XCTF mobile新手区解题记录(WP)以及一些总结和思考



windy_ll ① 于 2020-03-28 17:46:46 发布 ② 654 论 收藏 3 版权声明:本文为博主原创文章,遵循 <u>CC 4.0 BY-SA</u> 版权协议,转载请附上原文出处链接和本声明。本文链接: <u>https://blog.csdn.net/qq_41374107/article/details/105165212</u> 版权

XCTF mobile新手区解题记录以及一些总结和思考

- 前言
- 题目: app3
- 题目: easy-apk
- 题目: easy-java
- 题目: easy-jni
- 题目: easy-so
- 题目: app1
- 题目: Ph0en1x-100
- 题目: RememberOther
- 题目: app2
- 题目:黑客精神
- 题目: easy-dex
- 题目: 我是谁
- 一些总结
- 一些博客链接

前言

疫情当下,都已经耍了好几个月了,都不知道干啥了,好不容易等到我四川宣布开学时间了,结果四川高校一点动静都没有,无聊中,那就写一下解题记录吧,有些题先前已经在我个人博客上发过了,就不再重复写了,贴个链接就行了!!!

题目: app3

此题wp已经在个人博客写过,不在详细说明,详情请看链接:https://www.52pojie.cn/thread-1082706-1-1.html

题目: easy-apk

此题wp已经在个人博客写过,不在详细说明,详情请看链

接: https://www.cnblogs.com/aWxvdmVseXc0/p/11955006.html

此题wp已经在个人博客写过,不在详细说明,详情请看链

接: https://www.cnblogs.com/aWxvdmVseXc0/p/12207697.html

题目: easy-jni

- 此题wp已经在个人博客写过,不在详细说明,详情请看链
- 接: https://www.cnblogs.com/aWxvdmVseXc0/p/12198459.html

题目: easy-so

1、下载好题目,拖入夜神中,打开如下所示:

「OX 夜神機拟器 6.3.0.0 Android 5		압 Ξ ◎ − □ ×
– pingan2		
4222		
	СНЕСК	
	验证失败!	
	à	tps://blog.csdn.net/qc_41374107

2、查壳,发现无壳,用jeb反编译后,找到关键字 验证失败 所在类,发现调用了so层的 CheckString 函数进行了验证,传进去的参数为我们在输入框中输入的字符串,如下图所示:





◯ JEB2 - C:\Users\admin\Desktop	v\XCTF\easy-so\456c1dab04b2.apk	-		×
文件 编辑 Navigation 行为 Del	bugger 窗口 帮助			
ا دے کے آپ ایک ایک 🖌 🖌 😂	▶ /* ⊘ ඞ ⊕ ⊕ ⊕ 박 ¥ 號 棽 ⊪ □ ■ ᆳ ۞ ᇎ ᆐ			
🔩 Project Explorer 🛛 😫 🖱 🗖	🖷 Bytecode/Hierarchy 🖷 Bytecode/Disassembly 卫 MainActivity/Source 🖾	🖹 xml <unbound> 🕼 apk<unbound></unbound></unbound>	7	7 - 8
 ✓ C:\Users\admin\Desktop\X ✓ 456c1dab04b2.apk ✓ 456c1dab04b2.apk ☑ Manifest ✓ Certificate > ⊕ Bytecode > ➢ Resources 	<pre>package com.testjava.jack.pingan2; import android.os.Bundle; import android.support.v7.app.AppCompatActivity; import android.view.ViewSOnClickListener; import android.view.v; import android.widget.Toast; public class MainActivity extends AppCompatActivity { public MainActivity () { super(); }</pre>			
	<pre>protected void onCreate(Bundle arg3) { super.onCreate(arg3); this.setContentView(2131296283); this.findViewById(2131165218).setOnClickListener(new public void onClick(View arg6) { if(cyberpeace.checkString(MainActivity.this.</pre>	<pre>View\$OnClickListener() { findViewById(2131165233).getText().toString()) == t!", 1).show(); t!", 1).show();</pre>	1) {	
< > Filter: type "Enter" to validate	Console			
coord: (0,8,0) addr: Lcom/testjava	/jack/pingan2/MainActivity; loc: ?			5741107



	Description Source	Least Le
	📮 Logger 🖾 🖨 Console	
< >>		\$
Filter: type "Enter" to validate	<	>
coord: (0,2,0) addr: Lcom/testjav	a/jack/pingan2/cyberpeace; loc: ?	

3、用IDA打开so文件(要提取x86文件夹下面那个so文件,两个arm文件夹下面的so文件用ida打开有问题),找到 该静态函数,直接F5大法,静态分析该函数可知:首先将传入的字符串前16位与后16位互换,然后两两一组互换位置, 最后将得到的字符串与字符串 f72c5a36569418a20907b55be5bf95ad 比较返回比较结果!!!

💱 IDA - libcyberpeace.so C:\Users\admin\D) Desktop	CTF\easy-so\libcybe	peace.so														-	0	\times
File Edit Jump Search View Debugo	ger Opt	ions Windows Hel	0																
· ★ + + + + + h h h & 4 ↓ A	i 🗖 🥥			No debugger			• 🕸 🛃	90 g×											
							_, , .									:			-
Library function Kegular function	i 📕 Inst	ruction 🔛 Data 📕 Un	explored E	xternal symbol		_		_			_			_		_			
Functions window	3 8 X	IDA Viev-A		Hex View-1		A Struc	ures 🗵	æ	Enuns	×	₫ <u>₽</u>	Imports	×	P	Exports	× _			
Function name	Se ^	Name				Address	Ordinal												^
🗾cxa_atexit	.p	🛃 ChangeIndex				00001054													
🗾cxa_finalize	.p	📝aeabi_unvind_	cpp_pr0			00001C9C													
<u>f</u> strlen	. p	📝 TestDec				00001062													
f malloc	. p	📝aeabi_unvind_	cpp_pr1			00001CA4		_											
<u>f</u> aeabi_memclr	. p	📝 Java_con_testja	va_jack_pinga	n2_cyberpeace_	CheckString	000010BC													
faeabi_memcpy	. p	🔯aeabi_unvind_	cpp_pr2			00001CAC													
<u>f</u> j_TestDec	. p	📑 🚉gnu_Unvind_Re	store_VFP_D			00002038													
fstrcmp	. p	📑 🔤gnu_Unvind_Re	store_VFP			00002028					_								
fgnu_Unwind_Find_exidx	. p	gnu_Unvind_Re	store_VFP_D_1	6_to_31		00002048					ì	周田的此函	約 完崎	具—个着	静态函数				
<pre>jgnu_Unwind_Restore_VFP_D</pre>	. p	💇gnu_Unvind_Re	store_WMMXD			00002058					4	19/13131000033	以下 历史中国		BT/CALLER				
jgnu_Unwind_Restore_VFP	.p	🕎gnu_Unvind_Re	store_WMMXC			000020E0													- 8
jgnu_Unwind_Kestore_VFP_D_16_to_3	51 .p	restore_core_re	gs			00002014													- 18
jgnu_Unwind_Kestore_WMMAD	.p	_Unwind_GetCFA				00001564													
jgnu_unwind_kestore_www.kc	. p	gnu_Unwind_Ka	iseffiception			00001560													
abort	. p	gnu_onvind_ro	rceaunvina			00001610													
J_restore_core_regs	. 0	gnu_unvind_Ke	sune			00001620													
f i any Unwind BeigeEscention	. p	gnu_onvinu_ke	- sume_or_kernr	01		00001080													
f i Unwind VRS Cot	. p	_Unwind_Complete	e voortior			00001600													
f i Unwind VRS Set	. p	Unwind VPS Cot	ACEPTION			00001004													
f cya hegin cleanun	n	Inwind VRS Set				00001050													
f cya tyne match	. y	gou Howind Ba	cktrace			00001788													
f i gnu unwind execute	. n	gnu unwind es	ecute			00002248													
f i gnu Unwind Save VEP D	'n	Inwind VRS Por				00001CR4													
7 j gnu Unwind Save VFP	. p	snu Invind Sa	ve VEP D			00002040													
7 j gnu Unwind Save VFP D 16 to 31	. p	gnu Unwind Sa	ve VFP			00002030													
📝 j gnu Unwind Save WMMXD	α.	👼 enu Unvind Sa	ve VFP D 16 t	o 31		00002050													
J gnu Unwind Save WMMXC	. p	👼 gnu Unvind Sa	ve VINXD	-		0000209C													
📝 jgnu_Unwind_Resume	.p	gnu Unwind Sa	ve_VHHXC			000020F4													
📝 jgnu_Unwind_Resume_or_Rethrow	.p	📝Unwind_Raise	Exception			00002108													
📝 jgnu_Unwind_ForcedUnwind	.p	😿Unwind_Resum	e			0000212C													
f jgnu_Unwind_Backtrace	.p	📝Unwind_Resur	e_or_Rethrow			00002150													
j_Unwind_VRS_Pop	. p	📝Unwind_Force	dUnwind			00002174													
f sub_1018	.t	📝Unwind_Backt	race			00002198													
f ChangeIndex	.t	[📝gnu_unvind_fr	ane			000025E0													
7 TestDec		📑 _Unwind_GetRegi	onStart			00002620													
N	,	int Inwind GetLans	uageSnecificD	ata		00002630													
		Line 5 of 39																	
Output vindov		nanaran																	θ×
The initial autoanalysis has been f	inished	puguccu																	<u></u>
Durk un	20123000	•																	•
Fython																nttos://	bloccesdrun	et/aa 402	72PT07
AU: idle Down Disk: 26GB																			



4、写了一个python小脚本跑出flag,如下所示:

脚本代码:

```
string = 'f72c5a36569418a20907b55be5bf95ad'
strlist = list(string)
k = 0
for i in range(16):
    ch = strlist[1 + k]
    strlist[1 + k] = strlist[0 + k]
    strlist[0 + k] = ch
    k = k + 2
k = 0
for i in range(16):
    ch = strlist[0 + k]
    strlist[0 + k] = strlist[16 + k]
    strlist[16 + k] = ch
    k = k + 1
print(''.join(strlist))
运行截图:
```

题目: app1

★ 文件(F) 编辑(E) 选择(S) 查看(V) 转到(G) 运行(R) 终端(T)

	the set of a 1171197
	s so kirst exercit https://lightoliitibuluu
COX: 夜神傳說器 6.3.0.0 Android 5	
pingan2	
90705bb55efb59da7fc2a5636549812a	a
	СНЕСК
	验证通过!
	https://blog.asdm.nat/qq41374107

۲ <u>)</u>	🕏 addCodeUi.py		🕏 SQLMethod.py		addCode.py	JS testrule.js	🕏 easy-so.py ×			t1 ⊳	
م *	E:>py> ♥ easy-so. 1 string = 2 strlist = 3 4 k = 0	py> 'f72c5a36569418a list(string)	120907b55be5bf95ad'))((1))))((2))	
°⊐ °	5 6 for i in 1 7 ch = 9 8 strli 9 strli	range(16): strlist[1 + k] st[1 + k] = str] st[0 + k] = ch	list[0 + k]								
	10 K = K 11	+ 2									
Д	12 k = 0 13 for i in f 14 for i in f 15 ch = : 16 strli: 17 strli: 18 k = k 19 20 21 print(''.;	range(16): strlist[0 + k] st[0 + k] = strl st[16 + k] = ch + 1 join([strlist]))	list[16 + k]								
	输出终端通过的							2: Python	× + 1	T) 偷	~ ×
	PS C:\Users\admir 90705bb55efb59da7 PS C:\Users\admir	<pre>>> & D:/python/py /fc2a5636549812a >> ■</pre>	thon.exe e:/py/easy-s	o.py flag				2.1 juon			
553											

题目: Ph0en1x-100

1、拖进夜神中安装运行,主界面只有一个输入框和一个按钮,随便输入信息,点击按钮后,弹出信息 Failed!,如下图所示:

へつつで 夜神塘拟器 6.3.0.0 Android 5		
		▼⊿ 🖬 3:51
Android_crackme		
	Input flag	
	Input hag	
	4555222	
	GO!	
	Failed	
		https://blog.csdn.net/qc_4/1374/107

2、查壳后无壳直接使用JEB反编译,查看MainActivity.java文件,发现要弹出信息 Success 逻辑如下:首先在so层注 册了两个静态函数-encrypt(String)、getFlag()函数,然后在java层有个函数 getSecret(String),将so层函 数 getFlag 返回值经过 getSecret 函数加密后与我们在输入框中输入的字符串经过 encrypt 函数后在经过 getSecret 函数 加密比较,如果一致,则返回 Success,由于比较的两个字符串最外层都经过 getSecret 函数加密,所有我们不需要在 管 getSecret 函数,直接让内部两个字符串一直一致即可得到flag!!!

JEB2 - C:\Users\admin\Desktop\XCTF\phoe	11x100\f6adc401d0eb472892a4ac4481f76a85.apkjdb2	-	٥	×
文件 编辑 Navigation 行为 Debugger 窗] 帮助			
😑 🗟 🏄 🛆 📑 🗐 🗟 수 수 🎯 /* 🖉 🖻	● 伊 ↔ 책 其 钰 李 ୲▷ □ 圖 전 ☆ 관 ଏ			
🐮 Project Explorer 🛛 😫 🗖 🗖	🗣 Bytecode/Hierarchy 🗣 Bytecode/Disassembly 🗋 MainActivity/Source 🖾 🗷 xml <unbound> 🗊 apk<unbound></unbound></unbound>		~	- 0
 C:Users\admin\Desktop\XCTP\phoen1; 6fad:401d0eb472892a4ac448176a8 6ad:401d0eb472892a4ac448176a8 Certificate Fodd:401d0eb472892a4ac448176 Posters Fodd:401d0eb472892a4ac448176 Libraries 	<pre>package com.ph0enlx.android_crackme; import android.comtent.Context; import android.support.v7.app.AppCompatActivity; import android.visw.View; import android.wisyEt.DitText; import android.wisyEt.DitText; import java.io.UnupportedEncodingException; import java.security.MessageDigest; import java.security.MessageDigest; import java.security.tends AppCompatActivity {</pre>			
	<pre>static (System.loadLibrary("phem"); } public MainActivity() (super(); } public native String encrypt(String argl) (} public native String getFlag() (}</pre>			
	<pre>public String getSecret(String arg12) { String VT = "WEXTANGENGENGENGENGENGENGENGENGENGENGENGENGEN</pre>			>
	Description Source			_
				- 0
< >>	Unit "AnimRes" (java) was created			Ĵ
Filter: type "Enter" to validate				>
coord: (0,70,104) addr: Lcom/ph0en1x/andro	d crackme/MainActivity;->onGoClick(Landroid/view;/V+3Eh loc: ?			

Xit Haw Mudding frag Dubloger BL HB Xit Haw Mudding frag Dubloger BL HB Y: Policy Could Could PL / Stall	JEB2 - C:\Users\admin\Desktop\XCTF\phoen	1x100\f6adc401d0eb472892a4ac4481f76a85.apkjdb2	-	٥	×
<pre>Pipel: Diversion of Pipel: Pipel</pre>	文件 编辑 Navigation 行为 Debugger 窗口	一帮助			_
<pre>Protect Dublet a</pre>		● 문 수 대 왕 전 월 ▶ 프로그 수 요 네 		~ =	
<pre> protection durated (arg2); filesetContentTiew(213)998001; this.setContentTiew(213)998001; this.setContentTiew(213); public void onOcClick(View arg5) { if(this.getEcret(this.getEcret(this.entrypt(this.setFlag.getText().toString())))) (Toast.makeText(((Context)this), "Success", 1).show(); } else { Toast.makeText(((Context)this), "Failed", 1).show(); } } public void once { Description Source Description Source Description Source Discription Source</pre>	** Project Explorer 2 2 □ • © C\Users\admin\Desktop\UCT\phoentfile • • 6 fadad-001 doeb472892a4ac4481f7€ • © fadad-001 doeb472892a4ac4481f7€ • © fadad-001 doeb472892a4ac4481f7€ • © fadad-001 doeb472892a4ac4481f7€ • © ertificate • © Bytecode • © Resources • © Libraries	<pre> Bytecode/Hierarchy</pre>		0	
Filter: type "Enter" to validate	<	<pre>protected void onCreate(Bundle arg2) { super-onCreate(arg2); this.setContentTiew(2130960601); this.setFlag = this.findViewBJd(2131492943); } public void onSoclick(View arg5) { if(this.getSecret(this.getSecret(this.encrypt(this.setFlag.getText().toString())))) { Toast.makeText(((Context)this), "Success", 1).show(); } else { Toast.makeText(((Context)this), "Failed", 1).show(); } }</pre>		>	
	Filter: type "Enter" to validate	<			>

3、使用IDA打开so文件,静态分析一下 encrypt 函数,发现逻辑很简单,就是将传进来的字符串的每个字符的ASCII 码减一;对于 getFlag 函数,由于该函数没有输入只有输出,直接用frida Hook该函数得到返回值即可,如下图所示:

😯 IDA - libphcm.so C:\Users\admin\Desktop\XCTF\phoer	ntx100\libphcm.so — 🗇	×
File Edit Jump Search View Debugger Options	Windows Help	
👩 🔜 🗢 🕶 🕶 🍋 🌇 🏝 🗣 🕽 🌾 🗖 🕥 📑	🟦 ☞ 📌 ☞ 🖉 🕨 🗖 🕼 debugger 🔹 🗏 🗃 🔐 🎬	
		•
📃 Library function 📕 Regular function 📕 Instructi	ion 🔤 Data 📕 Unexplored 💭 External symbol	
📝 Functions window 🛛 🗗 🛪 []	IDA View-A 🖸 🗓 Fseudocode-A 😫 🧿 Hex View-1 🖸 🖪 Structures 🗊 🗒 Enums 🗊 📆 Inports 🗊 🛃 Exports 🗊	
Function name Seqn F sub_300 .plt cxa_staxit .plt cxa_staxit .plt cxa_finalize .plt f sub_420 .ter f sub_420 .ter	<pre>lintcdel Java_com_ph0enlx_android_lorackme_MainActivity_encrypt(_JMLEnv *al, int a2, int a3) { size_t v3; // esi cost cher *s; // edi cost cher *s; // edi for (= al->functions->GetStringUTEChars((JMLEnv *)al, (jstring)a3, 0); v3 < strlen(s);s[v3++]) return al->functions->NewStringUTE((JMLEnv *)al, s); return al->functions->NewStringUTE((JMLEnv *)al, s);</pre>	
	00000660 Java_coa_ph0en1x_android_lcrackme_MainActivity_encrypt:7 (660)	
Output vindov		□ # ×
The inicial aucoanalysis has been finished. D:\androidtools\jni.h: successfully compiled		\$
Python		
ur 111 p pil ocan	https://biog.esdin.nel/ga	2013720107

AU: idle Down Disk: 29GB



Frida代码:

```
import frida
import sys
jscode = """
Java.perform(function(){
    Interceptor.attach(Module.findExportByName("libphcm.so","Java_com_ph0en1x_android_1crackme_MainActivity_getF
lag"),{
        onEnter: function(args) {
        },
        onLeave: function(retval){
            var String_java = Java.use('java.lang.String');
            var args_4 = Java.cast(retval, String_java);
            send("getFlag()==>"+args_4);
        }
    });
});
.....
def printMessage(message,data):
    if message['type'] == 'send':
        print('[*] {0}'.format(message['payload']))
    else:
        print(message)
process = frida.get_remote_device().attach('com.ph0en1x.android_crackme')
script = process.create_script(jscode)
script.on('message',printMessage)
script.load()
sys.stdin.read()
```

4、得到以上信息后,使用python脚本跑出falg即可,如下所示:

C:\Windows\System32\cmd.exe	—		×
E:\py>python ph0en1x100.py flag{Ar3_y0u_go1nG_70_scarborough_Fair}			Â
E:\py>_			
http	os://blog.csdn.	net/qq_413	74107,

Input flag flag{Ar3_y0u_go1nG_70_sca rborough_Fair} Go!

python脚本:

```
Flag = 'ek`fz@q2^x/t^fn0mF^6/^rb`qanqntfg^E`hq|'
flaglist = list(Flag)
stringlist = []
for ch in flaglist:
    stringlist.append(chr(ord(ch) + 1))
print(''.join(stringlist))
```

题目: RememberOther

PS: 此题为脑洞题, 披了一个安卓的皮而已!!!

1、下载好题目后,查壳发现无壳,直接拖进夜神中,要求输入用户名和注册码,随便输入,弹出信息无效用户名或 注册码,如下图所示:





2、用JEB反编译后,查看 onCreate 方法,发现将我们输入的用户名和注册码作为参数调用 checkSN 函数,并且当该 函数返回 false 时不弹出信息 无效用户名或注册码,接着去看 checkSN 函数,该函数当用户名和注册码为空时返回 false,返回false后弹出了一串md5值,再看 checkSN 函数其他逻辑,发现将输入的用户名经过md5加密后返回16进制 字符串,然后取该字符串的奇数位拼接成一个新的字符串,再然后与我们输入的注册码进行比较,返回 true,最后也没 有发现这个跟flag没什么关系,想起之前还有一串md5值,进行解密,解密得出 YOU_KNOW_,输入,提示flag错误。。。 看了一下其他大佬的wp,才发现在后面加上 ANDROID 就行了。。。。。因为压缩包里面有个word文件,里面写了不懂安 卓。。。。。。。

JEB2 - C:\Users\admin\Desktop\XCTF\Remei	nberOther\476d9022bb0449c09c0b1e24f0686b66\RememberOther.apk -	- 0	>	K
文件 编辑 Navigation 行为 Debugger 窗口	帮助			
🗁 🗟 🧨 🛆 📑 📲 🖓 🖓 🗢 🗢 🏉 🧨 🖊 🖉 🖻	■ 串 今 礼 業 職 季 ● 四 画 画 3			
🐾 Project Explorer 🛛 🔰 🗖 🗖	🗣 Bytecode/Hierarchy 🐥 Bytecode/Disassembly 🖉 MainActivity/Source 🛛 🗷 xml <unbound> 🕼 apk<unbound></unbound></unbound>		▽ □	
** Project Explorer IS Image: Clusers/admin/Desktop/XCTF/Rememl Image: Clusers/admin/Yestop/XCTF/Rememl Image: Clusers/admin/Yestop/XCTF/Rememl Image: Clusers/admin/Yestop/Xest	<pre>Bytecode/Hierarchy @ Bytecode/Disassembly @ MainActivity/Source [3] @ xml<unbound> @ apk<unbound> if(v5.toString().equalsIgnoreCase(arg13)) { return v7; } oatch(NoSuchAlgorithmException v2) { goto label_f1; return false; label_f1; return false; label_f1; public void onCreate(mumfle arg3) { return false; } public void onCreate(stg3); this.setCreate(arg3); this.setTile(2131094122); this.setTile(213109412</unbound></unbound></pre>	<pre>crim())) {</pre>		
	<pre>tosst.makerest(MainActivity.this, Zi3u04122, 0).show(); else { Tosst.makerest(MainActivity.this, Zi3u04120, 0).show(); MainActivity.this.btn_register.setEnabled(false); MainActivity.this.setEntle(Zi3u04118); } public boolean onCreateOptionsMenu(Menu arg3) { this.getMenuInflater().inflate(Zi3u06184, arg3); return 1; } public boolean onCreateOptionsMenu(Menu arg3) { this.getMenuInflater().inflate(Zi3u06184, arg3); return 1; } commute attric String found() arg7_String arg8) { Comple Comple</pre>		>	
	se Logger a processore Unit "MainActivity" (iava) was created			-
< >	one managerey (jura) was ereaded			Ç
Filter: type "Enter" to validate	<		>	
coord: (0,56,60) addr: Lcom/droider/crackme0	201/MainActivity;->checkSN(Ljava/lang/String;Ljava/lang/String;)Z+76h loc: ?			

JEB2 - C:\Users\admin\Desktop\XCTF\Remer	nberOther\476d9022bb0449c09c0b1e24f0686b66\RememberOther.apk –	o ×
文件 编辑 Navigation 行为 Debugger 窗口	帮助	
<u>⊨⊌≁ ∆ ¤!⊴⊴⇔⇒ ₽∦∕⊵ </u>	●●◆☆ ★☆ ◆●□■≥◇ふふう	
12 Project Explorer 🛛 🔄 🗳 🗖	🗣 Bytecode/Hierarchy 🗣 Bytecode/Disassembly 🗋 MainActivity/Source 🔤 🖻 xml <unbound> 🔯 apk<unbound></unbound></unbound>	
 CAUserstadmin/Desktopt/XCTFRememl RememberOther.apk Manifest Certificate Potecode AniActivity Resources 	<pre>static Button access300(MeinActivity argl) { return arg1.btn_register; } private boolean checkBN(String arg12, String arg13) { boolean v7 = true; try { if(arg12.length() == 0 && arg13.length() == 0) { return v7; } if(arg12 == null arg12.length() == 0) { return false; } MessageDigest v1 = MessageDigest.getInstance("MD5"); v1.reset(); v1.reset(); v1.reset(); string v3 = MainActivity.toRsString(v1.digest(), ""); StringBvildr v5 = new StringBvildr(); int v4; for(v4 = 0; v4 < v3.length(); v4 += 2) { v5.append(v3.charAt(v4)); } if(v5.toString().equalsIgnoreCase(arg13)) { return v7; } catch(McSuchAlgorithmException v2) { // return v7; } // return v7; } } } } }</pre>	
Filter: type "Enter" to validate	<pre>return false; label_41; v2.printStackTrace(); return false; < Description Source © Logger & @ Console Unit "MainActivity" (java) was created </pre>	
coord: (0.56.66) Laddr: Lcom/droider/crackme0	2 2010 Advin Activity: scheck SN(Liava/Jang/Stringt) iava/Jang/Stringt)7+76h Lloc: 2	
coord, (0,56,66) addr. Lcom/droider/crackmed	zv r/man/scurry,->chccks/ng/ad/ang/sunng,gd/dd/dd/g/Sunng,jz+7011 [UC: f	





C ① ● SoundScom	C ○ ● somd5.com	6 题目	× 🕄	writeup 🗙	📸 md5_百度 🗙 📀	MD5免费 × 🚺 md5	在线 🗙 📔 😇 Jav	a equa 🗙 📔 法 javaĦ	anur × 🤌 Java的位に ×	🕻 📔 java.lang. 🗙 🛛 🖸	Android® ×	🖸 Java trim 🗙 📔 🕂	-		<
※ 該用 ■ Translate 業 医感觉解 ③ Planetle Syntax ③ 和子・監督目的 2. ● 教育・監督目的 2. ● 教育・予算者 2. ● 教育・予算者 2. ● 教育・日本学校的 2. ● 教育・自然学校的 2. ● 社内大小WWW.andro ● 教育会会社 C SDN-S装订技术 ④ Kload ●	● Will ● Translate ● Edition ● Planetle syntax. ● Planetle syntax. ● Raw - Softwarg - ● AAHO ● Mail = FR2 edity. ● Mail	$\leftarrow \rightarrow$	C 🗅 🌘	somd5.con									*	8	1
输入让你无语的MD5 b3241668ecbeb19921fdac5ac1aafa69 解◎ md5 YOU_KNOW_	输入让你无语的MD5 b3241668ecbeb19921fdac5ac1aafa69	1 应用	Translate	🔠 吾爱破解	S PlanetB Syntax	😍 登录 - 慌伴lmg	🛞 个人中心	🤄 博客园 - 开发者的	🛅 简书 - 创作你的创作	🔯 http://www.andro	🞄 看雪安全论坛	C CSDN-专业IT技术	📹 iCloud		`
						b3241668ecbe md5 You_KNC	输入 eb19921fdac5 w_	让你不	:语的MC	D5 解密					

	FoundItFun.docx (受保护的视图) - Word	
帮助 💡 告诉我你想要做什么		
,否则保持在受保护视图中比较安全。	启用编辑(E)	

题目: app2

1、将下载好的题目拖进夜神中,发现要求登陆,随便点击登陆后,如下所示:





2、将apk用jeb反编译后,首先看一下MainActivity这个入口文件,发现没什么,就是将我们输入的用户名和密码传入SecondActivity页面中,然后跳转到该页面,再来看一下SecondActivity,发现调用了so层函数 doRawData,使其返回 值和字符串 VEIzd/V2UPYNdn/bxH3Xig== 进行比较,如下所示:





3、用IDA查看一下该函数,发现至少将传入的字符串进行了 AES-128-ECB 加密,并且发现密钥 thisisatestkey==,将 开始我们发现的字符串 VEIzd/V2UPYNdn/bxH3Xig== 进行解密,得到的字符串输入后发现不是flag,但是在FileDataActivity 文件中发现另一串字符串 9YuQ2dk8CSaCe7DTAmaqAA==,对其解密,得到flag: Cas3_0f_A_CAK3

IDA - libJNIEncrypt.so C:\Users\admin\	\Desktop\>	XCTF\ap	o2\libJNIEncrypt.so		- 0	o ×
File Edit Jump Search View Debu	igger Op	tions \	Vindows Help			
📂 🔜 🗢 🗸 🛉 👻 🌆 🌇 🍓 💵 🛛	<u>6</u> I 🗖 🥥)				
		,		:		-
Tiburun America 🗖 Benden Ameri			Date Brandund Frederical ambal			
Library function kegular functi	ion ins	truction	Data Unexplored External Symbol			
Functions window	0 8 ×		IDA View-A 📧 🗓 Pseudocode-A 🖾 🙆 Hex View-1 🖾 🕼 Structures 🖾 🔃 Enums 🖾 🛅 Imports 📧 🛃 Expo	rts 🗵		
Function name	Se ^	1	intfastcall doRawData(_]NIEnv *env, int a2, int a3, int a4)			
<pre>7cxa_stexit 7cxa_finalize stack_chk_fail 7 strlen 7 j.b54_encode 7 j.b54_encode 7 j.b54_encode 7 j.b54_encode 7aabbi_BCB_encrypt 7 j.b54_BCB_dcoryt 9 j.b54_BCB_dcoryt 9 j.b54_BCB_dcoryt 9 j.b54_BCB_dcoryt 9 j.b54_BCB_dcoryt 9 j.chcdc 9 j.chcdc</pre>	.p .p .p .p .p .p .p .p .p .p .p .p .p	2 2 3 3 4 4 5 5 6 6 7 7 7 7 7 9 9 9 9 9 9 10 11 12 13 13 14 15 16 17 17 18 19 20 20 22 23 24 24 25 26 26 26 27 28 29 20 30 31 32 24 24 25 27 28 29 30 30 31 32 24 24 25 27 26 26 26 27 27 28 29 20 20 20 20 20 20 20 20 20 20	<pre>[JNHEnv *v4; // r4 int v5; // r9 int result; // r9 int v1; // [sp+18h] [bp-18h] will int v1; // [sp+18h] [bp-18h] will int v1; // [sp+18h] [bp-18h] v4 = env; v5 = ad; if (j_checkSignature((int)env, a2, a3) != 1 [(strcpy((char ~)&v12, ~thisiatestkey=="), [VIEnv *, int, _DWORD))v4->functions->GetStringUTFChars)(v4, v5, 0), v7 = j_AES_128_ECPR(SEPAGdidg_Encrypt(v6, (int)&v12), ((void (_fastcall *)(_JNIEnv *, int, const char *))v4->functions->ReleaseStringUTFChars)(v4, v5, v6),</pre>			
jgnu_Unwind_RaiseException	.p _	,				
<	>					
		_	00003972 doRayData:17 (3972)			
Output window						Пēх
1200: USING guessed type intTa	stcarr J	_004_er	code(_DworkD, _DworkD);			
6004: using guessed type char *UN	ISIGNATUR	E[3];				-
Python						
AU: idle Down Disk: 29GB						1_41374107

JEB2 - C:\Users\admin\Desktop	p\XCTF\app2\2554cf208cfb4cdf9218a840fa9bf237.apk	-		\times
文件 编辑 Navigation 行为 De	bugger 窗口 帮助			
🕞 🔚 🥕 💧 🔁 🚽 🖓 🗘 🖒 🌢	▶ /* 2 2 4 🖶 伊 4 項 ¥ 装 梦 🖤 🗉 🗏 조 🕫 🖻			
😘 Project Explorer 🛛 😫 🗖 🗖	🗣 Bytecode/Hierarchy 🗣 Bytecode/Disassembly 🖸 FileDataActivity/Source 🛛 🛚 xml <unbound> 🕼 apk<unbound></unbound></unbound>		~	- 8
 C. (Serstadmin/Desktop/v 2554cf208cfb4cdf9218a 2554cf208cfb4cdf92? Manifest Certificate Bytecode Esources 	<pre>import android.os.Bundle; import android.widget.TextView; import com.tencent.testvuln.c.Encryto; public class FileDataActivity extends a { private TextView c;</pre>			
> 🅞 Libraries	<pre>public FileDataActivity() { super(); } protected void onCreate(Bundle arg3) { super.onCreate(arg3); this.setContentView(2130903042); this.c.setText(Encryto.decode(this, "9YuQ2dkBCSaCe7DTAmakpA==")); } } </pre>			>
< >>				^
Filter: type "Enter" to validate	<			> *
coord: (0.17.64) addr: Lcom/tenc	ent/testvuln/FileDataActivity:->onCreate(Landroid/os/Bundle:)V+36h loc: ?	ps://blog.csdp.ne	t/cjcj_413	374107

\leftrightarrow \rightarrow C \triangle	▲ 不安全 tool.chacuo.net/cryptaes	*	6
🛄 应用 🧕 Translate	e 🔡 吾爰破解 😚 PlanetB Syntax 🔅 登录 - 慌伴Img 😮 个人中心 🛛 博客园 - 开发者的 🔞 简书 - 创作你的创作 🔛 http://www.andro 🐁 看雪安全论坛 🧲 CSDN-专业IT技术 🧃	iCloud	
 » RC5加密解密 » RC6加密解密 » Blowfish加密解密 » Twofish加密解密 » Serpent加密解密 	BVLCARI 宝格丽Fiorever咏绽系列作品 将象征幸福与快乐的花朵融入光彩夺目的钻石作品中 绽放隽永不思的璀璨光芒.		
» Gost加密解密 » Rijndael加密解密 » Cast加密解密 » Ytoobnsser	AES加密模式: ECB v 現充: zeropadding v 数据块: 128位 v 密码: thisisatestkey== 備移電: iv偏移量, ecb模式 輸出: base64 v 字符集: gb2312 v		
 ▲ ACEOULEAFEL 非对体性加速解密 * Isa公式加速解密 > Isa系括加速解密 > RSA系相密码降密 > RSA系相密码降波(● RSA系相密码降波(● RCSA等相转化 > 校验RSA密钥对 > AG4的案相对 > DSA密钥对 			4
	ALS加密、解始转换结束(Daseb47)(也) × P		
	flag		

题目:黑客精神

1、将下载好的题目拖进夜神模拟器中,发现要求注册,随便点击注册后,弹出一个弹框,提示 已注册。

○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○			
			V 3 :21
55555			
			注册
1			
	回复		
	你的注意可以去		
	恣的注册码已保仔		
		+ 7 nm	
		好吧	

2、用JEB反编译后,跟进MainActivity文件,发现点击按钮后就一个弹出弹框,点击弹框确定后,跳转到 RegActivity界面去,在该界面点击注册后,调用了so层函数 saveSN。



JEB2 - C:\Users\admin\Deskto	p\XCTF\黑客精神\4c1def82.apk.jdb2 - □ >	<
文件 编辑 Navigation 行为 De	bugger 窗口 帮助	
😑 🗟 🥕 🛆 🗠 🗂 🚽 🚭 🔶 🔶 🕯	❷ /* ⊘ ⊵ ⊕ ⊕ ⊕ ҵ Ӿ ҵ 券 ⊪ ഥ ⊟ ゑ ⊚ 爬 ⊴	
🖺 Project Explorer 🛛 🔄 🗳 🗖 🗖	🌸 Bytecode/Hierarchy 🌸 Bytecode/Disassembly 🖸 RegActivity/Source 🖾 🖻 xml <unbound> 📮 apk<unbound> 🗢 🌄</unbound></unbound>	
 C:\Users\admin\Desktop\X 4c1def82.apk 4c1def82.apk Manifest Certificate Bytecode Resources Charaises 	<pre>public class RegActivity extends Activity { private Button btn_reg; private EditText edit_sn; public RegActivity() { super(); } static EditText access\$000(RegActivity arg1) { return arg1.edit sn; } }</pre>	
	<pre>} public void onCreate(Bundle arg3) { super.onCreate(arg3); this.setContentView(2130968603); this.setContentView(2130968603); this.setT_reg = this.findViewById(2131427412); this.edit_sn = this.findViewById(2131427413); this.btn_reg.setOnClickListener(new ViewSonClickListener() { public void onClick(View arg5) { String v0 = RegActivity.this.edit_sn.getText().toString().trim(); if(v0 == null v0.length() == 0) { Toast.makeText(RegActivity.this, "\$M\$\Delta\De</pre>	
	Cescription Source	
		T
< >		^
Filter: type "Enter" to validate	()	~
coord: (0,14,0) addr: Lcom/gdufs	/xman/RegActivity; loc: ?	07

3、用IDA分析so文件发现在java层注册的native函数都是动态注册的,此时用 Ctr1+S 找到.data 段进入,发现对应的函数 n1 、 n2 、 n3 。

File Edit Jump Search View I	Debugger Options V	/indows Help	
🖆 🖬 🖉 🕶 🕶 👘 🛍 🗛	🕨 🍂 🛋 🖾		
1			-
📕 Library function 📕 Regular fu	unction 📕 Instruction	🔲 Data 📕 Unexplored 🛑 External symbol	
📝 Functions window	- 8 × 💽	IDA View-A 🛽 🖸 Hex View-1 💟 🖪 Structures 💴 🗄 Enums 🖾 🕅 Imports 🖾 😰 Exports 🖾	
<pre>Function name 7</pre>	Se ^ .p .p .p .p .p .p .p .p .p .p .p .p	LCAD: 090000000 ; LCAD: 09000000 ; LCAD: 090000000 ; LCAD: 0900000000 ; LCAD: 090000000 ; LCAD: 090000000 ; LCAD: 090000000 ; LCAD: 090000000 ; LCAD: 09000000 ; LCAD: 090000000 ; LCAD: 090000000 ; LCAD: 0900000000 ; LCAD: 0900000000 ; LCAD: 09000000000000 ; LCAD: 0900000000000000000000000000000000000	
<pre>/ freed / streap / streap / streap / raise /smu_Uwrind_Find_exidx / abort /cxa_begin_cleanup /cxa_type_match / n2 / sub_IDC / n2 / setValue / setValue / setValue / setValue / callWork / ab ab</pre>	.p .p .p .p .p .p .t .t .t .t .t .t	LOAD:eq G2. ARM. exitab 00002C34 00002C36 R F L dword 07 Lbd13-592632012012636 00 LOAD:eq G2. ARM. exitab 00002C350 R L L dword 09 public CONST 32 00 00 LOAD:eq G2. rodata 00002F30 R L L dword 00 public CONST 32 00 00 LOAD:eq G2. rodata 00002F30 R F L dword 00 public DATA 32 00 00 LOAD:eq G2. init_array 00004E54 0004E58 R F L dword 00 public DATA 32 00 00 LOAD:eq G2. init_array 00004E54 0004E58 R F L dword 00 public DATA 32 00 00 LOAD:eq G2. init_array 00004E54 0004E58 R F L dword 00 public DATA 32 00 00 LOAD:eq G2. init_array 00004E54 0004E58 R F L dword 00 public DATA 32 00 00 LOAD:eq G2. init_array 00004E50 00005000 R F L dword 00 public DATA 32 00 00 LOAD:eq G2. init_array 00004E50 0000500 R F L dword 00 public DATA 32 00 00 LOAD:eq G2. init_array 00004E30 0000500 R F L dword 00 public DATA 32 00 00 LOAD:eq G2. init_array 00004E30 0000500 R F L dword 00 public DATA 32 00 00 LOAD:eq G2. init_array 0000500 0000501 7 7 2 . L dword 00 public BATA 32 00 00 LOAD:eq G2. init_array 0000503 0000503 7 7 2 . L dword 11 public 32 00 10 LOAD:eq G2. init_array 0000508 00005094 7 7 7 . L dword 11 public 32 00 11 LOAD:eq G2. init_array 0000508 00005094 7 7 7 . L dword 11 public 32 00 11 LOAD:eq G2. init_array 0000508 00005094 7 7 7 . L dword 11 public 32 00 11	
JNI_OrnLoad J.JNI_OrnLoad J.JNI_OrnLoad J.JNI_OrnLoad J.JNI_OrnLoad J.JNI_OrnLoad J.JNI_OrnLoad	. t t t . t t . t t . t t . t t t * . * *	LOAD:00 LOAD:00 LOAD:00 LOAD:000 LOAD:0000000000 ; Byte sex : Little endian LOAD:0000000000 ; Segment type: Pure code LOAD:000000000 ; Segment type: Pure code LOAD:0000000000000 ; Segment type: Pure code LOAD:0000000000 ; Segment type: Pure code LOAD:000000000 ; Segment type: Pure code LOAD:00000000 ; Segment type: Pure code Segment type: Pure code Segm	
-		nnnnnnn nnnnnnn: Frenslakal (220culaursed aftu Hex Area-1)	~
Output window	DPPD TIDISDED		0 8 ×
D:\androidtools\jni.h: success	sfully compiled		0
Python			

17

AU: idle Down Disk: 26GB

n IDA - libmyjni.so C:\Users\admin\Desktop\XCTF\黑客精神\libmyjni.so 🕅

IDA - libmyjni.so C\Users\admin\Desktop\XCTF\黑客精件\libmyjni.so — 日 🗙										
file Edit Jump Search View Debugger Options Windows Help										
📑 🛃 🗢 🕶 👻 🆓 🆓 🐧	🕨 🙇 💽 🛋	🚮 📌 🖛 🧩 📷 🗙 🕟 💷 🔲 No debu	gger ·	• *: 🗗 🗊 🚏 🎬						
Library function 🗖 Regular fun	Library function 📕 Regular function 📕 Instruction 📕 Data 📕 Unexplored 🖷 External symbol									
Functions window	- 8 × 💽	IDA Viev-A 🛛 💽 Hex Vi	ew-1 🛛 🖪 Struct	ures 🗵 🗮 Enuns		Imports		Exports	X	
Function name 7cxs_atexit 7cxs_inalize 7 fopen 7 strlen 7 strlen 7stack_chk_fail 7stack_chk_fail 7stack_chk_fail 7stack_chk_fail 7stack_fail 7 felose 7 ftel 7 strcap 7 str	Se ∧ .p .p .p .p .p .p .p .p .p .p	.data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005001 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005000 .data:00005020 .data:00005028 .bss:0000508 .bss:0000508 .bss:0000508 .bss:0000508 .bss:000508 .bss:0000508 .bss:000508 .bss:0000508 .bss:000508 .bss:0	be: Pure data AREA .data, DATA CON C.C.CON DCB 0 DCB 0 DCB 0 DCB 0 DCD aV DCD aV DCD aV DCD aV DCD aV DCD aInitsn DCD aV DCD aVavalangStrin DCD a24 DCD aVavalangStrin DCD a24 DCD aV DCD A	<pre>; DATA XREF: sub_118C+41 ; .text:off_11C8fo ; DATA XREF: JNL OnLoad ; .text:off_1590fo ; "initSN" ; "()V" ; "saveSN" ; "()V" ; "uov/lang/String;)); ; "work" ; "()V" ; DATA XREF: LOAD:00000032 ; LOAD:0000067Cfo</pre>	o 381 		÷ ط م	ijava层中的ini 的函数	tSN等函数经动态	。 注册后对
<pre></pre>	.t .t .t .t *	.bss:0000502C native_class .bss:0000502C native_class .bss:0000502C ; .bss .bss:0000502C ; .bss .prgend:00005030 ; ========	ends	; DATA XREF: LOAD:0000031 ; JNI_OnLoad+36†o	LC†o					
		00004000 00005000: .data:unk_50	00 (Synchronized with Hex Vi	ev-1)						~
Output window										□ # ×
The initial autoanalysis has be	een tinisned.									^
D:\androidtools\jni.h: success	tully compiled									~
Python									hiteedia on order of the	
AU: idle Down Disk: 26GB										

4、先分析一些 n1 函数, F5 大法后,可以很轻易看出该函数作用是创建了一个文件 /sdcard/reg.dat, 然后读取该 文件,与字符串 EoPAoY62@E1RD 进行比较;再来看一下 n2 函数,该函数对应这java层的 saveSN 函数,首先前面做了一大 堆令人看不懂的操作,然后将我们输入的注册码的每个字符与另一个字符进行异或操作(该字符我们不知道,所以无法 得到最终异或结果);再来看一下 n3 函数,该函数首先调用了 n1 函数,若结果为真,则将一大串字符串赋值给 v4,双 击去看一些该字符串,使用 A 健将其转为ascii后,发现提示 输入即是flag,格式为xman{.....}!。

💱 IDA - libmyjni.so C:\Users\admin\Deskto	pp\XCTF\黑客精神\libmyjni.so	- o ×
File Edit Jump Search View Debug	ger Options Windows Help	
📂 🔜 🛛 🗢 🕶 🕶 🏪 🏝 🗛 🖡 🔬	a 🔺 🕒 📾 📾 🕼 📌 🕶 🗶 🕨 🔲 🖬 No debugger 🔹 🤨 🛃	
:		
Library function 🗖 Regular functio	n 📕 Instruction 📕 Data 📕 Unexplored 📕 External symbol	
T Functions window	🗆 🗗 🗙 🔯 IDA View-A 🗵 🗳 Pseudocode-A 🖾 💆 Hex View-1 🗵 🖪 Str	actures 🗵 🗄 Enums 🗵 📆 Imports 🗵 📝 Exports 🗵
Function name 7cxs_atexit 7cxs_inalize 7 fopen 7 strlen 7 strlen 7 strlen 7stack_chk_fail 7stack_chk_fail 7stack_chk_fail 7stack_fail 7 felote 7 felote 7 strcap 7 s	<pre>Se 12 p 13 y v1 = a1; y 2 = fopen("/sdcard/reg.dat", "r+"); p 15 y 3 = v2; p 17 f (y 4 = v1; p 19 v5 = v3; p 20 y 5 = v3; p 21 p 23 v6 = ftell(v3); p 24 v7 = malloc(v6 + 1); p 25 v5 = 0; p 26 f (1r7) p 26 f (lose(v3); p 27 v5 = 0; p 28 v4 = v1; v5 = 0; p 29 v5 = 0; p 29 v5 = 0; p 20 return setValue(v4, v5); p 21 seek(v3, 0, 0); t 33 fread(v7, v5, 1u, v3); t 34 return setValue(v4, v5); j 5 t 35 if (1strcmp((const char *)v7, "EoPAoY62@E1R0")) t 36 f (1strcmp((const char *)v7, "EoPAoY62@E1R0")) t 34 if 0 = 1; t 34 v10 = 1; t 34 v10 = 1; t 44 j 5 etValue(v9, v10); t 5 etValue(v9, v10); t</pre>	↑ 创建了一个文件,然后读取该文件的内容与*EopAoY62@E1RD*比
F euh 1804	• • • • • • • • • • • • • • • • • • •	
	00001412 n1:34 (1412)	v
Output window		0 8 ×
1304: Using guessed type inttas	tcall i felece(DHORD);	^ · · · · · · · · · · · · · · · · · · ·
Python	(Call J_(Close(_DWORD),	·
		Difect/Directo estre 205/2016

AU: idle Down Disk: 26GB

💱 IDA - libmyjni.so C:\Users\admin\l	Desktop\XCTF\黑	客精神\libmyjni.so									-	o ×	
File Edit Jump Search View [Debugger Optic	ons Windows Help											
i 📂 🛃 i 🦛 🕶 🗣 i 🌆 🌇 🦓 i 🎼	1 6 2 0	雨山は マーイ 山)	🗙 🕨 🔲 🗖 No debug	ger	- 🕫 🖬 👔 🕈	<u>و</u> ×							
						-							-
Library function Regular fu	unotion 📕 Insta	untian 📃 Data 📕 Una	mlanad External or	mhal						 			_
Transform window		action bata e one.	proreu — Excernar sy		N N N		(FT) -	-					
Punctions window		UE IDA Viev-A 🔛	us Pseudocode-A 🔝	Stack of n2	Hex Viev-1	🖪 Structures 🔛	Enuns		Imports	Exports			_
Function name	Se ^	■ 34 V8 += 8											^
<pre>fcxa_atexit</pre>	. p	 30 310 310	(_DWORD ~)V8 - 1);										
<u>f</u> cxa_finalize	. p	 37 37 37 	v10·										
7 Iopen	. p	 38 38 v11 = v 	7 + 2.										
f frute	. p	 39 v7 += 2 	,										
f stack chk fail	. p	40 }											
f android log print	.p	41 while (vi	3 != "E");										
7 fclose	. p	42 v12 = v3;											
f fseek	. p	43 v13 = 201	5;										
f ftell	. p	44 *v11 = *(_WORD *)v8;										
f malloc	. p	45 v14 = v3-:	<pre>>functions;</pre>										
f fread	. p	46 v15 = 0;											
f moise	. p	47 v16 = (coi v16 = (coi	nst char *)((int (_	_fastcall *)(_JNIEn	v *, int, _DWORD))	/14->GetStringUTFCh	nars)(v12, v4,	0);					
f am Inwind Rind oxids	. p	48 v17 = v16											
f abort	. p	49 v18 = str.	len(v16);										
f memcpy	.p	50 while (v)	15 < v18)										
📝cxa_begin_cleanup	. p	51 {	8 D 1 1										
🗾cxa_type_match	.p	52 IT (VI:) /o 5 == 1)										
f sub_11BC	.t	● 54 ↓ v13 -	(y13 + 5) % 16										
f n2	.t	 55 55 55 	$*(2y_{23} + y_{13} - 23)$										
f getValue	.t	56 }	(4125 - 115 - 25)	,									
f setvalue	.t	57 else if	(v15 % 3 == 2)										
f callWork		58 {											
f n3	. t	59 v13 =	(v13 + 7) % 15;										E
f JNI_OnLoad	.t	60 v19 =	*(&v23 + v13 - 22)	;									
f JNI_OnUnLoad	.t	61 }											
fdivsi3	.t	62 else											F
f sub_15F4	. t	63 {											
aeabi_idivmod	.t	64 v13 =	(v13 + 3) % 13;										
Jaeabi_ldiv0	.t	65 v19 =	*(&v23 + v13 - 21)	;									
f sub_1018	.t	66 }				将我们输入的字符	;串与v19逐位进	进行异或					
f sub_1784		67 v20 = *	/17;										
7 sub 1804	· * ~	68 ++v15;											
<	>	• 69 *((_BYTI	*)++v17 - 1) = v2	0 ^ v19;									
		000012CE n2:65	(12CE)										~
Output window												0 8	×
2014: USINg guessed type int	TASICALL J	_android_iog_print	(_рмокр, _рмокр, _р	жоко);									^
2C24: using guessed type int	tastcall j_f	close(_DWORD);								 			~
Python										https:/	//electescienc	ana zerozen	

AU: idle Down Disk: 26GB



💱 IDA - libmyjni.so C:\Users\admin	\Desktop\XCTF\黑客精神	libmyjni.so	- 🗆 ×
File Edit Jump Search View	Debugger Options V	indows Help	
📑 🔜 🖕 🖛 👻 🦓 🏀 🍇	1 🙀 🖾 🔘 📾 📾	🕼 🕈 🛫 😹 🗙 🕨 🔲 🗈 No debugger 💎 🤫 🛃 🔐 😭	
			-
Library function 🗖 Regular f	function 📕 Instruction	🔲 Data 📕 Unexplored 📒 External symbol	
📝 Functions window	- 8 × 💽	IDA View-A 💶 🗓 Pseudocode-A 🗵 💽 Hex View-1 🗵 🖪 Structures 🗉 🗒 Enums 📧 📆 Inports 📧 📝 Expo	ts 🗵
Function name	Se ^	.rodata:00002E40 ; .text:off_14C41o	^
<pre>Tcca_atexit /cca_atexit /cca_finalize / strlen / strlen / strlen / fuuts /stack_chk_fail /stack_chk_fail /strlen / felse / felse / felse / falloc / fread / strcap /cca_togin_cleanup /</pre>	2 .p .p .p .p .p .p .p .p .p .p	<pre>* crodata:00002E43 aljavalangStrin DCB "(Ljava/lang/String;)V",0 rodata:00002E45 ; DATA XXEF: callWork+420 rodata:00002E58 asc_2E58 DCB "还不行呢!",0 ; DATA XXEF: callWork+420 rodata:00002E58 asc_2E58 DCB "还不行呢!",0 ; DATA XXEF: n3+14fo rodata:00002E68 eFlagXelan DCB "输入即是flag,将式以xman(]!",0 rodata:00002E68 eFlagXelan DCB "输入即是flag,将式以xman(]!",0 rodata:00002E68 eFlagXelan DCB "输入即是flag,将式以xman(]!",0 rodata:00002E68 eFlagXelan DCB "和T_OnLoad()",0 ; DATA XXEF: n3+26fo rodata:00002E68 eFlagXelan DCB "和T_OnLoad()",0 ; DATA XXEF: DATE: DATE: rodata:00002E68 eFlagXelan DCB "NT_OnLoad()",0 ; DATA XXEF: DATE: rodata:00002E68 eFlagXelan DCB "NT_OnLoad()",0 ; DATA XXEF: DATE: rodata:00002E68 eFlagXelan DCB "RegisterNative()> nativeNethod() of:,0 rodata:00002E68 eFlagXelan DCB "RegisterNative()> nativeNethod() of:,0 rodata:00002E68 eFlagXelan DCB "RegisterNative()> nativeNethod() of:,0 rodata:00002E68 eFlagXelan DCB "RegisterNative()> nativeNethod() of:,0 rodata:00002E69 eFlagXelan DCB "RegisterNative()> nativeNethod() of:,0 rodata:00002E69 eFlagXelan DCB "RegisterNative()> nativeNethod() of:,0 rodata:00002E69 eFlagXelan DCB "NII_ONLOAd()",0 ; DATA XXEF: JNII_ONLOAd+S2To rodata:00002E69 eFlagXelan DCB "UnregisterNative() ->> nativeNethod() of:,0 rodata:00002E60 eFlagXelan DCB "UnregisterNative()",0 rodata:00002E60 eFlagXelan DCB "UnregisterNative()",0 rodata:00002E76 eFlagXelan DCB "InregisterNative()",0 rodata:00002E76 eFlagXelan DCB "InregisterNative()",0 rodata:00002E76 eFlagXelan DCB "InregisterNative()",0 rodata:00002E76 eFlagXelan EFlagXelan EFL DII_ONULoad+30to rodata:00002E76 eFlagXelan EFLAGXEFT DII_ONULOAd+30to rodata:00002E76 eFLAGXEFT DII_ONULOAD eFlagXelan:000002E76 eFLAGXEFT DII_ONULOAD+30to rodata:00002E76 eFLAGXEFT DI</pre>	
		00002E6B: 00002E6B: .rodata:aFlagIman (Synchronized with Hex View-1)	~
Output window			□ 8 ×
2024: Using guessed type int			0
Python		,	*
All: idle Down Disk: 26CB			https://blog.csdn.nel/qq_2PI372PI07

5、按照提示,我们将开始发现的字符串 EoPAoY62@E1RD (只发现这一个字符串,不输入这个输入啥)作为字符串进行输入,然后将文件 /sdcard/reg.dat 拷贝出来,打开文件一看,发现flag!!!



🧾 reg.dat - 记事本														-		×
文件(F) 编辑(E) 格: 2016084ml233	式(O) 査	语(V) 幕	昏助(H)													~
201000Am:200																
	0	μţ		0	K)	*	R	ර	•	2	∗		https://bl	log.csd	9 . Gq_@13	\$741 11 7

题目: easy-dex

1、下载好题目后,拖进夜神中,发现直接是黑屏的,在JEB中反编译后,发现在 AndroidManifest.xml 文
 件 application 和 activity 标签中存在 android:hasCode="false" 和 android:name="android.app.NativeActivity",说明
 这是个纯C++编写的,并且不含java代码,也就是 Native Activity。



AU: idle Down Disk: 27GB



2、既然是 Native Activity,直接找到so文件,拖进IDA中,寻找 android_main 方法(这是 Native Activity 的入口 方法,关于 Native Activity 的一些基础知识,我会在将一些我觉得写得比较好的博客链接附在文末),发现存在一 个'write'方法,结合包名'findmydex',大胆推测一波此处就是将dex写入某个文件中,那么直接开启动态调试,在关键地 方下好断点后,终于运行到了 write 函数处,结合传进去的参数,直接dump下来整个内存,用010打开一看,全都 是'00000...',。。。。。卒!!!!!



3、好吧,开玩笑的!!! 动态调试dump下来的内存有问题,一看就不是dex文件,开始还以为调试的时候哪里出来问题,导致dump出来的有问题,然后接着动态了一整个下午,发现好像dump下来的加密后的dex,去看了一下大佬们的wp,发现都是dump下加密后的dex,然后直接解密,好吧,再来看 android_main 函数,发现一开始就通过_aeabi_memcpy 函数将加密后的 dex 文件加载进来了,我们可以轻松看到加密后的dex文件首地址为 0x7004 (ida使用F5 后,要使用那一块内存空间地址直接是以&unk_地址命名的,所以首地址可以轻松看出来是 0x7004),大小为 0x3ca10,那么直接在静态下执行dump脚本即可,至于解密,把想应的c语言转换为python即可,至于最后为啥要进行一次zip的解压操作,是因为在 android_main 函数中解密完成后调用了 uncompress 函数进行了解压缩(更偷懒的可以直接把F5后的c 代码复制下来,替换一下就行了)。

👧 IDA - libnative.so C:\Users\admin\Desk	ktop\XCTF\e	easy-dex	(libnative.so											-	٥	×
File Edit Jump Search View Debu	igger Optio	ons W	indows Help													
1 🛋 🔲 1 (a) y ab y 1 Ab Ab Ab Ab 🚹 🚺		the star	1.1.1.1.1.X	No.	tehugger			• 🕸 🗈	🔐 🐏 😥							
					40046801											
· · · · · · · · · · · · · · · · · · ·	_															
📃 Library function 📃 Regular functi	on 📕 Instr	uction	📃 Data 📕 Ünexpl	ored Extern:	al symbol											
📝 Functions window	0 8 ×	1	IDA View-A 🖂	🖪 Pseudoco	ode-A 🖾	🖸 Hex	View-1 🛛	3 A	Structures	Enuns		Inports	Exports			
Function name	Se ^	46	int v45; //	sp+D4h] [bp-	8Ch]											^
<pre>/cxa_atexit /cxa_finalize /cxa_finalize /extak_chk_fail /aeabi_semcpy /aeabi_semcur8 / ASencorManager_getUntance / ASencorManager_getUntance / ASencorManager_getEventue /android_log_print / AComportanger_createRventQueue android_log_print / ALcompr_pollAll / remove / AsencorManager_getEvents / uncompress / open / vrite / uncompress / close / free access / addin / eglcetDipplay / eglcetDipplay / eglcetEventSuffersCecnetry / eglcetEventSuffersCecnetry / AMativeHindow_setEventSuffersCecnetry / AMativeHindow_setEventSuffersCecnetry / eglcetConfig trib / AMativeHindow_setEventSuffersCecnetry / eglcetEventSuffersCecnetry</pre>	.p .p .p .p .p .p .p .p .p .p .p .p .p	48 48 49 50 50 51 52 53 54 45 55 56 57 55 58 58 59 60 16 2 63 64 4 65 66 4 65 66 4 67 7 7 2 2 63 3 7 4 55 59 60 11 62 63 64 64 65 67 7 7 7 8 8 8 9 7 0 0 0 11 52 3 55 56 56 57 7 7 59 9 60 11 62 63 64 64 65 7 7 7 7 7 8 8 9 9 0 0 11 62 63 7 7 7 7 7 7 8 9 9 0 0 11 62 63 7 7 7 7 7 7 7 7 7 7 8 9 9 9 0 0 11 1 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	<pre>int vd3; // int vd3; // int vd3; // int vd3; // int filename; int v52; // int v53; // int v53; // int v53; // int v54; // int v55; // int v56; // int v56; // int v59; // int v59; // int v61; // int v61; // int v62; // int v62; // int v62; // int v63; // int v63; int v64; int v64;</pre>	<pre>sptCh1 [bp; sptCh1 [bp- sptEdb1 [bp- // [sptEdb1 [bp- // [sptEdb1 [bp- // [sptEdb1 [bp- sptFdb1 [bp- sptFdb1 [bp- sptFdb1 [bp- sptFdb1 [bp- spt10db1 [bp spt10db1 [bp spt10db1 [bp spt10db1 [bp spt10db1 [bp spt10db1 [bp spt11db1 [bp]]</pre>	bah] 8dh] 8dh] 8dh] 8dh] (bp-7Ch] -7Ah] (bp-7Ch] (bp-7Ch] 7dh] 6dh] 6dh] 6dh] 6dh] 6dh] 6dh] -6dh] -5ch] -5dh] -5dh] -5dh] -3dh] 1000000u); e_t)off_4; 0d, v2); b	3A18); л∎	密后的de	■x文件大	小, 点击进	 似看见大小	小为0x3ca	10				l
		• 81	v58 = -193687	9484;												
1. I.			000025A0 android	main:46 (25A0)												~
Output window																₽×
ZC14: USINg guessed type IntTa	Stcall Suc	J_ZC14(_DWORD);													^
43A18: using guessed type void *o	++_43A18;															~
Python																
AU: idle Down Disk: 28GB																

IDA - libnative.so C:\Users\admin\Desktop\XCTF\easy-dex\libnative.so File Edit Jump Search View Debugger Options Windows Help

📑 🖬 🛛 🗢 🕶 🖛 🛍 🛍 🐴 🗼 👧 🗖 🕥	init at at st ▼ at X > D □ No debugger	
📃 Library function 📒 Regular function 📕 Instr	ruction 📃 Data 📕 Unexplored 📕 External symbol	
📝 Functions window 🛛 🗗 🛪	🖪 IDA View-A 🛛 🖪 Pseudocode-A 🛛 🧿 Hex	View-1 🖸 🖪 Structures 📧 🗒 Enums 📧 🗺 Inports 🗵 📴 Exports 🗵
Function name Se^	46 int v45; // [sp+D4h] [bp-8Ch]	
7 cxa_atexit .p 7 cxa_finalize .p 7 stack_chk_fail .p 7 nalloc .p 7 aeabi_memcpy .p	47 imt v46; // [sp-t0h1] [bp-38h] 48 ter v47; // [sp-t0h1] [bp-38h] 49 imt v48; // [sp-t0h1] [bp-38h] 50 _int16 v49; // [sp-t6h1] [bp-7h] 51 char v50; // [sp-t6h1] [bp-7h]	Execute script
f j_app_dummy .p	52 int v52. // [sp+E(h] [bp-74h]	
ASensorWanager_getInstance .p ASensorWanager_getDefaultSensor .p ASensorWanager_createEventQueue .p android_log_print .p	54 int v53; // [sp+F0h] [bp-70h] 55 int v54; // [sp+F4h] [bp-6ch] 56 int v55; // [sp+F8h] [bp-68h] 57 int v56; // [sp+F6h] [bp-64h]	Spippet list Please enter script body Name Default snippe*** addr = 0x7004 size = 0x3cal0
// ALcoper_pollAll .p // remove .p // AsensorEventQueue_getEvents .p // time .p	58 int v57; // [sp+100h] [bp-60h] 59 int v58; // [sp+104h] [bp-5Ch] 60 int v59; // [sp+108h] [bp-58h] 61 int v66; // [sp+10Ch] [bp-54h]	<pre>with open('dump','wb') as f: f.write(get_bytes(addr,size))</pre>
f open .p f open .p f write .p f close .p	62 int v61; // [sp+110h] [bp-50h] 63 int v62; // [sp+114h] [bp-4Ch] 64 int v63; // [sp+118h] [bp-48h] 65 char v64; // [sp+11ch] [bp-44h]	Line:9 Column:22
f access	66 int v65; // [sp+124h] [bp-3Ch]	Scripting language Python 🔹 Iab size 4 🗸
< > > المعالم معالم معالم معالم معالم معالم معالم معالم معالم معالم معالم	<pre>0 / v1 = a1; 0 68 v1 = a1; 0 69 destLen = 0x100000; 0 70 dest = (Byttef *)malloc(0x100000u); 71 v2 - off 4348; 71 v2 - off v</pre>	Run Export Inport
	<pre>72 v3 = (Bytef *)malloc((size_t)off_43A18); 73aeabi_memcpy(v3, &unk_7004, v2);</pre>	
	000025A0 android_nain:46 (25A0)	
Output window		
1000: using guessed type intfastcall AS 2259: using guessed type intfastcall sul 2368: using guessed type intfastcall sul 2988: using guessed type int sub_2988(); 2800: using guessed type int sub_2980(); 280A: using guessed type intfastcall sul 2014: using guessed type intfastcall sul 2014: using guessed type intfastcall sul	ensorEventQueue_getEvents(_DWORD, _DWORD, _DWORD); b_2250(_DWORD); b_3368(_DWORD); b_28DA(_DWORD); b_28DA(_DWORD);	
[+] dump end		
Python AU: idle Down Disk: 28GB		https://blog.osdim.ne/dg_2413/41101

- o ×



🕅 IDA - libnative.so C:\Users\admin\Desktop\XCTF\easy-dex\libnative.so 🔹 🕫 🖻 🗊 🕈 🎬 Library function 📕 Regular function 📕 Instruction 📗 Data 📕 Unexplored 📒 External symbol n Data - view-A I Pseudocoau While (v17); 📝 Functions window 🖪 Pseudocode-A 🛛 🖸 Hex View-1 🗵 A Structures E **N** . Enuns Imports Exports Practions window Function name __cxa_finalize __stack_chk_fail __stack_chk_fail __stalloc __aeabi_seaclr8 ASensorManager_getInstance ASensorManager_getDefaultSensor ASensorManager_createZentQueue __adroit_loc_print ALooper_pollAll Teneove ASensorEventQueue_getEvents time uncompress open free access free access Function name } if (v14 == 89) while (v18 < (signed int)v16)
v3[v18++] ^= 0x59u;</pre> } v10 = v14 + 1;} if (v14 == 100)if ((signed int)(time(0) - v6) > 9) { _android_log_print(4, "FindMyDex", "OH~ You are too slow. Please try again"); _aeabi_memcpy(v3, &unk_7004, off_43A18); v1a = 0; } else 224 • 225 • 226 • 227 • 228 • 229 • 230 • 231 • 232 • 233 • 234 { v20 = v6; if (uncompress(dest, &destLen, v3, (uLong)off_43A18)) __android_log_print(5, "FindMyDex", "Dangerous operation detected."); v21 = open((conjint(Bytef *dest, uLongf *destLen, const Bytef *source, uLong sourceLen)) if (!v21) __android_log_print(5, "FindMyDex", "Something wrong with the permission."); wrlte(v21, dest, destLen); close(v21); free(dest); free(v3); v20 = v6; 解zip数据操作 .p v tree(dest); free(v3); if (access((const char *)&name, 0) && mkdir((const char *)&name, 0xIFFu)) _android_log_print(5, "FindMyDex", "Something wrong with the permission.."); sub_2368(v1); remove((const char *)&filename); _android_log_print(4, "FindMyDex", "Congratulations!! You made it!"); sub_2250(v1); 235
236
237
238
230 □ # × Å Graph overview ф П 239240 0000283A android_main:205 (283A) Dutput vindov ZCI4: USING guessed type Int ___Yastcall Sub_ZCI4(_UWUKU); 43A18: using guessed type void *off_43A18; . . Python All: idle Down Disk: 27GB



IDA dump脚本:

import idaapi

addr = 0x7004 size = 0x3ca10

```
with open('dump','wb') as f:
    f.write(get_bytes(addr,size))
```

print('[+] dump end')

python解密dex脚本(我电脑上运行环境为3.6):

```
with open('dump', 'rb') as f:
    data1 = f.read()
    data = list(data1)
    count = 0
    while True:
        if count <= 0x59:
            count tmp = (int)(count / 10)
            if count % 10 == 9:
                size = 0x3ca10
                size_tmp = (int)(size / 10)
                xor = (count_tmp + 1) * size_tmp
                if (size_tmp * count_tmp) < xor:</pre>
                    index = size_tmp * count_tmp
                    while size tmp:
                        data[index] = data[index] ^ count
                        index = index + 1
                        size_tmp = size_tmp - 1
                if count == 89:
                    while xor < size:
                        data[xor] = data[xor] ^ 0x59
                        xor = xor + 1
        else:
            break
        count = count + 1
filebytes = bytes(data)
with open('easy-dex.dex','wb') as f1:
```

f1.write(zlib.decompress(filebytes))
print('[+] decrypt end')

4、将解密后的dex文件拖进jeb中反编译,首先看一下 MainActivity.java 文件的 onCreate 函数,发现有一个按钮监 听事件,触发后调用了 a.java 里面的 onClick 函数,那么去看一下 onclick 函数,发现调用了 MainActivity 里面的 a 函 数,并且传入了两个字符串参数,第一个字符串是输入框中的值,第二个参数是 string.xml 资源文件中的字符串,然后 与一个字节数组进行比较,结合题目反编译后的 public.xml和 string.xml文件,该字符串为 I have a male fish and a female fish.,并且资源ID为 two_fish,好吧,都已经是明示了,这里是 TwoFish 加密(关于TwoFish加密我会将一下我 觉得写得还行的博客链接贴在最后面),并且该字符串就是密钥,与之比较字节数组就是加密后的结果。那么首先将字 节数组转为字符串再说吧,一看字符串里面还有负数,那就直接与一下0xff咯(关于为啥要与0xff,是因为数字在java中 是以补码形式表示的,与0xff相当于将一个有符号数转为了无符号数,看来我确实没有写博客的天赋,感觉说得不明不白 的,老规矩,就把我觉得写得可以的博客链接贴在文末),然后与完后,直接转为ascii拼接成字符串,发现有些根本无 法显示出具体字符来。。。。。又卡了,突然发现里面有个 / 符号,一下就想到了 base64,尝试一下base4解码,结果 解出来又是啥都不是。。。。。好气哦这个题。。。。。那就在来一个base64加密吧,得到字符 串 iE3y2hEF1izgbVUfGKWQrUCtgFQFop7iEkbmRwWdwsZ1HdQGCPxRVAkWzV/eDC9N (好吧,我承认我有赌的成分。。。。),随便 找了个在线解密的网站,解密即得到flag。



JEB2 - C:\Users\admin\Desktop	p\XCTF\easy-dex\easy-dex.dex	-		×
文件 编辑 Navigation 行为 De	bugger 窗口 帮助			
🗁 🔚 🥕 🛆 🗂 🚽 🖓 🗘 🔿 💰	▶ /* ∥ ⊵ ⊕ ⊕ ⊕ 悚 ¥ 钰 梦 ⊪ □ 圖 丞 ۞ 止 ऱ!			
飸 Project Explorer 🛛 😫 🗖 🗖	🗣 easy-dex.dex/Hierarchy 🌵 easy-dex.dex/Disassembly 🖸 a/Source 🖾 🗵 xml <unbound> 🕼 apk<unbound></unbound></unbound>		~	- 8
Project Explorer S S C.Users\admin\Desktop\X easy-dex.dex easy-dex.dex easy-dex.dex	<pre>easy-dex/dex/Herarchy = easy-dex.dex/Disassembly] a/Source % M xml<unbound> P apk<unbound> package com.a.sample.findmydex; import android.content.Context; import android.view.View@OnClickListener; import android.widget.EditText; import android.widget.EditText; import android.widget.EditText; import android.widget.EditText arg2, Context arg3) { this.o = arg1; this.a = arg2; this.b = arg3; super(); public void onClick(View arg5) { if(Arrays.equals(MainActivity.a(this.a.getText().toString(), this.o.getString(2131099683)), MainAct Toast.makeText(this.b, this.o.getString(2131099682), 1).show(); } else { Toast.makeText(this.b, this.o.getString(2131099682), 1).show(); } } // Cost.makeText(this.b, this.o.getString(2131099682), 1).show(); } // Cost.makeText(this.b, this.o.getString(2131099682), 1).show(); //) </unbound></unbound></pre>	tivity	·.i())))	
< >>				\$
Filter: type "Enter" to validate	<			>
coord: (0,9,0) addr: Lcom/a/samp	le/findmydex/a; loc: ? https://blog.c			







python获取twofish加密结果脚本(运行环境同上):

```
import base64
```

i = [-120, 77, -14, -38, 17, 5, -42, 44, -32, 109, 85, 31, 24, -91, -112, -83, 64, -83, -128, 84, 5, -94, -98, -30, 18, 70, -26, 71, 5, -99, -62, -58, 117, 29, -44, 6, 112, -4, 81, 84, 9, 22, -51, 95, -34, 12, 47, 77]

data = []

```
for k in i:
    data.append(k&0xff)
print(base64.b64encode(bytes(data)))
```

题目:我是谁

一些总结

1、以往反编译后的第一步是看 MainActivity 文件,很少看 AnroidMainifes.xml,结果在很多地方吃了大亏,比如 easy-dex 这个题, android_main 这个函数还是撞进去的,后来看见那两个标签觉得没见过,才去百度了一下,才知道是 Native Activity,感觉这个文件看似不起眼,结果能少走一些不必要的地方!!!

2、思想上的牛角尖比技术上的牛角尖更难受,不然也不会去傻乎乎动态调试了一下午了!!!

```
一些博客链接
```

关于Native Activity的:

https://blog.csdn.net/qq_19683651/article/details/82623717 https://blog.csdn.net/qq_21071977/article/details/77878252

关于TwoFish加密的:

https://blog.csdn.net/I540538550/article/details/5642435

关于与0xff的:

https://blog.csdn.net/csdn_ds/article/details/79106006?depth_1-utm_source=distribute.pc_relevant.none-task&utm_source=distribute.pc_relevant.none-task