

Slovenská technická univerzita v Bratislave Fakulta informatiky a informačných technológií

Systémové programovanie a asemblery — Zadanie 3

Statická a dynamická analýza programu

Autor: Nazar Meredov

Cvičiaci: Ing. Ján Hudec, PhD.

# Contents

1	Dĺžka akceptovaného reťazca					
2	2 Tvar akceptovaného reťazca					
3	Windows API: funkcie a ich použitie  3.1 DialogBoxParam	5				
4	Úprava programu pre vlastné heslo 4.1 Popis funkcie	7				

# 1 Dĺžka akceptovaného reťazca

Maximálna dĺžka vstupu je 255 znakov (limit funkcie GetDlgItemTextA), no správna dĺžka je práve 8 znakov, čo je overované v programe.

```
LAB 00401064
                                                                        XREF[1]:
                                                                                     00401041(j)
00401064 81 7d 0c
                                      dword ptr [EBP + 0xc],0x111
                          CMP
11 01 00 00
0040106b 0f 85 80
                                      LAB_004010f1
                          JNZ
         00 00 00
00401071 8b 45 10
                                      EAX, dword ptr [EBP + 0x10]
                          MOV
00401074 8b d0
                          MOV
                                      EDX, EAX
00401076 c1 ea 10
                          SHR
                                      EDX, 0x10
00401079 66 0b d2
                                      DX,DX
0040107c Of 85 8a
                          JNZ
                                      LAB_0040110c
00401082 66 83 f8 66
                          CMP
                                      AX,0x66
00401086 75 67
                                      LAB_004010ef
                          JNZ
00401088 68 ff 00
                          PUSH
0040108d 68 58 30
                          PUSH
                                      DAT_00403058
         40 00
00401092 6a 65
                          PUSH
                                      0x65
00401094 ff 75 08
                                      dword ptr [EBP + 0x8]
                          PUSH
00401097 e8 fa 00
                                                                                         UINT GetDlgItemTextA(HWND hDlg, ...
                          CALL
                                      USER32.DLL::GetDlgItemTextA
         00 00
0040109c 50
                          PUSH
0040109d 83 f8 08
                                      EAX,0x8
                          CMP
004010a0 74 07
                          JΖ
                                      LAB_004010a9
004010a2 b8 00 00
                          MOV
                                      EAX,0x0
004010a7 eb 21
                          JMP
                                      LAB_004010ca
```

Figure 1: Volanie funkcie GetDlgItemTextA

```
00401082 66 83 f8 66
                                      AX,0x66
00401086 75 67
                                      LAB_004010ef
                          JNZ
00401088 68 ff 00
                                      0xff
                         PUSH
0040108d 68 58 30
                          PUSH
                                      DAT_00403058
         40 00
00401092 6a 65
                          PUSH
                                      0x65
00401094 ff 75 08
                          PUSH
                                      dword ptr [EBP + 0x8]
00401097 e8 fa 00
00 00
                                                                                        UINT GetDlgItemTextA(HWND hDlg, ...
                                      USER32.DLL::GetDlgItemTextA
                         CALL
```

Figure 2: Porovnanie vstupu s hodnotou 8

## 2 Tvar akceptovaného reťazca

Reťazec FIITge<br/>ek slúži ako heslo. Je generovaný funkciou FUN\_00401146 zo vstupného buffera pomocou extrakcie znakov.

#### Generovanie hesla

- Zdrojový reťazec: I4561AsEmblerySuPOhodicka2x3Xzgv
- Extrahované znaky:

```
MOV AL, byte ptr [ESI+0x2a]; 'F' MOV AL, byte ptr [ESI]; 'I' ...

MOV AL, byte ptr [ESI+0x17]; 'k'
```

Výsledné heslo: FIITgeek

004010ca 6a 00 004010cc 83 f8 01 004010cf 75 0c	LAB_004010ca PUSH CMP JNZ	0x0 EAX,0x1 LAB_004010dd	XREF[1]:	004010a7(j)
004010d1 68 00 30 40 00	PUSH	s_Right_!_00403000		= "Right !"
004010d6 68 00 30 40 00	PUSH	s_Right_!_00403000		= "Right !"
004010db eb 0a	JMP	LAB_004010e7		
	LAB_004010dd		XREF[1]:	004010cf(j)
004010dd 68 08 30 40 00	PUSH	s_Wrong_!_00403008		= "Wrong !"
004010e2 68 08 30 40 00	PUSH	s_Wrong_!_00403008		= "Wrong !"

Figure 3: Správne vs. nesprávne heslo

#### Overovanie hesla

- 1. Kontrola dĺžky.
- 2. Porovnanie s heslom.
- 3. Výstup Right! alebo Wrong!.

## 3 Windows API: funkcie a ich použitie

## 3.1 DialogBoxParam

Zobrazí dialógové okno.

Návratové hodnoty: INT\_PTR (vráceno v registru EAX)

Volane Adresy: 0x0040101E (ukazuje na ptr [->USER32.DLL::DialogBoxParamA], adresa

skoku: 0x0040118A)

#### Parametre:

- hInstance inštancia aplikácie
- ID ID dialógu
- hWndParent rodič
- lpDialogFunc callback
- lParamInit voliteľné dáta

```
POINTER to EXTERNAL FUNCTION
                 INT_PTR
                   Stack[0x4]:4
   HINSTANCE
                               hInstance
   LPCSTR
                   Stack[0x8]:4
                               lpTemplateName
   HWND
                   Stack[0xc]:4
                               hWndParent
   DLGPROC
                   Stack[0x10]:4 lpDialogFunc
                   Stack[0x14]:4 dwInitParam
   LPARAM
                 146 DialogBoxParamA <<not bound>>
                 PTR_DialogBoxParamA_00402020
                                                          XREF[1]:
                                                                     DialogBoxParamA:0040118a
00402020 8c 20 00 00
                    addr
                              USER32.DLL::DialogBoxParamA
```

Figure 4: DialogBoxParam

## 3.2 GetDlgItemText

Získa text z ovládacieho prvku.

Návratové hodnoty: UINT (vráceno v registru EAX)

Volane Adresy: 0x00401097 (ukazuje na ptr [->USER32.DLL::GetDlgItemTextA], adresa

skoku: 0x00401196)

#### Parametre:

• hDlg, nIDDlgItem, lpString, cchMax

Figure 5: GetDlgItemText

### 3.3 MessageBox

Zobrazí správu používateľovi.

Príklad: MessageBox(..., "Right!", ...) pri správnom hesle.

Návratové hodnoty: int (vráceno v registru EAX)

Volane Adresy: 0x004010EA (ukazuje na ptr [->USER32.DLL::MessageBoxA], adresa

skoku: 0x004011A2)

#### Parametre:

- hInstance inštancia aplikácie
- lpText text správy ("Right!" alebo "Wrong!")
- lpCaption titulok okna (rovnaký ako text)
- uType 0x0 (štandardné okno)

```
POINTER to EXTERNAL FUNCTION
                 int __stdcall MessageBoxA(HWND hWnd, LPCSTR lpText, LPCS...
                               <RETURN>
                   Stack[0x4]:4
   HWND
                               hWnd
   LPCSTR
                   Stack[0x8]:4
                               lpText
   LPCSTR
                   Stack[0xc]:4
                               lpCaption
                   Stack[0x10]:4 uType
   UINT
                443 MessageBoxA <<not bound>>
                PTR_MessageBoxA_00402018
                                                         XREF[1]:
                                                                   MessageBoxA:004011a2
00402018 c8 20 00 00
                             USER32.DLL::MessageBoxA
                   addr
```

Figure 6: MessageBox

# 4 Úprava programu pre vlastné heslo

Program bol upravený tak, aby akceptoval heslo Meredovv.

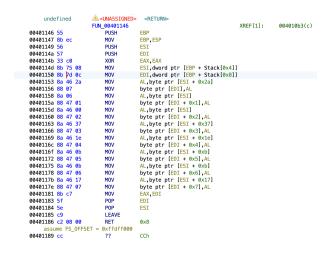


Figure 7: Pôvodná verzia

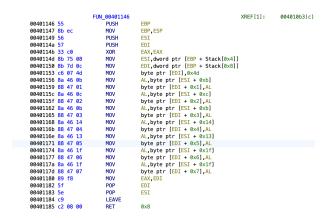


Figure 8: Upravená verzia

## 4.1 Popis funkcie

#### Iniciácia:

```
XOR EAX,EAX
MOV ESI,[EBP+0x8] ; vstup
MOV EDI,[EBP+0xC] ; výstup
```

#### Zápis hesla:

```
MOV byte ptr [EDI],0x4d ; 'M'
MOV AL,byte ptr [ESI+0xb] ; 'e'
MOV AL,byte ptr [ESI+0xc] ; 'r'
MOV AL,byte ptr [ESI+0xb] ; 'e'
MOV AL,byte ptr [ESI+0x14] ; 'd'
MOV AL,byte ptr [ESI+0x13] ; 'o'
MOV AL,byte ptr [ESI+0x1f] ; 'v'
MOV AL,byte ptr [ESI+0x1f] ; 'v'
```

#### Ukončenie:

MOV EAX,EDI

POP EDI

POP ESI

LEAVE

RET 0x8

Zaver: Funkcia zoberie 8 znakov od výstupu a porovna to z bufferomm a ak budu rovnake tak vypiše Right! ak nie tak Wrong!