MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE

NATIONAL TECHNICAL UNIVERSITY

"KHARKIV POLYTECHNICAL INSTITUTE"

Department of Computer Engineering and Programming

«Software Means of Information Protection »

*Laboratory work report No 6*

*Topic: «* **Self-modifying Code** *»*

Student:

Group. KH 919 I.E

Ayoub EL-HADDADI

Verified by:

Lecturer Viktor CHELAK

Kharkiv – 2022

***Purpose of work***:

To acquire practical skills in writing and applying self-modifying programs in masm64 environment.

***Individual task:***

Write the program in the environment of masm64 according to the variant of the job using the maximum number of methods of self-modification of the code.

Variant 8: A 3 x 6 matrix is specified. Define the elements of multiples of 3 in each row and place in their place an element whose number coincides with the row number.

**Algorithm of the program**

In first program we create a loop that goes through 3x6 array elements, and we set a variable that increases and reinitialized every row with increasing another variable every row. Next step is check array element if is a multiple of number 3, and if it is we add row number to result array if not we add the current element.

We separate string result into two parts since we have 18 result and cannot have 18 parameters in wsprintf function.

*#self-modification part:*

We initialize opcode to replace the written ones like the following:

MOV rax, r12 - opcode1 dd 0E08B4Ch - 3 bytes

MOVD rax, xmm1 - opcode2 dq 0C87E0F4866h - 5 bytes

ADD rsi, 8 - opcode3 dq 08C68348h - 4 bytes

INC r9 - opcode4 dd 0C1FF49h - 3 bytes

Loop m1 - opcode5 dd 0B4E2h - 2 bytes

And, we invoke WriteProcessMemory with addresses of lines to change.

**Source Code**

Full source code of this lab you can find it in:

[**https://github.com/Elh-Ayoub/RP\_Labs/tree/main/lab6**](https://github.com/Elh-Ayoub/RP_Labs/tree/main/lab6)

**Results of the program:**

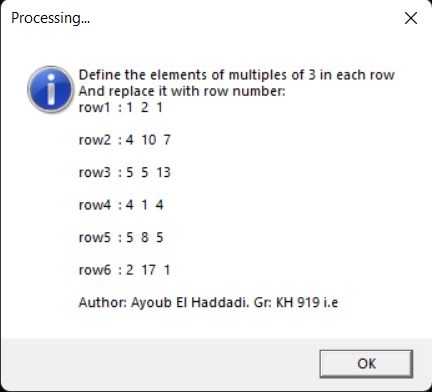


Figure 1 – Program results

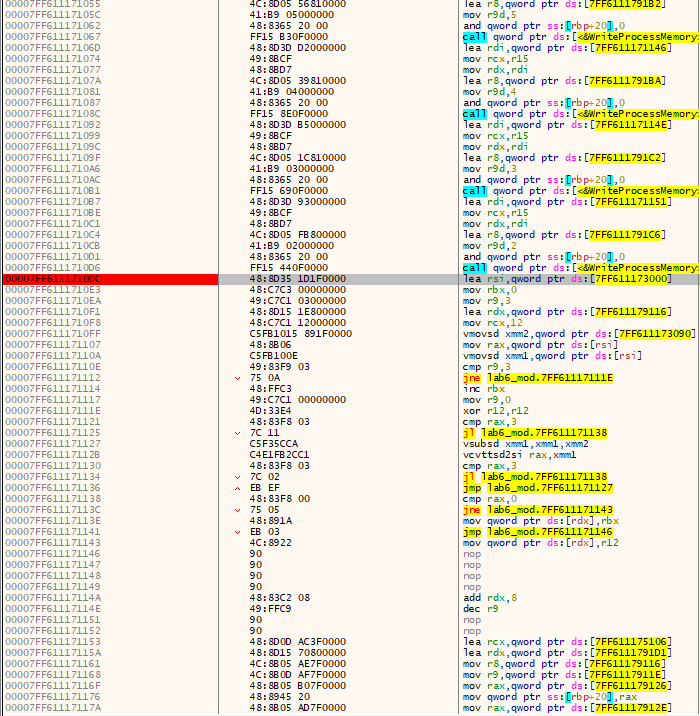


Figure 2 – Modifies program before invoke WriteProcessMemory

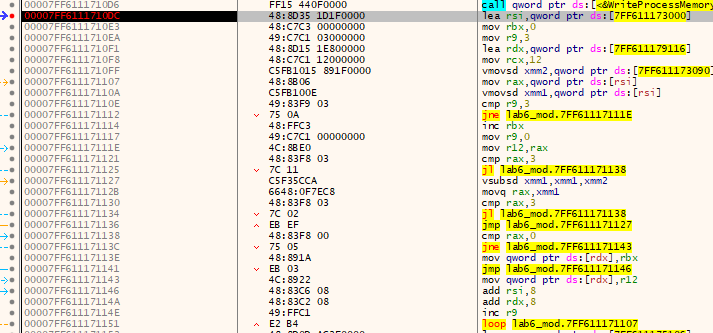


Figure 2 – Modifies program After invoke WriteProcessMemory

**Conclusions:**

As a result of laboratory work we gained a practical skills in writing and a self-modifying programs in masm64 environment, working with WriteProcessMemory and calculation of opcodes.

**You can also find this report in:**

<https://github.com/Elh-Ayoub/RP_Labs/tree/main/Docs>