CTF-Crypto学习1（软件加壳、反汇编、Babe64、Rijndael密码算法）

原创

魔云连洲 于 2021-10-20 21:53:57 发布 77 收藏 2

分类专栏： 笔记 文章标签： 算法 密码学

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。

本文链接：https://blog.csdn.net/m0_46335150/article/details/120876022

版权

笔记 专栏收录该内容

17 篇文章 0 订阅

订阅专栏

CTF-Crypto学习1（软件加壳、反汇编、Babe64、Rijndael密码算法）

1.软件加壳

定义：

加壳的全称应该是可执行程序资源压缩，压缩后的程序可以直接运行。

加壳的另一种常用的方式是在二进制的程序中植入一段代码，在运行的时候优先取得程序的控制权，之后再把控制权交还给原始代码，这样做的目的是隐藏程序真正的OEP（入口点，防止被破解）。

作用：

- 防止软件被破解，保护软件版权；
- 增加程序运行速度；
- 病毒制作，绕过一些杀毒软件的扫描，从而实现作为病毒的一些入侵或破坏一些特征；

工具：

PEiD：一款著名的查壳工具，其功能强大，几乎可以侦测出所有的壳，其数量已超过470 种PE 文档 的加壳类型和签名。

2.反汇编

定义：将源代码转换成二进制执行代码的过程叫编译，比如将C源代码编译成.exe可执行文件；那么把二进制执行代码转换成源代码的过程就叫“反编译”，比如把.exe文件转换为C源代码就叫反编译。

作用：把可执行的二进制文件转为汇编代码,进而可以研究该程序 。

反汇编常用工具：.net反编译工具Reflector、JD-GUI、procyon-decompiler等

3.Base64

定义：Base64是网络上最常见的用于传输8Bit字节码的编码方式之一，Base64就是一种基于64个可打印字符来表示二进制数据的方法。

要点：

Base64 使用US-ASCII子集的64个字符,即大小写的26个英文字母，0－9，＋，/。

编码总是基于3个字符，每个字符用8位二进制表示，因此一共24位，再分为4四组，每组6位，表示一个Base64的值。

Base64值为0就是A，为27的就是b。这样，每3个字符产生4位的Base64字符。如果被加密的字符串每3个一组，还剩1或2个字符，使用特殊字符"="补齐Base64的4字。

怎么判断一段字符串是否是经过了Base64编码？

看最后是否有“=”，但是没有“=”也可能是经过了Base64编码

4.Rijndael密码算法

Rijndael是由比利时密码学家设计的分组密码算法，于2000年被选为新一代的标准密码算法——AES。今后会有越来越多的密码软件支持这种算法。

Rijndael的分组长度和密钥长度可以分别以32比特为单位在128比特和256比特的范围内进行选择。不过在AES的规格中，分组长度固定为128比特，密钥长度只有128、192和256比特三种。

刚开始看到很懵，但是看到AES就熟悉了。。。