



南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



# 汇编语言与逆向技术

## 第10章 软件保护技术

王志

zwang@nankai.edu.cn

南开大学 网络空间安全学院

2021-2022学年

有哪些保护软件知识产权的技术？

正常使用主观题需2.0以上版本雨课堂

作答





允公允能 日新月异

## 本章知识点

- 序列号保护
- 警告窗口
- 时间限制
- 菜单功能限制
- KeyFile保护



南开大学  
Nankai University



南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



# 序列号保护

哪些软件使用了序列号技术？序列号技术的实现机理。

正常使用主观题需2.0以上版本雨课堂

作答



允公允能 日新月异

# 序列号保护

- 共享软件（shareware）
  - 使用时间和功能上的限制
  - 注册用户
  - 获得序列号
  - 输入注册信息和序列号，软件取消各种限制



南开大学  
Nankai University



允公允能 日新月异

# 序列号保护

- 序列号的验证
  - 验证用户名和序列号之间的数学映射关系
  - 映射关系是由软件公司设计
  - 映射关系越复杂，越不容易被破解



南开大学  
Nankai University



允公允能 日新月异

# 序列号保护

- 序列号检查的4种基本方法：
  - 将用户名等信息作为自变量，通过函数 $F$ 变换之后得到注册码
  - 通过注册码验证用户名的正确性
  - 通过对等函数检查注册码
  - 同时将用户名和注册码作为自变量(即采用二元函数)







允公允能 日新月异

## 通过用户名生成序列号

- 将用户名作为自变量，通过函数 $F$ 变换之后得到注册码
  - 将计算出的注册码和用户输入的注册码进行比较，以确定用户是否为合法用户
- 序列号= $F$ （用户名）



南开大学  
Nankai University

序列号保护有哪些脆弱性？

正常使用主观题需2.0以上版本雨课堂

作答



南开大学  
Nankai University



允公允能 日新月异

## 通过用户名生成序列号

- 序列号是以明文形式存储在内存中
- 序列号很容易被逆向分析出来，不安全
  - 注册码泄漏
  - 注册机：映射函数 $F$ 的泄漏，构造注册机
  - 破解版：修改注册码的比较指令，通过注册码检查



南开大学  
Nankai University



允公允能 日新月异

## 通过注册码验证用户名

- 生成注册码的公式：序列号= $F$ （用户名）
- 检查注册码，是利用 $F$ 的逆变换 $F^{-1}$
- 用户名= $F^{-1}$ （序列号）



南开大学  
Nankai University



允公允能 日新月异

## 通过注册码验证用户名

- 更加安全
  - 生成注册码的函数 $F$ 没有直接出现在软件代码中
  - 注册码的明文没有出现在内存中





允公允能 日新月异

## 通过注册码验证用户名

- 穷举法，暴力破解序列号
- 破坏注册码的验证过程
- 因为 $F^{-1}$ 在软件中的，通过 $F^{-1}$ 找出 $F$
- 给定一个序列号，利用公式得出一个用户名，从而得到一个正确的用户名/序列号对



南开大学  
Nankai University



允公允能 日新月异

## 对等函数验证注册码

- 验证公式:  $F_1$  (用户名) =  $F_2$  (序列号)
- 内存中不出现注册码的明文





允公允能 日新月异

## 二元函数

- 之前的3种注册码验证函数都是一元函数
- **二元函数**：同时将用户名和注册码作为自变量
- 验证公式： 特定值=  $F_3$ （用户名、序列号）



南开大学  
Nankai University





允公允能 日新月异

## 二元函数

- 用户名与序列号之间的关系不再清晰
- 必须维护用户名与序列号之间的唯一性





允公允能 日新月异

## 序列号面临的安全问题

- **破解版**：修改判断序列号之后的跳转指令
- **注册机**：逆向用户信息和序列号的映射关系，构造注册机



南开大学  
Nankai University



南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异

警告窗口

有哪些软件使用了警告窗口技术？警告窗口技术有哪些脆弱性？

正常使用主观题需2.0以上版本雨课堂

作答



允公允能 日新月异

## 警告窗口

- 警告(Nag)窗口是软件设计者用来提醒用户购买正式版本的弹出窗口
- 去除警告窗口常用方法
  - 修改程序的资源
  - 静态分析
  - 动态分析



南开大学  
Nankai University



允公允能 日新月异

## 修改程序的资源

- 将警告窗口的属性修改成透明或不可见
- 完全去掉Nag，需要定位窗口的创建代码



南开大学  
Nankai University



## 静态分析

- 资源分析发现启动画面窗口的ID是121（79h）
- 通过ID 79h找到 “Dialog: DialogID\_0079” 就是Nag

```
:0040104D  mov eax, dword ptr [esp+04]
:00401051  push 00000000          ;初始化值
:00401053  push 004010C4          ;对话框处理函数指针，指向一段子程序
:00401058  push 00000000          ;父窗口句柄
:0040105A  push 00000079          ;对话框 ID 为 DialogID_0079
:0040105C  push eax               ;应用程序实例句柄，即 Nag.exe 的基地址
:0040105D  mov dword ptr [0040119C], eax
* Reference To: USER32.DialogBoxParamA, Ord:0093h
:00401062  Call dword ptr [00401010] ;显示 Nag 对话框
:00401068  xor eax, eax
:0040106A  ret 0010
```





## 警告窗口

- DialogBoxParam函数一般和EndDialog函数配对使用
  - DialogBoxParam打开对话框
  - EndDialog关闭对话框
  - 不能简单地将DialogBoxParam函数屏蔽

```
int DialogBoxParam(  
    HINSTANCE hInstance,           //应用程序实例句柄  
    LPCTSTR lpTemplateName,        //对话框 ID  
    HWND hWndParent,              //父窗口句柄  
    DLGPROC lpDialogFunc,          //对话框处理函数指针  
    LPARAM dwInitParam             //初始化值  
);
```







# 警告窗口

开始

```
:0040104D  mov  eax, dword ptr [esp+04]
:00401051  push 00000000                ;初始化值
:00401053  push 004010C4                ;对话框处理函数指针, 指向一段子程序
:00401058  push 00000000                ;父窗口句柄
:0040105A  push 00000079                ;对话框 ID 为 DialogID_0079
:0040105C  push  eax                    ;应用程序实例句柄, 即 Nag.exe 的基地址
:0040105D  mov  dword ptr [0040119C], eax
* Reference To: USER32.DialogBoxParamA, Ord:0093h
:00401062  call dword ptr [00401010]     ;显示 Nag 对话框
:00401068  xor  eax, eax
:0040106A  ret  0010
```

结束

```
004010C4  mov  eax, dword ptr [esp+8]
004010C8  sub  eax, 110                 ;Switch (cases 110..111)
004010CD  je   short 00401103
004010CF  dec  eax
004010D0  jnz  short 004010FF
004010D2  mov  eax, dword ptr [esp+C]   ;Case 111 of switch 004010C8
004010D6  dec  eax
004010D7  jnz  short 004010FF
004010D9  push 0
004010DB  push dword ptr [esp+8]
004010DF  call [<&USER32.EndDialog>]    ;关闭对话框
004010E3  push 0                        ;初始化值
004010E7  push 00401109                ;主对话框处理函数指针
004010EC  push 0                        ;父窗口句柄
004010EE  push 65                       ;主对话框 ID 为 DialogID_0065
004010F0  push 0
```



有哪些方法可以去掉nag窗口？

正常使用主观题需2.0以上版本雨课堂

作答



南开大学  
Nankai University

# 去除警告窗口

- 跳过警告窗口代码
  - 修改开始位置push, 变成jmp, 直接到结束之后
- 使用主对话框的参数修改Nag的DialogBoxParam函数的参数

```
:00401051  push 00000000  
:00401053  push 00401109      ;将此处指向主窗口的子处理程序  
:00401055  push 00000000  
:0040105A  push 00000065      ;指向主对话框的 ID DialogID_0065  
:0040105C  push eax  
:0040105D  mov dword ptr [0040119C], eax  
* Reference To: USER32.DialogBoxParamA, Ord:0093h  
:00401062  Call dword ptr [00401010] ;该函数会调用主对话框窗口  
:00401068  xor eax, eax  
:0040106A  ret 0010           ;主对话框关闭后将从这里退出
```



南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



时间限制

此题未设置答案，请点击右侧设置按钮

有哪些软件使用了时间限制？时间限制技术有哪些绕过的方法？

正常使用填空题需3.0以上版本雨课堂

作答



允公允能 日新月异

## 时间限制

- 限制每次软件运行的时长
- 每次运行时长不限，但是有使用时间限制，例如可以免费使用30天



南开大学  
Nankai University



允公允能 日新月异

## 计时器

- 每次运行时都有时间限制，例如运行10分钟或20分钟就停止，必须重新运行程序才能正常工作。
- 这类程序里有一个**计时器**来统计程序运行的时间



南开大学  
Nankai University



允公允能 日新月异

# Windows计时器

- SetTimer函数, WM\_TIMER消息
- 高精度的多媒体计时器timeSetEvent
- GetTickCount函数
- timeGetTime函数



南开大学  
Nankai University





# SetTimer函数

- 函数原型
  - `UNIT SetTimer (HWND hwnd, UINT nIDEvent, UINT uElapse, TIMERPROC lpTimerFunc)`
- 回调函数
  - `void CALLBACK TimerProc(HWND hwnd, UNIT uMsg, UNIT idEvent, DWORD dwTime);`
- KillTimer函数删除计时器





## 高精度的多媒体计时器

- 多媒体计时器的精度可以达到1毫秒
- 应用程序可以通过调用timeSetEvent()函数来启动一个多媒体计时器
  - MMRESULT timeSetEvent( UINT uDelay, UINT uResolution, LPTIMECALLBACK lpTimeProc, DWORD\_PTR dwUser, UINT fuEvent);





允公允能 日新月异

## GetTickCount()函数

- 函数返回的是系统自成功启动以来所经过的时间(以**毫秒**为单位)
- 将程序开始和结束时的GetTickCount返回值相减，获得程序的执行时间



南开大学  
Nankai University



允公允能 日新月异

## timeGetTime函数

- 多媒体计时器函数timeGetTime返回Windows自启动后所经过的时间
  - 以毫秒为单位
- 一般情况下，不需要使用高精度的多媒体计时器
  - 精度太高会对系统性能造成影响



南开大学  
Nankai University



允公允能 日新月异

## 时间限制

- 演示版软件通常设置了使用时间限制，例如试用30天
- 超过试用期软件就不能运行，付费注册之后，才能继续使用



南开大学  
Nankai University



允公允能 日新月异

# 时间限制

- 软件安装或者第一次运行时，记录软件安装时间
- 软件每次执行，读取当前系统日期
- 计算当前系统日期与软件安装日期的差值
- 如果差值大于指定的值，例如30天，就停止软件运行
- 原理简单，但是安全性不够



为什么刚才介绍的时间限制方法的安全性不高？

正常使用主观题需2.0以上版本雨课堂

作答



南开大学  
Nankai University



允公允能 日新月异

## 软件的安装日期的修改

- RegMon、FileMon等监控软件可以找到日期的存储位置
  - 删除时间，破除时间限制保护
- 将软件的安装日期存储在多个位置



南开大学  
Nankai University





允公允能 日新月异

## 修改最近一次运行的系统时间

- 攻击者可以通过修改系统时间绕过软件的时间限制保护
  - 在软件打开、退出的时候都要进行日期检查
  - 使用多种方式获得系统时间
  - 常用的API: GetSystemTime、GetLocalTime
  - 读取被频繁修改的系统文件的最后修改日期



南开大学  
Nankai University



## 时间限制实例

- 实例程序Timer.exe采用SetTimer()函数计时
- 每秒发送1次WM\_TIMER消息。当应用程序收到消息时，将执行如下语句

---

```
case WM_TIMER :  
    if(i<=19)  
        i++;                                //i 的初值是 0  
    else  
        SendMessage(hDlg, WM_CLOSE, 0, 0); //关闭程序  
return 0 ;
```

---





## 时间限制实例

- 使用jmp直接跳过SetTimer()函数

```
004010C2  mov     esi, dword ptr [esp+8]
004010C6  push    0                      ;/Timerproc = NULL
004010C8  push    3E8                    ;|Timeout = 1000. ms
004010CD  push    1                      ;|TimerID = 1
004010CF  push    esi                    ;|hWnd
004010D0  call    [<&USER32.SetTimer>]   ;\SetTimer
004010D6  mov     eax, dword ptr [403004]
```





## 时间限制实例

- #define WM\_TIMER 0x0113
- 修改判断条件

```
00401175  cmp     eax, 113                ;Case 113 (WM_TIMER)
0040117A  jnz     short 00401148
0040117C  mov     eax, dword ptr [403008] ;[403008]处存放的是 i (定义了全局变量)
00401181  cmp     eax, 13                 ;超过 20 秒 (“13” 是十六进制数)
00401184  jg      short 00401137          ;超时就跳走退出, 直接 NOP
00401186  inc     eax                     ;i++
00401187  lea     ecx, dword ptr [esp+C]
0040118B  push    eax
0040118C  push    00403000
00401191  push    ecx
00401192  mov     dword ptr [403008], eax ;将 i 放进[403008]
```



南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



# 菜单功能限制

有哪些软件会限制非付费用户使用的功能？ 功能限制技术有哪些脆弱性？

正常使用主观题需2.0以上版本雨课堂

作答



允公允能 日新月异

## 菜单功能限制

- 通常试用版的软件，菜单或窗口中的部分选项是灰色的，无法使用。
- 这种功能受限的程序有两种
  - 试用版和正版软件是完全两个不同的文件（**推荐使用**）
  - 试用版和正版软件是同一个文件



南开大学  
Nankai University





允公允能 日新月异

## 菜单功能限制

- 将软件菜单和窗口变灰(不可用状态), 可以使用如下函数
  - EnableMenuItem函数
  - EnableWindow函数



南开大学  
Nankai University





## EnableMenuItem()函数

- 允许或禁止指定的菜单条目，原型如下
- `BOOL EnableMenuItem(HMENU hMenu, UINT uIDEnableItem, UINT uEnable)`
  - `hMenu`: 菜单句柄
  - `uIDEnableItem`: 允许或禁止的一个菜单条目的标识符
  - **`uEnable`**: 控制标志
  - 返回值: 返回菜单项以前的状态





允公允能 日新月异

## EnableWindow()函数

- 允许或禁止指定窗口,原型如下
  - `BOOL EnableWindow(HWND hWnd, BOOL bEnable)`
  - `hWnd`:窗口句柄
  - `bEnable`: "TRUE"为允许, "FALSE"为禁止
  - 返回值:非0表示成功, 0表示失败





## 菜单限制保护

- 当uEnable控制标志为0时，恢复菜单的功能

:004011E3 6A01	push 00000001	;控制标志
:004011E5 68459C0000	push 00009C45	;标识符 (Menu 的 ID=40005)
:004011EA 50	push eax	;菜单句柄
:004011EB FF1524204000	Call USER32.EnableMenuItem	





南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



KeyFile保护

有哪些软件使用了KeyFile的软件保护技术？ KeyFile技术有哪些脆弱性？

正常使用主观题需2.0以上版本雨课堂

作答



## KeyFile保护

- KeyFile也叫注册文件，是一种利用文件来注册软件的保护方式，包含加密或者未加密的数据
- 试用版的软件一般没有KeyFile
- 用户向作者付费后，会收到作者提供的KeyFile
- 将KeyFile放入指定的目录就可以完成注册





允公允能 日新月异

# KeyFile保护

- 采用较大的KeyFile文件
- 加入垃圾信息，干扰逆向分析
- KeyFile的合法性检查分成几部分，分散在软件的不同模块中进行判断。



南开大学  
Nankai University



允公允能 日新月异

# KeyFile保护

- KeyFile数据处理也要尽可能采用复杂的运算
- 可以让注册文件中的部分数据和软件中的关键代码或数据发生关联，使软件无法被暴力破解



南开大学  
Nankai University





允公允能 日新月异

# KeyFile保护

- KeyFile是一个文件，因此，所有与Windows文件操作有关的API函数都可作为动态跟踪破解的断点。
  - CreateFile
  - FindFirstFile
  - ReadFile
  - SetFilePointer



南开大学  
Nankai University



允公允能 日新月异

## 破解KeyFile

- 用Process Monitor等工具，监视软件对文件的操作，以找到KeyFile的文件名。
- 伪造一个KeyFile文件。用十六进制工具，编辑和修改KeyFile



南開大學  
Nankai University



允公允能 日新月异

# 破解KeyFile

- 在调试器中，对CreateFileA函数设断点
- 对ReadFile函数设断点
  - 分析传递给ReadFile函数的文件句柄和缓冲区地址
  - 逆向分析程序对KeyFile的**判断过程**



南开大学  
Nankai University



允公允能 日新月异

## 本章知识点

- 序列号保护
- 警告窗口
- 时间限制
- 菜单功能限制
- KeyFile保护



南开大学  
Nankai University



南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



# 汇编语言与逆向技术

## 第10章 软件保护技术

王志

zwang@nankai.edu.cn

南开大学 网络空间安全学院

2021-2022学年