实战介绍Windows下的PC客户端常见漏洞挖掘

FB客服 发表于 FreeBuf

0X00 为什么写这篇文章

对于小白来说，WEB安全方面似乎已经有了很完备的知识体系和漏洞发掘流程，刚刚入门的朋友总是喜欢选择web方向来作为自己的发展方向，因为针对web系统的渗透测试似乎获得的成就感要更高，也有很多小白认为web似乎更好学，然而对于PC客户端漏洞发掘，因为涉及到了一些计算机和操作系统底层的知识，很多人都不敢去碰，而实际上PC客户端的漏洞比大家想象中要容易的多，甚至你并不需要精通汇编语言就能很容易的挖到PC客户端漏洞，不过汇编语言是PC客户端漏洞发掘的基础，最好还是学好它。

另外，挖掘PC客户端漏洞和挖掘WEB漏洞是一样的，都需要细心和耐心，你要学会关注每一个细节，了解系统和软件是如何协同工作的。本文主要讲Windows下的PC客户端漏洞发掘，为了浅显易懂，不涉及ROP等高级内存攻击和内存溢出技术，大佬请绕道。

0x01 工具

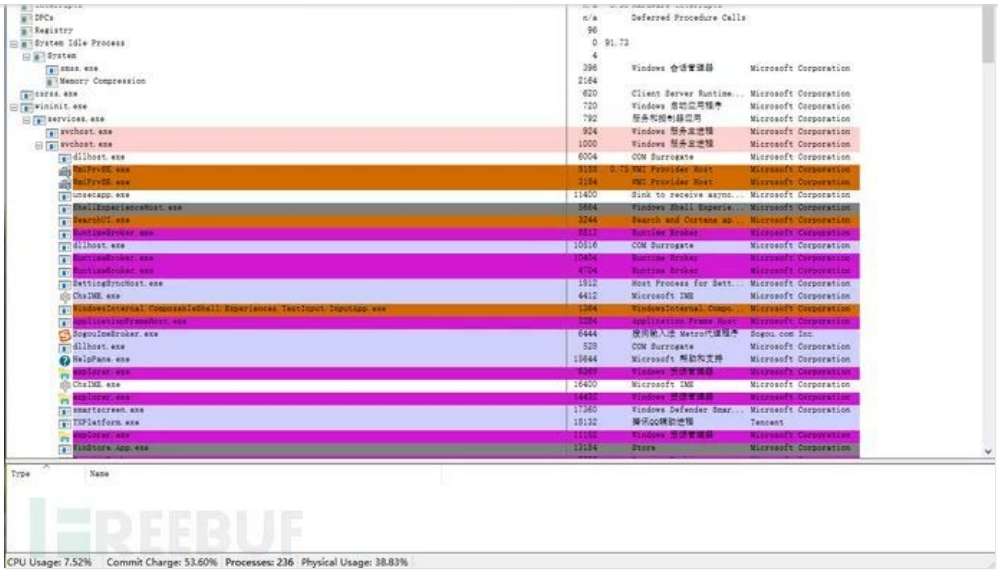
> “工欲善其事，必先利其器”。

PC客户端漏洞挖掘主要是逆向工程和进程监控为主。

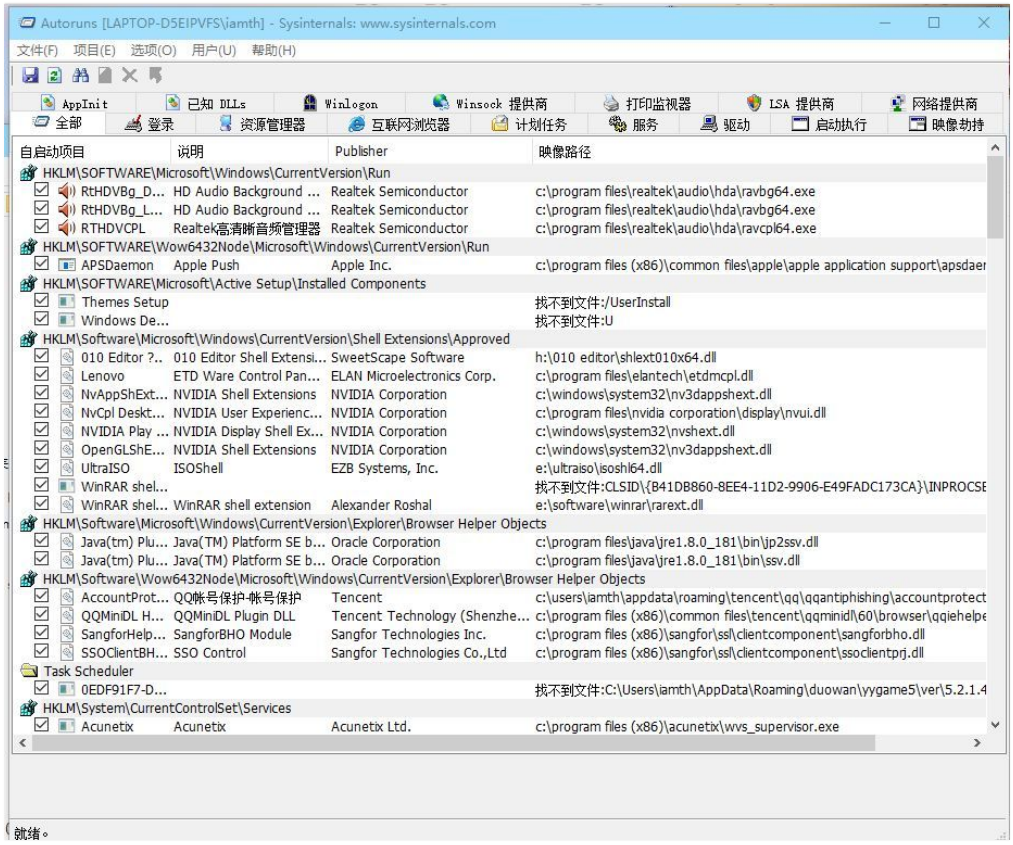
逆向工程方面我推荐两个工具，一个是静态分析之王：IDA pro，另一个是动态调试里面最好用的：Ollydbg（推荐大家用吾爱破解论坛版本的）这两个逆向分析工具一查就可以查到，在这里就不多介绍了。

进程监控工具主要分为进程**本地行为监控**和进程**网络行为监控。**

本地行为监控工具我推荐：**ProcessExplorer进程监控**和**Autoruns进程监控**，这两个工具知名度不高，但是很好用。



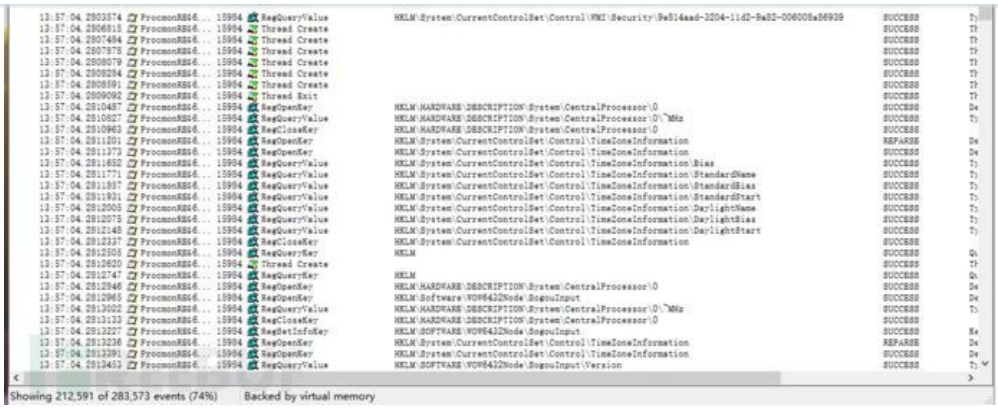
ProcessExplorer进程监控



Autoruns进程监控工具

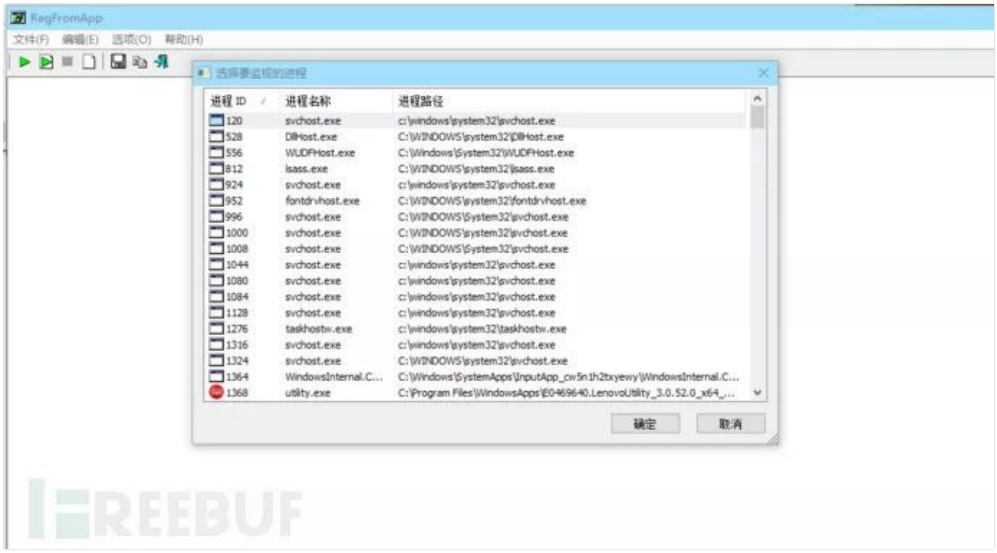
本地监控工具里还有一种工具是专门监控注册表的工具，这里推荐几个：

实战介绍Windows下的PC客户端常见漏洞挖掘



Process Monitor：一个强大的注册表监视工具，可以添加过滤规则，很方便。

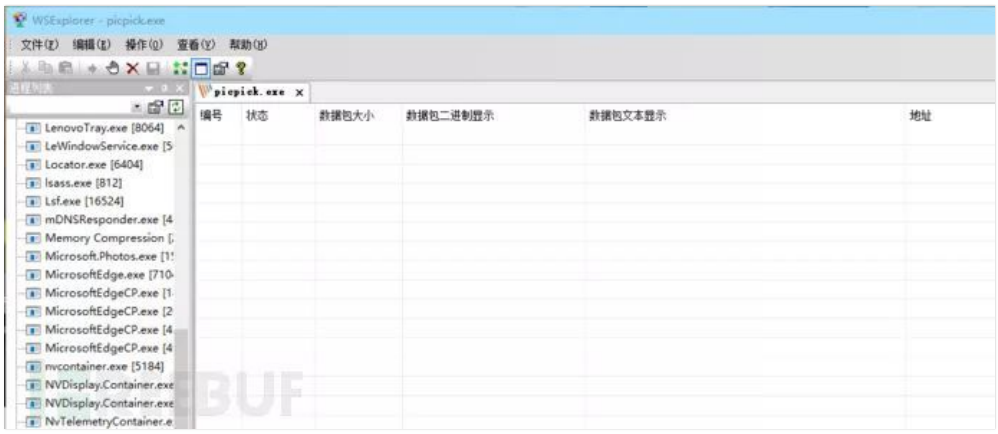
Regshot：一个注册表备份和比对工具，可以通过保存快照和比对快照的方式来找出注册表中哪些值发生了变化。



RegfromApp：也是一个进程监控工具，可以选择一个进程之后跟踪其对注册表的修改。

网络行为监控工具当然首推大名鼎鼎的Wireshark啦，当然还有一个工具很小众但是很好用，是岁月联盟的工具，叫WSExplorer(进程抓包)。

Wireshark



岁月联盟的进程抓包工具WSExplorer，非常方便，左侧是进程，右侧是抓到的数据包。

有了以上这些工具，我们便可以对程序在我们的计算机上做了些什么了如指掌，知己知彼方能百战百胜，便可以开始下一步的漏洞发掘了。

0x02 缺陷

对于开发者来说，开发一款完全没有漏洞的程序是不可能的，特别是这个程序的体量及其庞大时，则其必定存在漏洞，我们需要知道的就是哪些位置容易出现漏洞；

客户端的授权认证漏洞：

一般正版的客户端软件都设有授权认证模块，这些授权认证方式所需要达成的目的无非就是“买了的人能用，没买的人不能用”，一般验证采用注册码的形式并与个人计算机的机器码相互绑定，或者与某种个人认证机制相互绑定，以达到验证的目的。授权认证漏洞可以导致软件和功能被破解，盗版程序流通等严重后果。授权认证漏洞往往是开发者在开发时没有注重授权认证的保密性以及安全性所导致的。

客户端的网络服务漏洞：

这类漏洞一般是由于客户端在发送数据包或接收时没有进行严格的认证造成的，可导致无条件调用高级权限的服务。

客户端功能逻辑漏洞：

这类漏洞一般是由客户端功能设计不合理导致的，可以导致无授权的访问等严重后果。

客户端溢出漏洞：

这类漏洞包含属于逆向工程中比较高难度的一块，主要是由于开发时对内存的错误管理，或者程序本身的执行逻辑漏洞导致的。

本文仅介绍前三种漏洞。

0x03 实战

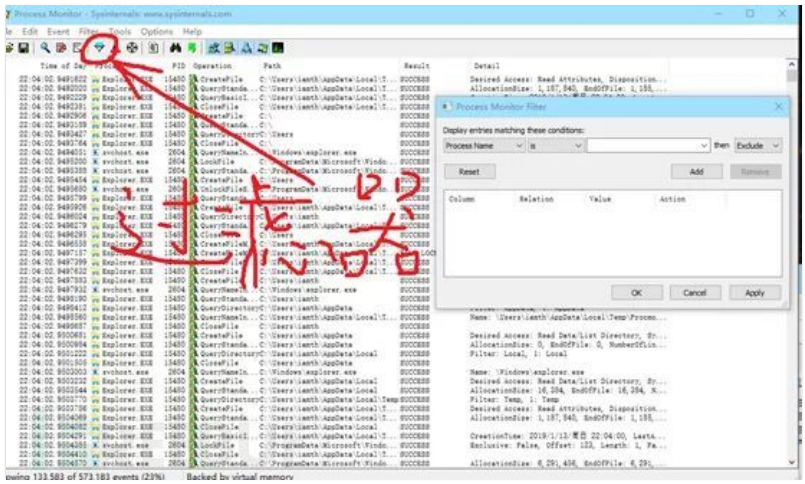
漏洞也包括在反编译之后的部分，一般情况下通过修改关键call函数之上的跳转逻辑来进行漏洞挖掘，主要成因是客户端逻辑过于简单，采用了较少的逻辑判断。这种漏洞在具有完备功能的客户端上并不常见。

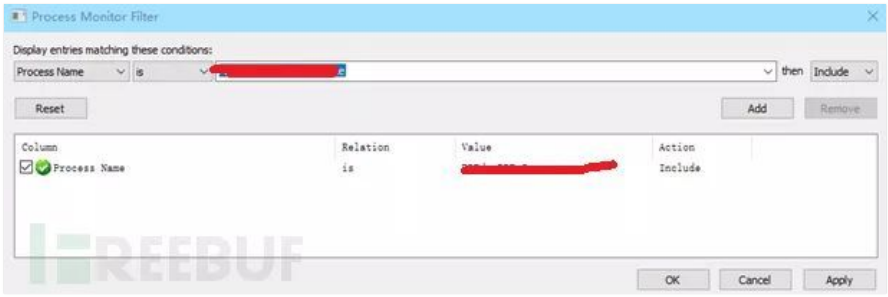
客户端的授权认证漏洞：

1.基于本地注册表的破解：

某些软件虽然使用网络进行授权验证，但是由于其试用次数设计的验证缺陷，可以导致通过修改注册表来实现无限次数的试用，导致“不付费也能用”，即出现了授权认证漏洞。下面这款客户端程序即是如此，我们在刚刚打开它的时候会提示试用次数还剩29次。

现在我们打开Process Monitor，使用过滤功能添加白名单使Pm仅显示该进程的相关信息。





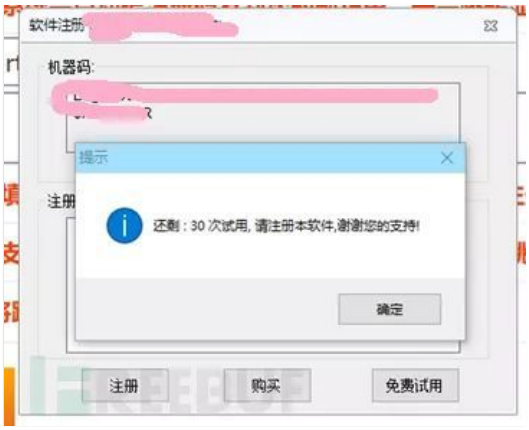
添加过滤白名单，仅显示该进程

关键部位做了处理。。

之后停止所有捕获，关闭并重启客户端，多次重复后我们监控到每次客户端打开时，会自动做一次RegsetValue(注册表值修改)，如下：

了是我们编写一个BAT脚本，修改该客户端指向的那个值，并让他在客户端启动的自动运行，即可锁定试用次数为30次，不会减少。


```
Reg add HKCU\Software\客户端名字\一个位置 /v Nowtimes /t REG_DWORD /d 0 /f
```



这个漏洞的成因主要是因为试用次数认的方法太简单了，不联网不加密直接写进注册表中，并且键名还那么浅显易懂叫做“Nowtimes”，这种存在在注册表的漏洞发掘和利用方法还是比较简单的，但是问题是这样的漏洞还蛮多的，所以大家在挖掘时注意关注注册表。另外如果注册表禁止监控，我们可以用REGshot来保存前后的快照进行比对分析。

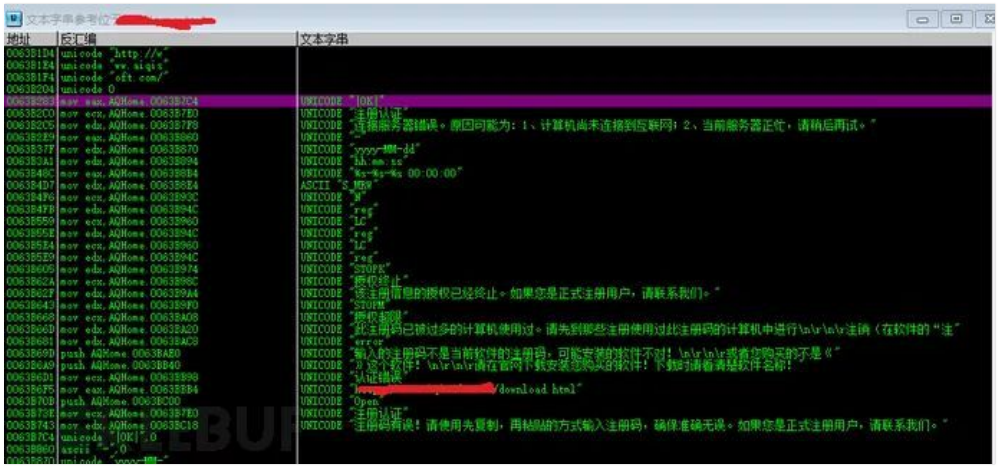
2.基于网络授权验证的hosts欺骗破解

这一部分内容需要用到一部分逆向工程的知识。这次破解的客户端没有设置试用机制，所以第一种路子行不通 ，我们转而把眼光放到它的网络验证模式上来，看看到底这个客户端的网络授权验证方式是如何工作的。

关键位置打码处理

随便输一个注册码进去

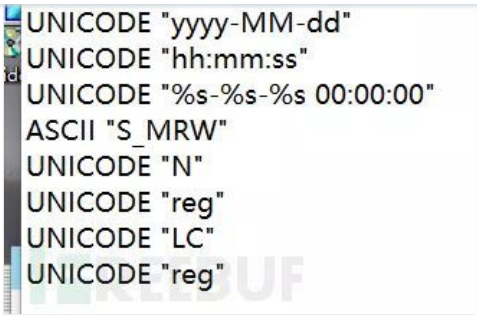
随便输入一个注册码然后确定，根据弹出的错误窗口来定位到客户端的注册检测验证的函数处。拖入OLLYdbg查找字串“注册码有误”，并跟踪到汇编窗口。



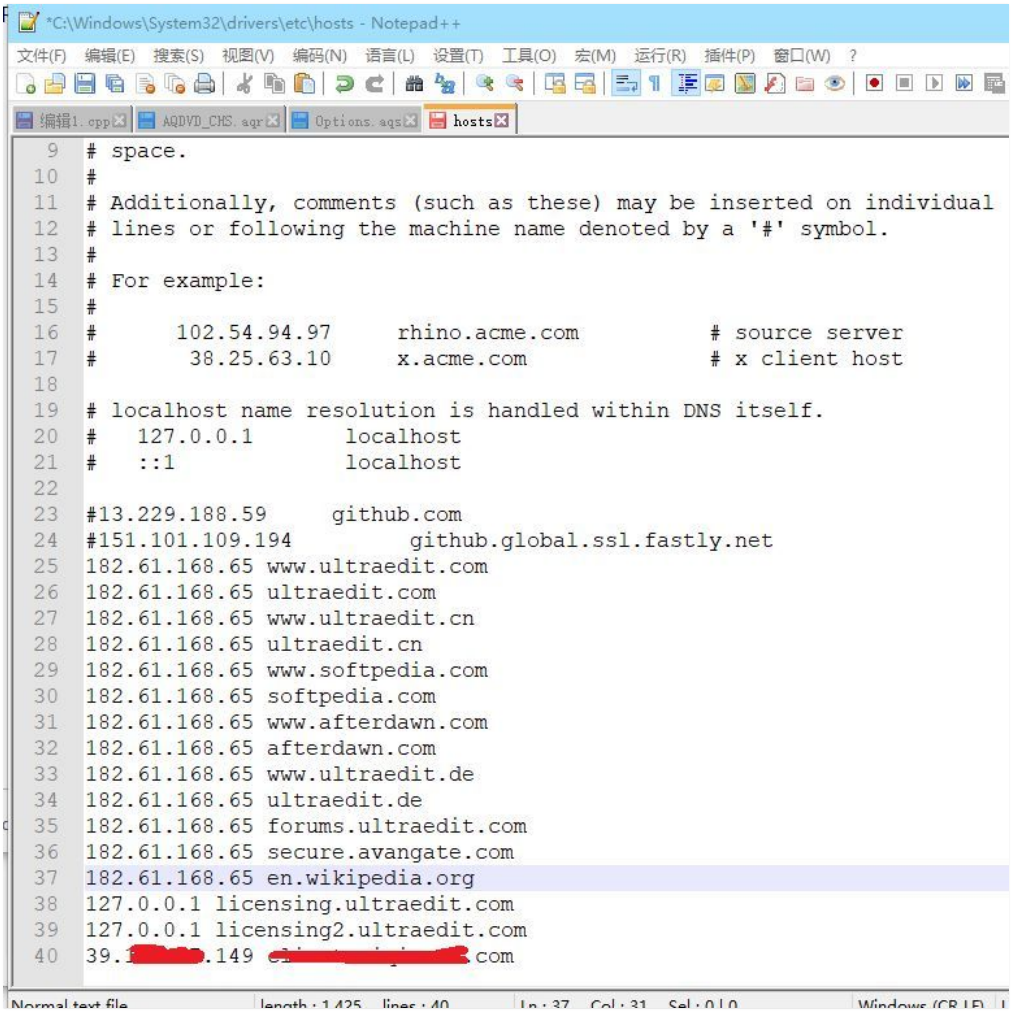
于是我们得到了注册授权的服务器地址。为了进一步验证，我们使用wireshark来分析这个客户端注册时的网络请求。

可以看出客户端携带着我们的机器码和几个其他数据请求了服务器的/verifycheck/login.php

再回到我们的汇编窗口中，我们可以看到几个unicode的编码，疑似服务器的返回，记录下来。

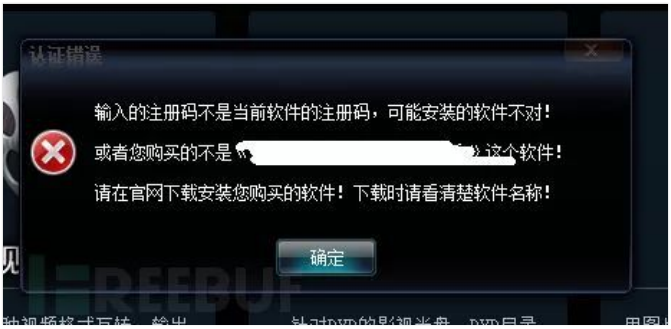


直接用浏览器访问，可以发现返回值和汇编窗口的记录值中的一条相同，所以我们猜测可以构造一个假服务器，修改主机的hosts文件来实现请求重定向，让我们的服务器返回注册成功的信息。



修改hosts文件，将服务器域名绑定到我们自己的假服务器的ip地址

在服务器上构造不同的payload，可以得到客户端不同的反应，说明漏洞成功了一半。



至此我们可以排除掉其他的payload，从而确定一个格式化日期返回值是注册成功的标志。所以我们构造一个格式化时间，并且重新打开客户端输入任意注册码注册，即可看到注册成功的窗口。

最终的payload

这一漏洞的成因为客户端软件在校验注册码返回时的数据太过简单，进而非常容易构造注册成功的返回。并且客户端的反编译能力也非常差，敏感信息在反编译后直接就能够看到。在挖掘这一类漏洞时，我们需要一些逆向工程的基本知识，以及计算机网络的一些基础知识，重点关注客户端与网络服务器之间的通讯数据，利用抓包工具来进行漏洞挖掘。

客户端的网络服务漏洞：

由于想拿来做例子的漏洞厂商还没有修复，所以这里不放例子了。网络服务漏洞发掘主要采用客户端网络请求分析的方式，主要的工具就是前面提到的进程抓包工具以及Wireshark, 大部分网络服务漏洞起因是由于在客户端的网络请求中没有采取验证方式或者采取了安全性非常低的验证方式，从而使得任何人都可以以客户端合法的名义来请求这个网络服务，实现没有权限的调用私有网络服务接口。常常出现在客户端vip付费资源的试听服务，客户端付费的查询功能接口处等，不安全的客户端请求可以被拦截并分析，进而实现越权调用无权限的资源或接口。在挖掘这类漏洞时，我们需要

实战介绍Windows下的PC客户端常见漏洞挖掘

*本文作者：18615253400，转载请注明来自FreeBuf.COM

原文发布于微信公众号 - FreeBuf（freebuf）
原文发表时间：2019-05-21
本文参与腾讯云自媒体分享计划，欢迎正在阅读的你也加入，一起分享。
发表于 2019-05-23

上一篇：Ubuntu16.04下CUDA的安装和卸载
下一篇：Ubuntu18.04下安装CUDA