## [xueqi]ISCC 2019 writeup 信息安全与对抗-解题思路xueqi





CTF 专栏收录该内容

2篇文章 0 订阅 订阅专栏 ISCC 2019 writeup-全国大学生信息安全与对抗技术竞赛 解题思路xueqi

## **WEB**

#### Web1

题目地址: http://39.100.83.188:8001

读了下代码,get传参,value和password。Value的值要等w3lc0me\_To\_ISCC2019,但ascii值不能在32和127 之间。

查php手册资料得知chr()函数是值除以256取余数。这就好办了,只需要在原有数值上加上256就ok了。

再看password, intval可以用科学技术法绕过, 如: 2e4

http://39.100.83.188:8001/index.php? value[]=375&value[]=307&value[]=364&value[]=355&value[]=304&value[]=364

Hello w3lc0me\_To\_ISCC2019!

```
<?php
error_reporting(0);
require 'flag.php';
$value = $_GET['value'];
$password = $_GET['password'];
$username = '';
for ($i = 0; $i < count($value); ++$i) {
    if ($value[$i] > 32 && $value[$i] < 127) unset($value);
    else $username .= chr($value[$i]);
    if ($username .= chr($value[$i]);
    if ($username .= 'w3lcOme_To_ISOC2019' && intval($password) < 2333 && intval($password + 1) > 2333) {
        echo 'Hello '.$username.'!', '<br/>br>', PHP_EOL;
        echo $flag, '<br/>hr>';
    }
highlight_file(_FILE_);
```

01

#### Web2

nayto

此题是要输入三位数值,还有验证码,要暴力破解试下。

自己从999倒着试试,还没到990就试出来了

不过大佬告诉我要抓包,删除cookie和user\_code的值,填上pwd的值就好了

<>C 0 5 %   0	39.100.83.188:8002
🙄 微信网页版 📘 微软 Bing 📄 虚拟化 🚺	🗿 百度翻译 🛅 工作 🔛 百度一下, 💙 loat 🎎 phpMyAd 💁 God
破解3位数字密码 用户名: admin	☐ 39.100.83.188:8002/login.php × +
密码: 990	< < < < < < < < < < < < < < < < < < <
验证码: bhb4	😬 微信网页版 ▶ 微软 Bing 📄 虚拟化 👔 百度翻译 📄 工作
bhb4 submit	flag is flag{

Web3

sqli-labs中的一道原题,好像加了过滤。自己做题时,admin好像被大佬改了弱口令,捡漏登陆admin直接爆出 flag,后来被修复。

思路是sql注入,注册帐号登陆后,修改密码页面,帐号填admin'or 1#修改admin的密码,然后登录admin获取 flag



## Web4

http://39.100.83.188:8066

```
<?php
error_reporting(0);
include("flag.php");
$hashed key = 'ddbafb4eb89e218701472d3f6c087fdf7119dfdd560f9d1fcbe7482b0feea05a';
$parsed = parse_url($_SERVER['REQUEST_URI']);
if(isset($parsed["query"])){
    $query = $parsed["query"];
    $parsed_query = parse_str($query);
    if($parsed query!=NULL){
        $action = $parsed_query['action'];
    }
    if($action==="auth"){
        $key = $_GET["key"];
        $hashed input = hash('sha256', $key);
        if($hashed_input!==$hashed_key){
            die("<img src='cxk.jpg'>");
        }
        echo $flag;
    }
}else{
    show_source(__FILE__);
}?>
```

读下代码, parse\_url这个函数引起注意, 百度了下, 原来存在变量覆盖漏洞。

Key要=0, key是被sha256加密过的, 传一个hashed\_key, 将原来的密文覆盖掉, hashed\_key的值是0经过 sha256加密的密文



#### Web5

访问,提示不是Union.373组织成员。看到这个就想到改请求头。

User-agent中加上Union.373,再次访问,然后页面提示要用户名,post方式提交,username和password的值试试admin访问,提示密码既为flag。问了大佬,说要在这里注入,有过滤,union没过滤,

#### Web6

一开始无从下手,顺着PasteBin查阅原来是考验JWT。注册一个帐号,随便鼓捣一番,有list功能呢,抓包发现 jwt到https://jwt.io/ 解码查看。 查看源码发现common.js文件有大料:

function getpubkey(){

```
/*
get the pubkey for test
/pubkey/{md5(username+password)}
*/
```

把jwt更改加密方式为HS256用公钥加密 然后拿到"admin:22f1e0aa7a31422ad63480aa27711277" 访问url/text/admin:22f1e0aa7a31422ad63480aa27711277获得flag

## MISC

#### Misc1 隐藏的信息

这是一个被混淆的文件,但是我忘记了这个文件的密码。你能够帮助我还原明文吗?

0126 062 0126 0163 0142 0103 0102 0153 0142 062 065 0154 0111 0121 0157 0113 0111 0105 0132 0163 0131 0127 0143 066 0111 0105 0154 0124 0121 060 0116 067 0124 0152 0102 0146 0115 0107 065 0154 0130 062 0116 0150 0142 0154 071 0172 0144 0104 0102 0167 0130 063 0153 0167 0144 0130 060 0113

看这些数字,像是某种进制,最大是7,可能是8进制,转换十进制再对应ascii表转字符串。得到结果。

#### Misc 倒立屋

拿到压缩包,里面是张图,还以为线索在倒立上,用Stegsolve打开,发现是lsb隐写。

Extract Preview	
497343635f323031 39a4026ef5224e75 Is .n."Nu	
5fe03bld8ed893b2 1fffff000038c3fa .;8	
ad4953b41b6d8ec4 ec4ec76d8edb6db6 .ISmN.mm.	
2713b1db6db62492 49db6d8e036db6fe 'm.\$. I.mm	
db4956d56db6db6d db6db6276272a5ba .IV.mm .m.'br	
ec0753d07f2c0d77 0c1fb55723007a94s.□,.wW#.z.	
6e2435ac161ec9b9 313a4e2d20790956 n\$5 l:N- y.V	
491e23f1f80381c0 92492491b8db9249 I.# I\$I	
2492492492492491 c8∈4924924924924 \$.I\$.I\$I\$.I\$	
7237239249249249 246db6db6db6db92 r7#.I\$.I \$mm	
ARAAL P.P. P.P.A. C. V. C. V. C. V. C. V. C. R. H.	
Bit Planes Order settings	
Alpha 7 6 5 4 3 2 1 0 Extract By Row Column	
Red 7 6 5 4 3 2 1 0 Bit Order MSB First LSB	First
Green 7 6 5 4 3 2 1 2 0	
Bit Plane Order	

选择 R G B 最低为勾选, 就可看到iscc字段, 不过flag提交有个脑洞就是要把找到的线索反序排列提交。

## **Misc Keyes' secret**

打开时,看到一大串字符头都大了,也不知道是什么加密什么算法。睡一觉起来,发现有一段{}括住的字符 串。

I	🥘 keyes.txt - 记事本	_		>
I	文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)			
I	RTY			
I	(WSX.WSXCDE.,QWERTYYHNMKJTGBNMJUCVGREDQWERTYYHNMKJTGBNMJUYTGBNMJUZAQWDVI	FRTYU	INTOIN	EF
	VTOWERTWIRVOVENER AND VORRECHUTORNIA UNVOVERED AND VORRECHUCORDVOVENETVUR		NIDV (O	

VTQWERTYNBVCXSWERFMNBVCDRTGHUTGBNMJUYCVGREDMNBVCDRTGHUGRDXCVBXSWEFTYHNIUYHNBVQW ERTYWSXZAQWDVFRQWERTYNBVCXSWERFMNBVCDRTGHUTGBNMJUYTRFVGWSXCDEIUYHNBVIUYHNBVWSXTG BNMJUYZAQWDVFRGRDXCVBWSXCVQWERTYIUYHNBVWSXCDETYUIOJMTYUIOJMWSXZAQWDVFRCVGREDIUYH NBV).QWERTYRFVGYHNWSXCDEMNBVCDRTGHUWSXCDEQWERTYGRDXCVBMNBVCDRTGHUWSXCDEQWERTYEF VTTGBNMJUYTGBNMJUMNBVCDRTGHUQWERTYTRFVGWSXCVGRDXCVBCVGRED {WSXIUYHNBVTRFVBTRFVBQWERTYQAZSCEWSXCDEEFVTYHNMKJTGBNMJUYGRDXCVBMNBVCDRTGHUWSXCFE QWERTYTRFVBWSXNBVCXSWERFRFVGYHNWSXCDEMNBVCDRTGHU}

解密无果,后来试了下键盘轨迹,发现轨迹前四个是iscc,此题需要有想象力,字母对照键盘画轨迹。

#### **Misc Aesop's secret**

打开是个gif图片,快速闪动,能看出来是个图案,拖进pscs6里,所有图层勾选可视。发现iscc图形旁边是黑 点,还以为线索在周围黑色像素点上,拖进hex编辑器发现gif尾部存在一串加密密文,怀疑是base64,复制解 密,无果。



拿去试AES解密,解不出,密钥不知道,想起gif的iscc字样,密码填入iscc,解密拿到明文。

#### Misc Welcome-流浪

这个题目拖了五天才有思路,期间大佬们一直调戏我这萌新,说电影里有线索,想到出题人的脑洞,我还真去 看了遍电影,专去找户口身上的付款二维码,条码,刘培强的号码牌条码(气到痛哭)。

然后有大佬提示我留意"户口""长条",然后就和小伙伴展开脑洞,长条是李一一,一一,1!!.户口,刘 启,口,0,???,二进制?

看密文,发现户口是重点,空格是隔断,我先把有户口的字符串替换为0,没户口的替换为1,拿去解密,无 果?哪里出了错?

朋友做的相反,是吧有户口的作为1,没户口的作为0,解密拿到了flag

#### 

长条多出来了, 删掉, 二进制转字符串: 得到结果

#### Misc他们能在一起吗

下载附近是个二维码后缀png,我觉得没那么简单,习惯性winhex16打开看,果然里面有压缩文件,打开,发现 需要密码。

扫下二维码,信息是:UEFTUyU3QjBLX0lfTDBWM19ZMHUIMjEIN0Q=,

Base64明文PASS{0K\_I\_L0V3\_Y0u!},群里好多人拿到这个就当flag提交还在群里问什么格式,怎么提交不对。 Png改成zip后缀,打开,解压文件,密码输入"0K\_I\_L0V3\_Y0u!",拿到flag

	🚹 Reply.png.zip - 自解压格式 ZIP 压缩文件, 解包大小为 30 字节
	🔒
18 A	You won't Wanna see this.txt *
-1	
	You won't Wanna see this.txt - 记事本
Re	文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
	ISCC {S0rrY_W3_4R3_Ju5T_Fr1END}

#### Misc无法运行的exe

附件里是个exe,winhex查看,明明是一串base64密文,拿去解base64,是乱码。

开头有png字样,想到了是不是图片转base64的。Exe文件winhex打开,编辑》转换文件》 base64->binary,保存成png,打开发现无图像,查看文件头

文件头是89 50 4E 47 0D 0A 1A 00

PNG文件头是89 50 4E 47 0D 0A 1A 0A

修改下头,就得到二维码,扫码拿到flag

innable.png																			
Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANS	I AS	CII
0000000	89	50	4E	47	OD	0A	1A	0A	00	00	00	OD	49	48	44	52	% PNG	I	HDR
0000016	00	00	01	29	00	00	01	29	01	00	00	00	00	E7	E2	E9	))	(	çãé
00000032	2F	00	00	02	02	49	44	41	54	78	9C	ED	9A	41	BE	98	/ IDAT:	xaiā	AŽ>
00000048	40	10	45	DF	07	4B	BO	6B	EF	B2	84	9B	EO	90	CC	F8	0 ES K°ki	" " >áo	zlø
00000064	66	E6	08	89	81	BD	CC	0E	16	91	40	32	FC	2C	20	9E	fæ * bl	1820,	, ž
08000000	59	8E	C4	64	70	48	F7	06	54	7A	52	7F	5A	A5	AF	AA	YŽĂdpH÷ T	zR Zł	¥- *
00000096	2E	64	3E	BO	2E	C9	47	28	88	58	C4	22	16	B1	88	7D	.d>".ÉG("	XĂ" :	±^}
00000112	18	EB	34	AF	B2	CB	47	89	8B	48	18	97	58	BE	B1	B6	44"*EGSK	J -X3	P±H
00000128	3D	60	99	6D	FB	44	E8	0E	00	55	73	04	D9	B6	DB	CD	=`PnûJê	Us Ù	tůr
00000144	<b>B</b> 5	ED	01	18	25	D5	DO	E5	C8	9C	2F	55	OB	20	A9	70	ui %ČĐảĖ	a/D	61

#### Misc解密成绩单

附件打开是个exe,文件头显示确实是exe,运行,需要帐号密码,拖进IDA分析 找到了admin像是用户名,下面字符串,有两行,我复制了第一行,怎么试都不对,后来两行都复制了连在一起 作为pass,成功获得flag

ISCO	C(YOu_FOUnD_ThE_P	4SSW0RD!}		管理         解密成绩单            应用程序工具										
		确定	J B B B B B B B B C C C C C C C List			_T	4							
	速访问" ← → · <b>↑</b>	★ 剪切 剪贴板 → 解密	Login Username Shi Password	: admin : *********	******	068	华							
	🛃 快速访问	^		Cane	۵۱	OK	_							
床回数 <b>三</b> 常美 回数室口	8近数 📕 指令 III 数据 📕 未知 III ロ	外部符号 ) & × 🚺	IDA Viev-A 🖸	○ 十六道制视型	1	组织体	<b>1</b>	603						
Score_List.sco Score_List.sco Score_List.sco Score_List.sco Score_List.sco Score_List.sco Score_List.sco Score_List.sco Score_List.sco	re_list_starel_Click re_list_ht&urel_Click re_list_ht&pin_Click re_list_dorderigConterved re_list_dorderigConterved re_list_dorderign re_list_dorderign re_list_bispee re_list_bispee			text "UTF-16LE", bAdmin: text "UTF-16LE", "IscatigeLighb6: Text "UTF-16LE", Text "UTF-16LE", BEthogin: text "UTF-16LE", OK: text "UTF-16LE", itext "UTF-16LE", text "UTF-16LE", Itext "UTF-16LE	Too many login att admin",0 ISCCq19pcIVhb6Sqtf eBo4ThSn",0 btnLogin",0 BOC.49/DIOG.4	<pre>cepts*,0 // DATA XREF: 5 // DATA XREF: 5</pre>	icore_List.score_ icore_List.score_ Qxtfallp000041im icore_List.score_ icore_List.score_ icore_List.score_	list_che list_che list_Ini list_Ini 7_4 list_Ini						

## Misc High起来

附件里是个png,没图像,发现末尾有1.MP3,一定又是包含了压缩文件,png改成zip后缀,打开发现01.MP3 看到MP3,专门去百度了音频隐写术,音频可视化工具也打开看了没有信息。 MP3Stego打开,MP3Stego-x-ppass1.mp3,不知道没密码,用不了。后来经经提醒让我再看看png图片为啥 不显示。 果然又是文件头损坏,改文件头89显示二维码图片,扫码拿到当铺密码 Pass:中口由羊口中中大中中中井 当铺密码解码: 201902252228 然后 MP3Stego -x -p 201902252228 1.mp3 解出信息放在txt中,拿到

flag{PrEtTy\_1ScC9012

然后Unicode转ascii就是flag了

Misc最危险的地方

## 下载附件: Misc-01.zip

发现里面一张jpg图片,打开无图像,16进制看到存在压缩文件头,以及50个图片名

改后缀为压缩文件或者直接winrar打开,解压,全是二维码,扫码。

由于想抢百血,一直很紧张,思路也不清晰,一直在和二维码较劲,都没仔细看文件大小和后缀名.

后来百血没了才冷静下来发现第50个图片35kb,其他的图片也才8百多字节,仔细一看后缀是jpg,拖进编辑器 发现了flag。

50.jpg	×																	
	Q	1	2	3	4	Ş	6	7	8	9	ą	þ	ç	þ	ę	ţ		
00001070h:	00	00	00	00	00	00	00	00	00	00	00	00	66	00	6C	00	;	
00001080h:	61	00	67	00	7B	00	31	00	35	00	63	00	43	00	39	00		a.g.{.1.5.c.(
00001090h:	30	00	31	00	32	00	7D	00	00	00	FF	E1	80	DD	68	74	;	0.1.2.}
000010a0h:	74	70	ЗA	2F	2F	6E	73	2E	61	64	6F	62	65	2E	63	GT	•	tp.//ms.udob
000010b0h:	6D	2F	78	61	70	2F	31	2E	30	2F	00	3C	3F	78	70	61	;	m/xap/1.0/.<
000010c0h:	63	6B	65	74	20	62	65	67	69	6E	ЗD	27	EF	BB	BF	27	;	cket begin='
000010d0h:	20	69	64	3D	27	57	35	4D	30	4D	70	43	65	68	69	48	;	id='W5M0MpCe
44.png	77	72	-65	019,	2/2:	20	214	63	77	A	DSe	ell	.6 P	N	35	31	1 K	BreSzMTczkc90
45.png			2	019/	2/2	5 20:	21			AC	DSe	e 10	.0 P	N			1 K	B leta
46.png			2	019/	2/2	5 20	21			AC	DSe	e 10	.0 P	N			1 K	B f:RI
47.png			2	019/	2/2	5 20	21			AC	DSe	e 10	.0 P	N			1 K	B "ht
🚾 48.png			2	019/	2/2	5 20	21			AC	DSe	e 10	.0 P	N			1 K	B rg/1
49.png			2	019/	2/25	5 20:	21			AC	DSe	e 10	.0 P	N			1 K	B .f-sy
🚾 50.jpg			2	019/	4/29	9 16	26			AC	DSe	e 10	).O JF	PE		3	1 K	B

#### Misc 碎纸机

找到了八张图片时,懵了,有大佬暗示要用opencv,或者gl,可是都不会用后来遇到大佬给思路,说是每个图 片后边插入的数据,很有规律,

> ÿÿÀ ð

> > € ð

ΫŸ

> € ð

00	01	FO	00	00	00	00	00	00	00	07	FF	FF	CO	0F	00		ð			ÿÿ	A
00	00	00	00	00	00	00	FF	FF	FC	00	FO	00	00	00	00				yyu	ð	
00	00	00	0F	FF	FF	CO	OF	00	00	00	00	00	00	00	00			ÿÿÀ			
FF	FF	FC	00	FO	00	00	00	00	00	00	00	OF	FF	FF	CO	ΫŸ	ü	ð			ÿÿÀ
OF	00	00	00	00	00	00	00	00	F8	00	00	00	FO	00	00				ø	1	ð
00	00	00	00	00	OF	80	00	00	OF	00	00	00	00	00	00			€			
00	00	F8	00	00	00	FO	00	00	00	00	00	00	00	OF	80		ø	ð			€
00	00	0F	00	00	00	00	00	00	00	00	F8	00	00	00	FO					ø	ð
00	00	00	00	00	00	00	OF	80	00	00	0F	00	00	00	00				€		
00	00	00	00	F8	00	00	00	FO	00	00	00	00	00	00	00		3	ø	ð		
0F	80	00	00	0F	00	00	00	00	00	00	00	00	F8	00	00	€				1	z
00	FO	00	00	00	00	00	00	00	OF	80	00	00	OF	00	00	ð				€	
00	00	00	00	00	00	F8	00	00	00	FO	00	00	00	00	00			ø		ð	
00	00	OF	80	00	00	OF	00	00	00	00	00	00	00	00	F8		€				ø
00	00	00	FO	00	00	00	00	00	00	00	OF	80	00	00	OF		ð			€	
00	00	00	00	00	00	00	00	F8	00	00	00	FO	00	00	00				ø	ð	
00	00	00	00	0F	FF	FF	80	0F	00	00	00	00	00	00	00			ÿΫ	E		
00	FF	FF	FC	00	FO	00	00	00	00	00	00	00	OF	FF	FF	ÿ	ÿü	ð			ΫŸ
C0	0F	00	00	00	00	00	00	00	00	FF	FF	FC	00	FO	00	À				yyu	ð
00	00	00	00	00	00	OF	FF	FF	80	0F	00	00	00	00	00			3	ÿÿ€		
00	00	00	FC	00	00	00	FO	00	00	00	00	00	00	00	OF		ü	ų.	ð		
80	00	00	OF	00	00	00	00	00	00	00	00	F8	00	00	00	€				ø	
FO	00	00	00	00	00	00	00	OF	80	00	00	OF	00	00	00	ð			€		
00	00	00	00	00	F8	00	00	00	FO	00	00	00	00	00	00			ø	ð		
00	OF	80	00	00	OF	00	00	00	00	00	00	00	00	F8	00		€				ø
00	00	FO	00	00	00	00	00	00	00	OF	80	00	00	OF	00		ð			€	
00	00	00	00	00	00	00	F8	00	00	00	FO	00	00	00	00				ø	ð	
00	00	00	OF	80	00	00	OF	00	00	00	00	00	00	00	00		1	E			
F8	00	00	00	FO	00	00	00	00	00	00	00	OF	80	00	00	ø	i.	ð		1	E
OF	00	00	00	00	00	00	00	00	F8	00	00	00	FO	00	00				ø		ð
00	00	00	00	00	OF	80	00	00	OF	00	00	00	00	00	00			€			
00	00	F8	00	00	00	FO	00	00	00	00	00	00	00	OF	80		ø	ð			€
00	00	lF	00	00	00	00	00	00	00	00	F8	00	00	01	FO					ø	ð
00	00	00	00	00	00	00	OF	80	00	00	1F	00	00	00	00				€		
00	00	00	00	F8	00	00	00	FO	00	00	00	00	00	00	00		1	2	ð		



## Mobile

## Mobile01

安卓逆向,安卓不了解,反编译了下看了java无果。 自己云里雾里。朋友提示我用check1爆破apk的 so文件。 然后拿到: 1234567836275184

## Reverse

## **RE-answer to everything**

IDA打开,就发现这个可疑,当作flag提交无果,想到题目上的提示sha1, 我就把这段字符加密,#kdudpeh进行sha1加密,做flag提交不对 去掉了#,加密80ee2a3fe31da904c596d993f7f1de4827c1450a 套上flag { },正确得分。

## **RE-dig dig dig**

原题,TWCTF的dec dec dec。 思路主要是base64 加密 rot13加密 uudecode加密 我们拿flag就需要反着来一边。 文件在IDA打开,shift+f12打开字符串窗口,找到一串密文: @1DE!440S9W9,2T%Y07=%<W!Z.3!:1T%S2S-),7-\$/3T 然后uudecode》rot13》base64 就可拿到flag 密文: @1DE!440S9W9,2T%Y07=%<W!Z.3!:1T%S2S-),7-\$/3T Uudecode解 密文: FIAQD3gvLKAyAwEspz90ZGAsK3I1sD== ROT13解 密文: SVNDQ3tiYXN1NjRfcm90MTNfX3V1fQ== base64解 结果: ISCC{base64\_rot13\_uu}

当然,还有python脚本:

```
targetString = "@1DE!440S9W9,2T%Y07=%<W!Z.3!:1T%S2S-),7-$/3T "</pre>
decodedBits = ""
for c in targetString[1:]:
    decodedBits += bin(ord(c) - ord(' ') + 64)[-6:]
decodedText = ""
for i in range(0, len(decodedBits), 8):
    decodedText += chr(int(decodedBits[i:i+8], 2))
print(decodedText)
rot13edText = ""
for c in decodedText:
    if ord(c) >= ord('a') and ord(c) <= ord('z'):</pre>
        rot13edText += chr((ord(c) - ord('a') + 13) % 26 + ord('a'))
    elif ord(c) >= ord('A') and ord(c) <= ord('Z'):</pre>
        rot13edText += chr((ord(c) - ord('A') + 13) % 26 + ord('A'))
    else:
        rot13edText += c
print(rot13edText)
from base64 import b64decode
print(b64decode(rot13edText))
```

Rev04它被flag污染了

IDA也不好使了,发懵,然后看到其他re类的题。在linux输出下可视的字符。 咱也死马当活马医。开虚拟机,kail打开 命令:strings bad 输出了好大一堆,我就慢慢一行一行看,终于一串有大小写和数字的字符串引起我注意

dWdnYzovL1ZGUFAyMDE5e2hleV9mcnJ6Zl9lYmdncmFfanZndX1wZ3MucGJ6:

复制了,感觉像base64,解码一下。

base64 >
uggc://VFPP2019{hey\_frrzf\_ebggra\_jvgu}pgs.pbz

什么东西? 某种协议? 内容会是 { } 里的? 想到ctf惯用加密, Rot13试了下

# rot13> http://ISCC2019{url\_seems\_rotten\_with}ctf.com

整体提交

-----未完待更------

2019-05-15作者首发于看雪论坛 转载请注明作者[Xueqi]与出处!