近期区块链领域黑客攻击事件频发，其中有一个很让Cocoa感兴趣的就是Wintermute 钱包因靓号地址的问题损失约 1.6
亿美元，具体说来话长，可以参见慢雾的这篇分析。

0.背景简介

关于什么是靓号地址这里简单解释一下，以太坊钱包的地址是一个由0x开头40个随机字符的16进制字符串，比如V神的钱包地址就是0xAb5801a7D398351b8bE11C439e05C5B3259aeC9B，很难记忆，也没有个性特色。

因此有人就别出心裁，通过暴力枚举的方式，试图找到好看的靓号地址，比如0x888888开头，或者为了节约合约部署的费用，使用0x00000开头的地址。

Profanity就是这样一个靓号地址生成工具（这里还有个谐音梗，在英文中靓号地址叫vanity address，vanity是“虚荣”的意思，Profanity的本意是“脏话”，但取了antiy的后缀谐音）。Profanity的特点就是使用了GPU，所以比其他工具更快地找到靓号地址。

总之，原项目因为存在漏洞，目前仓库已关闭，而且也不推荐使用，但发现漏洞的1inch帮人帮到底、送佛送上天，提供了船新版本的无漏洞Profanity2，继续满足大家的虚荣心。

但个人感觉1inch安全功夫了得，文档水平太烂，Cocoa贵为期末考试满分的密码学小王子，都看了5分钟才理解到底怎么用。废话少说，以下就是使用教程。

1.编译代码

考虑到私钥的安全性，这类项目建议从官方源码编译使用，不过Profanity2有个创新改进，下面会提。

1inch这次提供的是Linux下编译的代码，在Windows下编译需要改一个地方，主要是把Dispatcher.cpp中的以下代码修改掉。

#include

另外还涉及到OpenCL的SDK以及编译环境搭建的问题，总之这里就假设你已经拿到了可执行程序。

2.本地生成密钥对

Profanity2的一大改进就是将原来直接生成私钥、再计算公钥的步骤，改成：

先生成密钥对（私钥A+公钥A）然后把公钥A放到程序去跑，生成私钥B最后把私钥A和私钥B数学相加，得到私钥C

这个私钥C对应的公钥C就是你想要的的靓号了。

其中的数学原理Cocoa还没深入研究，大致猜测就是先用一个安全可靠的工具（如openssl）生成密钥对，再通过程序暴力试出一个偏差量，使私钥加上这个偏差量可以导出靓号公钥。

这样的一个好处就是Profanity2可以交由第三方或者云端来运行，因为最终的私钥C，是由私钥A和私钥B相加得到的（其中私钥A是你自己本地安全生成的，只要保护好私钥A，别人就猜不到私钥C，因此也就安全了）。

因此首先要生成私钥A和公钥A，官方提供了命令，在Linux下直接执行即可：

$ openssl ecparam -genkey -name secp256k1 -text -noout -outform DER | xxd -p -c 1000 | sed 's/41534e31204f49443a20736563703235366b310a30740201010420/Private Key: /' | sed 's/a00706052b8104000aa144034200/'$'\nPublic Key: /'

上述命令执行完成后，屏幕上便会显示出Private Key和Public Key，分别就是私钥A和公钥A，请注意私钥A一定要保管好。

3.使用公钥A去跑出私钥B

将上面步骤得到的Public Key去掉开头的04也就是公钥A，放到Profanity2中去跑，命令如下：

profanity2 --matching c0c0aXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX -z 公钥A（记得去掉前面的04）

稍等一会就会跑出来一个私钥B，过程和原版Profanity类似。

4.最终计算得到靓号地址对应的私钥C

拿到私钥B（这个公开也无所谓）后，我们只要加上私钥A（这个要保护好），即可得出最终靓号地址对应的私钥C了。

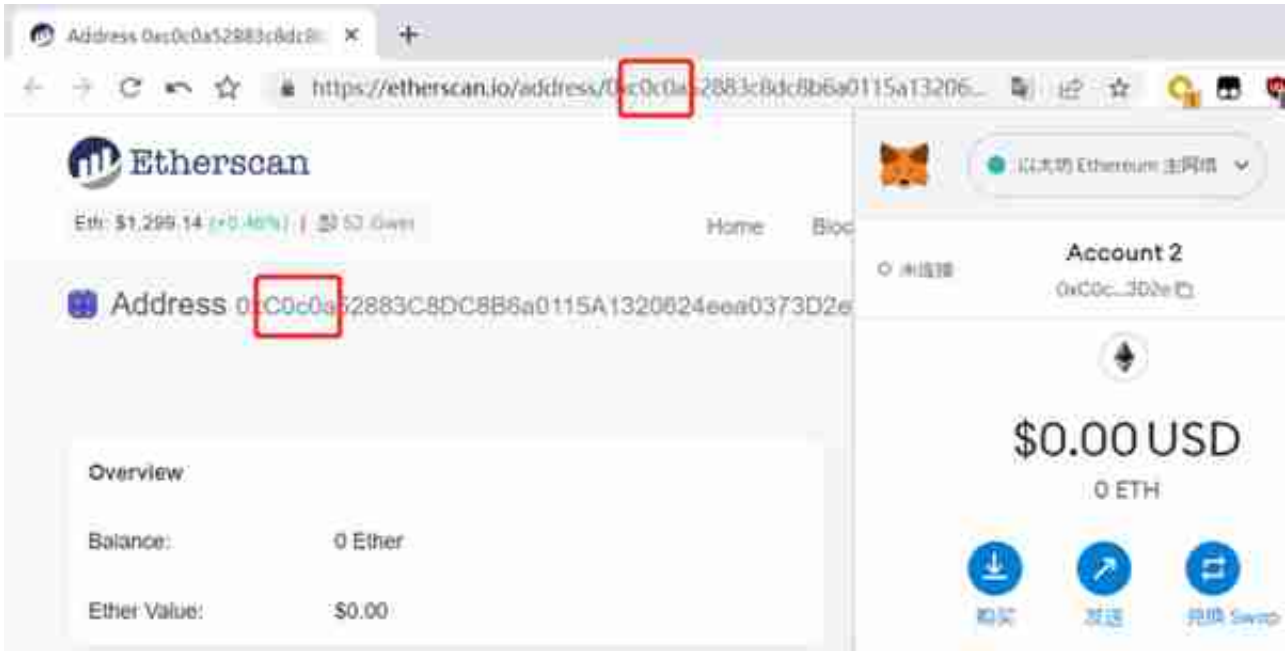
官方给了两个命令，分别是shell的和python的，因为我的kali好像没有bc，所以用了python的那个。其中私钥A记得前面加上0x。（Cocoa吐槽：尼玛那个PRIVATE_KEY_A + PRIVATE_KEY_B 我足足理解了1分钟才知道是数学上的加法）

(echo 'ibase=16;obase=10' && (echo '(PRIVATE_KEY_A + PRIVATE_KEY_B) % FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFEFFFFFC2F' | tr '[:lower:]' '[:upper:]')) | bc

$ python3

hex((PRIVATE_KEY_A + PRIVATE_KEY_B) % 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFEFFFFFC2F)

最后就得到一个0x开头的私钥C，导入metamask等钱包就可以看到我们的靓号地址啦。



附赠章节：漏洞原理简介

以太坊的私钥是32字节（也就是256位）的，但是原版Profanity在生成这个256位的私钥时，仅采用了4字节（也就是32位）的随机数作为伪随机数生成器的seed。