Міністерство освіти і науки України

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра Обчислювальна техніка та програмування

Звіт з лабораторной роботи з дисципліні « Реверсне програмування »

Лабораторна робота 8

Виконав ст.гр. КІТ–36

Надірян Г.О.

Перевірив

Челак В.В.

Харків 2019

**Лабораторна робота 8**

**Тема**: Самомодіфікація коду

**Мета**: Набути практичних навичок написання та застосування самомодифицирующихся програм в середовищі masm64.

Постановка задачі

1. Написати програму в середовищі masm64 відповідно до варіанта завдання з використанням максимально можливої ​​кількості прийомів самомодіфікаціі коду.

2. Вивести результати виконання програми через функцію MessageBoxIndirect з власної іконкою. Програму і результати привести в звіті.

**Завдання**

Задана матриця 3 х 4. Визначити рядок з максимальною сумою позитивних елементів.

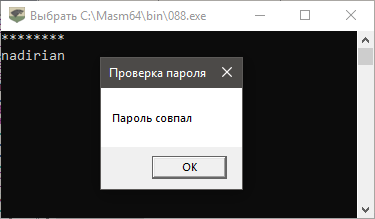


Рисунок 1 – результат роботи програми. Вводимо пароль

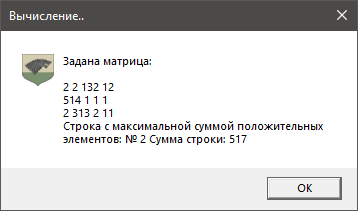


Рисунок 2 - Визначення рядка з максимальною сумою позитивних елементів.

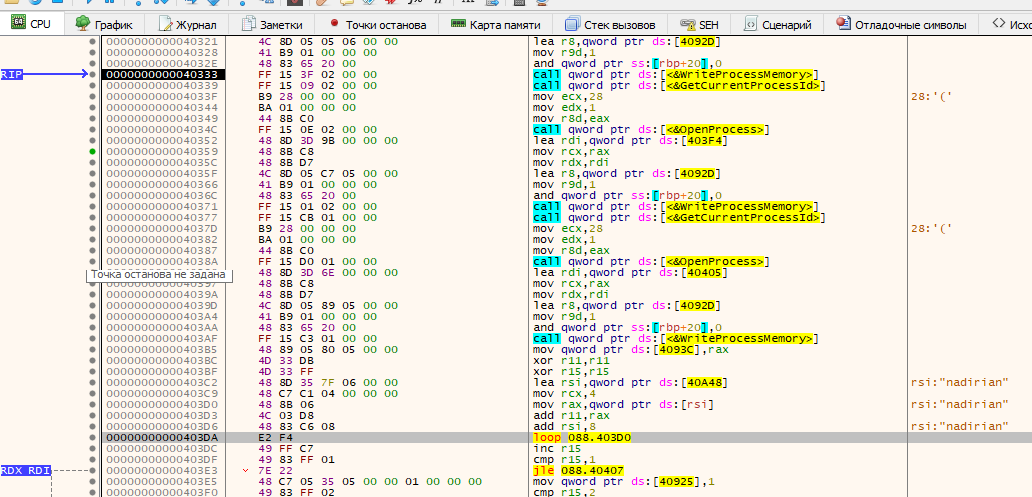


Рисунок 3 Робота програми до самомодіфікації

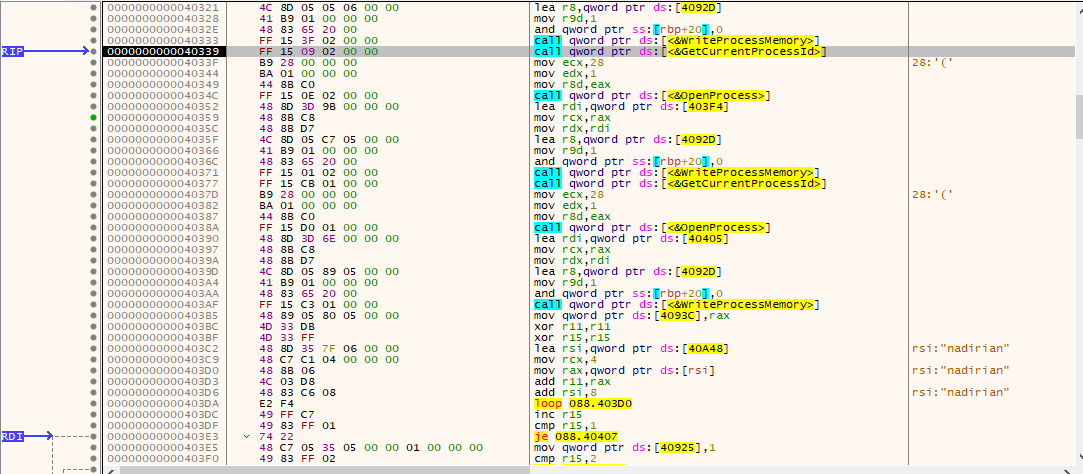


Рисунок4 - Робота програми після самомодіфікації.

Команда jz була замінена на jle. Але перед виконання перевірки jle замінюється на jz за допомогою функції WriteProcessMemory

**Код програми**

(Програма з самомодифікацією)

include win64a.inc

IDI\_ICON EQU 1001

MSGBOXPARAMSA STRUCT

cbSize **DWORD** **?,?**

hwndOwner **QWORD** **?**

hInstance **QWORD** **?**

lpszText **QWORD** **?**

lpszCaption **QWORD** **?**

dwStyle **DWORD** **?,?**

lpszIcon **QWORD** **?**

dwContextHelpId **QWORD** **?**

lpfnMsgBoxCallback **QWORD** **?**

dwLanguageId **DWORD** **?,?**

MSGBOXPARAMSA ENDS

.data

params MSGBOXPARAMSA **<>**

DATE1 STRUCT ; тип данных СТРУКТУРА с именем DATE1

elem1 dq **?** ; имя первого поля структуры

elem2 dq **?** ; имя второго поля структуры

elem3 dq **?** ; имя третьего поля структуры

elem4 dq **?** ; имя четвертого поля структуры

elem5 dq **?** ; имя четвертого поля структуры

elem6 dq **?** ; имя четвертого поля структуры

elem7 dq **?** ; имя четвертого поля структуры

DATE1 ENDS

Password db "nadirian" ; проверка только первых пяти символов

len1 equ **($-**Password**)/**type Password

Buf dq 8 ;

Err1 dq 0

Msg1 db "Пароль совпал"**,**0

Msg2 db "Пароль не корректен"**,**0

Title1 db "Проверка пароля"**,**0

Title2 db "Тсс,пароль Надирян (на латинице в нижнем регистре)"**,**0

stdout dq 0 ;

stdin dq 0 ;

cRead dq 0 ;

cWritten dq 0 ;

max1 dq 0

numstr dq 0

opc db 074h

\_c dd 3

Msg db "\*\*\*\*\*\*\*\*"**,**0**,**10 ;

mem dq **?**;

.code

Pas1 proc

**lea** **rsi,**Password ; адрес первого элемента строки

**lea** **rdi,**Buf ; адрес второго элемента строки

**mov** **rcx,**len1

**repe** **cmpsb** ; побайтно проверяется len раз

**jz** m2 ;

**inc** Err1 ; счетчик несовпадений

m2**:**

**ret**

Pas1 endp

WinMain proc

**sub** **rsp,**28h;

**mov** **rbp,rsp**

invoke GetStdHandle**,**STD\_OUTPUT\_HANDLE

**mov** stdout**,rax**

invoke GetStdHandle**,**STD\_INPUT\_HANDLE

**mov** stdin**,rax**

invoke WriteConsole**,**stdout**,ADDR** Msg**,**sizeof Msg**,ADDR** cWritten**,**0

invoke ReadConsole**,**stdin**,ADDR** Buf**,**8**,ADDR** cRead**,**0

invoke Pas1

.if **(**Err1**==**0**)**;

invoke MessageBox**,**0**,addr** Msg1**,addr** Title1**,**MB\_OK

invoke GetCurrentProcessId

invoke OpenProcess**,**PROCESS\_VM\_OPERATION **or** PROCESS\_VM\_WRITE**,** 1**,** **eax**

**lea** **rdi,**\_kk1111

invoke WriteProcessMemory**,rax,rdi,addr** opc**,**1**,**0

invoke GetCurrentProcessId

invoke OpenProcess**,**PROCESS\_VM\_OPERATION **or** PROCESS\_VM\_WRITE**,** 1**,** **eax**

**lea** **rdi,**\_kk1112

invoke WriteProcessMemory**,rax,rdi,addr** opc**,**1**,**0

invoke GetCurrentProcessId

invoke OpenProcess**,**PROCESS\_VM\_OPERATION **or** PROCESS\_VM\_WRITE**,** 1**,** **eax**

**lea** **rdi,**\_kk1113

invoke WriteProcessMemory**,rax,rdi,addr** opc**,**1**,**0

.data

dir db 256 dup**(**0**)** ;переменная для хранения пути к текущей директории

align 8 ; выравнивание памяти по адресам, кратным 8 байтам

str1 DATE1 **<**2**,**2**,**132**,**12**>** ; структура с именем str1

str2 DATE1 **<**54**,**1**,**1**,**1**>** ; структура с именем str2

str3 DATE1 **<**2**,**313**,**2**,**11**>** ; структура с именем str2

titl1 db "Вычисление.."**,**0

buf1 dq 10 dup**(**0**)**;

ifmt db "Задана матрица:"**,**0dh**,**0ah**,**0ah**,\**

"2 2 132 12"**,**0dh**,**0ah**,\**

"514 1 1 1"**,**0dh**,**0ah**,\**

"2 313 2 11"**,**0dh**,**0ah**,\**

"Строка с максимальной суммой положительных элементов: № %d Cумма строки: %d"**,**0

.code

**mov** mem**,rax**

**xor** **r11,r11** ; обнуление

**xor** **r15,r15** ; обнуление

**lea** **rsi,**str1 ; загрузка адреса первой строки структуры

**mov** **rcx,**4 ; количество элементов в строке

@1**:** **mov** **rax,qword** ptr**[rsi]** ; загрузка элемента из строки структуры

**add** **r11,rax** ; накапливаемая сумма элементов

**add** **rsi,**8 ; подготовка адреса нового элемента

**loop** @1 ; rсх := rcx – 1 и переход на m3, если не нуль

**inc** **r15** ; счетчик структур

**cmp** **r15,**1 ; это первая строка?

\_kk1111**:**

**jle** m1

;jz m1 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

**mov** numstr**,**1

**cmp** **r15,**2 ; это вторая строка?

\_kk1112**:**

**jle** m2

;jz m2 ;

**mov** numstr**,**2

**cmp** **r15,**3 ; это третья строка?

\_kk1113**:**

**jle** m3

;jz m3 ;

m1**:** **mov** **r10,r11** ; сумма элементов 1-й строки

**xor** **r11,r11** ; обнуление

**mov** **rcx,**4 ; количество элементов во 2-й строке

**lea** **rsi,**str2 ; загрузка адреса 2-й строки

**jmp** @1 ;

m2**:** **mov** **r12,r11** ; сумма элементов 2-й строки

**xor** **r11,r11** ; обнуление

**mov** **rcx,**4 ; количество элементов в 3-й строке

**lea** **rsi,**str3 ; загрузка адреса 3-й строки

**jmp** @1 ;

m3**:** **mov** **r13,r11**; сумма элементов 3-й строки

**cmp** **r10,r12**

**jg** @@1 ; если больше, то на метку @@1

**mov** max1**,r12**

**jmp** @@2

@@1**:** **mov** max1**,r10**

**mov** numstr**,**1

@@2**:** **cmp** max1**,r13**

**jg** m4

**mov** numstr**,**3

**mov** max1**,r13**

**jmp** m4

m4**:**

;invoke OpenProcess,PROCESS\_VM\_OPERATION or PROCESS\_VM\_WRITE, 1, eax

invoke wsprintf**,ADDR** buf1**,ADDR** ifmt**,**numstr**,**max1;

**mov** params.cbSize**,**SIZEOF MSGBOXPARAMSA ; размер структуры

**mov** params.hwndOwner**,**0 ; дескриптор окна владельца

invoke GetModuleHandle**,**0 ; получение дескриптора программы

**mov** params.hInstance**,rax** ; сохранение дескриптора программы

**lea** **rax,** buf1 ; адрес сообщения

**mov** params.lpszText**,rax**

**lea** **rax,**titl1 ;Caption ; адрес заглавия окна

**mov** params.lpszCaption**,rax**

**mov** params.dwStyle**,**MB\_USERICON ; стиль окна

**mov** params.lpszIcon**,**IDI\_ICON ; ресурс значка

**mov** params.dwContextHelpId**,**0 ; контекст справки

**mov** params.lpfnMsgBoxCallback**,**0 ;

**mov** params.dwLanguageId**,**LANG\_NEUTRAL ; язык сообщения

**lea** **rcx,**params

invoke MessageBoxIndirect

.else

invoke MessageBox**,**0**,addr** Msg2**,addr** Title2**,**MB\_ICONWARNING

.endif

invoke ExitProcess**,**0

WinMain endp

end

**Код програми**

include win64a.inc

IDI\_ICON EQU 1001

MSGBOXPARAMSA STRUCT

cbSize **DWORD** **?,?**

hwndOwner **QWORD** **?**

hInstance **QWORD** **?**

lpszText **QWORD** **?**

lpszCaption **QWORD** **?**

dwStyle **DWORD** **?,?**

lpszIcon **QWORD** **?**

dwContextHelpId **QWORD** **?**

lpfnMsgBoxCallback **QWORD** **?**

dwLanguageId **DWORD** **?,?**

MSGBOXPARAMSA ENDS

.data

params MSGBOXPARAMSA **<>**

DATE1 STRUCT ; тип данных СТРУКТУРА с именем DATE1

elem1 dq **?** ; имя первого поля структуры

elem2 dq **?** ; имя второго поля структуры

elem3 dq **?** ; имя третьего поля структуры

elem4 dq **?** ; имя четвертого поля структуры

elem5 dq **?** ; имя четвертого поля структуры

elem6 dq **?** ; имя четвертого поля структуры

elem7 dq **?** ; имя четвертого поля структуры

DATE1 ENDS

Password db "nadirian" ; проверка только первых пяти символов

len1 equ **($-**Password**)/**type Password

Buf dq 8 ;

Err1 dq 0

Msg1 db "Пароль совпал"**,**0

Msg2 db "Пароль не корректен"**,**0

Title1 db "Проверка пароля"**,**0

Title2 db "Тсс,пароль Надирян (на латинице в нижнем регистре)"**,**0

stdout dq 0 ;

stdin dq 0 ;

cRead dq 0 ;

cWritten dq 0 ;

max1 dq 0

numstr dq 0

opc db 074h

\_c dd 3

Msg db "\*\*\*\*\*\*\*\*"**,**0**,**10 ;

.code

Pas1 proc

**lea** **rsi,**Password ; адрес первого элемента строки

**lea** **rdi,**Buf ; адрес второго элемента строки

**mov** **rcx,**len1

**repe** **cmpsb** ; побайтно проверяется len раз

**jz** m2 ;

**inc** Err1 ; счетчик несовпадений

m2**:**

**ret**

Pas1 endp

WinMain proc

**sub** **rsp,**28h;

**mov** **rbp,rsp**

invoke GetStdHandle**,**STD\_OUTPUT\_HANDLE

**mov** stdout**,rax**

invoke GetStdHandle**,**STD\_INPUT\_HANDLE

**mov** stdin**,rax**

invoke WriteConsole**,**stdout**,ADDR** Msg**,**sizeof Msg**,ADDR** cWritten**,**0

invoke ReadConsole**,**stdin**,ADDR** Buf**,**8**,ADDR** cRead**,**0

invoke Pas1

.if **(**Err1**==**0**)**;

invoke MessageBox**,**0**,addr** Msg1**,addr** Title1**,**MB\_OK

.data

dir db 256 dup**(**0**)** ;переменная для хранения пути к текущей директории

align 8 ; выравнивание памяти по адресам, кратным 8 байтам

str1 DATE1 **<**2**,**2**,**132**,**12**>** ; структура с именем str1

str2 DATE1 **<**514**,**1**,**1**,**1**>** ; структура с именем str2

str3 DATE1 **<**2**,**313**,**2**,**11**>** ; структура с именем str2

titl1 db "Вычисление.."**,**0

buf1 dq 10 dup**(**0**)**;

ifmt db "Задана матрица:"**,**0dh**,**0ah**,**0ah**,\**

"2 2 132 12"**,**0dh**,**0ah**,\**

"514 1 1 1"**,**0dh**,**0ah**,\**

"2 313 2 11"**,**0dh**,**0ah**,\**

"Строка с максимальной суммой положительных элементов %d Строка № %d"**,**0

.code

**xor** **r11,r11** ; обнуление

**xor** **r15,r15** ; обнуление

**lea** **rsi,**str1 ; загрузка адреса первой строки структуры

**mov** **rcx,**4 ; количество элементов в строке

@1**:** **mov** **rax,qword** ptr**[rsi]** ; загрузка элемента из строки структуры

**add** **r11,rax** ; накапливаемая сумма элементов

**add** **rsi,**8 ; подготовка адреса нового элемента

**loop** @1 ; rсх := rcx – 1 и переход на m3, если не нуль

**inc** **r15** ; счетчик структур

**cmp** **r15,**1 ; это первая строка?

**jz** m1

**mov** numstr**,**1

**cmp** **r15,**2 ; это вторая строка?

**jz** m2

**mov** numstr**,**2

**cmp** **r15,**3 ; это третья строка?

**jz** m3

m1**:** **mov** **r10,r11** ; сумма элементов 1-й строки

**xor** **r11,r11** ; обнуление

**mov** **rcx,**4 ; количество элементов во 2-й строке

**lea** **rsi,**str2 ; загрузка адреса 2-й строки

**jmp** @1 ;

m2**:** **mov** **r12,r11** ; сумма элементов 2-й строки

**xor** **r11,r11** ; обнуление

**mov** **rcx,**4 ; количество элементов в 3-й строке

**lea** **rsi,**str3 ; загрузка адреса 3-й строки

**jmp** @1 ;

m3**:** **mov** **r13,r11**; сумма элементов 3-й строки

**cmp** **r10,r12**

**jg** @@1 ; если больше, то на метку @@1

**mov** max1**,r12**

**jmp** @@2

@@1**:** **mov** max1**,r10**

**mov** numstr**,**1

@@2**:** **cmp** max1**,r13**

**jg** m4

**mov** numstr**,**3

**mov** max1**,r13**

**jmp** m4

m4**:**

**lea** **rdi,**m1

invoke WriteProcessMemory**,rax,rdi,addr** opc**,**1**,**0

**lea** **rdi,**m2

invoke WriteProcessMemory**,rax,rdi,addr** opc**,**1**,**0

**lea** **rdi,**m3

invoke WriteProcessMemory**,rax,rdi,addr** opc**,**1**,**0

invoke wsprintf**,ADDR** buf1**,ADDR** ifmt**,**max1**,**numstr;

**mov** params.cbSize**,**SIZEOF MSGBOXPARAMSA ; размер структуры

**mov** params.hwndOwner**,**0 ; дескриптор окна владельца

invoke GetModuleHandle**,**0 ; получение дескриптора программы

**mov** params.hInstance**,rax** ; сохранение дескриптора программы

**lea** **rax,** buf1 ; адрес сообщения

**mov** params.lpszText**,rax**

**lea** **rax,**titl1 ;Caption ; адрес заглавия окна

**mov** params.lpszCaption**,rax**

**mov** params.dwStyle**,**MB\_USERICON ; стиль окна

**mov** params.lpszIcon**,**IDI\_ICON ; ресурс значка

**mov** params.dwContextHelpId**,**0 ; контекст справки

**mov** params.lpfnMsgBoxCallback**,**0 ;

**mov** params.dwLanguageId**,**LANG\_NEUTRAL ; язык сообщения

**lea** **rcx,**params

invoke MessageBoxIndirect

.else

invoke MessageBox**,**0**,addr** Msg2**,addr** Title2**,**MB\_ICONWARNING

.endif

invoke ExitProcess**,**0

WinMain endp

end

**Висновок**

Написали програму в середовищі masm64 відповідно до варіанта завдання з використанням ​​ прийомів самомодіфікаціі коду. Вивели результати виконання програми через функцію MessageBoxIndirect з власної іконкою