

**课程名称：**逆向工程实践

**报告题目：**使用Ollydbg破解程序

**课程教师：**何兴高

**学院：**信息与软件工程学院

**姓名：**杨博辰

**学号：**201522220234

**一、题目名称**

《使用Ollydbg破解程序》

**二、题目内容**

使用C++语言通过Windows API编写一个登录程序，当用户输入正确的用户名和密码之后，提示登录成功，否则提示登录失败。通过Ollydbg软件进行反汇编，对程序进行破解。程序界面如下图。

****

**三、知识点及介绍**

1.汇编语言介绍

汇编语言（Assembly Language）是面向机器的程序设计语言。在汇编语言中，用助记符（Mnemonics）代替机器指令的操作码，用地址符号（Symbol）或标号（Label）代替指令或操作数的地址，如此就增强了程序的可读性并且降低了编写难度，象这样符号化的程序设计语言就是汇编语言，因此亦称为符号语言。使用汇编语言编写的程序，机器不能直接识别，还要由汇编程序或者叫汇编语言编译器（即汇编器）转换成机器指令。汇编程序将符号化的操作代码组装成处理器可以识别的机器指令，这个组装的过程称为组合或者汇编。因此，有时候人们也把汇编语言称为组合语言。

汇编语言是一种功能很强的程序设计语言，也是利用计算机所有硬件特性并能直接控制硬件的语言。汇编语言，作为一门语言，对应于高级语言的编译器，需要一个“汇编器”来把汇编语言原文件汇编成机器可执行的代码。高级的汇编器如MASM，TASM等等为我们写汇编程序提供了很多类似于高级语言的特征，比如结构化、抽象等。在这样的环境中编写的汇编程序，有很大一部分是面向汇编器的伪指令，已经类同于高级语言。现在的汇编环境已经如此高级，即使全部用汇编语言来编写windows的应用程序也是可行的，但这不是汇编语言的长处。汇编语言的长处在于编写高效且需要对机器硬件精确控制的程序。

2.C++语言介绍

C++是在C语言的基础上开发的一种通用编程语言，应用广泛。C++支持多种编程范式 －－面向对象编程、泛型编程和过程化编程。最新正式标准C++14于2014年8月18日公布。其编程领域众广，常用于系统开发，引擎开发等应用领域，是至今为止最受广大受用的最强大编程语言之一,支持类：类、封装、重载等!

C++语言的主要特点表现在两个方面，一是尽量兼容C,二是支持面向对象的方法。它操持了C的简洁、高效的接近汇编语言等特点，对C的类型系统进行了改革的扩充，因此C++比C更安全，C++的编译系统能检查出更多的类型错误。另外，由于C语言的广泛使用，因而极大的促进了C++的普及和推广。

3.Windows API介绍

Windows 这个多作业系统除了协调应用程序的执行、分配内存、管理资源…之外， 它同时也是一个很大的服务中心，调用这个服务中心的各种服务（每一种服务就是一个函数），可以帮应用程式达到开启视窗、描绘图形、使用周边设备等目的，由于这些函数服务的对象是应用程序(Application)， 所以便称之为 Application Programming Interface，简称 API 函数。WIN32 API也就是Microsoft Windows 32位平台的应用程序编程接口。

当WINDOWS操作系统开始占据主导地位的时候，开发WINDOWS平台下的应用程序成为人们的需要。而在WINDOWS程序设计领域处于发展的初期，WINDOWS程序员所能使用的编程工具唯有API函数，这些函数是WINDOWS提供给应用程序与操作系统的接口，他们犹如“积木块”一样，可以搭建出各种界面丰富，功能灵活的应用程序。所以可以认为API函数是构筑整个WINDOWS框架的基石，在它的下面是WINDOWS的操作系统核心，而它的上面则是所有的华丽的WINDOWS应用程序。

程序员想编写具有Windows风格的软件，必须借助API，API也因此被赋予至高无上的地位。但是，如若没有合适的Windows编程平台，那么Windows开发是一项很复杂的工作。在可视化编程IDE出来之前，那时的WINDOWS程序开发还是比较复杂的工作，程序员必须熟记一大堆常用的API函数，而且还得对WINDOWS操作系统有深入的了解。然而随着软件技术的不断发展，在WINDOWS平台上出现了很多优秀的可视化编程环境，程序员可以采用“所见即所得”的编程方式来开发具有精美用户界面和功能强大的应用程序。

**四、工具及介绍**

1.Ollydbg介绍

OLLYDBG是一个新的动态追踪工具，将IDA与SoftICE结合起来的思想，Ring 3级调试器，非常容易上手，己代替SoftICE成为当今最为流行的调试解密工具了。同时还支持插件扩展功能，是目前最强大的调试工具。

OllyDbg 可以以在任何采用奔腾处理器的 Windows 95、98、ME、NT 或是 XP（未经完全测试）操作系统中工作，但我们强烈建议您采用300-MHz以上的奔腾处理器以达到最佳效果。还有，OllyDbg 是极占内存的，因此如果您需要使用诸如追踪调试[Trace]之类的扩展功能话，建议您最好使用128MB以上的内存。

OllyDbg 的最大特点之一就是分析。它会分析函数过程、循环语句、选择语句、表[tables]、常量、代码中的字符串、欺骗性指令[tricky constructs]、API调用、函数中参数的数目，import表等等.. 这些分析增加了二进制代码的可读性，减少了出错的可能性，使得我们的调试工作更加容易。

2.MINGW介绍

MinGW，是Minimalist GNU for Windows的缩写。它是一个可自由使用和自由发布的Windows特定头文件和使用GNU工具集导入库的集合，允许你在GNU/Linux和Windows平台生成本地的Windows程序而不需要第三方C运行时库。MinGW 是一组包含文件和端口库，其功能是允许控制台模式的程序使用微软的标准C运行时间库（MSVCRT.DLL）,该库在所有的 NT OS 上有效，在所有的 Windows 95发行版以上的 Windows OS 有效，使用基本运行时间，你可以使用 GCC 写控制台模式的符合美国标准化组织（ANSI）程序，可以使用微软提供的 C 运行时间扩展，与基本运行时间相结合，就可以有充分的权利既使用 CRT（C Runtime）又使用 WindowsAPI功能。

**五、源程序**

#include<windows.h>

LRESULT CALLBACK WindowProc(HWND hwnd,UINT uMsg,WPARAM wParam,LPARAM lParam );

int CALLBACK WinMain(HINSTANCE hInstance,HINSTANCE hPrevInstance,LPSTR lpCmdLine,int nCmdShow){

WNDCLASS wndClass;

wndClass.style=CS\_HREDRAW|CS\_VREDRAW;

wndClass.lpfnWndProc=WindowProc;

wndClass.cbClsExtra=0;

wndClass.cbWndExtra=0;

wndClass.hInstance=hInstance;

wndClass.hIcon=LoadIcon(NULL,IDI\_APPLICATION);

wndClass.hCursor=LoadCursor(NULL,IDC\_ARROW);

wndClass.hbrBackground=(HBRUSH)(COLOR\_WINDOW+1);

wndClass.lpszMenuName=NULL;

wndClass.lpszClassName="主程序界面";

RegisterClass(&wndClass);

HWND hwnd=CreateWindow(

wndClass.lpszClassName,

TEXT("逆向工程"),

WS\_OVERLAPPED|WS\_CAPTION|WS\_SYSMENU,

CW\_USEDEFAULT,

CW\_USEDEFAULT,

400,

300,

NULL,

NULL,

hInstance,

NULL

);

ShowWindow(hwnd,SW\_SHOW);

UpdateWindow(hwnd);

MSG msg;

while(GetMessage(&msg,NULL,0,0)){

TranslateMessage (&msg) ;

DispatchMessage (&msg) ;

}

return msg.wParam;

}

LRESULT CALLBACK WindowProc(HWND hwnd,UINT uMsg,WPARAM wParam,LPARAM lParam ){

static HWND hEditUserName;

static HWND hEditPassword;

static HWND hButtonLogin;

HDC hdc;

PAINTSTRUCT ps;

RECT rect;

switch(uMsg){

case WM\_CREATE:

hButtonLogin=CreateWindow(TEXT("button"),TEXT("登录"),WS\_CHILD|WS\_VISIBLE|BS\_PUSHBUTTON,

170,130,60,40,hwnd,NULL,((LPCREATESTRUCT)lParam)->hInstance,NULL);

hEditUserName=CreateWindow(TEXT("edit"),NULL,WS\_CHILD|WS\_VISIBLE|ES\_LEFT|WS\_BORDER,

130,55,150,25,hwnd,NULL,((LPCREATESTRUCT)lParam)->hInstance,NULL);

hEditPassword=CreateWindow(TEXT("edit"),NULL,WS\_CHILD|WS\_VISIBLE|ES\_LEFT|WS\_BORDER|ES\_PASSWORD,

130,85,150,25,hwnd,NULL,((LPCREATESTRUCT)lParam)->hInstance,NULL);

return 0;

case WM\_PAINT:

hdc=BeginPaint(hwnd,&ps);

rect.left=150;

rect.top=100;

rect.right=50;

rect.bottom=30;

DrawText(hdc,TEXT("用户名："),-1,&rect,DT\_SINGLELINE|DT\_CENTER|DT\_VCENTER);

rect.left=150;

rect.top=160;

rect.right=50;

rect.bottom=30;

DrawText(hdc,TEXT("密 码："),-1,&rect,DT\_SINGLELINE|DT\_CENTER|DT\_VCENTER);

return 0;

case WM\_COMMAND:

if((HWND)lParam==hButtonLogin){

char userName[20];

char password[20];

GetWindowText(hEditUserName,userName,20);

GetWindowText(hEditPassword,password,20);

if(strcmp(userName,"杨博辰")==0&&strcmp(password,"201522220234")==0){

MessageBox(hwnd,TEXT("登录成功"),TEXT("提示"),MB\_OK);

}else{

MessageBox(hwnd,TEXT("登录失败"),TEXT("提示"),MB\_OK);

}

}

return 0;

case WM\_DESTROY:

PostQuitMessage(0);

return 0;

}

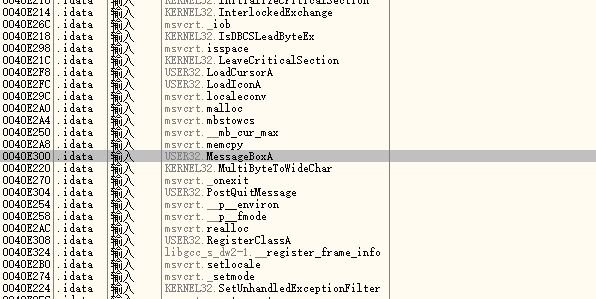
return DefWindowProc (hwnd, uMsg, wParam, lParam) ;

}

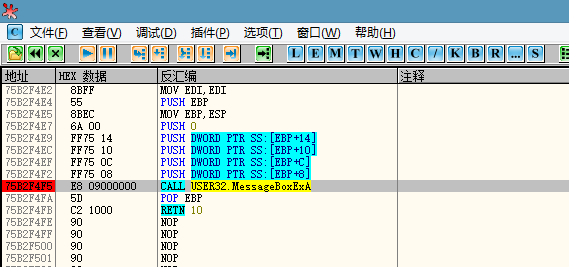
**六、过程及分析**

破解方法一：

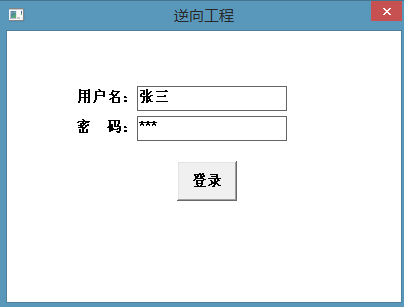
程序在输入用户名和密码后点击登录，如果用户名和密码正确，则会弹出登录成功的消息框，否则会弹出登录失败的消息框。在反汇编窗口中单击右键选择“查找”，然后选择“当前模块中的名称（标签）”，按下键盘上的M键，找到USER32.MessageBoxA，如下图所示。



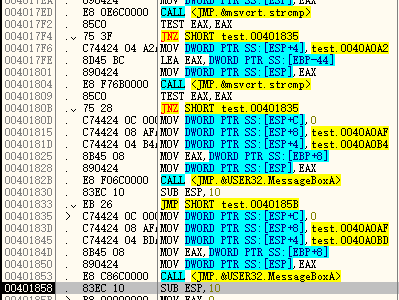
在该行单击右键，选择“反汇编窗口中跟随输入函数”，转到该函数的汇编代码处，在返回处加上断点。如下图所示。



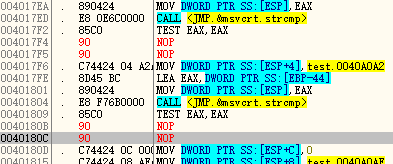
然后按F9运行程序，输入用户名和密码，如下图所示。



点击登录，弹出登录失败消息框。程序执行到断点处暂停，按F7单步步入，程序执行到调用MessageBox函数的下一条指令处。如下图所示。



灰色指令的上一条指令是当用户名和密码不正确时执行，提示登录失败。004017ED处和00401804处的CALL <JMP.&msvcrt.strcmp>指令是比较字符串，分别比较用户名和密码是否相等。004017F4处和0040180B处的JNZ SHORT test.00401835指令是如果字符串不相等，跳转到00401835处执行，提示用户登录失败。将两处JNZ指令换成NOP指令，无论用户输入的用户名和密码是否正确，都会提示登录成功。如下图所示。



将修改后的程序保存为test2.exe后运行，与上述分析一致，无论用户输入的用户名和密码是否正确，都提示登录成功。如下图所示。



破解方法二：

在反汇编窗口中单击右键，选择“超级字符串参考+（U）”,然后选择“查找UNICODE”，会弹出一个“超级字串参考+”窗口。如下图所示。

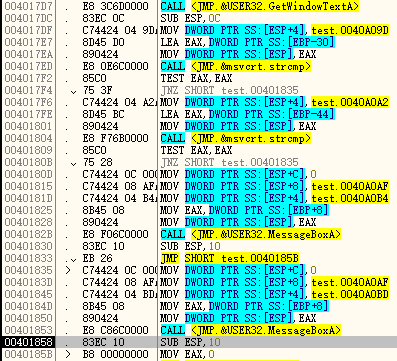


在上图中发现“杨博辰”和“201522220234”可能是用户名和密码，尝试后发现登录成功，如下图所示。

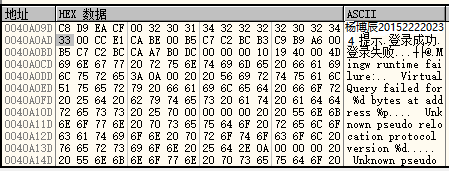


破解方法三：

按照破解方法一中的方式，执行到登录失败后的那一条语句。如下图所示。



在004017DF处有一条指令，MOV DWORD PTR SS:[ESP+4],test.0040A09D，在DUMP窗口中转到地址0040A09D，发现0040A09D地址开始的4个字节存储的是“杨博辰”。在004017F6处有一条指令，MOV DWORD PTRSS:[ESP+4],test.0040A0A2,在DUMP窗口中转到地址0040A09D开始的12个字节存储的是“201522220234”。如下图所示。



猜测用户名是“杨博辰”，密码是“201522220234”，尝试登录后发现登录成功，如下图所示。



**七、心得体会**

通过学习《逆向工程实践》这门课程，能够从更高的层面上认识软件的安全性，熟练掌握各种逆向工具，对软件进行逆向分析。通过这门课，认识到了汇编语言的重要性，汇编语言就是逆向工程语言。熟练掌握了汇编语言，才能更好地对软件进行逆向工程。在学习这门课的过程中，学到了许多以前没有接触的的概念和技术，拓宽了我的知识面。在今后的工作中，逆向工程会对我有很大的帮助。