XCTF-Reverse-ExerciseArea-003-writeup





ctf 专栏收录该内容

35 篇文章 **0** 订阅 订阅专栏

0x00介绍

本题是xctf攻防世界中Reverse的新手第三题。

对给定的helloctf.exe进行逆向分析,找到serial。

实验环境: IDA Pro 7.0

0x01 解题过程

1.1 前期的分析

运行helloctf.exe,可以看到需要输入serial,如果输错的话会进入下一次循环判断

Ē	cmd - he	elloctf.exe				_ [×
🕅 <1> cmd - helloctf		5	earch		Ρ 🛨 🗸		1 🗌 🖃
振祥@LENOVO-HZY D:\ctf-learni \$ helloctf.exe please input your serial:abcd wrong! please input your serial:abc wrong! please input your serial:	ng\reverse	-engineeri	ng\xc	tf\Exe	rciseAr	ea\003	
helloctf.exe*[32]:7336	« 190108[64]	1/1 [+] NUM	PRI‡	80x25	h(\$6;96159)	ćsdn.n275	<mark>4_31008%</mark> 292

我们还是从字符串 please input your serial: 入手。用IDA打开helloctf.exe, Shift+F12, 显示在该二进制文件中的所有字 符串。在顶部导航栏中: Search→Search,输入字符串 please input your serial:, 然后双击跳转

٠	IDA - hell	octf.exe D:\ctf-lea	rning\reverse-eng	ineering\xctf\Exercise/	Area\003\hel	lloctf.exe			_ 🗇 🗙
<u>File Edit Jump Search View Debugger</u>	Options Windows	Help		, i i i i i i i i i i i i i i i i i i i					
		X No debugge	r 🔸 🍬 🕏	• • • • •					
			1 1					:	•
Library function 📃 Regular function	Instruction 📃 Data	Unexplored Exter	nal symbol						
f Functions window 🗆 🗗 🛪	🖪 IDA View-A 🔳	🖬 Strings window 🛛	Hex View-1 ⊠	🖪 Structures 🗵 🗮	Enums 🔳	🛅 Imports	Exports	×	
Function name	Address	Length	Type String				i		^
f_main ffilbuf f_sprintf	's' .rdata:00407244 's' .rdata:0040726C 's' .rdata:004072A4	00000026 00000035 00000029	C R6025\r\n- pure C R6024\r\n- not C R6019\r\n- unab	virtual function call\r\n enough space for _onexit/at le to open console device\r	exit table\r\n \n				
f _scanf f sub_40134B	's' .rdata:004072D0 's' .rdata:004072F4	00000021 0000002D	C R6018\r\n- unex C R6017\r\n- unex C R6016\r\n- not	spected heap error\r\n spected multithread lock error enough space for thread dat	or\r\n				
<pre>famsg_exit ffast_error_exit f _nullsub_1</pre>	s'.rdata:00407350 rdata:00407374 rdata:00407374	00000021 0000002C 0000002A	C \r\nabnormal pr C R6009\r\n- not C R6008\r\n- not	rogram termination\r\n enough space for environmen enough space for arguments\:	t\r\n r\n				
fioinit fread fgetbuf fwh 401889	s . rdata:004073CC s . rdata:004073F4 s . rdata:00407420 s . rdata:00407440	00000025 00000025 0000001A 00000017	C R6002\r\n- floa C Microsoft Visua C Runtime Error!\ C <program name="" td="" u<=""><td>ting point not loaded\r\n 1 C++ Runtime Library n\nProgram: nknown></td><td></td><td></td><td></td><td></td><td>- 1</td></program>	ting point not loaded\r\n 1 C++ Runtime Library n\nProgram: nknown>					- 1
f = fcloseall $f = cinit$ $f = cinit$ $f = cinit$. rdata:00407458 . rdata:0040746C . rdata:0040746C . rdata:0040747C	00000013 00000010 0000000C	C GetLastActivePo C GetActiveWindow C MessageBoxA	рир	双击,	跳转			
f exit f doexit f initterm	 s'.rdata:00407488 rdata:0040789E data:00408030 	0000000B 0000000D 00000008	C user32.dl1 C KERNEL32.dl1 C wrong!\n						
f _flush f _flush	s'.data:00408038	0000000A 0000001A	C success!\n C please input yo	our serial:					
f sub_401B2B v <	<pre>'s' . data:00408068 's' . data:00408328 's' . data:00408326</pre>	00000023 00000006 00000006	C 437261636b4d654 C \t-\r] C @\t	la757374466f7246756e					
Line 1 of 121	's' . data:004083EE 's' . data:0040842E data:0040843E	00000006 00000006	C @\n C @\x1B C @x						
	s'.data:00408446 s'.data:00408446 s'.data:0040844E	00000006	C ··· @y C ··· @z	u					
uu	Line 37 of 46	0000042	<u> </u>	n					·
Output window									□ <i>8</i> ×
Command "OpenStrings" failed									^
Python									•
AU: idle Down Disk: 6GB									

再双击,跳到引用该字符串的地方。

9	IDA - helloctf.exe D:\ctf-learning\reverse-engineering\xctf\ExerciseArea\003\helloctf.exe 🗕 🗖 🗙	
Eile Edit Jump Search View Debugge	r Options Windows Help	
👩 🖬 🖕 🖛 🕶 🚽 🦓 🆓 🖓 📮 🔬 🗖		
		•
Library function Regular function	Instruction Data Unexplored External symbol	
🗗 Functions window 🗆 🗗 🛪	🕼 IDA View-A 🖸 🐼 Strings window 🔹 🖸 Hex View-1 🔹 🖪 Structures 🔹 🗒 Enums 🔹 📆 Imports 🔹 📝 Exports	
Function name	align 4	~
f_main	.data:00408048 ; char aS[3]	
ffilbuf	.aata:00408048 aS db %s',0 ; DATA XREF: _main+39To	
<u>f</u> _sprintf		
f_scanf	. cata:00408044 aPleaseInputYou db 'please Input your serial:',0	
f sub_40134B	. data: 0040804C ; DATA XREF: _main+25To	
f start	.data:00408066 align 4	
<pre>famsg_exit</pre>	.data:00408068 a437261636b4d65 db '437261636b4d654a757374466+7246756e',0	
<pre>f _fast_error_exit</pre>	.data:00408068 ; DATA XREF: _main+Cfo	
f nullsub_1	data:0040808B align 10h	
fioinit	.data:00408090 ; FILE stru_408090	
fread	data:00408090 stru_408090 FILE <offset 0="" 0,="" 1000h,="" 101h,="" offset="" unk_40ae80,=""></offset>	
fgetbuf	.data:00408090 ; DATA XREF: _main+12Bfw	
f sub_401889	.data:00408090 ; _main:loc_401135^o	
ffcloseal1	.data:004080B0 ; FILE stru_4080B0	
fcinit	.data:004080B0 stru 4080B0 FILE <0, 0, 0, 2, 1, 0, 0, 0>	
f_exit	.data:004080B0 ; DATA XREF: sub 401348+210	
fexit		
<pre>f _doexit</pre>	data:004080D0 unk 4080D0 db 0 : DATA XREE: flshuf+58to	
<u>f</u> initterm		
f_f1ush	dta:00408001 db 0	
fflush		
f sub_401B2B		
f_flsall		
fflsbuf		
f sub 401CB6	.data:004080D5 db 0	
f _write_char	.data:004080D6 db 0	
f write multi char	db 0	
f write string	db 0	
F get int arg	db 0	
< >>	db 0	
Line 1 of 121	0000804C 0040804C: .data:aPleaseInputYou (Synchronized with Hex View-1)	Y
Output window		×
Command "JumpOpXref" failed		¢
Python		
AU: idle Down Disk: 6GB	http://i.log.codu.net/igr.3505523	2

可以看到是在main中引用的

R	II	DA - hello	octf.exe D:\ctf-lea	rning\reverse-engi	neering\xctf\Ex	erciseArea\003\h	elloctf.exe		_ 🗇 🗙
Eile Edit Jump Search View Debu	gger <u>O</u> ptions	Windows	Help	<u> </u>	.				
👩 🔜 🖕 🗢 👻 🦓 🆓 🆓 🖕 🗼 👧	🔺 🥥 🖬 📩 💣	े 🕶 🖈 🖬 🕽	🗙 🕨 💷 🗈 No debugge	r 🔹 🐮 🛃	🗊 🕈 🕅				
				1					•
Library function 📃 Regular functi	on 📕 Instructio	on 📃 Data 🛛	Unexplored Exter	nal symbol					
📝 Functions window 🗆 🕫	× 📑 IDA Vi	ew-A 🛛	🖾 Strings window 🖂	Hex View-1 ⊠	🖪 Structures 🖂	🖽 Enums 🗵	🛐 Imports	Exports	×
Function name	^			1					
<pre>7 _main 7 _main 7 _filbuf 7 _sprintf 7 _scanf 7 _printf 7 _start 7 _amsg_exit 7 _fast_error_exit 7 _nullsub_1 7 _ioinit 7read 7getbuf 7 _stb_401889 7fcloseal1 7cinit 7exit 7exit 7exit 7exit 7exit 7exit 7initterm 6filbut 7initterm 6filbut 7initterm 6filbut 7initterm 6initterm 6init</pre>		var_5E= var_5C= var_4A= var_24= argc= dw argv= dw envp= dw sub mov push push push push push push push push	byte ptr -5Eh byte ptr -5Ch byte ptr -48h byte ptr -48h word ptr 4 word ptr 8 word ptr 0Ch esp, 60h ecx, 8 ebx ebx ebz esi edi esi, offset a43720 edi, [esp+70h+var] d	51636b4d65 ; " <mark>41720</mark> _24]	1636b4d654a7573	74466f7246756e ^r			
<u>f</u> _flush				Ľ.					
f sub_401B2B	~		🛄 🚄 🖂						
< >>	·		100 1010101						
Line 1 of 121			mov ecx.	. 8					
Craph overview	×	7 1005) (12	xor eax lea edi push offs rep stosd	, eax , [esp+70h+var_48] set aPleaseInputYou	; "please input	t your serial:"			
Output window		., 2000) (10	,,						□ # ×
									A
commana "JumpOpXret" tailed									*
Python									
AU: 1010 Down Disk: 668								h	ttps://blog.cscin.net/gg_35056292_

并且可以看到,在main函数的开始,将字符串 437261636b4d654a757374466f7246756e 赋值给了寄存器esi

1.2 具体基本块的分析

用Ollydbg来运行helloctf.exe。可以看到,地址0x40101A-0x401057是读取用户的输入,并对输入字符串的长度进行限制。

R	IDA - helloctf.e	exe D:\ctf-learning\reverse	-engineering\xctf\Exe	rciseArea\003\hello	octf.exe		_ 🗇 🗙
<u>File Edit Jump Search View Debugg</u> e	er Options <u>W</u> indows Help		<u> </u>				
📂 🔜 🛛 🗢 🔻 🐴 👘 🍓 🐴 🗼 👧	🍳 🗟 📾 🗟 🖑 🔻 📽 📈 🕒	No debugger •	12 🛃 🕄 🕈 🖀				
						2	-
Library function 📃 Regular function	📕 Instruction 📗 Data 📕 Une	explored 📕 External symbol					
📝 Functions window 🗆 🕫 🗙	📑 IDA View-A 🛛 🐼 St	rings window 🗵 🛛 Hex View-1	🗵 🚺 Structures 🗵	🖽 Enums 🗵	🛅 Imports 🗵	🛃 Exports	
Function name	•		Y Y Y				
<pre>f _main ffilbuf f _sopintf f _soanf f _printf f _scanf f _printf f start fdass_exit fdast_error_exit f nullsub_1 fifast_error_exit f nullsub_1 fifullsub_1 fifullsub_1 fifullsub_1 fexit fexit fexit fexit fexit fexit fflush fflush fuflush fufl</pre>		<pre>loc_40101A: mov ecx, 8 xor eax, eax lea edi, [esp+70h+var_ push offset aPleaseInpu rep stosd stosw stosb call _printf ; lea eax, [esp+74h+var_ push offset aS ; call _scanf lea edi, [esp+7Ch+var_ or ecx, 0FFFFFFh xor eax, eax add esp Ch repne scasb ; j not ecx ; dec ecx ;</pre>	<pre>48] tYou; "please input y, printf函数 5C] "%s" 5C]; [esp+0x20], 用户\$ acx会增加, Up3我输入"hzy111", ecx 取非: 00000007 输入字符率的长度,包括10 el减,仅取用户输入的字符;</pre>	our serial:" 輸入的字符串首地址 字符串结束 会变成FFFFFFB, 串长度			
Å Graph overview □ # ×		cmp ecx, 11h ia loc 40110D :	输入的字符串不能超过17个	字符			
	100. 00% (303, 1506) (1088, 576	5) 00001057 00401057:main+57 (Sy	mchronized with Hex View-1)				
Output window							□ <i>8</i> ×
Command "JumpOpXref" failed							<u>^</u>
Python							v
AU: idle Down Disk: 6GB							
No. 1410 DOWN DISK. 00D							pswbioglesan.navqq_aa056292

地址0x40105F将用户输入字符串的第一个字符赋值给了寄存器al,也就是寄存器eax的低八位。接下来,如果al保存的值为0则 直接跳到与字符串 437261636b4d654a757374466f7246756e 比较的基本块。显然不符合要求,因此第一个字符不能为0

9		IDA - hell	octf.exe D:\ctf-lear	ning\reverse-eng	gineering\xctf\Exe	rciseArea\0	03\helloo	tf.exe			_ 🗇 🗙
<u>File Edit Jump Search View</u>	/ Deb <u>u</u> gger	r <u>O</u> ptions <u>W</u> indows	Help								
📂 🔒 🖕 🖛 🕶 👘 🍓 🚳	1 🔬	🎱 🗄 🏥 💣 🖑 🔻 🧩 🕍	🗙 🕨 🖬 🖬 No debugger	• 12	🛃 🚼 🕈 🌮						
										:	•
📃 Library function 📃 Regular	r function 📕	📕 Instruction 📗 Data	Unexplored Extern	nal symbol							
📝 Functions window	□ <i>ē</i> ×	🔄 IDA View-A 🗵	🖾 Strings window 🗵	Hex View-1 ⊠	🖪 Structures 🖂	🖽 Enums		Imports	🖂 📝 Expe	orts 🗵	
Function name	^		_	pusn	onisec awrong ,	wiong: (ii					
f main				mov	_princi eax. stru 408090.	cnt					
ffilbuf				add	esp, 4						
<u>f</u> _sprintf				dec	eax						
f_scanf	_			рор	edi						
f _printi				рор	esi						
f amsg exit				рор	ebp stru 408090 ont						
ffast_error_exit				pop	ebx	Cux					
f nullsub_1				js	short loc_401135						
f101n1t											
fgetbuf		+			* *		_				
f sub_401889							-				
$f_{f_{f_{f_{f_{f_{f_{f_{f_{f_{f_{f_{f_{f$		1135:	; FILE * loc_40105	F: ;	[esp+0x20],用户输	入字符串的首切	也址				
ff		offset stru_4080	390 mov a	1, [esp+ebx+70h+	var_5C]						
fexit		filbuf	test a	l, al	and a statistic						
fdoexit		esp, 4	JZ S	hort 10c_401080	; al为0才跳转						
finitterm		esp, oon									
f _fluch		endp									
f sub_401B2B											
6											
Line 1 of 121			1								
& Cuenta energian		mov	/sx ecx, al								
HR Graph overview	0 8 4	pus	sh ecx								
		lea	edx, [esp+/4h+	Fvar_5E] ; [esp+0 2044 · "%∨"	x16]						
		pus	sh edx	: char *							
		100 00% (252 0046) (1	125 400) 00001065 004010	65. main±65 (Cunchus	migod with How View-1)						
		100.00% (333, 2240) (1.	123, 409/ 0001003 004010	05main+05 (Synchro	mized with nex view-1/						
Communa Sampoparer raii											X
Command "JumpOpXref" fail	.ed										÷
Python											
AU: idle Down Disk: 6GB						-	_			https://blc	g.csdn.net/gg_35056292

然后是最主要的部分,地址**0x401067**开始,对于用户输入的字符串,在循环中,将每个字符对应的十六进制数拼接到寄存器**edx** 中保存,比如我输入的是 hzyll1,那么得到的结果是 687a796c6c6c

CllyDbg -	- helloctf.exe - [CPU - main thread, module helloctf] 英学家
Opdition BD7C24 IE LLA BLL [LOCAL 23*2] 0040107E 8300 FF 00 BCX, IFFFFFF 004010631 8300 NO BCX, IFFFFFF 00000083 8300 004010631 8300 NO BCX, IFFFFFF 00000083 8300 004010631 8300 NO BCX, IFFFFFF 0000008 8000 004010681 F2: AB REPHE SCAS BTTE PTR ES: [EDI] 0000008 004010802 80859 NOV ECX, EDI I 0000000 004010802 80859 MOV ECX, EDI I 0000000 004010802 80859 MOV EDI, ECX 0000000 004010802 80859 MOV EDI, ECX 00000000 004010804 8087 MOV EDI, EDX 000000000 004010804 80804 MOV EDI, EDX 000000000000000000000000000000000000	A Registers (FPU) A EAX 0000000 EAX 0000000 C EDX 0016FEFC ASCII "687.796.66.66" C C EDX 0016FEFC ASCII "cyll1" C C EDX 0016FEFC ASCII "ryll1" C C EDX 0010AE C C C C 1 ES 002B 32bit 0(FFFFFFFF) C D Z 0 DS 002B 32bit 0(FFFFFFFF) C D D 0 D D D
004010A5 - 83E1 03	0 0 LastErr 0000000 FERDR_SUCCESS FFL 0000287 (NO. B. NE. BE, S. FE, L. LE) 570 empty 0.0 571 empty 0.0 573 empty 0.0 574 empty 0.0 576 empty 0.0 576 empty 0.0
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	ODIFFEN OODIFFEN OODIFFEN
	https://biog.csci.net/o

从地址**0x4010B0**开始,让上面循环中得到的十六进制字符串跟一开始的字符串 437261636b4d654a757374466f7246756e 进行比 较,二者一致才输出 success 的字样

🕷 OllyDbg - helloctf.exe - [CPU - main thread, n	nodule helloctf] 英学之
C File View Debug Irace Plugins Options Windows Help	
EMX FIL SERIE OU LEMWICRK BMH	
00401096 - 83C0 FF (OR ECX, FFFFFFFF 00401098 - 85CD MOW ECX, EEF 00401098 - 85CD MOW ECX, EEF 00401090 - 4F DEE EDI 00401091 - CIEB 02 SHR ECX, 2 00401004 - F3: A5 FEF MOVS BOODD FTR ES: [EDI], DWORD FTR DS: [ESI] 004010A3 - 83EL 03 MOW ECX, EEF 004010A5 - 83EL 03 ANN ECX, 00000003 004010A6 - 83FB 11 DWE EEX, 11 004010A6 - 83FB 11 DWE EEX, 11 004010A7 - 72. AF TH, SIGNET 0040105F 004010A7 - 83A4 HPF MOVS BYTE FTR ES: [EDI] 004010A6 - 83A42 HEA EEX, [LOCAL. 17] 004010B7 - 8A12 HPF MOV DL ATTER THE ES: [EDI] 004010B7 - 8A12 HPV DL ATTER THE ES: [EDI] 004010B7<	Prejisters (FPU) EAX 0010FEFC ASCII "667x796c6c6c" EXX 0010FEFC ASCII "667x796c6c6c" EIX 0010FEFC ASCII "667x796c6c6c" ESX 0010FEFC ASCII "667x796c6c6c" ESX 0010FEFC ASCII "437261636b4d654x757374466f7246756c" EIX 0010FEFC ASCII "437261636b4d654x757374466f7246756c" EIX 0010FFC ASCII "437261636b4d654x757374466f7246756c" EIX 0010FFC ASCII "437261636b4d654x757374466f7246756c" EIX 0010FFC ASCII "437261636b4d554x757374466f7246756c" EIX 0010FFC ASCII "437261636b4d554x757374466f7246756c" EIX 0010FFC ASCII "537261636b4d554x757374466f7246756c" EIX 0010FFC ASCII "537261636b4d554x757374466f72475
Undition - 76 if INF SHORT ORADIDEC Undition - 76 if INF SHORT ORADIDEC Undition - 74 if INF SHORT ORADIDEC Undition - 74 if INF SHORT ORADIDEC Undition - 74 if INF SHORT ORADIDEC Undition - 8450 01 How DL STTE FTR DS [EXT+] Undition - 8462 How CL STTE FTR DS [EXT+] Undition - 8463 How CL STTE FTR DS [EXT+] Undition - 8463 How CL STTE FTR DS [EXT+] Undition - 8464 How CL STTE FTR DS [EXT+] Undition - 8464 How CL STTE FTR DS [EXT+] Undition - 8464 How CL STTE FTR DS [EXT+] Undition - 8464 How CL STTE FTR DS [EXT+] Undition - 8464 How CL STTE FTR DS [EXT+] Undition - 8464 - 8464 Stack [0010767 - 8464 Stack [00107687 - 8464 Junp - 766 - 766 Junp - 766 - 766 Junp - 766 <td>D 0 0 0 LastErr 00000000 ERROR_SUCCESS EFL 00000246 (NO, NE, E, BE, NS, FE, GE, LE) ST0 empty 0.0 ST1 empty 0.0 ST2 empty 0.0 ST4 empty 0.0 ST4 empty 0.0 ST6 empty 0.0 ST6 empty 0.0</td>	D 0 0 0 LastErr 00000000 ERROR_SUCCESS EFL 00000246 (NO, NE, E, BE, NS, FE, GE, LE) ST0 empty 0.0 ST1 empty 0.0 ST2 empty 0.0 ST4 empty 0.0 ST4 empty 0.0 ST6 empty 0.0 ST6 empty 0.0
Address Hex June ASCII D0187EPT 61 37 39 63 63 60 00	∧ OUISTED4 (PO040137C) [96] helloctf (ModuleEntryFeint>) ∧ OUISTEDC
	https://blog.csdn.net/ab <mark>P35638</mark> 292

R	IDA - helloctf.exe D:\ctf-learning\reverse-engineering\xctf\ExerciseArea\003\helloctf.exe	_ 🗇 🗙
Eile Edit Jump Search View Debugger	Options Windows Help	
👩 🕞 🖕 🖛 🕶 🔸 🍓 🖡 🙀 🗛 🗛	■ int	
		•
Library function Regular function	Instruction Data Unexplored External symbol	
Functions window □ # ×	IDA View-A 2 Strings window □ O Hex View-1 □ A Structures □ E Enums □ 3 Imports □ @ Exports □	
Function name		
f _main f _filbuf f _sprintf _scanf	loc_401880: lea esi, [esp+70h+var_24] lea eax, [esp+70h+var_48]	
<pre>f _ printf f start f start f _ amsg_exit f _ fast_error_exit f _ ioinit fioinit fioinit fread fgetbuf f sub_401889 ffloseall fexit fdeexit fdeexit fdeexit fdeexit fflush f</pre>	■ ■ ■ loc_40086: mov bl, [esi] mov cl, dl cmp dl, bl jnz short loc_401080; itest cl, cl jz short loc_401000 iz short loc_401000	
Line 1 of 121		
A Graph overview	add esi, 2 ; 相当于住省储号一个字符 add esi, 2 ; inz short loc_401085 , 3 80.00% (134, 3559) (1057, 320) 00001067 00% (134, 3559) (1057, 320) 00001067	
Output window		0 8 ×
Command JumpOpYrci failed		^
Dether		~
Python		
AU: 1416 Down Disk: 6GB		dn.nei/gg_35056292

结论:分析到这里,题目的意图很明显了,给定一个字符串 437261636b4d654a757374466f7246756e,你的输入转成16进制字符 串以后必须跟它一样。只要把代码中给的字符串转成ASCII码形式即可

解密脚本:

-*- coding:utf-8 -*-

import binascii

src_str = '437261636b4d654a757374466f7246756e'
res = binascii.a2b_hex(src_str)

print(res)

脚本运行结果为: CrackMeJustForFun

1.3 结果

输出 success! 信息,说明拿到了flag

