# CTF学习之路-攻防世界: MISC, 入门篇



 <u>**R**丝小帅的逆袭</u>
 ● 12020-11-26 11:24:17 发布
 ● 1592
 ◆ 收藏 25

 分类专栏:
 <u>**R**丝小帅的CTF学习过程</u> 文章标签:
 <u>安全 经验分享 恰饭</u>

 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/xuhc25/article/details/110167384</u>

 版权



屌丝小帅的CTF学习过程 专栏收录该内容

4 篇文章 0 订阅 订阅专栏

## 目录

- 一、感谢和安利
- 二、MISC-新手练习篇Writeup
- 1、this\_is\_flag
- 2、pdf
- 3、如来十三掌
- 4、give\_you\_flag
- 5、坚持60S
- 6、gif
- 7、掀桌子
- 8、stegano
- 9、SimpleRAR(比较麻烦)
- 10、base64stego
- 11、ext3
- 12、功夫再高也怕菜刀

## 一、感谢和安利

我是从攻防世界入门的,感谢攻防世界的免费靶场 靶场地址:攻防世界 https://adworld.xctf.org.cn/

二、MISC-新手练习篇Writeup

# 1、this\_is\_flag



### WP:

明显,入门用的,了解flag格式,直接提交题目例子的flag就可以了 flag{th1s\_!s\_a\_d4m0\_4la9}

## 2、pdf



WP:

题目给的一个PDF文件,使用PDF编辑器打开(我用的是万兴PDF编辑器)



移出来看到就是flag了



https://blog.csdn.net/xuhc25

3、如来十三掌





附件是一份文档:

夜哆悉諳多苦奢陀奢諦冥神哆盧穆皤三侄三即諸諳即冥迦冥隸數顛耶迦奢若吉怯陀諳怖奢智侄諸若奢數菩奢集遠俱 老竟寫明奢若梵等盧皤豆蒙密離怯婆皤礙他哆提哆多缽以南哆心曰姪罰蒙呐神。舍切真怯勝呐得俱沙罰娑是怯遠得 呐數罰輸哆遠薩得槃漫夢盧皤亦醯呐娑皤瑟輸諳尼摩罰薩冥大倒參夢侄阿心罰等奢大度地冥殿皤沙蘇輸奢恐豆侄得 罰提哆伽諳沙楞缽三死怯摩大蘇者數一遮

文档结尾 ■

一段经文,那不就是与佛论禅加密? http://www.keyfc.net/bbs/tools/tudoucode.aspx 记得加"佛曰"前缀





得到一串字符:

MzkuM3gvMUAwnzuvn3cgozMIMTuvqzAenJchMUAeqzWenzEmLJW9 看着这段字符,百思不得其解,最后题目:"如来十三掌",十三??试试rot13解码,得到一串字符串: ZmxhZ3tiZHNjamhia3ptbmZyZGhidmNraWpuZHNrdmJramRzYWJ9

	I CTF-	Tools V 1	.2.4 202005	19 By qiar	nxiao996					_		×
	Encode	Decode	e Encrypt	Decrypt	Binary	Website	Plug	gins	Oth	iers		
-								Repla	ace	Clear	Сору	Paste
行	MzkuM3g	vMUAwnzuv	vn3cgozM1MTu	ıvqzAenJchl	MUAeqzWer	nzEmLJW9						
位刀运												
a												
I												
I												
l												
I												
a								Renla	ace	Clear	Conv	Paste
I	7mmh72+	;7UN jomb	i o 2 n + h m 7 1 7 C k	i dm\mo\wow	7UNindim Tino	mPaVWT0		Nepr		oroar	00pj	14500
I	ZIIIXIIZƏt	ıznıyanın.	rasptbilizyzer	namvrawpu.	2UNLOUDLS	umvz i w j 9						
I												
										https://	blog.csdn.i	net/xuhc25

最后丢到解密工具里一键解码,得到flag: flag{bdscjhbkzmnfrdhbvckijndskvbkjdsab}

── [随波逐流]CTF	偏码工具 V1.0 2020	1022					—		$\times$
Base加解密 字符	加解密 字符编码转换	已知key解密	进制转换	其他工具	赞赏作者				
需要解密的文本	、↓ 密钥(ke	y) :			字数统计	一键解码	粘贴剪切板	清空内	容
ZmxhZ3tiZHNjamh	ia3ptbmZyZGhidmNra	¥puZHNrdmJram	RzYWJ9						
解密结果↓					复	制内容	↑解密结果转	至文本机	Ξ↑
— 键 解 码: base64解码: base32解码:	结  果 flag{bdscjhbkzmnf;	rdhbvckijndsk	vbkjdsab}				https://blog.cs/	dn.net/xuhc/	25

4 give\_you\_flag



WP: 附件是个GIF图片,看到闪的飞快,最后有一帧黑乎乎的,那就是可疑的地方了,思路肯定是工具逐帧分解GIF

可以用"GIF动画帧提取器",或者PS分解出来。 PS打开:找到可疑图层,打开图层显示,就可以看到可疑的那一帧。



#### GIF动画帧提取器打开,找到可疑帧



小恐龙手上拿着的是二维码,但是少了三个定位点,要PS回去。



扫描二维码得到flag



flag{e7d478cf6b915f50ab1277f78502a2c5}

# 5、坚持60S

World of Attack&Defense. 答题 竞赛 排行榜 ~	队伍
坚持60s 12 最佳Writeup由不要让我起名提供	
难度系数: ◆★★★★ 4.0	
题目来源: 08067CTF	
题目描述:菜狗发现最近菜猫不爱理他,反而迷上了菜鸡	
题目场景:暂无	
题目附件: 附件1	
https://blog.csdn.	net/xuhc25

下载下来是一个jar包



运行起来是一个游戏,没啥有用的信息





## 使用gd-gui工具反编译看看 查找关键字flag

PlaneGameFrame.class - Java Decompiler File Edit Navigation Search Help

(금) (철 🔗   🗇 🔶						
9dc125bf1b84478cb14813d9bed6470c. jar⊠						
<pre>     # META-INF     MANIFEST.MF     c.bjsxt     fb plane     fb plane     fb plane     fb plane.class     fb GameObject.class     fb plane.class     fb plane.class     fb of GameUtil     fb GameUtil     fb GameUtil     fb GameUtil     fb GameUtil     fb fb</pre>	MANIFEST. MF ☆ 38 boolean 39 if (peng 40 this.g 41 if (th 42 this 45 this.b } 49 if (!this.j 51 grintInfn 53 int perint 54 grintInfn 56 switch (j 58 grint breat 61 grint 61 grint 56 case 2 64 grint 57 case 3 58 grint 59 case 3 59 case 3 50 case 3 50 case 3 50 case 3 50 case 3 51 grint 54 grint 55 case 3 56 case 3 57 case 3 58 case 3 59 case 3 59 case 3 50 case 3 50 case 3 50 case 3 50 case 3 51 grint 54 grint 55 case 3 56 case 3 57 case 3 58 case 3 59 case 3 59 case 3 50 case 3 50 case 3 50 case 3 51 grint 51 grint 53 case 4 59 case 3 50 case 3 51 grint 51 grint 54 case 3 57 case 3 58 case 3 59 case 3 59 case 3 59 case 3 50 case 4 50 case 5 50 case 5	Bullet.class     b.getRect().intersects     (     is.bag = new Date();     bag = new Date();     bag = new Explode(this.p.x.     bag = new Explode(this.p.x.     flag     class     Search string (*     flag     class     flag     class     flag     class     iflag     is     info(s     iflag     iflag     iflag     class     iflag     class     field ✓ Met     info(s     iflag     field ✓ Met     class     iflag     class     field ✓ Met     class     clas     class     class     clas     class     cla	plode. class∺ (this.p.getRect() this.p.y): * = any string structor	GameObject.c Game	lass <sup>⊠</sup> Plane. class Acter): Limit To ✓ Declarations ✓ References	DelaneGameFra



看到还真能找到

#### 🝵 PlaneGameFrame.class - Java Decompiler

File Edit Navigation Search Help

(승) 🖉 🖉 🗢 🔶	
9dc125bf1b84478cb14813d9bed6470c. jar⊠	
• ● MATA-INF         • ● MANIFEST. MF         • ● MANIFEST. MF         • ● Dane         • ● DaneCtals         • ● DaneCass         • ● DaneCameFrame.class         • ● GameUtil         • ● MAN         • ● Mane(String) : Image         • ● Images         • ●         • ●         • ●	<pre>ITFEST. MF ※</pre>

flag{RGFqaURhbGlfSmlud2FuQ2hpamk=} 拿去提交,发现是不对的

🛩 [随波逐流	]CTF编码工具	V1.0 2020	1022		
Base加解密	字符加解密	字符编码转换	已知key解密	进制转换	其他工
需要解密的	〕文本↓	密钥(ke	ey):		

RGFqaURhbG1fSmlud2FuQ2hpamk=



## 6、gif

flag{DajiDali\_JinwanChiji}





下载下来,发现是个压缩包,打开看看,发现都是黑、白两色块。脑袋里啪的想到:白:0,黑:1,为啥不是白:1,黑:0??做久了开发就知道,0是白黑是1,1都是反派的那个。



一番比对后得到一串数字:

然后发现总数是104个,这要么是2、4、8一组,计算机中以字节为单位存储和解释信息,规定一个字节由八个二进制位构成,那 么8个一组没跑了。

8个一组得到:

01100110 01101100 01100001 01100111 01111011 01000110 01110101 01001110 01011111 01100111 01101001 01000110 01111101

二进制转字符得到flag(用的随波逐流提供的解密工具): flag{FuN\_giF}

 -  $\Box$   $\times$ 

Base加解密	字符加解密	字符编码转换	已知key解密	进制转换	其他工具	赞赏作者			
需要解密的文本↓		密钥(ke	y):		_	字数统计	一键解码	粘贴剪切板	清空内容

01100110 01101100 01100001 01100111 01111011 01000110 01110101 01001110 01011111 01100111 01101001 01000110 011 11101

解密结果↓	复制内容	↑解密结果转至文本框↑
flag{FuN_giF}		

https://blog.csdn.net/xuhc2

\*\*

7、掀桌子

World of Attack& Defense	<ul> <li>※</li> <li>答题</li> <li>売赛</li> </ul>	<b>排行榜</b> ~	队伍	商城	×
	天2时22分50秒				
F 掀桌子 105 最佳Writeup由flag	<b>{not_here} ●</b> 渣渣禹提供			🔲 WP	● 建议
难度系数: •★★★★ 4.0 题目来源: DDCTF2018					
<b>题目描述:</b> 菜狗截获了一份报文如下c8e9aca 5e7e9f3e4e3e8eaf9eaf3e2e4e6f2,生气地掀翻	ı0c6f2e5f3e8c4efe7a1a0d4e8e 翻了桌子(╯°□°) ╯⌒ ┺━┺	5a0e6ece1e7a0e9	f3baa0e8eafae3f§	9e4eafae2eae4e3ea	ebfaebe3f
题目场景: 暂无 题目附件: 暂无					
				https://blog.cs	sdn.net/xuhc25

这题没啥附件,只有一串字符,看字符是0-9, a-f组成的,就知道这是16进制字符串咯,那么2个16进制可以代表一个字符串,就用python进行转换下。



发现结果字符块都大于128,ASCII码范围是0-127

											٦	$\checkmark$	X
•	string = flag = '' for i in s = ' i = i print	<pre>"c8e9ad range(@ 0x'+str int(s,1@ c(str(i))</pre>	ca0c6f2e5f3e8 0,len(string) ring[i]+strin 5) ))	c4efe7a1a0d ,2): g[i+1]	4e8e5a0e6ec	e1e7a0e9f3baa0	e8eafae3	f9e4eafae	2eae4e3eaebfa	ebe3f5e7e9f	3e4e3e8	eaf9eat	fi ,
<													>
Search	n in Files	Search	Stack Data	Debug I/O	Exceptions	Debug Console	Watch	Modules	Python Shell	Bookmarks	Breakpo	ints 🖣	T
<u>ش</u> [p:	id 11068] 攻	P - De	ebug I/O (stdin,	stdout, stde	rr) appears be	low						<u>O</u> ptions	s •
200 233 172 160 198 242 229 243 232 196 239 231 160 212 232 229													
100													c25

<u>r</u> iie	<u>E</u> an	Source	<u>к</u> егаси	i Fiole	ect <u>D</u> e	bug	resung		window	<u>н</u> ер									
			8	4 🗖	•	¢	III <sup>I</sup> :				۹ 🏯	≣→	Ξ←	₽⇔	ę		•		ł
Help	攻防	世界脚本.	ру																
																	۶	•	>
	stri flag for :	ng = "c8 = '' i in ran s = '0x' i = int( print(st	e9aca0c6 ge(0,len +string[ s,16)-12 r(i))	f2e5f3e8 (string) i]+strir 8	3c4efe7a (,2): ng[i+1]	a1a0d4	<b>¦e8e5a0e6</b>	ece1e7a	0e9f3baa0	e8eafae	3f9e4eafa	e2eae4	e3eael	bfaeb	e3f5e7	e9f3e4	4e3e8e	af9eaf	Fg
<																			>
Searc	n in Fi	les Sea	arch St	ack Data	Debug	I/0	Exception	s Debi	ıg Console	Watch	Modules	Pyth	on Shel	11 E	Bookmark	s B	reakpoi	nts ٵ	T
₩ [p	id 9492	]政防 •	Debug I	/O (stdin,	stdout,	stder	r) appears	below									-	<u>O</u> ptions	;
105 44 32 70 114 101 115 104 68 111 103 33 32 84 104 101																			
32 100																			

## 这就对了。跟着感觉走没毛病,把进制转换成字符。

												۶	$\checkmark$	×
	<pre>string = "c8e9aca0c6f2e5f3e8c4efe7a1a0d4e8e5a0e6ece1e7a0e9f3baa0e8eafae3f9e4eafae2eae4e3eaebfaebe3f5e7e9f3e4e3e8eaf9e flag = '' for i in range(0,len(string),2):     s = '0x'+string[i]+string[i+1]     i = int(s,16)-128     flag+=chr(i)#chr是转换, flag+=的意思是拼接字符</pre>													fi 🔨
	-	print(flag	g)											~
<														>
Se	earch	in Files	Search	Stack Data	Debug I/O	Exceptions	Debug Console	Watch	Modules	Python Shell	Bookmarks	Breakp	oints 🖣	<b>h</b> -
ار ۲	∦ [pi Hi, ∣	id 3684] 攻防 FreshDog! T	- Deb The flag	ug process ter g is: hjzcyd	minated jzbjdcjkzkc	ugisdchjyjs	bdfr					×	<u>O</u> ption:	s •

nups.//biog.csun.nev/uni



转换后得到以下结果,可以得到flag咯: Hi, FreshDog! The flag is: hjzcydjzbjdcjkzkcugisdchjyjsbdfr

## 8、stegano



下载得到附件

Lorem insum dolor sit amet, consectetur adipiscing elit. Cras faucibus odio ut metus vulnutate, id laoreet m	agna
volutpat. Integer nec enim vel arcu portitior egestas. Vestibulum suscipit lorem sed sem faucibus rutrum. Nunc	diam
orci, convallis vitae auctor vehicula, interdum ut mi. Maecenas nec urna at dolor mattis dictum sit amet at	orci.
Mauris condimentum adipiscing erat nec feugiat. Curabitur scelerisque varius ligula, iaculis adipiscing dui. Duis	eget
ullamcorper arcu. In facilisis et tortor commodo aliquam. Nulla feugiat, sem eu molestie bibendum, leo nisi port massa id accumsan sapien libero id tellus. In enim lacus, sollicitudin a felis quis blandit porta insum. Dones ed	nibh
eresta, tristique mauris eu rutrum insto. Nulla facilisi. Duis gravida semper dui laoreet vulputate. Aenean quis te	nibor
orci. Cras placerat lectus nulla, eu bibendum metus interdum in.Lorem ipsum dolor sit amet, consectetur adipi	scing
elit. Cras faucibus odio ut metus vulputate, id laoreet magna volutpat. Integer nec enim vel arcu porttitor ege	stas.
Vestibulum suscipit lorem sed sem faucibus rutrum. Nunc diam orci, convallis vitae auctor vehicula, interdum ul Maccanae neo una at dolor mattie distum eit amet at orei. Maurie condimentum adicitatione act auc fouriet. Consi	t mi.
scelerisque varius ligula, iaculis adipiscing dui. Duis eget ullamcorper arcu. In facilisis et tortor commodo alici	uam.
Nulla feugiat, sem eu molestie bibendum, leo nisi porttitor massa, id accumsan sapien libero id tellus. In enim	acus,
sollicitudin a felis quis, blandit porta ipsum. Donec sed nibh egestas, tristique mauris eu, rutrum justo. Nulla fac	cilisi.
Duis gravida semper dui laoreet vulputate. Aenean quis tempor orci. Cras placerat lectus nulla, eu bibendum n istardum in Lorem incum delor sit amet, conceptatur adiniscing alit. Cras fausibus adia ut matur unbrutata id la	aetus
magna volutpat. Integer nec enim vel arcu porttitor egestas. Vestibulum suscipit lorem sed sem fancibus rutrum.	Nunc
diam orci, convallis vitae auctor vehicula, interdum ut mi. Maecenas nec urna at dolor mattis dictum sit amet at	orci.
Mauris condimentum adipiscing erat nec feugiat. Curabitur scelerisque varius ligula, iaculis adipiscing dui. Duis	eget
ullamcorper arcu. In facilisis et tortor commodo aliquam. Nulla feugiat, sem eu molestie bibendum, leo nisi port	nibh
egestas, tristique mauris eu, rutrum justo. Nulla facilisi. Duis gravida semper dui laoreet vulputate. Aenean quis ter	mpor
orci. Cras placerat lectus nulla, eu bibendum metus interdum in.Lorem ipsum dolor sit amet, consectetur adipie	scing
elit. Cras faucibus odio ut metus vulputate, id laoreet magna volutpat. Integer nec enim vel arcu porttitor ege	stas.
Vestibulum suscipit forem sed sem faucibus rutrum. Nunc diam orci, convalits vitae auctor vehicula, interdum ul Maecenae neg urna at dolor mattic dictum sit amot at orci. Mauris condimentum adiniscing orat neg faugiat. Cura	t mi.
scelerisque varius ligula, iaculis adipiscing dui. Duis eget ullamcorper arcu. In facilisis et tortor commodo aliq	uam
Your flag is not here jolestie bibendum, leo nisi porttitor massa, id accumsan sapien libero id tellus. In enim le	acus,
sollicitudin a felis quis, blandit porta ipsum. Donec sed nibh egestas, tristique mauris eu, rutrum justo. Nulla fac	cilisi.
Dus gravida semper dui faoreet vulputate. Aenean quis tempor orci. Uras piacerat lectus nulla, eu bibendum n interdum in Lorem insum dolor sit amet, consectetur adipiscing elit. Cras faucibus odio ut metus vulputate, id la	oreet
magna volutpat. Integer nec enim vel arcu porttitor egestas. Vestibulum suscipit lorem sed sem faucibus rutrum. !	Nunc
diam orci, convallis vitae auctor vehicula, interdum ut mi. Maecenas nec urna at dolor mattis dictum sit amet at	orci.
Mauris condimentum adipiscing erat nec feugiat. Curabitur scelerisque varius ligula, iaculis adipiscing dui. Duis	eget
mancorper arcu. In facilities et fortor commodo anquam. Auna leugiat, sem eu molestie bibendum, leo nisi port massa, id accumsan sapien libero id tellus. In enim lacus, sollicitudin a felis quis, blandit porta insum. Donec sed	nibh
egestas, tristique mauris eu, rutrum justo. Nulla facilisi. Duis gravida semper dui laoreet vulputate. Aenean quis ter	mpor
orci. Cras placerat lectus nulla, eu bibendum metus interdum in.Lorem ipsum dolor sit amet, consectetur adipi	scing
elit. Cras faucibus odio ut metus vulputate, id laoreet magna volutpat. Integer nec enim vel arcu portitor ege Vartibulum sussinit loram and som fausibus rutrum. Numa diam arci somullis vitas austas unlimba interdum u	stas.
Maecenas nec urna at dolor mattis dictum sit amet at orci. Mauris condimentum adipiscing erat nec feugiat. Cura	bitur
scelerisque varius ligula, iaculis adipiscing dui. Duis eget ullamcorper arcu. In facilisis et tortor commodo aliquam. ?	Nulla
feugiat, sem eu molestie bibendum, leo nisi porttitor massa, id accumsan sapien libero id tellus. In enim lacus, sollicit	tudin



a felis quis, blandit porta ipsum. Donec sed nibh egestas, tristique mauris eu, rutrum justo. Nulla facilisi. Duis gravida semper dui laoreet vulputate. Aenean quis tempor orci. Cras placerat lectus nulla, eu bibendum metus interdum in.Cras placerat lectus nulla, eu bibendum metus interdum in.Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras faucibus odio ut metus vulputate, id laoreet magna volutpat. Integer nec enim vel arcu porttitor egestas. Vestibulum suscipit lorem sed sem faucibus rutrum. Nunc diam orci, convallis vitae auctor vehicula, interdum ut mi. Maecenas nec urna at dolor mattis dictum sit amet at orci. Mauris condimentum adipiscing erat nec feugiat. Curabitur scelerisque varius ligula, iaculis adipiscing dui. Duis eget ullamcorper arcu. In facilisis et tortor commodo aliquam. Nulla feugiat, sem eu molestie bibendum, leo nisi portitor massa, id accumsan sapien libero id tellus. In enim lacus, solfactudin a felis ouis blandit porta ipsum. Donec sed nibh egestas tristique mauris eu rutrum insto. Nulla facilisi. Duis eravida semper



#### PDF编辑器,删掉挡在前面的那对英文,看到还剩下几个文本框。



复制文本出来,贴在记事本上

stoma	ilias	es	8	iew 3																									
XXX	xxx	XXXX	xxxx	XXX																									
XXX	xxx	XXXX	XXXX	XXXXX	XXXXX	xxxx	xxxx	XXXXX	xxxxx	XXXXX	XX	XXX	(XXXX)	XXXX	XXXXX	xxxxxxx	XXXXXXX	xxxxxx											
Cl																													
b																													
t																													
til	1																												
th																													
1.1																													
ose	÷ -																												
u																													
s																													
no																													
ere	2																												
BAB	BA	BBB	BA	BBA	ABA	AB	в	AAB	ABAA	AB	в	AA	BBB	BA	AAA	BBAABB	AABA	ABAA	AB	BBA	BBBAAA	ABBBB	BA	AAAB	ABBBB	AAAAA	ABBBB	BAAA	AI
																													mba25
																													unsza

#### 很明显, ABAB那串就是密文了。复制出来

BABA	BBB	BA	BBA	ABA	AB	В	AAB	ABAA	AB	В	AA	BBB	ΒA	AAA	BBAABB	AABA	ABAA	AB	BBA	BBBAAA	ABBBB
BA	АААВ	ABBBB	B AA	AAA	ABBB	В	BAAA	ABAA	AAA	BB	BB	AAABI	B A	AAAA	ΑΑΑΑΑ	AAAAB	BBA	AAABI	В		

试下转换为摩斯密码,记事本替换下A和B,A替换为.,B替换为-。 到如下结果。

	. 🔍	<b>N</b>	<b>X</b> /	<b>X</b>   «	V> 💙	V	'⊠	Δ,	🌂	¥ 💔	-	• _ •	•  TT	\$≻ `	5	Ľ
ust	malias	ses	new	3 📙 п	iew 4											
1														 	 	et/xuhc25

放到解密工具上解密

✓ [随波逐流]CTF编码工具 V	1.0 20201022				_		$\times$
Base加解密 字符加解密 号	字符编码转换 已知key	露 进制转换 其他工	具 赞赏作者				
需要解密的文本↓	密钥(key):		字数统计 -	- 键解码     粘则	占剪切板	清空内	容
	::: :-::	· ···		· ·-·· ·			-

解密结果↓	复制内容	↑解密结果转至文本框↑
CONGRATULATIONS, FLAG: 1 N V 1 5 1 B L 3 M 3 5 5 4	G 3	,

去掉空格,得到一串字符 CONGRATULATIONS,FLAG:1NV151BL3M3554G3 题目说flag是小写,转换为小写,得到flag。 congratulations,flag:1nv151bl3m3554g3 那么flag就是flag{1nv151bl3m3554g3}

Fift 💻

# 9、SimpleRAR(比较麻烦)

N	orld of Attack&Defense			元女	54r L FIFI	МШ	
<del>()</del> 返回		本题用时:	13时0分4秒				
Simple	RAR 🖬 🖆 62	最佳Writeu	p由它山提供				
难度系数:	• <b>*</b> ***5.	0					
题目来源:	08067CTF						
题目描述:	菜狗最近学会了拼图	图,这是他刚	拼好的, 可是却	搞错了一块(ps:X	双图层)		
题目场景:	暂无						
题目附件:	附件1						
						https://blog.csdn.net	/xuhc25

#### 下载是1个压缩包,360解压

	🚦 18с5326аас	a0499eafbe	) 360压缩			文件	操作	工具	帮助 1	3 🗆	_ [	3 23
	添加	<b>留</b> 解压到	一键解压	副除	四月						安	·全
	<b>☆ 🔲 🗄</b>	<b>4</b> 18c5326	Saada0499eafb	e03ad8a52	e40c.rar - 解包,	大小为 1 KB		Ŧ	搜索包内文	件		Q
2	3称 (上级目录)				j	压缩前	玉缩后	类型 文件夹	ſ	多改日期		
	flag.txt					1 KB	1 KB	文本文档	2	017-10-1	14 23:27	,
1	🗐 flag.tx	t - 记事本								-	_	
ſ	文件(F) 编	辑(E)格式(C	D) 查看(V) 帮	勂(H)								
	flag is n	ot here										
t												

当然是用rar啦,360压缩不会提示错误的,winrar才会。



📕	🔺 WinRAR:诊断信	息					_		×
	信息 <u> 小</u> 文件头已损坏:s	ecret.png	压缩文件 e03ad8a52e4	40c.rar (C:\Users\XUH	C\Downloads	\18c5326aada0499	eafbe03ad8	3a52e40c.ı	ar)
	<								>
<b>-</b>		关闭 <b>(C)</b>		中断操作(B)		夏制到剪贴板 <b>(L)</b>			
	错误总数:1								.csdn.ne <mark>/xu</mark> ł

文件头损坏,那就用winhex打开修复。 首先了解下rar块头的知识:

第二块为压缩文件头(MAIN HEAD),和标记块一样

# 压缩文件头块

000
xt□□
ot her
3 🗆 🗖
g□□

这里的头类型是0x73表示压缩文件头块,位标记为0x0000 没有 位 被置为1,如果块头被加密,则位标记应该为: 0x8000,文件头大小为0x0D00,由上图可以看出这个压缩文件头块占13个字节。

在这里怎么定位坏的块呢? 1、定位文件位置 rar提示secret.png文件头已损坏,winhex先定位到文件所在。 通过搜索文本,可以定位到文件所在了。

UEV -

WinHex - [18c5326aada0499eafbe03ad8a52e40c.rar]

🚟 又件(ト	) 编辑(E) 弢家 <b>声</b> ) 导航(N)	L具(I) 选坝(O)
帮助(H)		20.0 SR-2 x6
	3 🏟 🗊 📑 👘 🐚 🕲 🔂 🐚 🐎 丨 👬 🦀 🎄 🐇 🗎 → ⊣	🗑 🗲 🔿 🛛 🕹 🔹 🛛 secret.png 🗸 🗸
Offset	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	ANSI ASCII
00000048	20 00 00 00 66 6C 61 67 2E 74 78 74 00 B0 57 00	flag.txt °W
00000064	43 66 6C 61 67 20 69 73 20 6E 6F 74 20 68 65 72 Cfla	ag is not her
00000080	65 A8 3C 74 20 90 2F 00 3A 15 00 00 42 16 00 00 e"<1	t / : B 🖌 🛛 ASCII/Code page 🗸
00000096	02 BC E9 8C 2F 6E 84 4F 4B 1D 33 0A 00 20 00 00 460	Œ/n"OK 3
00000112	00 73 65 63 72 65 74 2E 70 6E 67 00 F0 40 AB 18 sec	cret.png ð@«      用作通配符(U):    ?
00000128	11 C1 11 55 08 D1 55 80 0D 99 C4 90 87 93 22 19 Å T	U ŇU€ ™Ă ≠ * * " ///////////////////////////////
00000144	4C 58 DA 18 B1 A4 58 16 33 83 08 F4 3A 18 42 0B LXU	±¤X 3f ô: B
00000160	04 05 85 96 21 AB 1A 43 08 66 EC 61 0F A0 10 21	-!« C fia !
00000176	AB 3D 02 80 B0 10 90 C5 8D A1 1E 84 42 B0 43 29 «= €	€° A ; "B°C)
00000192	08 10 DA OF 23 99 CC F3 9D C4 85 86 67 73 39 DE 0	#*10 Atgs9₽ 搜索(e): 全部 ∨
00000208	17 73 FE CD CF 77 3D OD FE 77 00 70 70 OF DE 3F 80 3	Ínu zale ő
00000224	AA AS ST CD OE // SD OD ET /A 99 A9 A9 OF DS ST -E	Inw, 12mee 0, 二 n ěentetyto
00000240	57 34 7B 8B 8F F9 C0 F7 F6 30 F3 25 60 55 58 00 W4/	
00000272	9A CC E6 CD CB ED 19 24 43 83 30 46 D6 97 30 0C str	ffý ścfořč=0
00000288	ED 2D 4D 8D E8 E6 3F 1A FB 23 10 0D 8D 1F A8 5F 1-M	
00000304	41 55 3D 55 70 4C 69 6B 6C 50 78 71 69 5B 78 56 AU=0	
00000320	5C 08 F0 DA 11 11 A0 C5 25 20 02 30 80 62 03 38 \ 80	ć Å% 0€b 8
00000336	06 FB D5 98 07 E8 6E 6F 72 FD 6F DD EC CD 01 F9 ûÕ	~ ènorýoÝìÍ ù
00000352	02 07 CB 9F F7 DE 3C E4 0F F8 4E DC DB 7E D0 95	Ÿ÷₽<ä ⊘NÜÛ~Đ•
页1/18	偏移地址: 113 = 115 选块:	113 - 122 大/ MA 确定(Q) 取消(A) 《帮助(H)

题目用到的是这两个知识点

# 文件头块

在网上找了图片资源,然后我再举个例子,由于现在rar版本不断更新,图片资源有点老了,部分内容现在可能遇不到。 HEAD\_CRC 2 字节 从 HEAD\_TYPE 到 FILEATTR 的 CRC 结构和文件

名

HEAD_TYPE	1 字节	头类型: 0x74	
-----------	------	-----------	--

HEAD\_FLAGS 2 字节 位标记:

- 0x01 文件在前一卷中继续
- 0x02-文件在后一卷中继续
- 0x04 文件使用密码加密
- 0x08 文件注释存在

RAR 3.x 使用分开的注释块,不设置这个标记。

- 0x10 前一文件信息被使用(固实标记)
  - (对于 RAR 2.0 和以后版本)
  - 765 位(对于 RAR 2.0 和以后版本) https://http://https://https://https//https://https://https://https://https://https//https//https://https://https://https://https/

附上常见的块类型(HEAD\_TYPE)如下:

记块: HEAD_TYPE=0x72
缩文件头:HEAD_TYPE=0x73
件头: HEAD_TYPE=0x74
风格的注释头:HEAD_TYPE=0x75
风格的用户身份信息:HEAD_TYPE=0x76
风格的子块: HEAD_TYPE=0x77
风格的恢复记录: HEAD_TYPE=0X78
风格的用户身份信息:HEAD_TYPE=0X79
块:HEAD_TYPE=0x7A
后的结束块: HEAD_TYPE=0x7B

### 2、找到块头

上一个文件结束,是 flag is not here,就是那个迷惑的txt文件内容结尾啦。那么跟着的就是下个文件,图片文件的开头。看下 图,从上面知识,知道第三个字节开始,就是文件块头咯,那么我们要改的就是这个字节。提示文件头损坏,那就给他改成对应 的 文件头: HEAD\_TYPE=0x74

文件礼体	🎬 文件(F) 编辑(E) 搜索(S) 导航(N) 查看(V) 工具(T) 专业工具(I) 选项(O) 窗口	(W)
	帮助(H) 20.0 SR-2 x64 -	ø
在网上找了图片资源,然后我再举个例子,由于现在rar版本不断更新,图片资源有点老了,部分内容现在可能遗不 一	□ 7	ا 🔍
	Offset 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 ANSI ASCII	
HEAD_CRC 2 字节 从 HEAD_TYPE 到 FILEATTR 的 CRC 结构和文件	00000000 52 61 72 21 1A 07 00 CF 90 73 00 00 0D 00 00 00 Rar! Î s	
8	00000016 00 00 00 00 D5 56 74 20 90 2D 00 10 00 00 10 OVt -	
	00000048 20 00 00 02 C7 00 07 50 05 55 55 42 45 15 50 00 00 C C C C C C C C C C C C C C C	
HEAD_IYPE 1 子中 头尖型: UX/4 =	00000063 13 66 6C 61 67 20 69 73 20 6E 6F 74 20 68 65 72 Cflag is not her	
HEAD_FLAGS 2 字节 位标记:	00000080 65 A8 3C 74 20 90 2F 00 3A 15 00 00 42 16 00 00 e < / / : B	
0x01 - 文件在前一卷中继续	000000006 02 BC E9 08 2F 6E 84 4F 4B 1D 33 0A 00 20 00 00 460/n,,CK 3	
0.00 + 4+ + F + + + + + + + + + + + + + + +	00000112 00 70 65 63 72 65 74 2E 70 6E 67 00 F0 40 AB 18 secret.png 80«	
UXU2 · 义针在后一卷中继续	00000128 11 C1 11 55 08 D1 55 80 0D 99 C4 90 87 93 22 19 A U NUE "A #""	
0x04 - 文件使用密码加密	00000111 10 50 DA 10 DI A1 50 10 53 03 00 FT 5A 10 12 0D LAU TA 53 0; D	
0x08 - 文件注释存在	00000176 AB 3D 02 80 B0 10 90 C5 8D A1 1E 84 42 B0 43 29 «= €° Å ; "B°C)	
	00000192 08 10 DA OF 23 99 CC F3 9D C4 85 86 67 73 39 DE Ú #™Ìó Ä…†gs9Þ	
RAR 3.x 使用分开的注释块,不设置这个标记。	00000208 47 63 91 DE C4 77 ED A8 DC 46 F4 C5 54 CD 55 6A Gc BÄwi "ÜFÔÅTÍUj	
0x10 - 前一文件信息被使用(固实标记)	00000224 AA A3 5F CD 6E 77 3B 8D EF 7A 99 A9 A9 8F D5 3F <sup>*</sup> £_ĺnw; ïz™©© Õ?	
(对于 RAR 20 和以后版本)	00000240 0A AA F9 55 7F 02 9E A2 9C 86 88 CC 59 CC FF 0C <sup>a</sup> ùU žoαt↑ÍYÍŸ	
(1) 1 100 20 100/10001	00000256 57 34 78 88 87 F9 C0 F7 E6 30 E3 25 60 55 58 00 W4{ (uA+æ0a* UX	
765 位(对于 RAR 2.0 和以后版本) https://kea.uetogrobi/mmetal/hc25	000002/2 FR CC LC CD CD FD 19 21 13 03 30 16 D6 97 30 0C SIGILY \$CJURC-0	
[在这里插入图片描述](https://img-	00000304 41 55 3D 55 70 4C 69 6B 6C 50 78 71 69 5B 78 56 AU=DpliklPxg1 [xv	

改完后保存,如下图

🚟 WinH	ex - [18c53	26aada0499	eafbe03ad8a52e40c .ra	ar] — 🗆 🗙
🎏 文件(F	·) 编辑(E) [	搜索(S) 导航	ī(N) 查看(V) 工具(T) €	₹业工具(I)选项(O)窗口(W)
帮助(H)				20.0 SR-2 x64 – 🖉 🛪
🗅 7 🖬 🍕	) 🗟 😭 📑	🔊 🖣 🖪 🖪	🖻 1012 🛛 👬 👫 👫 🖧 🙀	→ + + + + + + - 2 3 3 2 = 2 4 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 0 0 0 0
Offset	0 1 2 3	4 5 6 7	8 9 10 11 12 13 14 15	ANSI ASCII
00000000	52 61 72 21	1A 07 00 CF	90 73 00 00 0D 00 00 00	Rar! Ïs
00000016	00 00 00 00	D5 56 74 20	90 2D 00 10 00 00 00 10	ÕVt −
00000032	00 00 00 02	C7 88 67 36	6D BB 4E 4B 1D 30 08 00	Ç^g6m»NK 0
00000048	20 00 00 00	66 6C 61 67	2E 74 78 74 00 B0 57 00	flag.txt °W
00000064	43 66 6C 61	67 20 69 73	20 6E 6F 74 20 68 65 72	Cflag is not her
00000080	65 A8 3C 74	20 90 2F 00	3A 15 00 00 42 16 00 00	e" <t :="" b<="" th=""></t>
00000096	02 BC E9 8C	2F 6E 84 4F	4B 1D 33 0A 00 20 00 00	ιé∉/n"OK 3
00000112	00 73 65 63	72 65 74 2E	70 6E 67 00 F0 40 AB 18	secret.png ð@«
00000128	11 C1 11 55	08 D1 55 80	OD 99 C4 90 87 93 22 19	Á U ÑU€ ™Ä ‡""
00000144	4C 58 DA 18	B1 A4 58 16	33 83 08 F4 3A 18 42 0B	LXÚ ±¤X 3f ô: B
00000160	04 05 85 96	21 AB 1A 43	08 66 EC 61 0F A0 10 21	!« C fìa !
00000176	AB 3D 02 80	B0 10 90 C5	8D Al 1E 84 42 B0 43 29	«=€° Å ; "B°C)
00000192	08 10 DA OF	23 99 CC F3	9D C4 85 86 67 73 39 DE	Ú #™Ìó Ä…tgs9Þ
00000208	47 63 91 DE	C4 77 ED A8	DC 46 F4 C5 54 CD 55 6A	Gc ÞÄwí ÜFôÅTÍUj
00000224	AA A3 5F CD	6E 77 3B 8D	EF 7A 99 A9 A9 8F D5 3F	ª£_ĺnw; ïz™©© Õ?
00000240	0A AA F9 55	7F 02 9E A2	9C 86 88 CC 59 CC FF 0C	≞ùU ž¢œ†^ÌYÌÿ
00000256	57 34 7B 8B	8F F9 C0 F7	E6 30 E3 25 60 55 58 00	W4{< ùÀ÷æ0ã%`UX
00000272	9A CC E6 CD	CB FD 19 24	43 83 30 46 D6 97 30 OC	šÌæÍËý \$Cf0FC−0
00000288	ED 2D 4D 8D	E8 E6 3F 1A	FB 23 10 0D 8D 1F A8 5F	í-M èæ? û# "_
00000304	41 55 3D 55	70 4C 69 6B	6C 50 78 71 69 5B 78 56	AU=UpLiklFxqi[xV 💙
页1/18		偏移地址:	83 = 116 选块:	「「「PS://D 无」大小: BI// 「DO25天



打开图片,发现空白的,所以再winhex打开,发现是gif的图

💹 winhex常见的文件头.txt - 记事本	WinHex - [secret.png] —	
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)	🎇 文件(F) 编辑(E) 搜索(S) 导航(N) 查看(V) 工具(T) 专业工具(I) 选项(	0) 窗[
'fileformat': '.tar.gz', 'stahex': '1F8B', 'endhex': ''}	帮助(H) 20.0 SR-2	2 x64 -
''fileformat': '.tar', 'stahex': '30373037', 'endhex': ''}	🔁 😂 📽 🚺 👘 🐿 🖬 📾 📾 📾 📾 🖓 🛔 🏘 🎄 🌿 🛔 → 🕀 ← → 🗍 🍣	🕹 🧼 🖿
'fileformat': '.gz', 'stahex': '1F8B08', 'endhex': ''}	Offset         0         1         2         3         4         5         6         7         8         9         10         11         12         13         14         15         ANSI ASCII           00000000         9         7         49         46         38         39         61         18         01         91         02         00         FE         FF         FF         GIF89a         bÿÿ	^
'fileformat': '.rar', 'stahex': '52617221', 'endhex': ''}	00000016 FF FF FF FF FF FF FF FF 00 00 00 21 FF 0B 58 4D 50 20 ÿÿÿÿÿÿ !ÿ XMP	
'fileformat': '.zip', 'stahex': '504B0304', 'endhex': '504B'}	00000048 20 62 65 67 69 6E 3D 22 EF BB BF 22 20 69 64 3D begin="iw; id=	
'fileformat': '.bz&bz2', 'stahex': '425A68', 'endhex': ''}	00000064 22 57 35 4D 30 4D 70 43 65 68 69 48 7A 72 65 53 "WSMOMPCeniHzres 00000080 7A 4E 54 63 7A 6B 63 39 64 22 3F 3E 20 3C 78 3A zNTczkc9d"?> <x:< td=""><td></td></x:<>	
'fileformat': '.ipeg', 'stahex': 'FFD8FF', 'endhex': 'FFD9'}	00000096 78 6D 70 6D 65 74 61 20 78 6D 6C 6E 73 3A 78 3D xmpmeta xmlns:x= 00000112 22 61 64 6F 62 65 3A 6E 73 3A 6D 65 74 61 2F 22 "adobe:ns:meta/"	
'fileformat': '.png', 'stahex': '89504E47', 'endhex': 'AE426082'}	00000128 20 78 3A 78 6D 70 74 6B 3D 22 41 64 6F 62 65 20 x:xmptk="Adobe 00000144 58 4D 50 20 43 6F 72 65 20 35 2F 33 2D 63 30 31 XMP Core 5.3-c01	
'fileformat': '.gif' 'stahex': '47494638', 'endhex': '003B'}	00000160 31 20 36 36 2E 31 34 35 36 36 31 2C 20 32 30 31 1 66.145661, 201	
"fileformat': 'tif' 'stahex': '49492A00' 'endhex': ''}	000001/6 32 2F 30 32 2F 30 36 2D 31 34 3A 35 36 3A 32 37 2/02/06-14:56:27 00000192 20 20 20 20 20 20 20 20 22 3E 20 3C 72 64 66 3A "> <rdf:< td=""><td></td></rdf:<>	
'fileformat': 'exe' 'stahex': '4D5A' 'endhex': '}	00000208 52 44 46 20 78 6D 6C 6E 73 3A 72 64 66 3D 22 68 RDF xmlns:rdf="h 00000224 74 74 70 3A 2F 2F 77 77 77 2E 77 33 2E 6F 72 67 ttp://www.w3.org	
'fileformat': 'iar' 'stahey': '5E27A889' 'endhey': ''}	00000240 2F 31 39 39 39 2F 30 32 2F 32 32 2D 72 64 66 2D /1999/02/22-rdf- 00000256 73 79 6E 74 61 78 2D 6E 73 23 22 3E 20 3C 72 64 syntax-ns#"> <rd< td=""><td></td></rd<>	
'fileformat': 'evt' 'stahev': '03000000' 'endhev': '}	00000272 66 3A 44 65 73 63 72 69 70 74 69 6F 6E 20 72 64 f:Description rd	
"fileformat': 'xml' 'stahex': '3C3F786D' 'endhex': '}	00000304 3A 78 6D 70 4D 4D 3D 22 68 74 74 70 3A 2F 2F 6E :xmpMM="http://n	/xu <b>№</b> 25

把后缀改为gif。



https://blog.csdn.net/xuhc25

用stegsolve打开,得到半截二维码。

🛓 St	—	×	<
File Analyse	Help		
Gray bits			



题目说是分层图片,ps打开,发现有图层是隐藏的。



开启图层再另存为。







< >

两半二维码拼合



还是缺定位点没有,ps上去,得到完整的二维码。



扫码得到flag flag{yanji4n\_bu\_we1shi}

10、base64stego

< ← 返回	本题用时: 26分57秒
base64	stego 127 最佳Writeup由CTFshow • zEr0_0提供
难度系数:	• * * * * 5.0
题目来源:	olympicCTF
题目描述:	菜狗经过几天的学习,终于发现了如来十三掌最后一步的精髓
题目场景:	暂无
题目附件:	附件1
	https://blog.csdn.net/xuhc25

下载完提示有密码,但是这是伪加密,360压缩可以无视。 360压缩打开压缩包是这么一大串东西

<table-of-contents> a2eb7ceaf5a</table-of-contents>	ab49f7a	cb33 36	60压缩				文件	操作	工具	帮助	$\mathbf{\Phi}$		- 0	× .	1
添加	解 氏 到				西								未知		
										****	<del></del> ///		$\sim$		
	aze	b/ceat5ab	04917acb3	3de2e7eed	/4a.zip - 解包	大小万 6	9 KB	() -	V		内又1年			q	
						压缩前		压缩后	类型		修改	日期			
I (上级日求)						6.	9 KB	3.5 KB	又1 <del>11火</del> 文本文档		201	6-04-2	7 22.59		
stego.txt	Γ	/ stego	tvt - 记車	· • 木		0		5.5 KD			2010	0 04 2	1 23.35		1
		الع stegu													
		又件(ト) 绯	扁钼(E) 格	(C) 查看	i(V) 帮助(H)	)									
		U3RIZ2	Fub2dy	YXBoeS	BpcyB0a	GUgY)	COOL	5FuZCBz	Y2IIbmN	IIIG9m					
		IHdyaX	Rpbmc	gaGlkZG	VulG1lc3	3NhZ2	VziC	JUIHN1	2ggYSE	3YXkg	dGhr	IdCB	ubyBvb	pmV=	
		LCBhcG	FydCB	mcm9tIF	IRoZSBz	ZW5kz	ZXIg	YW5kIGI	udGVuZ	GVkIH	JIY2h	waW	VudCw	gc3Vz	1
		Y3RzIH	RoZSB	leGlzdGV	/uY2Ugb	2YgdG	ihllG	illc3M=							
		YWdILC	BhlGZ	vcm0gb2	2Ygc2Vjd	XJpdH	kgd	Ghyb3Vr	naCBvYr	nNjdXJ	pdHk	ulFS	=		
		aGUgd2 IYW==	29yZCE	BzdGVnY	W5vZ3Jh	cGh5l	Glzl	G9mlEdy	ZWVrIG	i9yaWo	dpbiB	hbm	NQgbW	Vhbnl	١
		bGVkIH	ldyaXR	pbmcilG	Zyb20gd	GhllEc	lyZV	VVrIHdvo	mRzIH	NOZW0	dhbm	9zIG	1lYW5p	obmco	2
		dmVvZ	WQab	3lacHJvd	lGVidGVk	(liwaY	Ŵ5k	GdvYXB	oZWlul	G1IYW	5pbm	ncalr	RvIHc=	: .	•
		cml0ZS	lulFRo	ZSBmaXJ	zdCBvZV	VNvcr	nRIZ	CB1c2U	ab2Yad	Ghlihr	lcm0	ad2F	zIGluID	E0OT	•
		0			,				J J			<u> </u>			
		YW5uZ	XMaVH	JpdGhl	oWl1cvBr	biBoa	XMd	uuuuuuuuuuuu	Fub2dv)	(XBoa)	NEslG	iEad	HJIYV=	=	
		dGlzZS	BvbiBic	nlwdG9r	ncmFwaF	lkaYW	/5klF	- HN0ZWd	hbm9nd	mFwa	HkaZ	GlzZ	[8==		
大小: 3.5 KB 共	1 个文件	dWlzZV	VOaYX	MaYSBib	29rlG9u	IG1hZ	2liLil	BHZW5lc	mFsbHl	csIG1lc	3P=	ttne•//bl	og cedn not		
					1 100	D'700								nuni620	

<b>鸟工具</b> WEB 在线编辑器 SVG 在线编辑器 实例归档 菜鸟教程	输入关键字 <b>Q</b>
★ 编码	☎ Base64 编码/解码
98 ZXItcHJvcGFnYW5kYS4gSW4gMTk2OCwgY3JldyBtZW1iZW== 99 cnMgb2YgdGh1IFVTUyBQdWVibGSgkEFHRVItMikgaW50ZWxsaWdlbmN1HHoaXAgaGVsZ( 04 aXNvbmVycyBieSB0b3J0aCBLb3J1YSwgY29tbXVuaWNhdCVkIG1uHNpZ25= 101 IGxhbmd1YWdl1GR1cmluZyB2dGFnZWQgCGhvdG8gb3Bwb3J0 102 dW5pdG11cywgaW5mb3JtaW5n1HRoZSBVbm10ZWQgU3RhdGVzIHRoZXkg 103 d2VyZSBu52QgZGVmZWN0b3JzIGJ1dCByYXRoZXIgd2VyZSB1ZW1u2yBoZWxkIGNh 104 cHRpdmUgYnkgdGh1IE5vcnRoIEtvcmVhbnMuIEluIG90aGVyIHBo53Rv 105 cyBwcmVzZW50ZWQgdG8gdGh1IFVTLCBjcmV3IG11bWJ1cnMgZ2F2ZSAidGh1IGZpbmd1c: 106 dGh1IHVuc3VzcGVjdG1uZyB0b3J0aCBLb3J1VW52LCBpb1Bhb1BhdHR1bXB01HRvIE== 107 ZG1zY3J1ZG101HBo53RvcyB0aGF01HNob3d1ZCB0aGVtHNtaQ== 108 bG1uZyBhbmQgY29tZm9ydGFibGUuDQoNCi0tDQpodHRw018vZW4ud21raXB1ZG1hLm9yZv L3dpa2kvU3R1Z2Fub2dyYXBoeQ0K	Steganography is the art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message, a form of security through obscurity. The word steganography is of Greek origin and means "concealed writing" from the Greek words steganos meaning "covered or protected", and graphein meaning "to write". The first recorded use of the term was in 1499 by Johannes Trithemius in his Steganographia, a treatise on cryptography and steganography disguised as a book on magic. Generally, messages will appear to be something else: images, articles, shopping lists, or some other covertext and, classically, the hidden message may be in invisible ink between the visible lines of a private letter.

E

就一段隐写技术的说明,没啥用。 题目中说base64解密,最后一步精髓(最后一步是精髓,不就是提示你取行尾的密文),所以提示很明显啦,base64隐写。 这篇文章说的很好啦,base64隐写原理。 https://blog.csdn.net/xnightmare/article/details/103774379

然后解题思路是,边解密边把隐写的密文拿到再解密。

依次读取每行,从中提取出隐写位。

```
如果最后没有'=',说明没有隐写位,跳过。
如果最后是一个'=',说明有两位隐写位,将倒数第二个字符转化为对应的二进制索引,然后取后两位。
如果最后是两个'=',说明有四位隐写位,将倒数第三个字符转化为对应的二进制索引,然后取后四位。
将每行提取出的隐写位依次连接起来,每8位为一组转换为ASCII字符,最后不足8位的丢弃。
贴上大佬的代码:
 -*- coding: cp936 -*-
import base64
b64chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'#base64字典库
with open(r'C:\Users\XUHC\Documents\代码段\stego.txt', 'rb') as f: #读取文件内容
   for line in f.readlines():
      stegb64 = str(line, "utf-8").strip("\n")
      rowb64 = str(base64.b64encode(base64.b64decode(stegb64)), "utf-8").strip("\n")
      offset = abs(b64chars.index(stegb64.replace('=','')[-1])-b64chars.index(rowb64.replace('=','')[-1]))
      equalnum = stegb64.count('=') #no equalnum no offset
          bin str += bin(offset)[2:].zfill(equalnum * 2)
   print(''.join([chr(int(bin_str[i:i + 8], 2)) for i in range(0, len(bin_str), 8)])) #8 位一组
```

	<b>♥ =</b> -   •	– 👩 ma		
<b>凌</b> 攻	, 防世界-脚本1.py			
🛃 ma	in.pv		b64	chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'#bas
			⊝wit	:h open(r'C:\Users\XUHC\Documents\代码段\stego.txt', 'rb') as f: #读取文件内容
IIIII EXT	ernai Libraries	5 (		bin_str = ''
🔊 Scr	ratches and Consoles			for line in f.readlines():
				<pre>stegb64 = str(line, "utf-8").strip("\n")</pre>
				<pre>rowb64 =str(base64.b64encode(base64.b64decode(stegb64)), "utf-8").strip</pre>
				offset = <code>abs(b64chars.index(stegb64.replace('='太'')[-1])-b64chars.index(r</code>
				equalnum = stegb64.count('=') <i>#no equalnum no offset</i>
				<i>if</i> equalnum:
				<pre>bin_str += bin(offset)[2:].zfill(equalnum * 2)</pre>
				<pre>print(''.join([chr(int(bin_str[i:i + 8], 2)) for i in range(0, len(bin_str),</pre>
un: 🧃	🗦 main 🗙			
- 2	C:\Users\XUHC\AppData\Local\Pro	grams\	Python\	
<del>ک</del> ک	Base_sixty_four_point_five			
<b>V</b>	Dresson finished with suit and	0		
↓ Į	Process finished with exit code	U		https://blog.csdn.net/xuhc25

## 得到flag Base\_sixty\_four\_point\_five

# 11、ext3



提示很明显: ext3文件系统, Linux环境。

直接mount试试看,然后再进到目录,试着查找flag文件,没想到还真有,查看flag.txt,得到密文:



解密后得到flag

🛩 [随波逐流	E]CTF编码工具	V1.0 20201	1022					—		$\times$
Base加解密	字符加解密	字符编码转换	已知key解密	进制转换	其他工具	赞赏作者				
需要解密的文本↓ 密钥(key):						字数统计	一键解码	粘贴剪切板	清空	内容
ZmxhZ3tzYWpiY2lienNrampibmIoc2I2Y2pianN6Y3N6Ymt6an0=										

解密结果↓		复制内容	↑解密结果转至文本框↑
— 键 解 码: base64解码: base32解码:	结  果 flag{sajbcibzskjjcnbhsbvcjbjszcszbkzj}		https://blog.csdn.net/xuhc25

flag{sajbcibzskjjcnbhsbvcjbjszcszbkzj}

## 12、功夫再高也怕菜刀



附件后缀很奇葩:acfff53ce3fa4e2bbe8654284dfc18e1.pcapng pcap是wareshark流量的抓包,png是图片,从这两方面下手了。 1、万变不离其宗,先分解提取。 foremost 提取



> Internet Protocol \		CHM	发送到(N)	>	G Foxmail	
> Transmission Contre	<ul> <li>▶ 音乐</li> </ul>	Editor(ch m文档生成 编辑器) ray			<ul> <li></li></ul>	
		× <u>*</u>			<ul> <li>● 使用Foremost分解</li> <li>◎ 文档</li> </ul>	
0000 00 50 56 +5 c2 0010 00 3c d3 cf 40 0 0020 2b 53 ba e4 00 5	58 个项目 选中1 イ 10 40 06 a0 c8 ( 10 81 3a 60 44 6	○项目 2.25 MB ○ a8 19 80 C ○ a8 00 00 a ○ a 00 00 a ○ a 00 00 a	删除(D) 重命名(M)		→ 入雪 正缩(zipped)文件夹	
acfff53ce3fa4e2bbe8	3654284dfc18e1		属性(R)		<ul> <li>Mathematical State</li> <li>Mathematical Sta</li></ul>	載时间: 0:0.57    配置文件: Default
= 🛛 🖾	<mark>۵ ھ</mark>	🧿 😰 í	🎓 🥼 📀		- 桌面快捷方式	9:55 <i>底</i> 4× 英 ④ 155

得到压缩包,解压要密码,确认不是伪加密后,进行下一步。

> customtoo	ls > foremostlrb > output > zip >	~	U	搜索"zip"	_
石寺	📑 正在解压: 00002778.zip -360压缩			- 23	
白你	📑 输入密码			23	
000027 📜					r 
4 000027	为加密的文件输入密码:				缩 ZIP :
	flag.txt				
				×	
	□ 显示密码				
	管理密码 确定	耳	以消		

https://blog.csdn.net/xuhc25

密码估计要在数据流里找了。

wareshark打开流量数据包,ctrl+f,查找flag关键字。

😈 cmd.exe (Admin) 🌃 Wireshark.exe (Admin)	na na se na se la 🙆 na na na na se la cal 🖬 💌 🗷 🎽 📖 🚍 👝
文件(E) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(I) 帮助(H)	
📶 🔳 🖉 📵 📕 🖹 🕱 🗳 🗢 🛸 🐐 🖢 🜉 🜉 🗨 Q, Q, Q, III	
应用显示过滤器 ··· <ctrl-></ctrl->	
	賞找 以语
No. Time Source Destination Protocol Length Info	
1138 49.859343921 219.216.128.25 192.168.25.128 TCP 1434 80 → 58432 [PSH	, ACK] Seq=172501 Ack=145 Win=64240 Len=1380 [TCP segment
1139 49.859392526 192.168.25.128 219.216.128.25 TCP 54 58432 → 80 ACK	] Seq=145 Ack=173881 Win=65535 Len=0
1140 49.896847126 219.216.128.25 192.168.25.128 TCP 1434 80 → 58432 PSH	, ACK] Seq=173881 Ack=145 Win=64240 Len=1380 [TCP segment
1141 49.896906030 192.168.25.128 219.216.128.25 TCP 54 58432 → 80 [ACK	] Seq=145 Ack=175261 Win=65535 Len=0
1142 49.897175062 219.216.128.25 192.168.25.128 TCP 1434 80 → 58432 [PSH	, ACK] Seq=175261 Ack=145 Win=64240 Len=1380 [TCP segment
1143 49.897215668 192.168.25.128 219.216.128.25 TCP 54 58432 → 80 [ACK	] Seq=145 Ack=176641 Win=65535 Len=0
1144 50.098716397 192.168.43.83 192.168.25.128 HTTP 300 HTTP/1.1 200 OK	(text/html)
1145 50.098792302 192.168.25.128 192.168.43.83 TCP 54 47856 → 80 [ACK	] Seq=205239 Ack=247 Win=30016 Len=0
1146 50.134447510 192.168.25.128 192.168.43.83 TCP 290 47856 → 80 [PSH	, ACK] Seq=205239 Ack=247 Win=30016 Len=236 [TCP segment
1147 50.138633287 192.168.43.83 192.168.25.128 TCP 60 80 → 47856 [ACK	] Seq=247 Ack=205475 Win=64240 Len=0
- 1148 50.138903657 192.168.25.128 192.168.43.83 HTTP 777 POST /upload/1.	<pre>php HTTP/1.1 (application/x-www-form-urlencoded)</pre>
1149 50.140816842 192.168.43.83 192.168.25.128 TCP 60 80 → 47856 [ACK	] Seq=247 Ack=206198 Win=64240 Len=0
- 1150 50.147576455 192.168.43.83 192.168.25.128 HTTP 515 HTTP/1.1 200 OK	(text/html)
V line-based text data: text/html	
-> _/\t2017-12-08 11:42:11\t0\t077\n	
1.php\t2017-12-08_11:33:16\t33\t0666\n	
6666 ing/t2017-12-08 11:42:11\t10226\t9666\n	
flag.txt\t2017-12-08 11:35:29\t17\t0666\n	
alba 36 36 36 92 66 60 61 67 20 74 78 74 09 32 30 31 666 flag tyt 201	
01c0 37 2d 31 32 2d 38 20 31 31 3a 33 35 3a 32 39 7-12-08 11:35:29	
01d0 09 31 37 09 30 36 36 36 0a 68 65 6c 6c 6f 2e 7a .17.0666 .hello.z	
01e0 69 70 09 32 30 31 37 2d 31 32 2d 30 38 20 30 39 ip.2017- 12-08 09	
01f0 3a 33 32 3a 33 36 09 32 32 34 09 30 36 36 36 0a :32:36.2 24.0666.	

	_	×
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)		
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H) { fileformat': '.tar.gz', 'stahex': '1F8B', 'endhex': ''} { fileformat': '.tar', 'stahex': '30373037', 'endhex': ''} { fileformat': '.gz', 'stahex': '1F8B08', 'endhex': ''} { fileformat': '.gz', 'stahex': '52617221', 'endhex': ''} { fileformat': '.gr', 'stahex': '52617221', 'endhex': ''} { fileformat': '.gr', 'stahex': '52617221', 'endhex': ''} { fileformat': '.gr', 'stahex': '504B0304', 'endhex': ''} { fileformat': '.gr', 'stahex': '504B0304', 'endhex': '504B'} { fileformat': '.bz&bz2', 'stahex': '425A68', 'endhex': ''} { fileformat': '.gr', 'stahex': '425A68', 'endhex': 'FFD9'} { fileformat': '.gif', 'stahex': '47494638', 'endhex': 'AE426082'} { fileformat': '.gif', 'stahex': '47494638', 'endhex': 'O03B'} { fileformat': '.gif', 'stahex': '49492A00', 'endhex': ''} { fileformat': '.gir', 'stahex': '49492A00', 'endhex': ''} { fileformat': '.gir', 'stahex': '495A', 'endhex': ''} { fileformat': '.gir', 'stahex': '5F27A889', 'endhex': ''} { fileformat': '.gir', 'stahex': '3C3F786D', 'endhex': ''} { fileformat': '.mth', 'stahex': '68746D6C', 'endhex': ''} { fileformat': '.mdb', 'stahex': '5374616E', 'endhex': ''} { fileformat': '.doc', 'stahex': 'D0CF11E0', 'endhex': ''} { fileformat': '.doc', 'stahex': '25504446', 'endhex': ''}		
{ fileformat': 'vis' 'stabev': '09040600' 'endbev': '}		
{'fileformat': 'mov' 'stahex': '6D646174' 'endhex': '}		
Windows (CRIF) 第7行 第42列	100%	

🧧 Wireshark · 追踪 TCP 流 (tcp.stream eq 7) · acfff53ce3fa4e2bbe8654284dfc18e1 POST /upload/1.php HTTP/1.1 User-Agent: Java/1.8.0 151 Host: 192.168.43.83 Accept: text/html, image/gif, image/jpeg, \*; q=.2, \*/\*; q=.2 Connection: keep-alive Content-type: application/x-www-form-urlencoded Content-Length: 204999 aa=@eval. (base64\_decode(\$\_POST[action]));&action=QGluaV9zZXQoImRpc3BsYX1fZXJyb3JzIiwiMCIpO0BzZXRfdGltZV9saW1 pdCgwKTtAc2V0X21hZ2ljX3F1b3Rlc19ydW50aW11KDApO2VjaG8oIi0%2BfCIpOzskZj1iYXNlNjRfZGVjb2R1KCRfUE9TVFsi ejEiXSk7JGM9JF9QT1NUWyJ6MiJd0yRjPXN0cl9yZXBsYWN1KCJcciIsIiIsJGMpOyRjPXN0cl9yZXBsYWN1KCJcbiIsIiIsJGM pOyRidWY9IiI7Zm9yKCRpPTA7JGk8c3RybGVuKCRjKTskaSs9MikkYnVmLj11cmxkZWNvZGUoIiUiLnN1YnNØcigkYywkaSwyKS k7ZWNobyhAZndyaXR1KGZvcGVuKCRmLCJ3IiksJGJ1Zik%2FIjEiOiIwIik7O2VjaG8oInw8LSIpO2RpZSgpOw%3D %3D&z1<del>\_RDpcd2</del>FtcDY0XHd3d1x1cGxvYWRcNjY2Ni5qcGc 

🧲 Wireshark · 追踪 TCP 流 (tcp.stream eq 7) · acfff53ce3fa4e2bbe8654284dfc18e1

0D8FFF0040155A4EABFF0060C97F9BD76D35B79E9B2F25D7D34F92D526A5C7525A6D7D2FF8297FEDD66EDAB7292B36B9684
84301B49954C876E724DEDD9FBCE47FCF08893C93CFD58E2AB92436EFDEA993F78464FDB2EB90B12918630C59C1C7CA474C
6E515607DEB3FF00AF5B8FE52D578BFE61FF00F5CE7AECA6AED5FB2FC7D9FF00F27AF751D7E295F9A52B5FADBBF973BFBFD
CF95FFBB1E5AD211F3EFF0098061F68653CCD3E014B58881FEAD38DFB41C633D90567C843799E63908306F1D78C95FF0055
650FAFDD01B1D31CF09CDE1F76C3FEBACDFF00A10ACDFF0097783FEBFDFF00F65AEDA6B48F5BD9EBE6A0FE7F1EADEFEF759
B6B96A371BABBDD2EDF69EAB7B7C1A745EE696824EACEBBE40F2E12487E72E3836303C525BF911631B65B88679619003931
4AF1B0C33815E56EC7F74DE5E71938B4B53CE31FC53CA08E4E09DDD8BE04D27FAA97FEC27FD64AA97BFF002FDFF5F917FEC
D5DD4A0AEDD97335772B6AF953B27FF0080E9D9C9B56B47979E527CA936DA8D9455F45774AF65AA4AF2D5755149B779B954
9189DA41F2C843E483D2D2D8925E67FF00A6D367823E63D41F996B3252C4A6C014E18DB8603F7518CF9B792F6F30F3B4B1C
82011C0506FDE74BEFAC1FCAB3EE7ADFF00FD7083F9C75D0FDD83B74D7D6DCDE565FC3EDBB5FC91B73D2F7A4BCF937D6DCF
EC97CECAADB5DD295FE393322E24C6C1183925BECCAC4F5E7CDBC9BD48DADB41C018E32139C694801447F38DCDE486FF009
6F3747B997B08D3076EEF61D9C56A5DFDFB8FFB0745FF00B4EB1E6EABFF0060E7FE42BCCAD75A76F2B6B74AE97CB4E8BDD4
EE9352F630E95A3E76FC545FFEDDD2DBCDAB392E4CD99B0A31FBD05FE4CF2D79723EF48D93930C648EBC608CE0935893BE7
71399079983839FB4DD374553FF003CA33D80E79EEC00D49BEF41FF005E32FF0027ACA3D6D3FEB85CFF00296BC6ACF5F9A5
F7FB3FFE4FE6959FC53E6F770E96AFB5FCB6E67D366DC3A689B6D2D236C8BA9000DB89705BF798CE679C1F96043FF3CE3EA
5BA7191FC158B70E4F98643BB0409D971F3B7FCB2B58F03855C7CDB41039FEEA83A72FDFB3FF7DFFF0066AC83D2D3FEBEA4
FF00D0C578F8896CBBC53FBD53767DFF0089D775CDFCEEDEEE192BFA36BE69DAFE5F0E96DBDDB690465CEF8326FE08C7DA1
97A20E365B47D831C00DE9DF201CE3DCB677866DADB71291FF2C61E8B02FABBF1B8638DD83CEEAD297FD5A7FD7F37F4AC8B
8E92FF00D7E8FE46BC5AD276BFCFEFF7B57F87CE4FAAB7BD865771E9ADBD2D251FD36DBE156693E6C9BA90E580C21D9866F
F009F780F1B467A4926E00FF176E0B3118731C7CD8D876128A738B780F2646FFA6B2678C1E49E30480356EFEEDCFF00D7D2
7F37ACAD47A5FF00FBF07F235E3577A7AB7F85FF001F7775B37756B452F6B0DAB827BC9C75F5F651DB6D39FEE56FB526F0E
E1CB30D9F28018C00E3F751F57B8931FF002D1B191E99E32319C999F681B7A90C6156C60609DD73276E31F2E7B8183F2F37
EE7EFDCFFD7383F9C758F77D65FF00AF78FF00F408EBC6C4DD4ADE4F5F4EDBD97BBA2E9A7F2A3E830EB45E89F7DD41EBD5F
C7AF7F7BAC9B59E658327293B9CF2E1BEF7FB5FF02EBF8D15763FF571FF00B8BFFA08A2B86DFE1FFC05797F93FBFEFE9E65
DA5FF81BFF002F5FE96 <mark>8</mark> FFD9H <mark>TTP/1.1 200 OK</mark>
Date: Fri, 08 Dec 2017 11:42:07 GMT
Server: Apache/2.4.23 (Win64) PHP/5.6.25
X-Powered-By: PHP/5.6.25
Content-Length: 7
Keep-Alive: timeout=5, max=100
<u> </u>
Naa 11000 oo <del>ya</del> y ya Naa, 110 Mkuyaa Naa, 11 UUII(3), mUUA2)+0

### winhex生成图片

从FFD8FF开头,FFD9结尾复制,打开winhex 点击文件,新建,大小设置1111

WinH	ex -	(末	命名]					$\times$
<b>経</b> 文件(F	) 绯	扁钼	i(E) 搜索(S)	导航(N) 查看(V)	) 工具(T)	专业工具(I) 选项	(0) 窗	口(W)
帮助(H)		7=				20.00	764	- 8 ×
		建.	业新又作			>	13	🗰 🔎 🙉 🔬
Offset.	0							
00000000	FF	Γ					^	
00000016	00		▲文件	+1. 1	111	Butos V		
00000032	01		<b>U</b> AH			Dytes		
00000048	01							
00000064	01							
00000080	01							
00000096	01							
00000112	01							
00000128	01							
00000144	01							
00000160	00							
00000176	01							
00000192	00	-						
00000208	AO	Г						
00000224	05		确定(O)	取消(/	4)			
00000240	31		40/LL( <u>C</u> )	-10117	-/			
00000256	42							
00000272	18	19	1A 25 26 27 28	29 2A 34 35 36	37 38 39 3A	A %&'()*456789		
00000288	43	44	45 46 47 48 49	AA 53 54 55 56	57 58 59 5A	CDEFGHIJSTUVWXY	Z	
00000304	63	64	65 66 67 68 69	0 6A 73 74 75 76	77 78 79 74	A cdefghijstuvwxy:	z 🗸	

\_

.

页1/323	偏移地址:	0	= 255 选块:	0-102,225 大小: 102,226

粘贴,提示编码选择,选ASCII Hex

	HEX Mind	ov I+						$\sim$
	HUINA BEEF	ex - [オ	DataObject					X
	. 🎏 文件(F	) 编辑	UTF-16 Unic ANSI/OEM -	ode 简体中文 GBK			(O) 窗口(	(W)
	邦助(H)		HTML Format				2 x64	- x
	(יי) נאינחי		application	/vnd.oasis.opend	oc		2 101 -	- n
,	, 🗅 🖱 🖬 🎕	) 📚 😭	Ole Private	Data			» 🗳 🍣 🎟	🔎 🦚 💕
	Offset	0 1	Locale					
	00000000	00 00	IBM/OEM					
	00000016	00 00	ASCII Hex					
	00000032	00 00						
	00000048	00 00						
	00000064	00 00						
	00000080	00 00						
	00000096	00 00						
	00000112	00 00						
	00000128	00 00						
	00000144	00 00						
	00000160	00 00						
ł	00000176	00 00						
	00000192	00 00						
	00000208	00 00						
	00000224	00 00						
1	00000240	00 00						
	00000256	00 00						
	00000272	00 00						
	00000288	00 00						
	00000304	00 00					$\sim$	
	页1/4	_					大小:	无
			🖪 确定( <u>O</u> )	取消( <u>A</u> )		帮助( <u>H)</u> ps://bl		

## 另存为,加后缀.jpg

件名(N):	1. jpg	$\sim$
存类型	所有文件	

### 得到一张图,估计是解压密码了





Th1s\_1s\_p4sswd\_!!! 拿去解压,得到flag.txt文件

📑 正在解压: 00002778.zip -360压缩	
📑 输入密码	23
为加密的文件输入密码:	
flag.txt	et
Th1s_1s_p4sswd_!!!	~
☑ 显示密码	
管理密码 确定	取消
	pad/otagoaadmonetikotho25

打开flag.txt,得到flag flag{3OpWdJ-JP6FzK-koCMAK-VkfWBq-75Un2z}

🧾 flag.txt - 记事本

文件(F)编辑(E)格式(O)查看(V)帮助(H)

# flag{3OpWdJ-JP6FzK-koCMAK-VkfWBq-75Un2z}

至此,新手的misc模块题目全部完成了。