**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 №2**

з дисципліни  
«Технології розроблення системних програм»

на тему  
«Дослідження структури програм формату COM.»

Виконав: Перевірив:

студент групи ІП-84 доц. кафедри ОТ  
Голубов Іван Олегович Павлов В. Г.  
номер залікової книжки: 8404

Київ 2020

**Мета роботи**

Вивчення прийомів написання, компіляції і відладки програм формату COM в середовищі Masm32. Здобуття навичок читання лістингу і розшифрування кодів команд.

**Порядок виконання роботи**

1. Вивчити структуру програм формату COM і застосування переривань BIOS та MS-DOS при роботі з консоллю.
2. Розробити програму на мові Асемблер, за допомогою якої на екран системної консолі по введеному паролю виводяться персональні дані студента – ПІБ, дата народження, номер залікової книжки тощо (див. лаб. роботу 1) кожне з нового рядка з попереднім очищенням вікна системної консолі. Для правильного відображення при виведенні символів кирилиці підключити відповідну кодову сторінку.
3. Вивчити опції компілятора і лінковщика і сформувати BAT-файл, в якому передбачити завдання назви вихідного файлу .asm, як параметра. Шлях до файлу має бути визначений в результаті сканування логічного диску і служити для вказівки розміщення відповідних йому об'єктного і виконуваного файлів
4. Виконати компіляцію розробленого файлу у формат COM.
5. Перевірити роботу програми шляхом введення як правильного, так і невірного паролів.
6. Отриманий виконуваний файл дослідити за допомогою програми HEX-редактору HIEW32 або HIEW.
7. Перемикаючи послідовно режими перегляду (**Text** – **Hex** – **Decode**), зняти три відповідних скріншоти програми і привести їх в звіті по лабораторній роботі.
8. Переконатися, що текст оригінала пароля, який міститься в тексті програми, може бути легко виявлений за допомогою HEX-редактора.
9. Виконати шифрування пароля за допомогою функції XOR, знову скомпілювати COM-файл і переконатися, що тепер вони не виявляються явним чином в тексті виконуваного COM-файлу. Привести скріншоти цієї програми в режимах «**Text**» та «**Decode**» у звіті по лабораторній роботі.
10. Порівняти текст програми, який набирався в редакторові, з текстом програми в скомпільованому вигляді, який формує HEX-редактор HIEW. Виявити розбіжності і відобразити їх в звіті по лабораторній роботі.
11. На отриманому у п. 9 в режимі «**Decode**» скріншоті знайти всі команди **MOV**, виписати їх коди і розібрати по окремих полях, відповідно їх формату. Результати привести в звіті по лабораторній роботі.
12. Зробити висновки по лабораторній роботі.

**Bat-file**

@echo off

setlocal

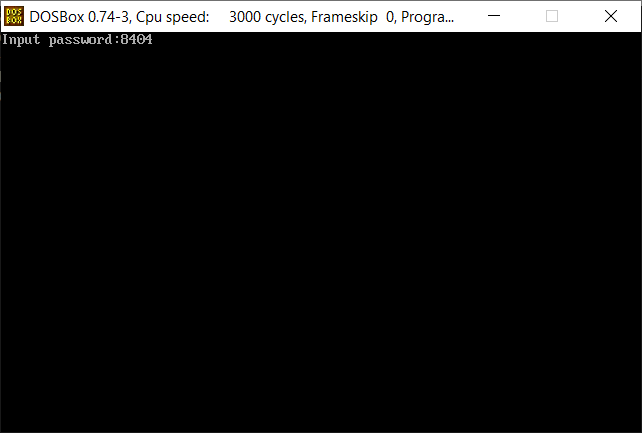
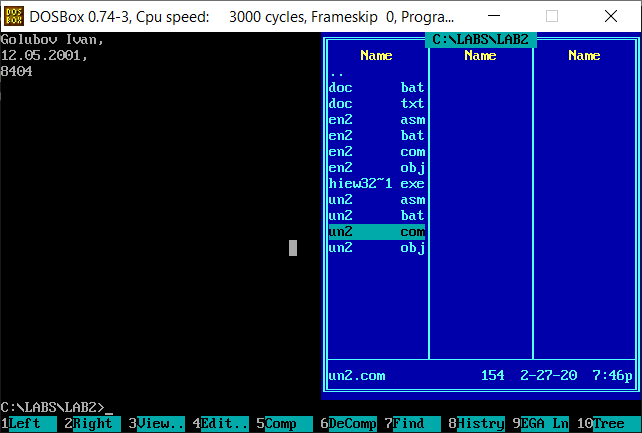
cd /d %1

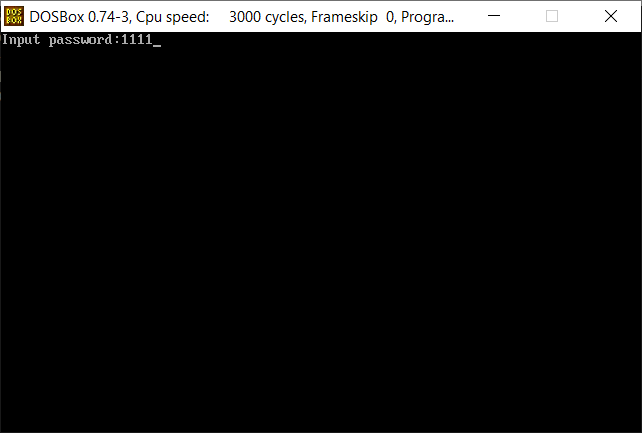
for /f "delims=" %%a in ('dir /s /b en2.asm') do F:\masm32\bin\ml /Bl F:\masm32\bin\link16.exe "%%a"

endlocal

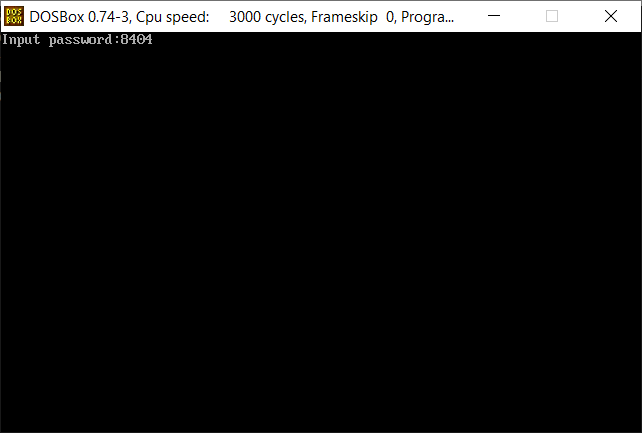
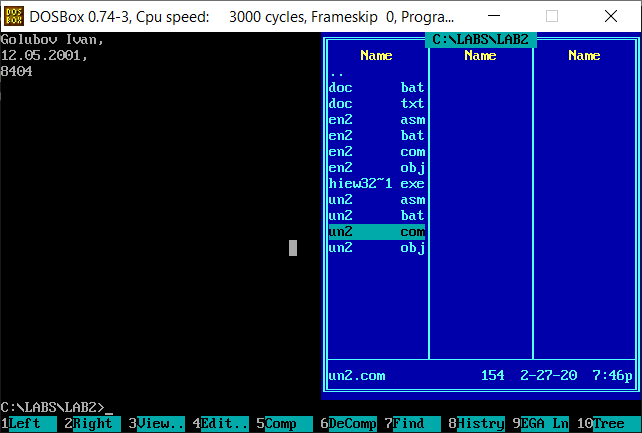
**Результати роботи програми**

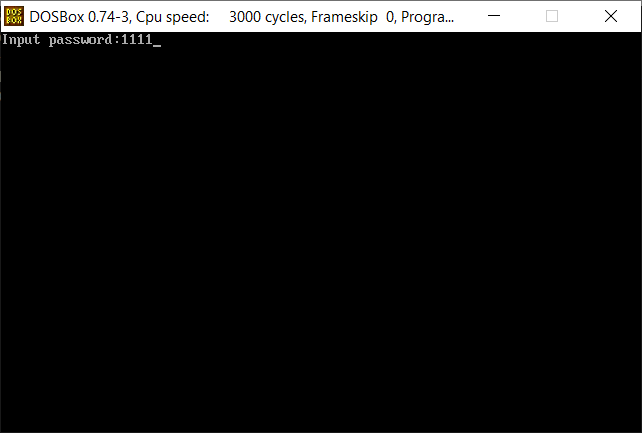
**Без шифрування**



**З шифруванням**



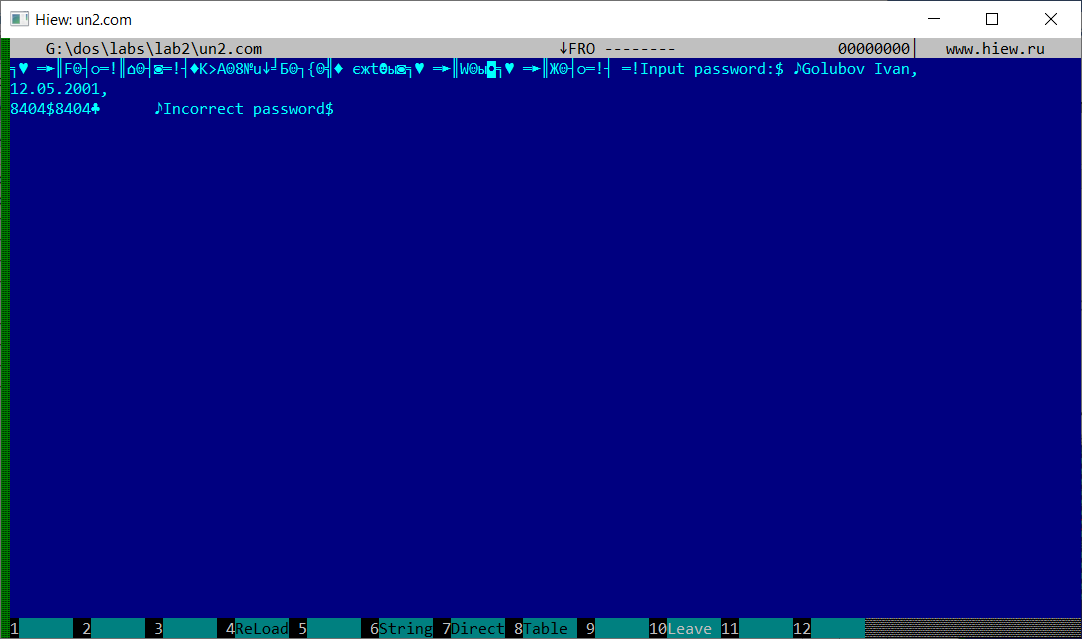
**Вихідний код**

|  |  |
| --- | --- |
| **Default** | **Decode Default** |
| .model tiny  .data  InpMassage db "Input password:$ ";  Data db 13, "Golubov Ivan, ",  13, 10, "12.05.2001, ", 13, 10, "8404$";  password db 38h, 34h, 30h, 34h;  PassLen=$-password  MaxInput db 5;  InputLen db 0;  buffer db 5 dup (0);  error db 13, "Incorrect password$";    .code  org 100h    main:  mov ax, 03h  int 10h  mov dx, offset InpMassage;  mov ah,09h;  int 21h    mov dx, offset MaxInput;  mov ah, 0Ah;  int 21h    mov ah, PassLen;  mov bh, InputLen;  cmp ah, bh;  jne false  mov si, offset buffer;  mov di, offset password;  mov cx, PassLen;  repe cmpsb;  je true  jmp false    true:  mov ax, 03h  int 10h  mov dx, offset Data;  jmp exit    false:  mov ax, 03h  int 10h  mov dx, offset error;  exit:  mov ah, 09h;  int 21h;    mov ah, 0;  int 21h    end main | mov ax,3 int 010  mov dx,00146 ;'☺F'  mov ah,9 int 021 ;'!'  mov dx,0017F ;'☺⌂'  mov ah,00A int 021 ;'!'  mov ah,4 mov bh,[00180] cmp ah,bh jnz 000000036 --↓1  mov si,00181 ;'☺Б' mov di,0017B ;'☺{' mov cx,4 rep cmpsb jz 00000002C --↓2 jmps 000000036 --↓1  2mov ax,3 int 010 mov dx,00157 ;'☺W' jmps 00000003E --↓3  1mov ax,3 int 010 mov dx,00186 ;'☺Ж'  3mov ah,9 int 021 ;'!'  mov ah,0  int 021 ;'!' |

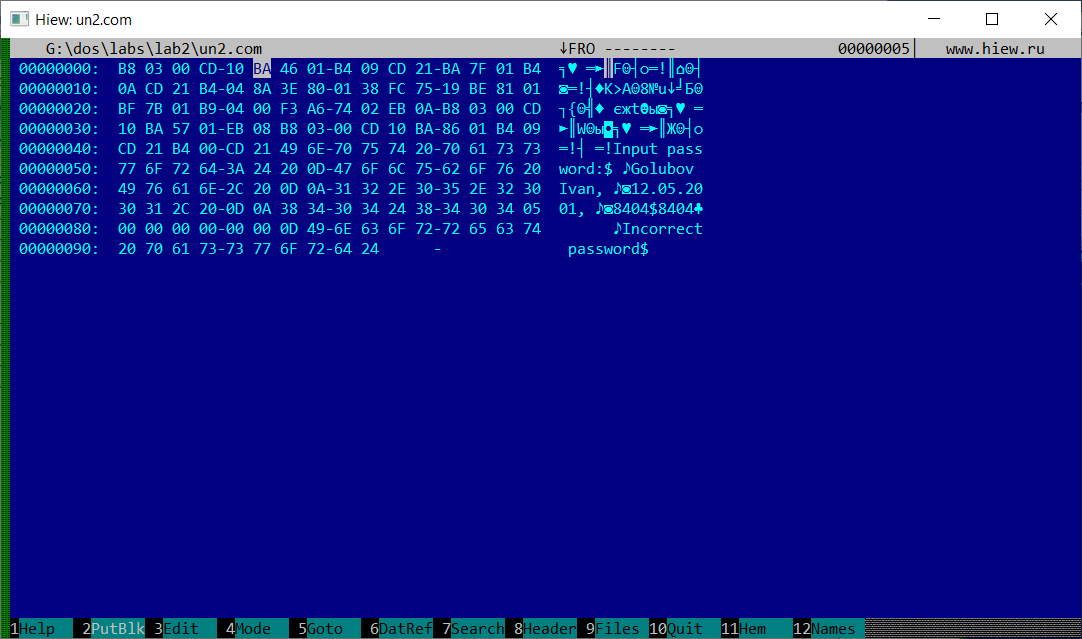
|  |  |
| --- | --- |
| **XOR** | **Decode XOR** |
| . .model tiny  .data  password db 03Ah, 036h, 032h, 036h;  PassLen=$-password  key equ 002h;  Data db 13, "Golubov Ivan, ",  13, 10, "12.05.2001, ", 13, 10, "8404$";  InpMassage db "Enter your password:$ ";  MaxInput db 5;  InputLen db 0;  buffer db 5 dup (0);  error db 13, "Incorrect password$";  .code  org 100h    main:  mov ax, 03h  int 10h    xor di, di  hash:  xor password[di], key  inc di  cmp di, 4  jne hash  xor di, di  cycle:  xor buffer[di], key  inc di  cmp di, 4  jne cycle    mov dx, offset InpMassage;  mov ah,09h;  int 21h    mov dx, offset MaxInput;  mov ah, 0Ah;  int 21h    mov ah, PassLen;  mov bh, InputLen;  cmp ah, bh;  jne false  mov si, offset buffer;  mov di, offset password;  mov cx, PassLen;  repe cmpsb;  je true  jmp false    true:  mov ax, 03h  int 10h  mov dx, offset Data;    jmp exit    false:  mov ax, 03h  int 10h  mov dx, offset error;    exit:  mov ah, 09h;  int 21h    mov ah, 0;  int 21h;    end main | mov ax,3 int 010  xor di,di  1xor b,[di][00160],2  inc di cmp di,4 jnz 000000007 --↑1  xor di,di  2xor b,[di][001A0],2 inc di cmp di,4 jnz 000000014 --↑2  mov dx,00188 ;'☺И' mov ah,9 int 021 ;'!'  mov dx,0019E ;'☺Ю' mov ah,00A int 021 ;'!'  mov ah,4 mov bh,[0019F] cmp ah,bh jnz 000000050 --↓3  mov si,001A0 ;'☺а' mov di,00160 ;'☺`' mov cx,4 rep cmpsb jz 000000046 --↓4 jmps 000000050 --↓3  mov ax,3 int 010 mov dx,00164 ;'☺d'  jmps 000000058 --↓1  mov ax,3 int 010 mov dx,001A5 ;'☺е'  1mov ah,9 int 021 ;'!'  mov ah,0 int 021 ;'!' |

**Скріншоти**

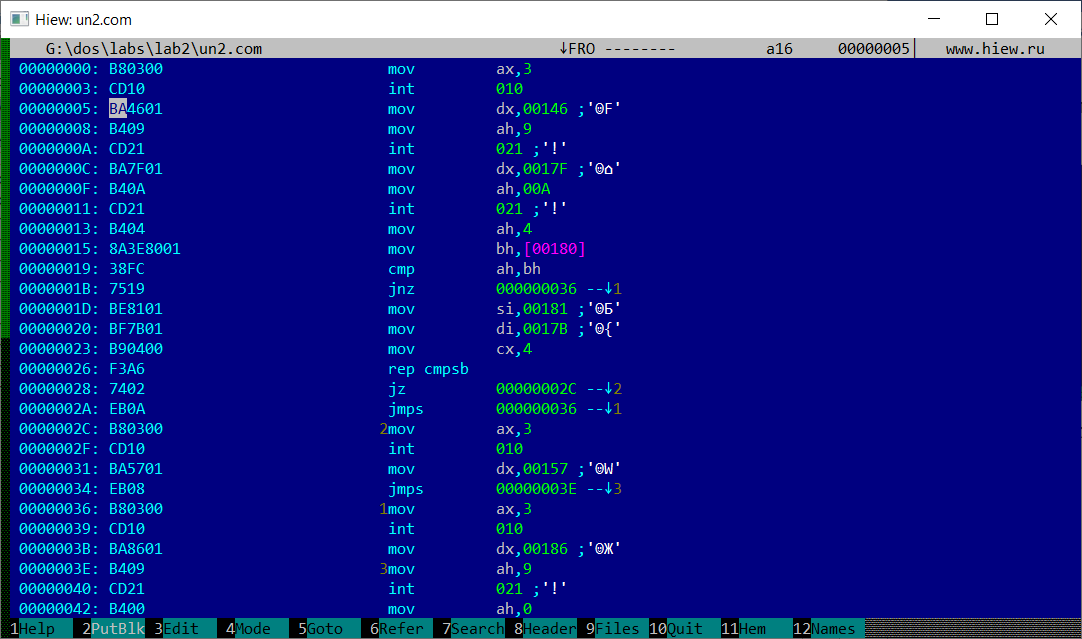
Default-Text:



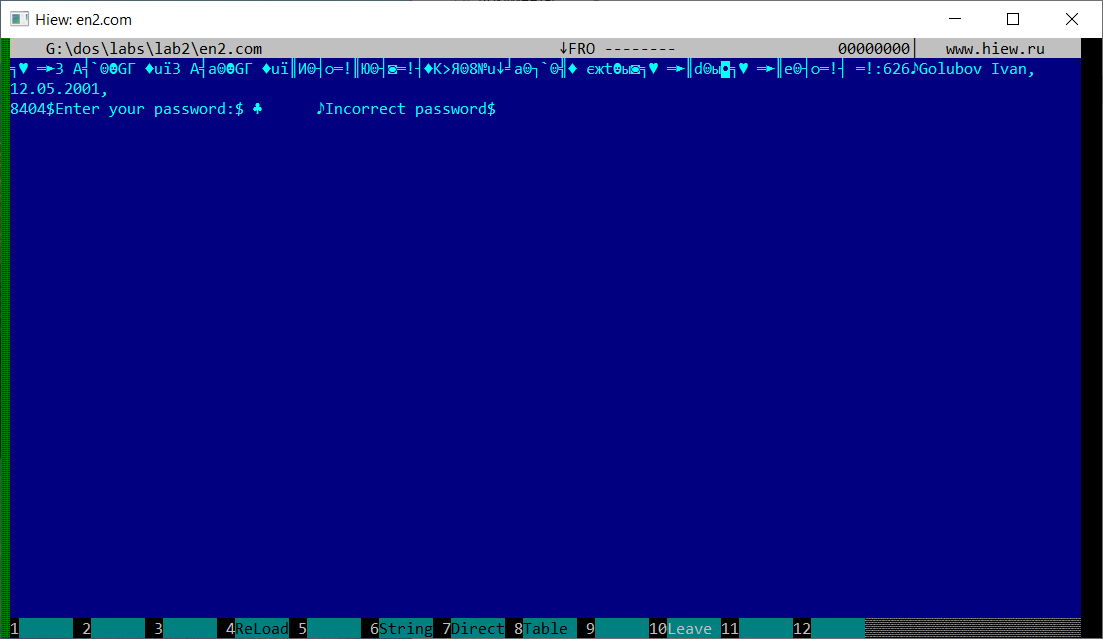
Default-Hex:



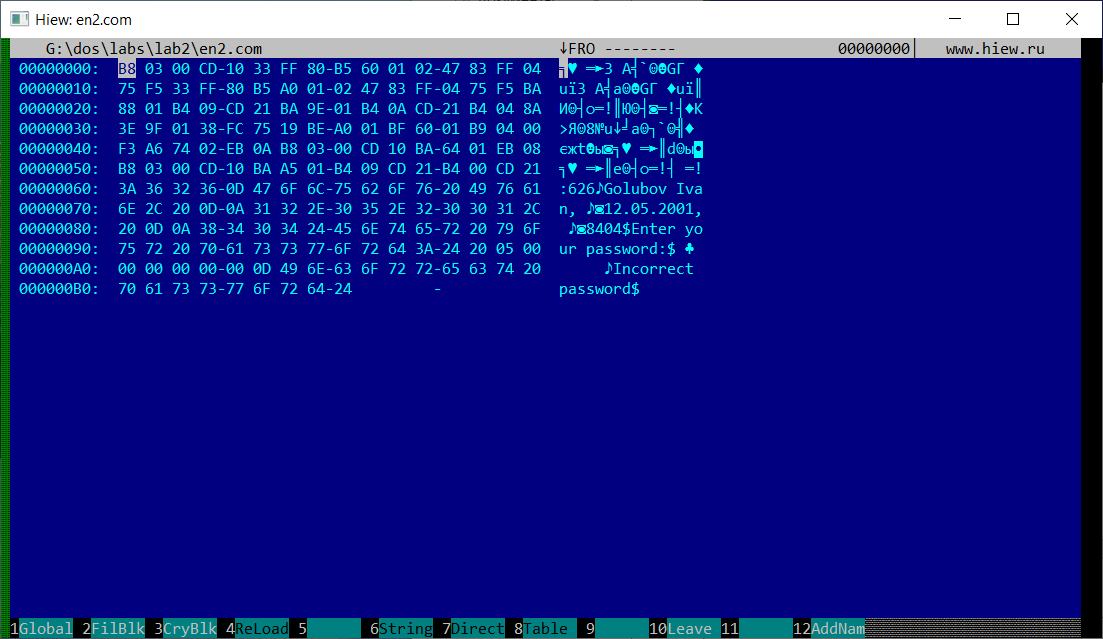
Default-Decode:



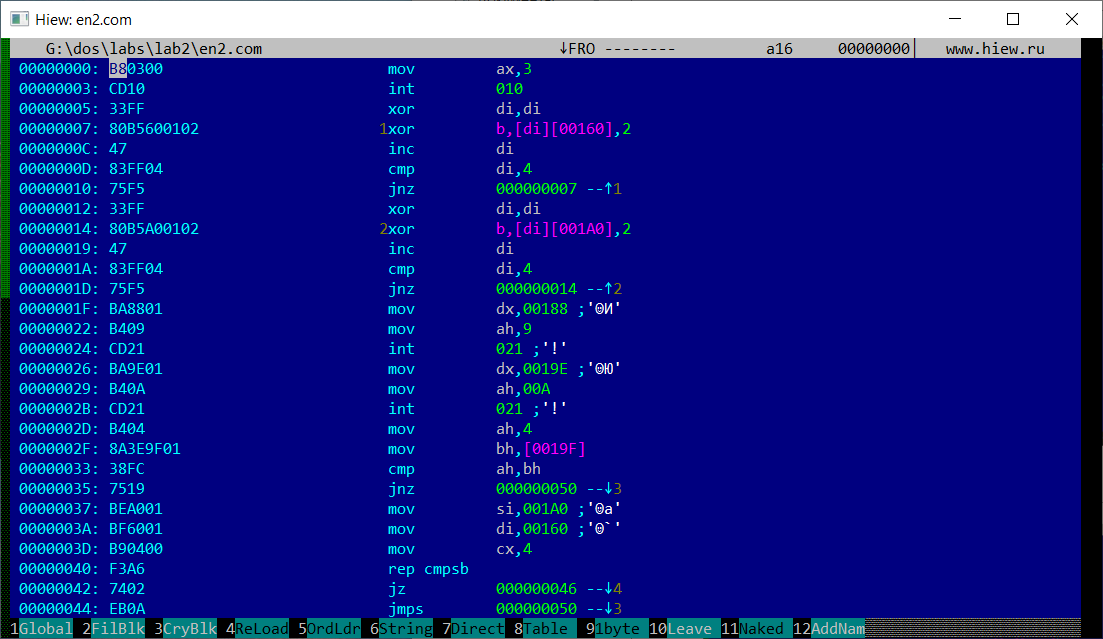
XOR-Text:



XOR -Hex:



XOR-Decode:



**Команди MOV**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Код** | **Код операції** | **Mod R/M** | **SIB** | **Зсув** | **Безпосер. операнд** |
| **mov** ax, 3 | B8 | - | - | - | 03 00 |
| **mov** dx,00188 | BA | - | - | - | 88 01 |
| **mov** ah, 9 | B4 | - | - | - | 09 |
| **mov** dx,0019E | BA | - | - | - | 9E 01 |
| **mov** ah,00A | B4 | - | - | - | 0A |
| **mov** ah,4 | B4 | - | - | - | 04 |
| **mov** bh,[0019F] | 8A | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 0 | 0 | 1 | 1 | **3** | | 1 | 1 | 1 | 0 | **E** | | - | 9F 01 | - |
| **mov** si,001A0 | BE | - | - | - | A0 01 |
| **mov** di,00160 | BF | - | - | - | 60 01 |
| **mov** cx,4 | B9 | - | - | - | 04 00 |
| **mov** ax, 3 | B8 | - | - | - | 03 00 |
| **mov** dx,00164 | BA | - | - | - | 64 01 |
| **mov** ax, 3 | B8 | - | - | - | 03 00 |
| **mov** dx,001A5 | BA | - | - | - | A5 01 |
| **mov** ah,9 | B4 | - | - | - | 09 |
| **mov** ah,0 | B4 | - | - | - | 00 |

**Висновки**

Склавши дві версії програми та провівши аналіз виконуваного файлу в редакторі HIEW32, ми здобули навички написання MS-DOS застосунків,  
а також роботи з системними перериваннями. Існуючу програму було удосконалено шифруванням функцією XOR для приховування явного паролю.

Таким чином, було доведено, що числові константи, рядки, змінні, команди,  
а також чутливі дані можуть бути явно виявлені у вихідному коді .COM-файлів. Крім того, було виявлено, що введений у редакторі masm-код link16.exe інтерпретує у 16-бітний формат.

Лістинг, згенерований HIEW32 містить значні відмінності від початкового коду у asm-файлі