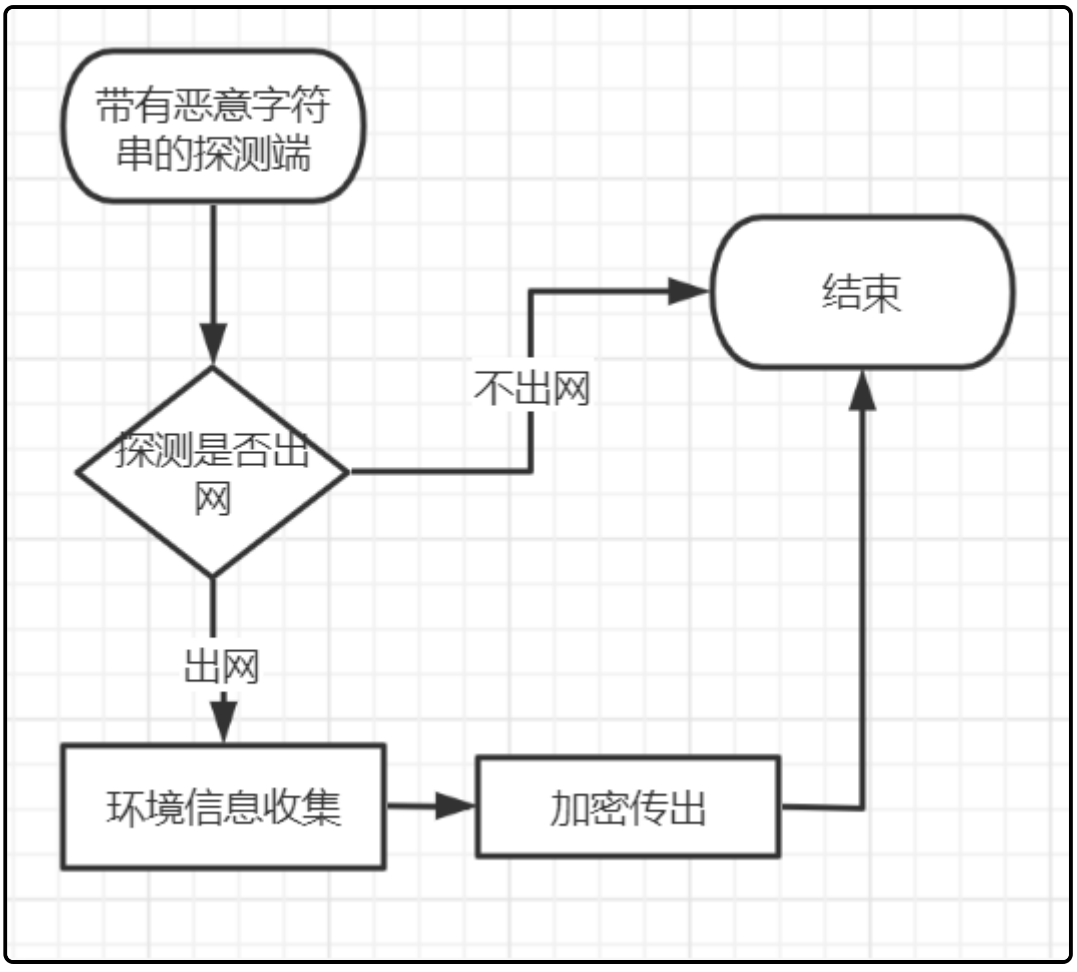
动态云沙箱的原理

在现代攻防实战中，云端的动态沙箱往往在恶意软件识别中占有重要地位，作为红队也经常头疼于刚写好的免杀客户端就被云沙箱抓到恶意特征。在学习了多个反沙箱开源项目的代码后，发现大部分的反沙箱项目都是针对沙箱环境的特征进行排查，收集足够多而准确的沙箱特征是重点，便产生了动手写一个针对动态云沙箱的主动探测 demo 的想法，主动的去探测并且向外传输云沙箱的环境特征。



如何实现 demo ?

首先，现代沙箱都是 纯黑盒程序 ，扔进一个文件，输出一个结果，而且还在云端部署，无法对其进行逆向分析定位特征，只能通过扔进去运行的程序主动收集并传出相关环境信息。

这个 demo 共分为七个部分：

第一个部分是检查沙箱的出网情况，由于本 demo 的初衷是为了研究C2客户端的反沙箱方式，所以对于内部不出网的云沙箱直接阻止程序运行。

第二部分为沙箱基础环境收集，最开始是锚定了：用户名、文件名、进程数、主机名、C盘大小四个因素，后来又加入了绝对路径的探测。

注：在与群友沟通后，又提供了@xrayteam（dns 缓存，arp 缓存探测）；@haya （麦克风探测）的想法，感谢！

第三部分是信息传出，对收集的数据进行异或和 base64 后以 http(s) 请求发送到指定的 server 进行接收，此处为了方便采用的是 GET 请求方式。

第四部分为 server 端，这里写的比较粗糙，采用 flask 框架，主要功能是对于第一个出网探测包进行回应以及接受传出数据并写入文件。

第五个部分是对于文件数据的解码脚本，把密文数据解码为明文并转换为更易于分析的 json 格式。

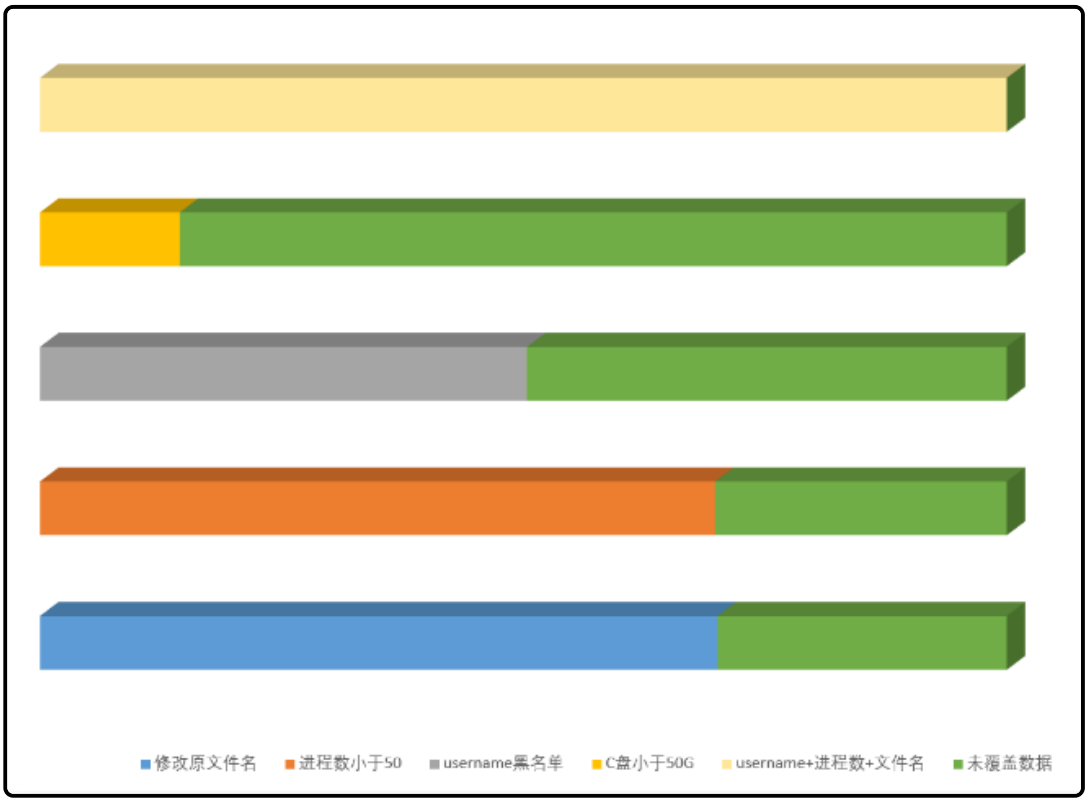
第六个部分为数据分析，这也是最为重要并且目前最欠缺的部分，由于个人能力的限制，只做到了单类型信息的提取，并未做到信息之间的关联分析。此处后续可借鉴威胁情报中的 IOC 的部分分析思路。

第七个部分为恶意诱饵，在检测器中硬编码恶意 shellcode ，作用是引诱云沙箱对样本进行主动运行。

把 demo 完成后扔到各大沙箱，virustotal，各大杀软云检测平台进行检测并等待数据传出，在测试几天的过程中一共收到了401个沙箱环境样本去重数据。

注：这一步骤是可以做成自动化的，即定时任务自动生成恶意样本，自动对接api提交样本，自动提取特征文件并去重添加，由于时间仓促并未能实现。

粗略的特征提取过滤结果如下图所示：



https://github.com/timwhitez