关于逆向入坑

0x0:基础

1. 编程(C语言)
2. 汇编语言
3. PE/ELF文件结构

0x01:资料

1. 《加密与揭秘》(有点老)
2. 《逆向工程核心原理》(挺不错)
3. 《汇编语言》 《intel汇编语言程序设计》(看得懂汇编即可，开始要求不高)

0x02：工具

1. OllyDbg(关于这个工具，看雪论坛的安于此生翻译了一个系列，很不错)
2. x64dbg(开源调试器，支持x86,x64)
3. IDA
4. gdb(Linux下调试可以使用，配合peda或者gef插件效果更佳)

0x03:网站推荐

- 学习的网站 看雪论坛 吾爱破解 tust4you KSSD
- 开始可以找些crackme练习下 crackme.de reversing.kr

0x04:其他

- Q：我的汇编要到什么程度啊?

- A：开始看得懂汇编代码就好了，到后面你自己会慢慢摸索的。

> 多动手实践，多动手实践，多动手实践。

0x05:学习过程

相比web方向而言，逆向的入门门槛的确高些，所需要的基础知识略多。这里分享一些经历，希望可以让学弟学妹们少走点弯路。 首先，基本功真的很重要。扎实的基本功能决定你以后可以走多远。基本功有哪些呢？上面我们提到的编程和汇编就是其中的一部分，还有对系统的一些机制的了解，也会对你的学习有所帮

助。我在知乎上看到过关于逆向学习的一个回答，答主说：++那些人肉逆向机无外乎都是基本功十分扎实的，编程能力好，熟练掌握操作系统相关的知识++。然而对于刚要入坑的人来说，操作系统什么的难度是很大，所以这个可以稍往后放放，先学习编程，在这个过程中你可以了解到一些基本的操作系统的机制；建议从Win平台开始学习，多动手就好。 其次，对于二进制方向，漏洞挖掘/利用这个部分也很有意思，相对应的是CTF比赛中的pwn类型的题目。这类要求你审计出程序中的漏洞，然后写出利用代码(exp)。然而这个对基础的要求很高，比如程序编译链接的过程，系统堆分配的机制，系统怎么处理不同大小的堆块等。很多地方都可以衍生出巧妙地漏洞利用姿势。 不论是学习什么技术，都不能浮躁，二进制前期是枯燥点，学习曲线不太好，而且出成绩需要一定的时间，这个时候不要怀疑自己，踏踏实实学习就好了。 分享一个真事:一天，有人问LateRain学长：++怎么挖漏洞啊?++ 他得到的回答是：++你会写代码吗？++ 不论是web还是二进制，编程能力都是必须的，试问，你了解开发的相关知识，怎么去做逆向分析呢？比如现在一些开发模式、框架，如果不做一些了解，在做逆向分析的时候就会很吃亏。同样的，十个人写一个功能，可能一半的人写法都一样，如果你要挖掘这个功能模块的漏洞，是不是需要了解开发相关的知识呢？这样可能就又有人提问了.....

- Q：我要学到什么程度啊？顺序呢?
- A：我个人的主张是，需要什么的时候再去搞什么。如果你一路按所谓的顺序学下去，早晚你的热情要耗光的。

最后分享一句当时学长对我说的话:==整就牛。==