



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
3	Windows API: funkcie a ich použitie	4
3.1	DialogBoxParam	4
3.2	GetDlgItemText	5
3.3	MessageBox	6
4	Úprava programu pre vlastné heslo	7
4.1	Popis funkcie	8

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    CMP      dword ptr [EBP + 0xc],0x111
          11 01 00 00
0040106b 0f 85 80    JNZ      LAB_004010f1
          00 00 00
00401071 8b 45 10    MOV      EAX,dword ptr [EBP + 0x10]
00401074 8b d0      MOV      EDX,EAX
00401076 c1 ea 10    SHR      EDX,0x10
00401079 66 0b d2    OR       DX,DX
0040107c 0f 85 8a    JNZ      LAB_0040110c
          00 00 00
00401082 66 83 f8 66 CMP      AX,0x66
00401086 75 67      JNZ      LAB_004010ef
00401088 68 ff 00    PUSH     0xff
          00 00
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    CALL     USER32.DLL::GetDlgItemTextA      UINT GetDlgItemTextA(HWND hDlg, ...
          00 00
0040109c 50          PUSH     EAX
0040109d 83 f8 08    CMP      EAX,0x8
004010a0 74 07      JZ       LAB_004010a9
004010a2 b8 00 00    MOV      EAX,0x0
          00 00
004010a7 eb 21      JMP      LAB_004010ca
```

Figure 1: Volanie funkcie GetDlgItemTextA

```

00401082 66 83 f8 66 CMP      AX,0x66
00401086 75 67      JNZ      LAB_004010ef
00401088 68 ff 00    PUSH     0xff
          00 00
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    CALL     USER32.DLL::GetDlgItemTextA      UINT GetDlgItemTextA(HWND hDlg, ...
          00 00
```

Figure 2: Porovnanie vstupu s hodnotou 8

2 Tvar akceptovaného reťazca

Reťazec FIITgeek 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

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

Figure 3: Správne vs. nesprávne heslo

Overovanie hesla

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

3.2 GetDlgItemText

Získa text z ovládacieho prvku.

Návratové hodnoty: UINT (vráteno v registru EAX)

Volané Adresy: 0x00401097 (ukazuje na ptr [->USER32.DLL::GetDlgItemTextA], adresa skoku: 0x00401196)

Parametre:

- hDlg, nIDDlgItem, lpString, cchMax

```
*****
*               POINTER to EXTERNAL FUNCTION               *
*****
UINT __stdcall GetDlgItemTextA(HWND hDlg, int nIDDlgItem..
EAX:4          <RETURN>
Stack[0x4]:4   hDlg
Stack[0x8]:4   nIDDlgItem
Stack[0xc]:4   lpString
Stack[0x10]:4  cchMax
258 GetDlgItemTextA <<not bound>>
PTR_GetDlgItemTextA_0040200c XREF[1]: GetDlgItemTextA:00401196
0040200c aa 20 00 00 addr USER32.DLL::GetDlgItemTextA
```

Figure 5: GetDlgItemText

3.3 MessageBox

Zobrazí správu používateľovi.

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

Návratové hodnoty: int (vráteno 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, LPCSTR lpCaption, UINT uType)
EAX:4          <RETURN>
Stack[0x4]:4   hWnd
Stack[0x8]:4   lpText
Stack[0xc]:4   lpCaption
Stack[0x10]:4  uType
443 MessageBoxA <<not bound>>
PTR_MessageBoxA_00402018 XREF[1]: MessageBoxA:004011a2
00402018 c8 20 00 00 addr USER32.DLL::MessageBoxA
```

Figure 6: MessageBox

4 Úprava programu pre vlastné heslo

Program bol upravený tak, aby akceptoval heslo Meredovv.

```
undefined      <UNASSIGNED> <-RETURN-
FUN_00401146    XREF[1]: 004010b3(c)
00401146 55      PUSH     EBP
00401147 8b ec    MOV      EBP, ESP
00401149 56      PUSH     ESI
0040114a 57      PUSH     EDI
0040114b 33 c0    XOR      EAX, EAX
0040114d 8b 75 08 MOV      ESI, dword ptr [EBP + Stack[0x4]]
00401150 8b 7d 0c MOV      EDI, dword ptr [EBP + Stack[0x8]]
00401153 8a 46 2a MOV      AL, byte ptr [ESI + 0x2a]
00401156 88 07    MOV      byte ptr [EDI], AL
00401158 8a 06    MOV      AL, byte ptr [ESI]
0040115a 88 47 01 MOV      byte ptr [EDI + 0x1], AL
0040115d 8a 46 00 MOV      AL, byte ptr [ESI]
00401160 88 47 02 MOV      byte ptr [EDI + 0x2], AL
00401163 8a 46 37 MOV      AL, byte ptr [ESI + 0x37]
00401166 88 47 03 MOV      byte ptr [EDI + 0x3], AL
00401169 8a 46 1e MOV      AL, byte ptr [ESI + 0x1e]
0040116c 88 47 04 MOV      byte ptr [EDI + 0x4], AL
0040116f 8a 46 0b MOV      AL, byte ptr [ESI + 0xb]
00401172 88 47 05 MOV      byte ptr [EDI + 0x5], AL
00401175 8a 46 0b MOV      AL, byte ptr [ESI + 0xb]
00401178 88 47 06 MOV      byte ptr [EDI + 0x6], AL
0040117b 8a 46 17 MOV      AL, byte ptr [ESI + 0x17]
0040117e 88 47 07 MOV      byte ptr [EDI + 0x7], AL
00401181 8b c7    MOV      EAX, EDI
00401183 5f      POP      EDI
00401184 5e      POP      ESI
00401185 c9      LEAVE
00401186 c2 08 00 RET      0x8
      assume FS_OFFSET = 0xfffff000
00401189 cc      ??      CCh
```

Figure 7: Pôvodná verzia

```
FUN_00401146    XREF[1]: 004010b3(c)
00401146 55      PUSH     EBP
00401147 8b ec    MOV      EBP, ESP
00401149 56      PUSH     ESI
0040114a 57      PUSH     EDI
0040114b 33 c0    XOR      EAX, EAX
0040114d 8b 75 08 MOV      ESI, dword ptr [EBP + Stack[0x4]]
00401150 8b 7d 0c MOV      EDI, dword ptr [EBP + Stack[0x8]]
00401153 c6 07 4d MOV      byte ptr [EDI], 0x4d
00401156 8a 46 0b MOV      AL, byte ptr [ESI + 0xb]
00401159 88 47 01 MOV      byte ptr [EDI + 0x1], AL
0040115c 8a 46 0c MOV      AL, byte ptr [ESI + 0xc]
0040115f 88 47 02 MOV      byte ptr [EDI + 0x2], AL
00401162 8a 46 0b MOV      AL, byte ptr [ESI + 0xb]
00401165 88 47 03 MOV      byte ptr [EDI + 0x3], AL
00401168 8a 46 14 MOV      AL, byte ptr [ESI + 0x14]
0040116b 88 47 04 MOV      byte ptr [EDI + 0x4], AL
0040116e 8a 46 13 MOV      AL, byte ptr [ESI + 0x13]
00401171 88 47 05 MOV      byte ptr [EDI + 0x5], AL
00401174 8a 46 1f MOV      AL, byte ptr [ESI + 0x1f]
00401177 88 47 06 MOV      byte ptr [EDI + 0x6], AL
0040117a 8a 46 1f MOV      AL, byte ptr [ESI + 0x1f]
0040117d 88 47 07 MOV      byte ptr [EDI + 0x7], AL
00401180 89 f8    MOV      EAX, EDI
00401182 5f      POP      EDI
00401183 5e      POP      ESI
00401184 c9      LEAVE
00401185 c2 08 00 RET      0x8
```

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 bufferom a ak budu rovnake tak vypíše Right! ak nie tak Wrong!