Міністерство освіти і науки України

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра Обчислювальна техніка та програмування

Звіт з лабораторной роботи з дисциплиі « Реверсне програмування »

Лабораторна робота 5

Виконав ст.гр. КІТ–36

Надірян Г.О.

Перевірив

Челак В.В.

Харків 2019

**Лабораторна робота 5**

**Тема**: «Перекриття коду»

##### Мета: Набути практичних навичок написання та застосування програм перевірки пароля і виправлення пароля в ехе-файлах в середовищі masm64 з розміщенням даних на сторінках коду і з перекриттям коду.

##### Постановка задачі

##### Виконати завдання в кілька етапів:

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

##### 2. Використовувати процедуру перевірки пароля.

##### 3. Використовувати перевірку на підтримку команд AVX, яку оформити у вигляді виклику окремого ехе-файлу. У разі, якщо команди AVX мікропроцесором не підтримуються, то використовувати команди SSE.

##### 4. Вивести результати виконання програми через функцію MessageBoxIndirect. Програму і результати привести в звіті.

##### 5.Переіменовать секцію коду на несистемне ім'я.

##### 6. Привести скріншоти послідовності пошуку пароля.

##### Завдання

##### За допомогою команд AVX виконати рівняння, де змінна а приймає п'ять значень і задана масивом. Решта числа також задані в масиві {b, c, d, e}. Вивести результат через MessageBoxIndirect. Числа задані в масивах. Застосувати максимально можливу кількість прийомів з перекриттям коду

##### √a – cd – √(e\*b);

Результат виконання програми

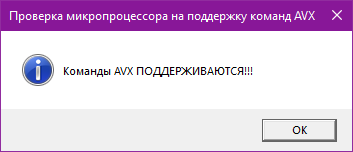


Рисунок 1- Результат виконання програми.

Натискаємо на кнопку меню ОК та вводимо правильний пароль

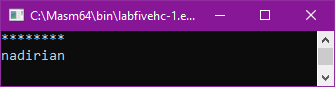


Рисунок 2 - вводимо пароль

Перевірка пароля, ОК.

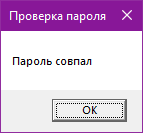


Рисунок 3 - Перевірка пароля

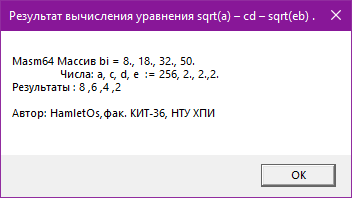


Рисунок 3 – Результат

1. Застосована технологія перекриття коду з переходом назад за кодом програми

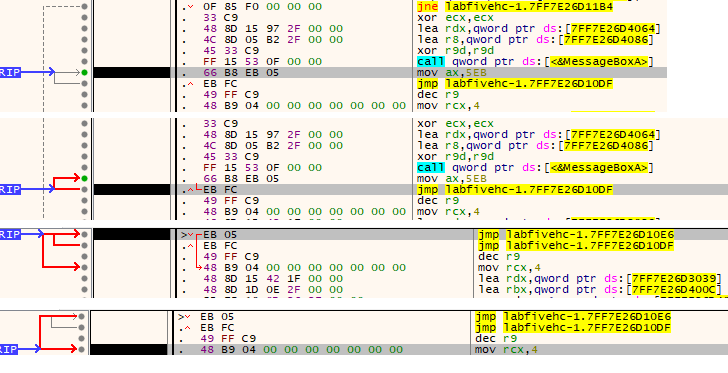


Рисунок 4 - Застосована технологія перекриття коду з переходом назад за кодом програми

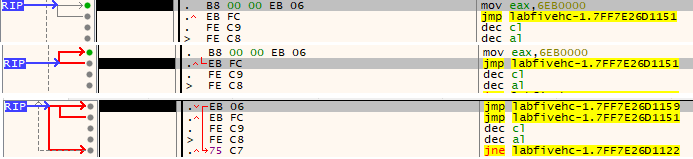


Рисунок 5 - Застосована технологія перекриття коду з переходом назад за кодом програми

2.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 00007FF6560311AD | EB 1C | |  |  |  |  | | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | |  | jmp labfivehc 1.7FF7F51411CB |  | | |

|  |  |  |
| --- | --- | --- |
| 00007FF7F51411CB | 33 С9 | xor ecx, ecx |

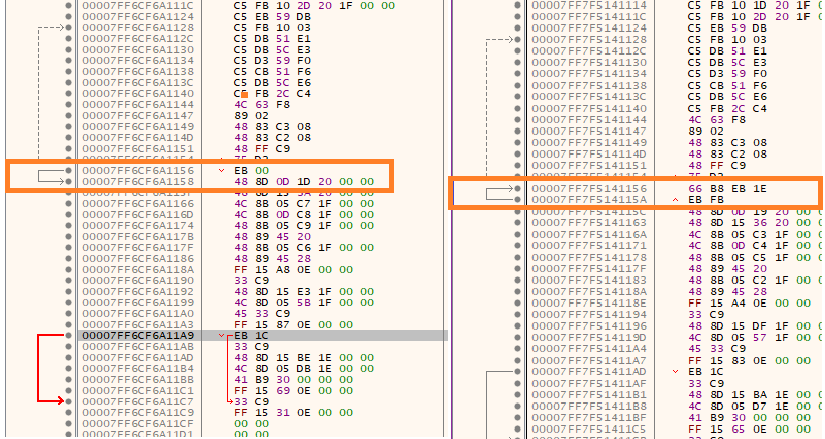
Ця команда розміщена за адресою 00007FF6560311AD а передає управління на адресу 00007FF6560311CB. Отримуємо різницю між адресами 1Eh

mov ax,1eebh

jmp $-3

;jmp m4

;m4:

Рисунок 6 - Команда jmp замінена на рядки коду

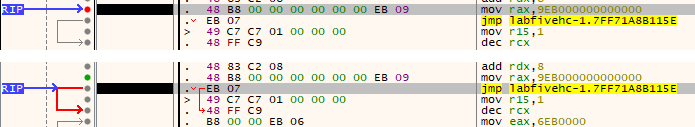
Застосована технологія перекриття коду з переходом вперед за кодом програми.

Рисунок 7 - Перекриття коду з переходом вперед за кодом програми.

Код програми

include win64a.inc

include kernel32.inc

mySg1 segment READ WRITE EXECUTE alias**(**"Ham"**)**

;.data

CmdCommand db "avxtest.exe"**,**0

mas1 **real8** 8.**,**18.**,**32.**,**50. ; массив чисел B

len2 equ **($-**mas1**)/**type mas1

mas2 **real8** 256.**,**2.**,**2.**,**2. ; a, c, d,e

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 ;

\_a dd 0;

\_b dd 0;

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

mySg1 ends

.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 WinExec**,addr** CmdCommand**,**SW\_SHOW ; юзать SW\_HIDE и окно будет скрыто

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

titll db "Результат вычисления уравнения sqrt(a) – cd – sqrt(eb) ."**,**0

res dq len1 DUP**(**0**),**0 ;

buf1 dd len1 DUP**(**0**),**0 ; буфер вывода сообщения

ifmt db "Masm64 Массив bi = 8., 18., 32., 50."**,**10**,**

9**,**"Числа: a, c, d, e := 256, 2., 2.,2."**,**10**,**

"Результаты : %d ,%d ,%d ,%d "**,**10**,**10**,**

"Автор: HamletOs,фак. КИТ-36, НТУ ХПИ"**,**10**,**

9**,**0

.code

**mov** **ax,**05EBh

**jmp** **$** **-** 2 ; пропуск 2 байтов EB 05 и остановка на EB

;EB 05 – это jmp на 5 байтов вперед с учетом своих 2-х байтов

**dec** **r9** ; пропускаем, 3 байта

**mov** **rcx,**len2

**lea** **rdx,**res

**lea** **rbx,**mas1

vmovsd **xmm1,**mas2**[**0**]** ; xmm1 — a — переслать real8

vmovsd **xmm2,**mas2**[**8**]** ; xmm2 — c

vmovsd **xmm3,**mas2**[**16**]** ; xmm3 — d

vmovsd **xmm5,**mas2**[**24**]**; ; xmm4 - e

vmulsd **xmm3,xmm2,xmm3**; ;xmm3 - c\*d

@@**:**

vmovsd **xmm0,qword** ptr**[rbx]**; - xmm0 - b

vsqrtsd **xmm4,xmm4,xmm1** ;xmm4 - a

vsubsd **xmm4,xmm4,xmm3** ;xmm4 - a-cd

vmulsd **xmm6,xmm5,xmm0**; xmm6 - (e\*b)

vsqrtsd **xmm6,xmm6,xmm6** ;xmm6 - sqrt(e\*b)

vsubsd **xmm4,xmm4,xmm6**; a-cd-sqrt(e\*b)

;vcvttsd2si eax,xmm4 ; от

vcvttsd2si **eax,xmm4** ; для отладки

**movsxd** **r15,eax** ;

**mov** **[rdx],eax** ; сохранение результата

**add** **rbx,**8

**add** **rdx,**8

**mov** **rax,**09EB000000000000h

**jmp** **$** **+** 9 ; перепрыгивает свои 2 байта и следующие 7

**mov** **r15,**1 ;

**dec** **rcx**

**mov** **eax,**06EB0000h

**jmp** **$** **-** 2 ; пропуск 2 байтов 06 и EB и остановка на EB

; EB 06 – это jmp на 6 байтов вперед с учетом своих 2-х байтов

**dec** **cl** ; пропускаем, 2 байта: FE C9

**dec** **al** ; пропускаем, 2 байта: FE C8

**jnz** **@b** ; ссылка на предыдущую метку @@ (наверх)

**mov** **ax,**1eebh

**jmp** **$-**3

;jmp m4

m4**:**

invoke wsprintf**,addr** buf1**,addr** ifmt**,**res**,**res**[**8**],**res**[**16**],**res**[**24**]**

invoke MessageBox**,**0**,addr** buf1**,addr** titll**,**MB\_OK

.else

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

.endif

invoke ExitProcess**,**0

WinMain endp

end

**Висновок**

Набули практичні навички написання та застосування програм перевірки пароля і виправлення пароля в ехе-файлах в середовищі masm64 з розміщенням даних на сторінках коду і з перекриттям коду.