网安实践逆向分析报告

1. 查找success字符串，通过strings表查找。并且通过跳转，找到输出success字符串的函数地址。如图1，2，3所示。

文本

描述已自动生成

图1 通过string查找success字符串



图2 找到success字符串所在位置后跳转到对应函数位置

图形用户界面, 文本

描述已自动生成

图3 success对应的函数位置,注意到函数地址为0x004011A0

注意：这里一开始ida并未将其识别为一段函数，需要手动将其转换为函数然后反编译，选取从0x004011A0-0x00401210的代码段然后按P,然后反编译即可。

图形用户界面, 文本

描述已自动生成

图4 输出success的C函数

1. 找到输出函数后，接着寻找溢出点，由实例可知，应该是在strcpy这个函数进行溢出操作，同时包含了整数溢出。首先寻找整数溢出点，在sub\_4012E0函数中，如图5所示。

文本

描述已自动生成

图5 整数溢出点

1. 接着进入sub\_401005和sub\_4014C0函数，发现整数需要在3-8之间，则构造低字节在3-8的输入，这里选择长度0x107，如图6所示。



图6 整数范围限制，以及溢出点函数

1. 查看sub\_4014C0的栈，如图7所示。发现Destination到r的距离为0x10，所以在输入文件中的0x10处输入输出函数地址0x004011A0,同时是小端存储，内容如图8所示。

文本

描述已自动生成

图7 sub\_4014C0栈分布

表格

描述已自动生成

图8 输入文件二进制显示

1. 打开可执行文件，选择文件输入，并输入自己的学号，输出Success结果，实验成功！如图9所示。

图片包含 文本

描述已自动生成

图9 输出结果截图