Reverse入门指东

走过路过，不要错过嘞。逆向工程入门指东，不要998，更不要98，现在免费送啦 ~

你们肯定好奇，为啥要咱叫指东，当然是因为指南指北都有了（）。

咳咳，回归正题。我们先来介绍下什么是逆向工程。

逆向工程

先来看下万能的Wiki怎么说：

逆向工程（英语：Reverse Engineering），又称**反向工程**，是一种技术仿造过程，即对一项目标产品进行逆向分析及研究，从而演绎并得出该产品的处理流程、组织结构、功能性能规格等设计要素，以制作出功能相近，但又不完全一样的产品。逆向工程源于商业及军事领域中的硬件分析。其主要目的是，在无法轻易获得必要的生产信息下，直接从成品的分析，推导产品的设计原理。

我们一开始先接触的呢，是从CTF竞赛角度出发的逆向工程。什么是CTF呢？请大家自行点击链接查看https://ctf-wiki.org/，这里不再赘述

CTF竞赛中的逆向工程指的都是软件逆向技术，所谓的软件包括但不限于windows或者linux平台的**二进制文件**（如常见的exe文件、dll文件等）、安卓平台下的**APK**文件。总的来说，一切能隐藏逻辑并具有执行特定功能的文件，都是我们逆向的对象。我们通过使用特定的工具对程序进行分析，从而分析程序的加密逻辑、执行流程，进一步的得到我们想要的答案。

这时候大家可能会好奇，二进制文件到底是什么？

二进制文件

拿我们比较熟悉的exe文件为例，当你双击进行运行它的时候，可以看到程序成功运行起来。但是如果你使用记事本来打开这个文件时，你会发现你只能看到一堆乱码，这是因为二进制文件有着自己的编码解码规则；大部分的二进制文件都是给计算机看的，计算机通过解析二进制文件来运行程序，而我们是无法直接看到文件内容的。

在一个二进制文件生成前，首先需要程序员进行软件开发，由程序员编写的代码称之为`源代码`；当我们编写完成一个程序时，CPU是无法理解我们所编写的代码的，我们需要借助一个“翻译”工具来将代码转换成CPU能够理解并运行的语言，翻译的过程称为`编译`，进行编译的工具就叫做`编译器`。

以一个C语言程序为例（可能你还没有学习C语言，不过并不要紧），程序代码会先被编译器翻译成接近底层的`汇编语言`，继续被`汇编器`转换成机器语言目标文件，最后使用`链接器`生成可执行文件(exe)，然后CPU才能够看懂该程序，并成功运行。

你可能对上述的流程看的不是很懂，当我们学习逐渐深入时，便能够理解上面的流程。

说了这么多，我们该如何进行逆向分析呢？

逆向分析

我们刚刚了解了`源文件`如何变成`二进制文件`，不过通常我们在对一个软件进行逆向分析时，只能够拿到一个软件的`二进制文件`，但是我们要分析它的设计原理，就必须查看我们能够看懂的`汇编代码`或者`源代码`，也就是逆转上面的流程。由此，诞生了与`编译器`和`汇编器`相对应的两个工具

- 反汇编器：将二进制文件中的机器码(CPU所能看懂并运行的指令)转换成汇编代码
- 反编译器：通过解析汇编代码，将其转换成接近源代码的高级语言

借助上述两个工具我们就可以通过二进制文件来得到源代码，开始分析程序的具体逻辑了。

我们在进行逆向分析时常用的工具有：

- IDA PRO：集反汇编器和反编译器于一体，可以对大部分的可执行文件进行分析。
- Ollydbg / DBG：汇编级的调试器，可以快速高效的调试汇编代码。

如何进行下载呢，我们一般在52pojie、看雪等论坛来下载相关工具。

具体如何使用上述工具，希望大家自行搜索。

基础知识，以及怎么学，请参考Pwn入门指北，写的很详尽了，这里不再赘述。

> 接下来，使用IDA来进行你的第一次逆向之旅吧
>
> 题目附件: https://pan.baidu.com/s/1kexk5YMhAUpl7QofUBSb2w?pwd=grad