**Міністерство освіти і науки України**

**Національний технічний університет України**

**«Київський політехнічний інститут імені Ігоря Сікорського»**

**Факультет інформатики та обчислювальної техніки**

**Кафедра обчислювальної техніки**

**Лабораторна робота №3**

з дисципліни

«Системне програмування»

на тему

«Дослідження структури програм формату EXE»

Виконала: Перевірив:

студент групи ІП-93 доц. кафедри ОТ

Узунлу Гамзенур Павлов В. Г.

номер залікової книжки: 9822

Київ 2021

Код програми

include \masm32\include\masm32rt.inc

.data

pwd db "qwerty"

info db "Узунлу Гамзенур, 9882, IP-93", 0

mbTitle db "info", 0

start db "Insert password: ", 0

bad db "Wrong password. Try again", 0

good db "Correct password", 0

inserted db 64 dup(0)

.const

IDC\_OK equ 1001

IDC\_FIELD equ 1002

IDC\_TEXT equ 1003

.code

DlgProc Proc hWnd:HWND, uMsg:UINT, wParam:WPARAM, lParam:LPARAM

.if uMsg == WM\_COMMAND

.if wParam == IDC\_OK

invoke SetDlgItemText, hWnd, IDC\_TEXT, addr bad

invoke GetDlgItemText, hWnd, IDC\_FIELD, addr inserted, 16

cmp eax, 6

jne \_exit

mov edi, 0

cmpSymb:

mov bl, pwd[edi]

mov bh, inserted[edi]

cmp bl, bh

jne \_exit

inc edi

cmp edi, 6

jne cmpSymb

jmp \_good

\_good:

invoke SetDlgItemText, hWnd, IDC\_TEXT, addr good

invoke MessageBox, hWnd, addr info, addr mbTitle, 0

\_exit:

.endif

.elseif uMsg == WM\_CLOSE

invoke EndDialog, hWnd, NULL

.else

xor eax, eax

.endif

ret

DlgProc endp

begin:

invoke GetModuleHandle, NULL

Dialog "Лабораторна робота 3", "MS Sans Serif", 10, WS\_OVERLAPPED+DS\_CENTER+WS\_SYSMENU, 3, 0, 0, 100, 50, 1024

DlgStatic "Insert password: ", SS\_CENTER, 0, 0, 100, 10, IDC\_TEXT

DlgButton "Get info", WS\_TABSTOP, 20, 20, 60, 10, IDC\_OK

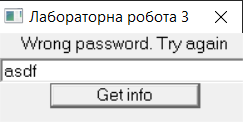
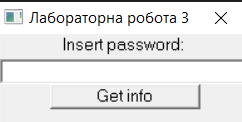
DlgEdit WS\_TABSTOP+WS\_BORDER, 0, 10, 100, 10, IDC\_FIELD

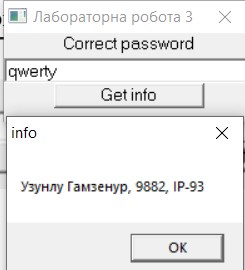
CallModalDialog NULL, NULL, DlgProc, NULL

invoke ExitProcess, NULL

end begin

Скриншот програми

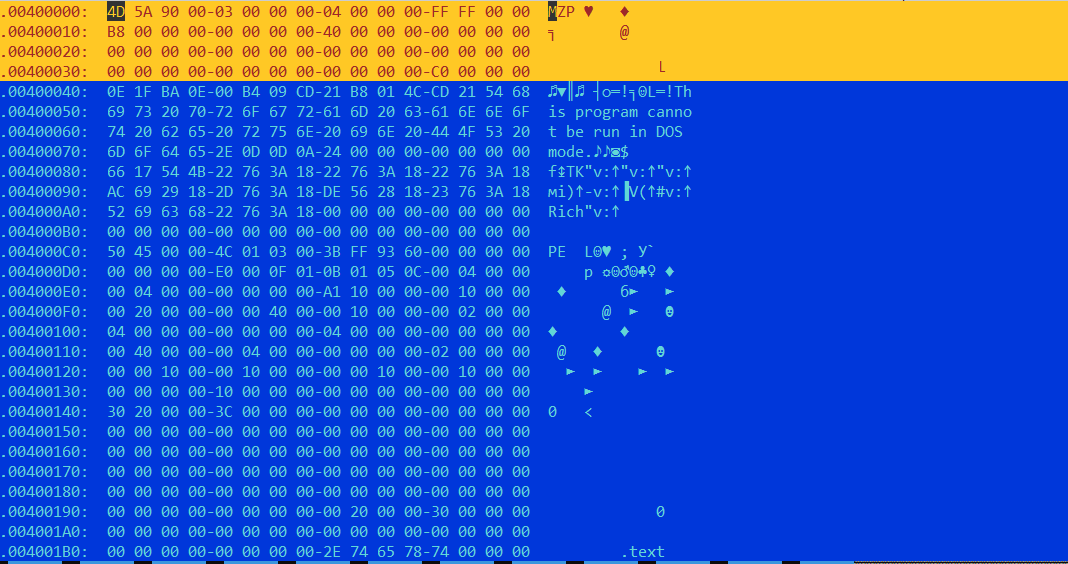




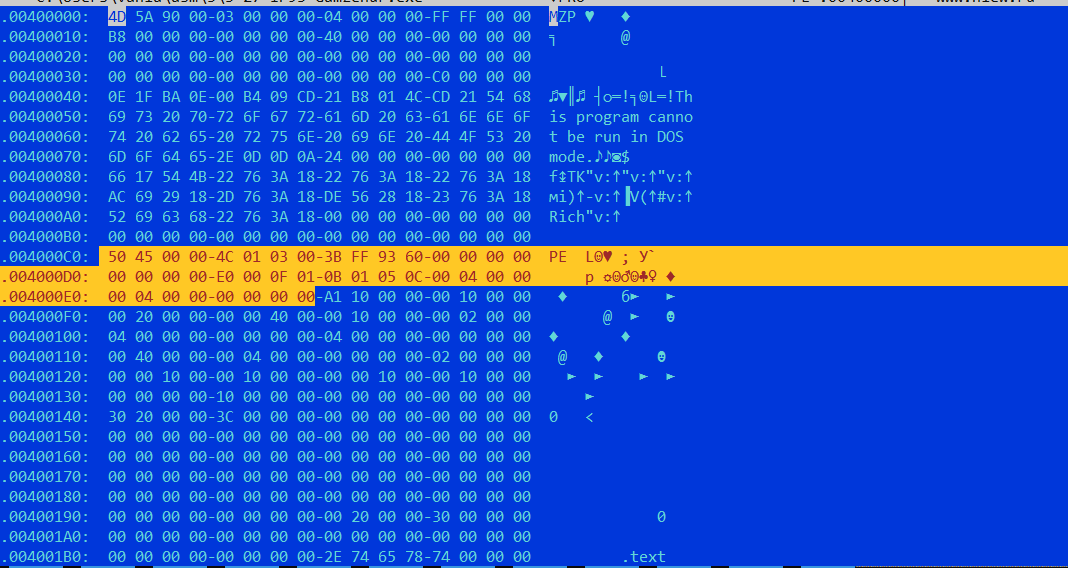
HIEW32



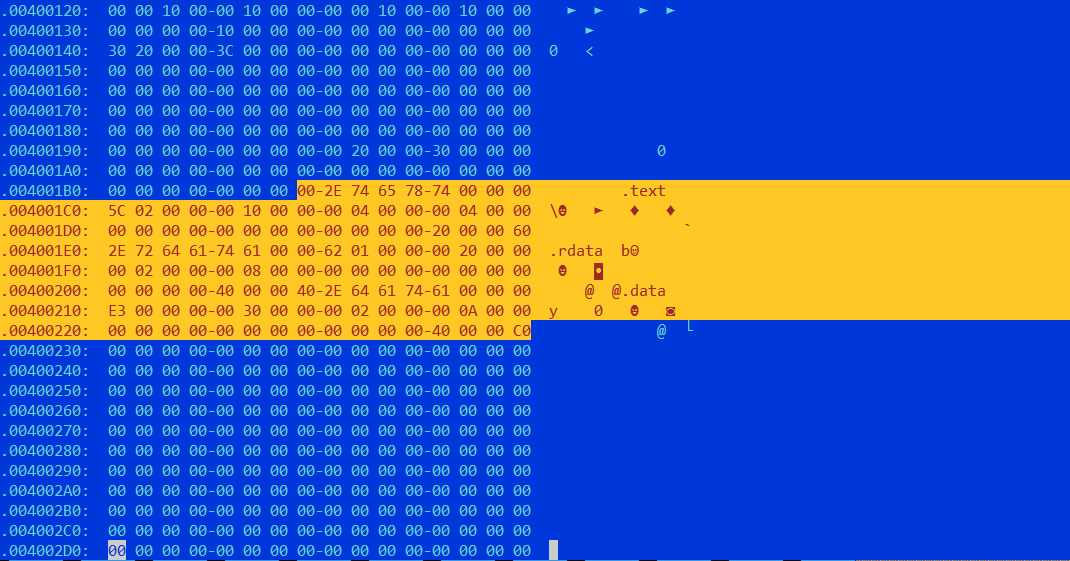
Dos header

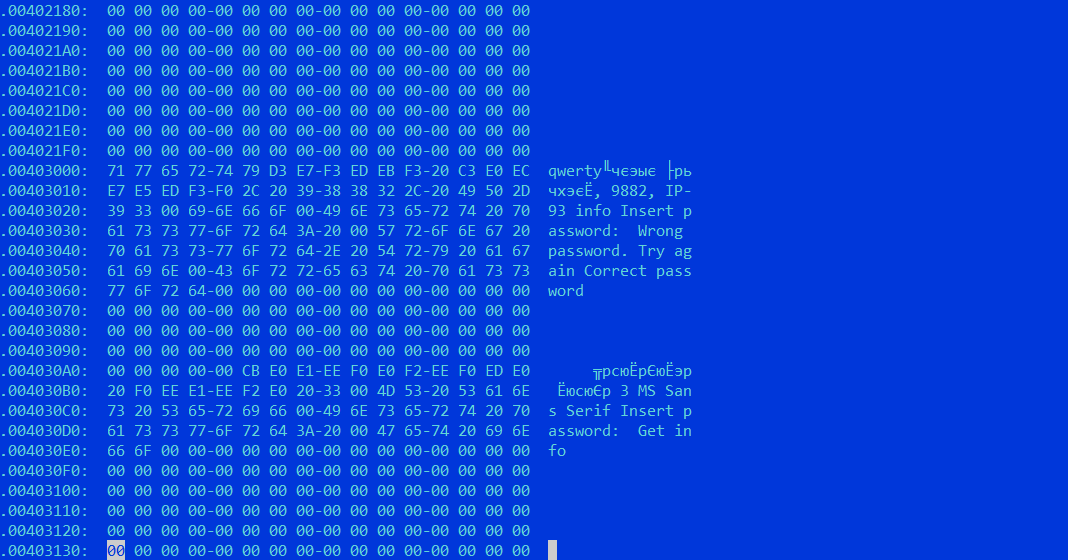


PE header



Sections

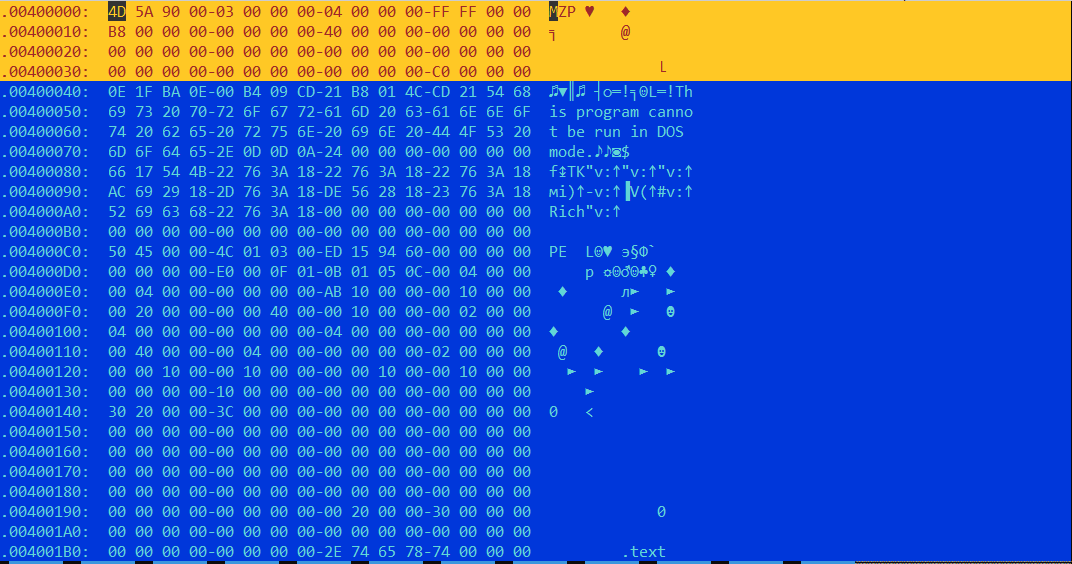




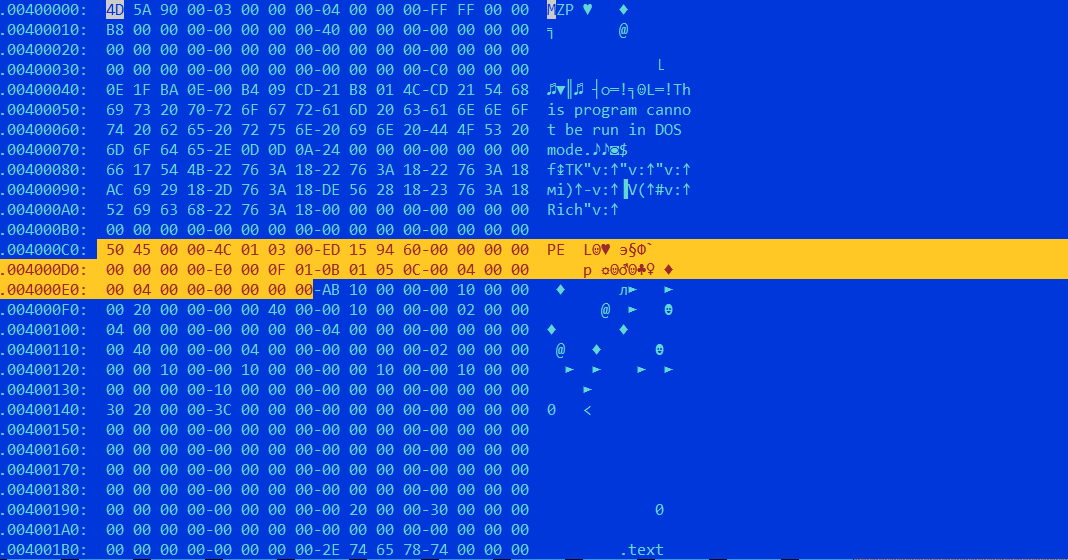
2 версія програми



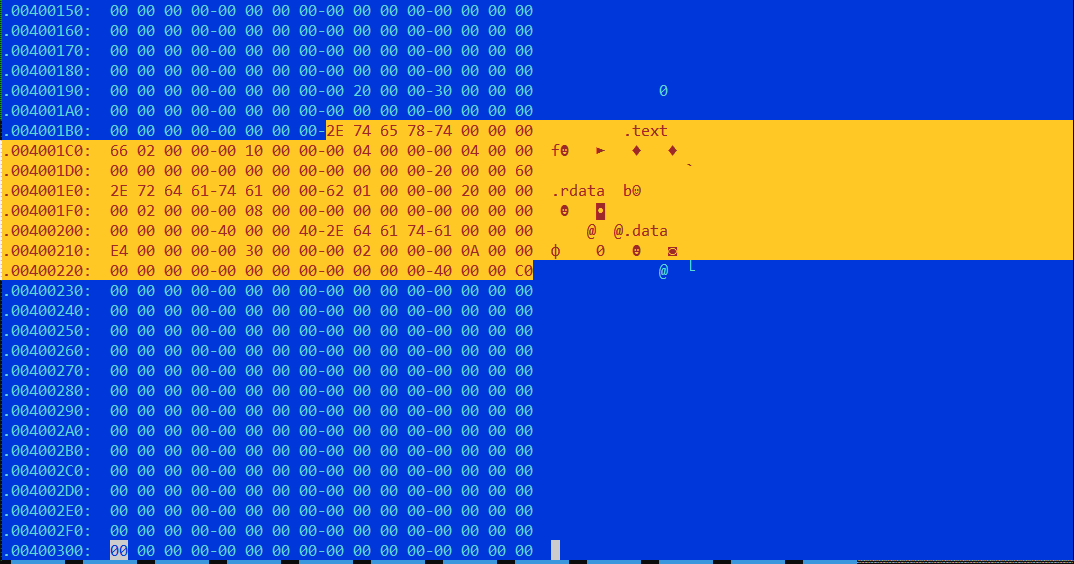
Dos header

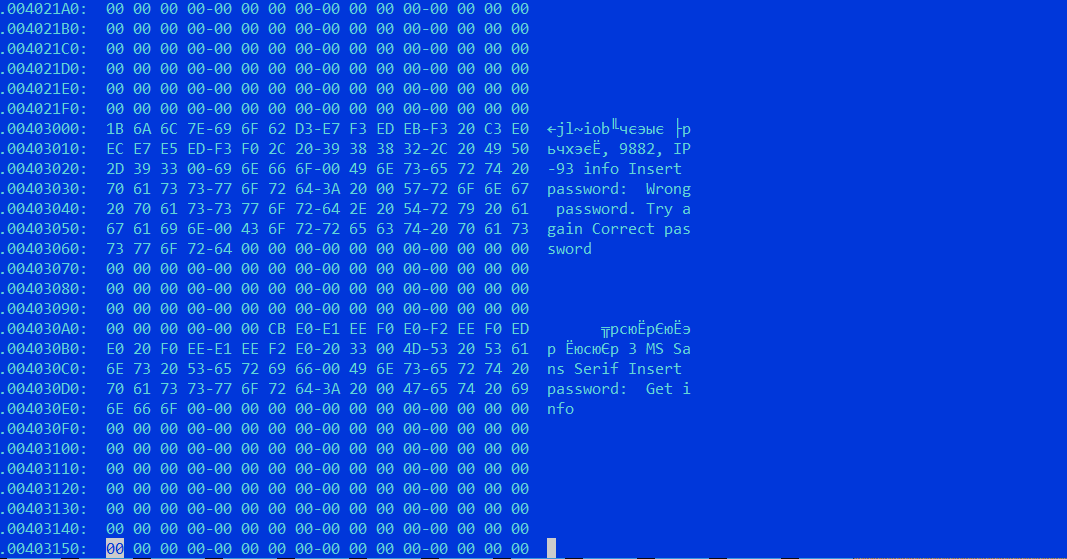


PE header



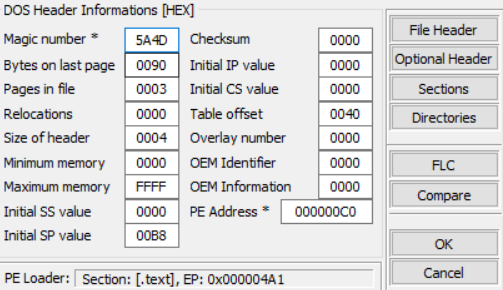
Sections

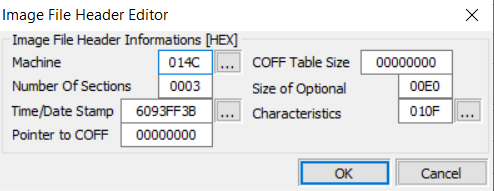


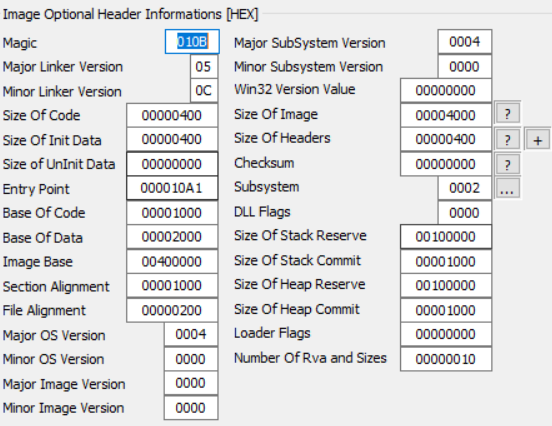


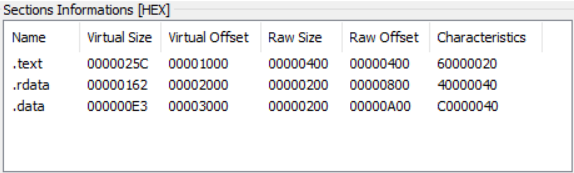
Бачимо, що в першій версій програми пароль видно у вкладках TEXT та HEX. А у другій версії - він зашифрований.

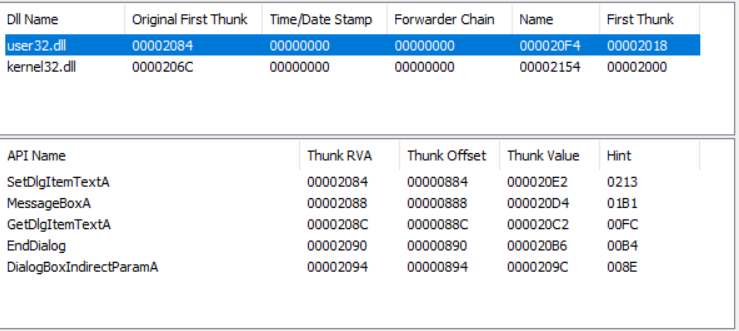
PE TOOLS

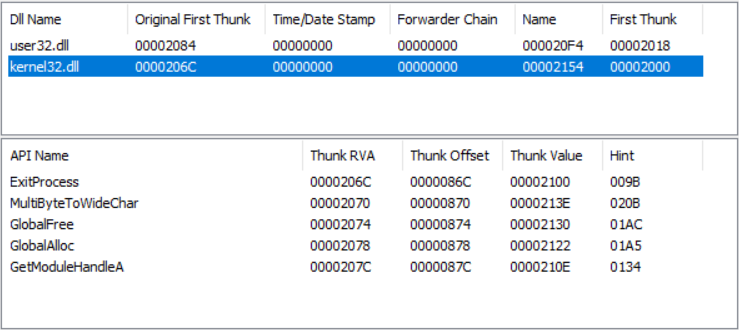












Параметри заголовків

|  |  |  |
| --- | --- | --- |
| Field | Value | Description |
| Machine type | 014C | Number identifying type of target machine. |

|  |  |  |
| --- | --- | --- |
| Field | Value | Description |
| Magic | 010B | Unsigned integer identifying the state of the image file. The most common number is 0413 octal (0x10B), identifying it as a normal executable file. 0407 (0x107) identifies a ROM image. |
| MajorLinkerVersion | 05 | Linker major version number. |
| MinorLinkerVersion | 0C | Linker minor version number. |
| SizeOfCode | 00000400 | Size of the code (text) section, or the sum of all code sections if there are multiple sections. |
| SizeOfInitializedData | 00000400 | Size of the initialized data section, or the sum of all such sections if there are multiple data sections. |
| SizeOfUninitializedData | 00000000 | Size of the uninitialized data section (BSS), or the sum of all such sections if there are multiple BSS sections. |
| AddressOfEntryPoint | 000010A1 | Address of entry point, relative to image base, when executable file is loaded into memory. For program images, this is the starting address. For device drivers, this is the address of the initialization function. An entry point is optional for DLLs. When none is present this field should be 0. |
| BaseOfCode | 00001000 | Address, relative to image base, of beginning of code section, when loaded into memory. |
| BaseOfData | 00002000 | Address, relative to image base, of beginning of data section, when loaded into memory. |

Sections

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Value | | | Description |
| .text | .rdata | .data |
| Name | 2E746578740000 | 000061746164722E | 000000617461642E | An 8-byte, null-padded ASCII string. There is no terminating null if the string is exactly eight characters long. For longer names, this field contains a slash (/) followed by ASCII representation of a decimal number: this number is an offset into the string table. Executable images do not use a string table and do not support section names longer than eight characters. Long names in object files will be truncated if emitted to an executable file. |
| VirtualSize | 00000266 | 00000162 | 000000E4 | Total size of the section when loaded into memory. If this value is greater than Size of Raw Data, the section is zero-padded. This field is valid only for executable images and should be set to 0 for object files. |
| VirtualAddress | 00001000 | 00002000 | 00003000 | For executable images this is the address of the first byte of the section, when loaded into memory, relative to the image base. For object files, this field is the address of the first byte before relocation is applied; for simplicity, compilers should set this to zero. Otherwise, it is an arbitrary value that is subtracted from offsets during relocation. |
| SizeOfRawData | 00000400 | 00000200 | 00000200 | Size of the section (object file) or size of the initialized data on disk (image files). For executable image, this must be a multiple of FileAlignment from the optional header. If this is less than VirtualSize the remainder of the section is zero filled. Because this field is rounded while the VirtualSize field is not it is possible for this to be greater than VirtualSize as well. When a section contains only uninitialized data, this field should be 0. |
| PointerToRawData | 00000400 | 00000800 | 00000A00 | File pointer to section’s first page within the COFF file. For executable images, this must be a multiple of FileAlignment from the optional header. For object files, the value should be aligned on a fourbyte boundary for best performance. When a section contains only uninitialized data, this field should be 0. |
| PointerToRelocations | 00000000 | 00000000 | 00000000 | File pointer to beginning of relocation entries for the section. Set to 0 for executable images or if there are no relocations. |
| PointerToLinenumbers | 00000000 | 00000000 | 00000000 | File pointer to beginning of line-number entries for the section. Set to 0 if there are no COFF line numbers. |
| NumberOfRelocations | 0000 | 0000 | 0000 | Number of relocation entries for the section. Set to 0 for executable images. |
| NumberOfLinenumbers | 0000 | 0000 | 0000 | Number of line-number entries for the section. |
| Characteristics | 60000020 | 40000040 | C0000040 | Flags describing section’s characteristics. See Section 4.1, “Section Flags,” for more information |

Висновки

Було розроблено 2 програми на мові асемблер. В програмах ми виводимо у вікні дані студента при вводі правильного пароля. Програми скомпілювалися у формат exe. Контент програм було переглянуто у редакторах HIEW32 та PE TOOLS. З заголовків було виписано значення полів для таблиці. Ці ж значення було знайдено у вкладках PE TOOLS. Значення полів співпадають.