攻防世界 misc高手进阶区



<u>~流氓没有兔</u> ● 于 2020-08-10 00:02:41 发布 ● 1895 ☆ 收藏 3
 分类专栏: CTF
 版权声明:本文为博主原创文章,遵循 CC 4.0 BY-SA 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/Bok_choy/article/details/107900844</u>
 版权



CTF 专栏收录该内容

2篇文章 0 订阅 订阅专栏

攻防世界 misc高手进阶区

每日打卡~~~

base64÷4

embarrass

神奇的Modbus

wireshark-1

pure_color

Aesop_secret

a_good_idea

Training-Stegano-1

can_has_stdio?

János-the-Ripper

Test-flag-please-ignore

Banmabanma

reverseMe

Hear-with-your-Eyes

What-is-this

MISCall

Reverse-it

something_in_image

打野

倒立屋

2017_Dating_in_Singapore

simple_transfer

Erik-Baleog-and-Olaf

hit-the-core

glance-50

Ditf

4-1

适合作为桌面

心仪的公司

每日打卡~~~

小萌新正在探索的路上~ 这里会记录自己做题的过程和方法,会持续更新的~ 争取每日打卡哒哒哒~

base64÷4

base64÷4 🔟 🛙 🖩	佳Writeup由admin提供
难度系数: ◆★1.0	
题目来源: 暂无	
题目描述: 暂无	
题目场景: 暂无	
题目附件: 附件1	https://blog.csdn.net/Bok_choy

由题目可以得出关键信息——64÷4=16,所以该加密方式应该是base16~

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H) 666C61677B45333342374644384133423834314341393639394544444241323442363041417D

编码

解码 清空

将文本复制后进行base64解密得到原文:

666C61677B45333342374644384133423834314341393639394544444241323442363041417D

https://blog.csdp.pat/Rok_chov

flag{E33B7FD8A3B841CA9699EDDBA24B60AA}

flag{E33B7FD8A3B841CA9699EDDBA24B60AA}

embarrass





压缩包解压得到:

54dcee33c07745f39f43a094a1b61dc
imisc_02.pcapng

方法一: 用记事本打开搜索flag

🥘 misc_02.pcapng - 记事本 X 文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H) 8?b鵣□??贋p夹T囷?钹4?□ ?t寫? ?韴F顯?ly枾H>B□O?□仙Y2aU□厝?z?jcZ□□U冽2iQ寪拖3 ?劀&p訾^?毂D嶁??`蒂墔T蠧5?\$ □筦圜□S? J!: ^ □ 鄔燆鵒ho珣□ +'迟0 ? □ □ ? ? □ ? □ ? □ ? □ ? □ ? □ ? □ . Do 🛧 Po \□ □ h□ □ p□ □ x□ □ €□ □' flag{Good b0y W3ll Done} □ □ Lncken □ □ A Normal.dotm □ o@ @□預 d⊡P doや/ Po^空v奃€oo両? ooo □b鶇 8? d □ ▲□ □ 捐瞗? ? \$?盉 ♠)瞮? E □躑茾 €□ ?d□?d□P P?lv晹□^垁□□ ? □□□ 8?b鶇碜 □ 筩□?禕 騆阾痢□鏤□y?D~鑩}{渋lt?j∵付E <€0□rU□甀:?`′渠焕;蚁□ yO坡5摵玹e誔N软j?b鲹U□I9蘸届□爔□釐鵺W?迆 ??□?□V□-u □?□印4?G瘞 嗫9] □ {岷瓴i □~??□黅廠?慣宱罖??□?V□X□霉?□□钐?裊翯L焁□瑇鋐9輯焔轇丂崌鮎挧? 汛 繟F<s懅锰 ?# N?Y蛺#1鵨□羽琍威2魕w E旎 ?狂D鉌 逛?歮o)劝□?"醇訂□6鏉av宾桃? ?□ M?厶烼?釜挶D+????^?5砓□繍*嘉SH? ?蔸F.□□□鹀□瓢OYZJS(18)N7秊鎡鷎□F绺鞙?□Z⊿瓉?焧詤jL ?診意呤@弛鍑泪€7絴蟅K檂CO4Ac*V#□忮樖B5懸□?jy笐n□铥莨%.%v%Q?□□t_ {控□6}e(□0 T□?□K改 叨欁j纱s剝Q♠總?Oiw□插♠S?帤纾肺揹 . 睸4n?U潛v3V? □}z列楊w烩颂(%??発?香他Zf醱袺<□&?T6屌論 B 欵膮耷埘T ?敋癄贎均]o?? a0癍?师? 焞 Za□?よ输8釦銝rY♠勢啶< 3E+s鑉2:{?;?? ??遅>Qセ□□\"i!跺雾g! 咊6□5z雨?□藭?*Q "N 矮?枡~€儴@?rZN 8?b鵣□??贋p夹T囷?钹4?□?t寪??韴F顯?ly枾H>B□O?□仙Y2aU□厝?z?jcZ□□U冽2iQ愙拖3?劀&p訾^?毂D嶁??`薔堆T蠧5?\$ □筦圛□S?」J! □ 鄔矯鴿h□珣□ + '迟0 ? □ □ ? ? □ ? □ ? □ ? □ ? □ ? □ ? □ . Do 🛧 Po \oohoopooxoo€oo' <mark>flag</mark>{Good b0y W3ll Done} □ □ Lncken □ □ ▲ Normal.dotm □ 8?b鵣 诵 ~b|C &P□€ 岕 \□ \ /P□□?? < ♠)瞮? PV? □□ E (*鰼 €□?括?括艀</p> 诵 ~blC-禤□€ d \□ \ /P□□?? < ♠)瞮? PV? □□ E (*鳣 €□?括?括艀</p> 诵 ~b|C:侾□€ r? \ □ \ /P□ 淁0? < ▲)瞮? PV? □□ E (*鴅 €□?括?括艀 诵 ~b|CH誠□€ dD \ □ \ /P□ 消0? < ♠)瞮? PV? □□ E (*鶣 €□?括?括艀 < 第 112406 行, 第 60 列 100% Unix (LF) https://blog.ANSI nat/

方法二: 放进虚拟机跑命令得到flag:

strings misc_02|grep flag

root@kali:~# cc	
root@kali:strings	misc_02.pcapng grep flag
GET /flag.php HTTP/1.1	
GET /flag.doc HTTP/1.1	
<pre>flag{Good_b0y_W3ll_Done}</pre>	
flag{Good_b0y_W3ll_Done}	
flag{Good_b0y_W3ll_Done}	
flag{Good_b0y_W3ll_Done}	

神奇的**Modbus**



由题目描述得到信息——提交格式为sctf{},大胆搜寻sctf{}:

\sim	· (<u>1</u>)	uumi(<u>x</u>) wu≪(<u>x</u>) imu	······································	-004-11/ 2004	NTT)						
	🔳 🖉 🔘 📜	🛅 🔀 🎑 🔍 🖛 I	🔸 🕾 不 👱 📃 📃	••••							
	立用显示过滤器	··· (Ctrl-/>									
	分组字节流	> 宽窄	~ □ 区分大小3	写 字符串	 ✓ sctf 					查找	
No.	Time	Source	Destination	Protoc Lengt	h Info						
	4 2480.2	192.168.130.1	192.168.130.1	Modb 10	3 Response: Trans:	1; Unit	: 1, Func:	3: Read Holding Registers			
	4 2480.2	192.168.130.1	192.168.130.1	TCP 5	4 62234 → 502 [ACK]	Seq=11353	Ack=16105 Win	=524032 Len=0			
	4 2480.9	192.168.130.1	239.255.255.2	SSDP 21	6 M-SEARCH * HTTP/1	.1					
	4 2484.5	192.168.130.1	192.168.130.1	Modb 6	6 Query: Trans:	1; Unit	: 1, Func:	1: Read Coils			
	4 2484.5	192.168.130.1	192.168.130.1	Modb 6	5 Response: Trans:	1; Unit	: 1, Func:	1: Read Coils			
	4 2484.6	192.168.130.1	192.168.130.1	TCP 5	4 62234 → 502 [ACK]	Seq=11365	Ack=16116 Win	=524032 Len=0			
	4 2484.9	192.168.130.1	192.168.130.1	SSH 10	6 Client: Encrypted	packet (le	n=52)				
	4 2484.9	192.168.130.1	192.168.130.1	TCP 6	0 22 → 61957 [ACK]	Seq=1 Ack=2	185 Win=251 L	en=0			
	4 2486.4	192.168.130.1	192.168.130.1	Modb 6	6 Query: Trans:	1; Unit	: 1, Func:	1: Read Coils			
	4 2486.4	192.168.130.1	192.168.130.1	Modb 6	7 Response: Trans:	1; Unit	: 1, Func:	1: Read Coils			
	4 2486.5	192.168.130.1	192.168.130.1	TCP 5	4 62234 → 502 [ACK]	Seq=11377	Ack=16129 Win	=524032 Len=0			
	4 2491.1	192.168.130.1	192.168.130.1	Modb 6	6 Query: Trans:	1; Unit	: 1, Func:	1: Read Coils			
	4 2491.1	192.168.130.1	192.168.130.1	Modb 6	5 Response: Trans:	1; Unit	: 1, Func:	1: Read Coils			
	4 2491.2	192.168.130.1	192.168.130.1	TCP 5	4 62234 → 502 [ACK]	Seq=11389	Ack=16140 Win	=524032 Len=0			
	4 2493.8	192.168.130.1	192.168.130.1	Modb 6	6 Query: Trans:	1; Unit	: 1, Func:	1: Read Coils			
	4 2493.8	192.168.130.1	192.168.130.1	Modb 6	7 Response: Trans:	1; Unit	: 1, Func:	1: Read Coils			
	4 2493.8	192.168.130.1	192.168.130.1	TCP 5	4 62234 → 502 [ACK]	Seq=11401	Ack=16153 Win	=524032 Len=0			
	4 2495.4	192.168.130.1	192.168.130.1	Modb 6	6 Query: Trans:	1; Unit	: 1, Func:	Read Input Registers			
	4 2495.4	192.168.130.1	192.168.130.1	Modb 11	9 Response: Trans:	1; Unit	: 1, Func:	4: Read Input Registers			
	4 2495.4	192.168.130.1	192.168.130.1	TCP 5	4 62234 → 502 [ACK]	Seq=11413	Ack=16218 Win	=524032 Len=0			
	∨ Register	3 (UINT16): 102									_
	Regist	er Number: 3									
	Regist	er Value (UINT1	5): 102								
	✓ Register	4 (UINT16): 123									
00	00 00 50 5	6 c0 00 08 00 0	c 29 02 23 7c 0	8 00 45 00	• PV • • • • • • + • • E						_
00	10 00 59 7	8 37 40 00 40 0	6 3c 93 c0 a8 8	2 82 c0 a8	·Yx7@·@· <·····						
00	20 82 01 0	01 f6 f3 1a ab 2	3 23 f9 fd ff 7	0 66 50 18	•••••# #•••pfP						
00	30 00 e5 3	89 <u>18 00 00 00</u> 0	1 00 00 00 2b 0	1 03 28 00	•••9••••••(•	·					
00	40 73 00 6	63 00 74 00 66 0	0 7b 00 45 00 6	1 00 73 00	s·c <mark>·t·f</mark> ·{·E·a·s·	· .					
00	50 79 00 5	if 00 4d 00 64 0	0 62 00 75 00 7	'3 00 7d 00	y·_·M·d· b·u·s·}·	· .					
00	00 00 00 0	00 00 00 00 00									

然后就得到flag啦,但是这道题很坑,因为你找到的flag中是mdbus没有o,而真正的flag是有o的,可能这就是"神奇的modbus 吧"hhh

i					The summer
1000	序号	解题思路	点赞数	上传者	操作
	1	xctf-wp	6	admin	

很详细的解题思路:
SCTF 2018:神奇的Modbus
[目标]
了解modbus协议
[环境]
无
[工具]
Wireshark
[分析过程]
在数据包中寻找flag就行,flag是明文形式存储。

工业设备消息传输使用modbus协议。所以我就采集了modbus的通信数据包。在这些数据传输中存在着flag。接替思路是这样的,利用该modbus读取设备的相关数据。首先读取数据的话modbus 有01,02,03,04功能码,01是读取线圈状态:取得一组逻辑线圈的当前状态(ON/OFF),02是读取输入状态:取得一组开关输入的当前状态(ON/OFF),03是读取保持寄存器:在一个或多个保持寄存器中取得当前的二进制值,04读取输入寄存器:在一个或多个输入寄存器中取得当前的二进制

wireshark输入modbus过滤追踪tcp流

	Wireshark ·	·追踪 TCP 流	(tcp.stream	eq 4) · modbus
--	-------------	-----------	-------------	------	------------

E ^ ^
•••••
74"
·····}·····
••••••••j•
······
••••••••••••••••••••••••••••••••••••••
······
••••••••••••••••••••••••••••••
K=:
6.
//////////-//////////
••••••
······]·····
%"
#c.t.f.
{.E.a.s.yM.d.b.u.s.}
3
1]
·····.N·····!
·····y····
·····
74EE
E.a.s.yM.d.b.u.s.}GG.
574
·····/····/·····/
N
\$
7 升组 2103。2.032 客户端 分组、2.032 服务器 分组、4.061 turn(s)、点击选择。
Entire conversation (58 kB)
查找:
滤掉此流 打印 Save as… 返回 Close Help
riiparratograsch.rarBok_ono/

sctf{Easy_Mdbus} (记得加一个o哦)

wireshark-1



 \times

_

题目附件: 附件1

"黑客通过wireshark抓到管理员登陆网站的一段流量包(管理员的密码即是答案)。" 由于是管理员登录网站然后被抓捕了信息,所以我们可以通过网站入手,追踪http流

	dianli_	ibctf_MIS	C_T10075_20150707	_wireshark.pcap									- 0	×
3	て件(F)	编辑(E) 礼	见图(V) 跳转(G) 捕获	茨(C) 分析(A) 统计(S) 电话(Y)	无线(W) 工具(T) 帮助(H)								
4		۰]	🗎 🕅 🏹 🔍 🖛	🗕 🕾 T 🛓 🗔 🗮		A 11								
	tcp.st	ream eq 2											F	+ ~ 💶 🛛
N	о. Т	ine	Source	Destination	Protoc L	ength Info								^
	- 13 2	.6718	192.168.1.102	115.231.236.1	TCP	66 22494 → 80 [SYN]] Seq=0 Win=8192 Len=0 MS	S=1460 WS=4 SAG	CK_PERM=1					
	18 2	.6845	115.231.236.1	192.168.1.102	TCP	66 80 → 22494 [SYN,	, ACK] Seq=0 Ack=1 Win=29	200 Len=0 MSS=:	1440 SACK_PERM=1 WS=10	24				
	19 2	.6845	192.168.1.102	115.231.236.1	TCP	54 22494 → 80 [ACK]] Seq=1 Ack=1 Win=66240 L	en=0						
_	> 20 2	.6849	192.168.1.102	115.231.236.1	HTTP	863 POST /user.php?a	action=login&do=login HTT	P/1.1 (appling	Aarly (Upmanly Darlasta)	Chal M				
	21 2	.6967	115.231.236.1	192.168.1.102	TCP	54 80 → 22494 [ACK]] Seq=1 Ack=810 Win=31744	Len=0	Mark/Unmark Packet(s)	Ctrl+M				
	22 2	.7399	115.231.236.1	192.168.1.102	TCP	990 80 → 22494 [PSH,	, ACK] Seq=1 Ack=810 Win=	31744 Len=93	ignore/Onignore Packet(s)	Ctrl+D				
	23 2	.7401	115.231.236.1	192.168.1.102	TCP	111 80 → 22494 [PSH,	, ACK] Seq=937 Ack=810 Wi	n=31744 Len=	以且/4X/月以且 111月参考 时间平移	Ctrl Shift T]			
	24 2	.7402	192.168.1.102	115.231.236.1	TCP	54 22494 → 80 [ACK]] Seq=810 Ack=994 Win=652	44 Len=0	公组注释	Ctrl+Alt+C				
	25 2	.7403	115.231.236.1	192.168.1.102	TCP	590 80 → 22494 [PSH,	, ACK] Seq=994 Ack=810 Wi	n=31744 Len=	/J>H/L##	Currante	1]			
*	- 26 2	.7403	115.231.236.1	192.168.1.102	HTTP	74 HTTP/1.1 200 OK	(text/html)		编辑解析的名称					
	27 2	.7403	192.168.1.102	115.231.236.1	TCP	54 22494 → 80 [ACK]] Seq=810 Ack=1550 Win=66	240 Len=0	作为过滤器应用	•				
	29 2	.7519	115.231.236.1	192.168.1.102	тср	74 [TCP Spurious Re	etransmission] 80 → 22494	[PSH, ACK]	Prepare as Filter	•	assembly erro	r, protocol TCP:	New fragm	l
	30 2	.7520	192.168.1.102	115.231.236.1	тср	66 [TCP Dup ACK 27#	#1] 22494 → 80 [ACK] Seq=	810 Ack=1550	对话过滤器	•				
1	48 5	.8630	192.168.1.102	115.231.236.1	HTTP	676 GET /user.php?ac	ction=login&email=flag HI	TP/1.1	对话着色	•				
	49 5	.8786	115.231.236.1	192.168.1.102	TCP	54 80 → 22494 [ACK]	Seq=1550 ACK=1432 Win=3	2/68 Len=0	SCTP	•				
	50 5	.9114	115.231.236.1	192.168.1.102	TCP	989 80 → 22494 [PSH,	, ACK] Seq=1550 ACK=1432	WIN=32768 Le	追踪流	•	TCP 流	Ctrl+Alt+Shift+T	1	
	51 5	.9127	115.231.236.1	192.168.1.102	TCP	297 80 → 22494 [PSH,	ACK] Seq=2485 ACK=1432	Win=32768 Le	复制	•	UDP 流	Ctrl+Alt+Shift+U		
	52 5	.9127	115.251.250.1	192.108.1.102	TCP	1404 80 → 22494 [PSH,	, ACK] SEY=2728 ACK=1432	2769 Lon=144	14.301-94-14-725		TLS 流	Ctrl+Alt+Shift+S		
	545	0127	115.251.250.1	102,108,1,102	тср	1494 80 7 22494 [ACK]	ACK] Sog=4405 Ack=1432	Lin=22769 Lo	例以自远坝 Dasada As		HTTP 流	Ctrl+Alt+Shift+H		
	54.5	. 9127	115.251.250.1	192.108.1.102	TCF	419 80 9 22494 [F30]	, ACK] SEQ-4455 ACK-1452	WIII-32708 LE	Decode As 本転空口目テム組のの		HTTP/2 Strea	.m		~
	Ac	cept-La	nguage: zh-CN,z	h;q=0.8,en-US;q	=0.5,en;o	q=0.3\r\n			113/1841132/3/35 11(VV)		QUIC Stream			^
	Ac	cept-Er	coding: gzip, o	leflate\r\n										
	Re	ferer:	http://www.wooy	/un.org/user.php	?action=1	login\r\n								~
<														>
4	0000	30 89 1	7 a6 82 df 74 d	e 2b 8d 1f 98 0	08 00 45	00 ·····t· +·····	E							^
	0010	03 51 6	1 15 40 00 40 0	6 <u>64 27</u> c0 a8 6	01 66 73	e7 ·Qa·@·@· •••	S*							
	2020	≗c 74 5 10 b0 d	7 ue 00 50 90 u d 44 aa aa 5a 4	a u9 55 41 88 4 f 52 54 20 2f 7	27 88 50	18 · LW · P · · · UU · · I 72 @ · · D · · PO ST / us	er.							
	3040	+0 00 u 2e 70 6	8 70 3f 61 63 7	4 69 6f 6e 3d 6	5 6f 67	69 .nhp?act ion=lo	a i							
	0050	5e 26 6	4 6f 3d 6c 6f 6	7 69 6e 20 48 5	54 54 50	2f n&do=log in HTT	P/							
(0060	31 2e 3	1 0d 0a 48 6f 7	3 74 3a 20 77 7	77 77 2e	77 1.1 Hos t: www	. W							
-	0070	5f 6f 7	9 75 6e 2e 6f 7	2 67 0d 0a 55 7	73 65 72	2d ooyun.or g. Use	r -							
1	080	11 67 6	5 6e 74 3a 20 4	d 6f 7a 69 6c 6	5c 61 2f	35 Agent: M ozilla,	/5							
1	0090	2e 30 2	0 28 57 69 6e 6	4 6f 77 73 20 4	1e 54 20	36 .0 (Wind ows NT	6							
	00a0	2e 31 3	b 20 57 4f 57 3	6 34 3b 20 72 7	76 3a 33	38 .1; WOW6 4; rv:	38							
	0000	2e 30 2	9 20 47 65 63 6	D 6T 2T 32 30 :	31 30 30	31 .0) Geck 0/2010	01							
(🕽 🗹 d	ianli_jbo	tf_HISC_T10075_201	50707_wireshark.pcap)					分组: 356 ·	已显示: 47 (13.2	%)	配置	: Defaul
		0	🖌 🛆 🖿		a	7					~ *	🧑 n) 🖛 nh. 🧲	22:31	
		10										maps 🌆	10g.2620/899	SOK_CIOY

描述又说了flag就是管理员密码,于是得到的这串字符就是flag了



<head>

<meta content="text/html; charset=utf-8" http-equiv="Content-Type"/>
<meta content="3; url=/user.php?action=login&email=flag" http-equiv="refresh"/>
<title> WooYun.org </title>
<meta content="80sec" name="author"/>
<meta content="http://www.wooyun.org/" name="copyright"/>
<meta <="" name="keywords" th=""/>
content=",wooyun,,web,
<meta <="" name="description" th=""/>
content="WooYun,,,,,,,,,
/link hpsf="/css/style_css" nel="stylesheet" type="text/css"/\

pure_color



下载附件后得到了一张png图片,打开后什么都没有,猜测是图片隐写

https://blog.csdn.net/Bok_choy

遇到图片隐写的题一般就会用到工具stegsolve 放进去跑一下

StegSolve 1.3 by Caesum File Analyse Help Blue plane 1 - 🗆 ×

Flag is	
	https://blog.csdn.net/Bok_choy



得到flag

Aesop_secret



同样,下载下来后解压是一个gif的动图,同样需要放进stegsolve里

<u></u>	legouve 1.5 by Caes	um			ш	\sim
File	Analyse Help					
Norm	File Format					
	Data Extract					
	Stereogram Solver					
	Frame Browser					
	Image Combiner					
			< >			

然后我们把图片的每一帧都保存下来

4			_		×
Frame : 1 of 9					,
-1 -					
	r				
	<	>itos:	Save	net/Bol	c choy
				1	

- 🔳 frame1.bmp
- 🔳 frame2.bmp
- 🔳 frame3.bmp
- 🔳 frame4.bmp
- 🔋 frame5.bmp
- 🔋 frame6.bmp
- 🔳 frame7.bmp
- 🔳 frame8.bmp
- frame9.bmp https://blog.csdn.net/Bok_choy

然后拼图得到 ISCC(这里是看到其他大佬拼图拼出来的,菜鸡我拼不出来.....ο(π----π)o) 图片隐写我们还会查看是否夹带着文件或者其他什么信息,于是扔进虚拟机里跑一下

root@kali:	# binwalk -e	e Aesop.gif
DECIMAL	HEXADECIMAL	DESCRIPTION
0 1004 <rdf:descrip< td=""><td>0x0 0x3EC ption rdf:about="</td><td>GIF image data, version "89a", 124 x 70 Unix path: /www.w3.org/1999/02/22-rdf-syntax-ns#"> " xmlns:xmpMM="http://ns.adobe.com/xap/1.0/mm/" xml</td></rdf:descrip<>	0x0 0x3EC ption rdf:about="	GIF image data, version "89a", 124 x 70 Unix path: /www.w3.org/1999/02/22-rdf-syntax-ns#"> " xmlns:xmpMM="http://ns.adobe.com/xap/1.0/mm/" xml
ns:stRef=		

没有发现什么信息,打开010editor,捕获到了一串密文

_								·																		
文件(<u>F</u>)	编辑	(<u>E</u>)	搜索	Ē(<u>S</u>)	视图	ষ্র(⊻)	格	i <u>⊐</u> t(C	<u>)</u>]	却本(<u>l)</u> t	莫板(<u>L)</u>	工具	(<u>T</u>)	窗[](⊻	<u>N</u>) 帮助(<u>H</u>)								
i 📄 🗸 🕑	- ~		1	1	1	1 1 1	‰ ∣	<u>h</u> [n e) (ι,	P 7	в 🖢	⊳ -)		A ₄		. Hex 🎇 ¶ 🔢 🔤 I	🖗 🏟	* 1	🍝 🔬					
启动	≜ eso	p. gi	if×																							
▼ 编辑	₩: ተ	-六讲	‡制(H) ~	/ 运	行朋	本 、	/ iž	- 行権	表板:	GIF	ht	\vee b													
-101-17			2		4	5				9	A	В	Ċ	D	Е	F		0123456789ABCDEF								
1830h:	FO	1E	8C	06	BD	90	49	67	7F	15	00	25	41	54	BD	18		ð.Œ.½.Iq%AT½.								
1840h:	60	5B	18	78	C6		04	03	Е6	в8	65	85	18	1E		97		`[.xÆæ.eÁ—								
1850h:	73	A9	31		1D	9A		E1		82	01	4A	F2	85		67		s©1šìá‡,.Jò…€q								
1860h:	6B	в0	38	5E	9C	F 8	91	93	C9	26	Е7	1D	24	DE	9B	49		k°8^œø``É&ç.\$₽>I								
1870h:	C6	A 8	E3	63	1A	84	37	DD		F7	7D				7C	8D		ƨãc."7Ýž÷}ÉW~ .								
1880h:	59	A6	A2	56	0D				E3	51		61	A1	84	7D	В5		Y¦¢V.0À`ãQua;"}µ								
1890h:	DC	40		D0	A5	D4	8A		4D	D9	Е0	04		4D	48	CE		Ü@.Đ¥ÔŠ~MÙà.ÃMHÎ								
18A0h:	07	26	14	E4	A 5	41	AF	26	1F	29	EΒ	$\mathbf{A}\mathbf{C}$	в4	D6	6A	EB		.&.ä¥A¯&.)ë¬′Öjë								
18B0h:	AD	в8	Eб	AA	EΒ	AE	BC	Fб	EA	EΒ	AF			2B	EC	в0		– "æªë®¼öêë¯À.+ì°								
18C0h:	C4	16	6B		В1	в2		04	00	3в	55	32	46	73	64	47		Ä.kì±²; <mark>U2FsdG</mark>								
18D0h:	56	6B	58	31	39	51	77	47	6B	63	67	44	30	66	54	6A		VkX19QwGkcgD0fTj								
18E0h:	5A	78	67	69	6A	52	7A	51	4 F	47	62	43	57	41	4C	68		ZxgijRzQOGbCWALh								
18F0h:	34	73	52	44	65	63	32	77	36	78	73	59	2F	75	78	35		4sRDec2w6xsY/ux5								
1900h:	33	56	75	6A	2F	41	4D	5A	42	44	4A	38	37	71	79	5A		3Vuj/AMZBDJ87qyZ								
1910h:	4C	35	6B	41	66	31	66	6D	41	48	34	$4 \mathrm{F}$	65	31	33	49		L5kAf1fmAH4Oe13I								
1920h:	75	34	33	35	62	66	52	42	75	5A	67	48	70	бE	52	бA		u435bfRBuZgHpnRj								
1930h:	54	42	6E	35	2B	78	73	44	48	4 F	4E	69	52	33	74	30		TBn5+xsDHONiR3t0								
1940h:	2B	4 F	61	38	79	47	2F	74	4 F	4B	4A	4D	4E	55	61	75		+Oa8yG/tOKJMNUau								
1950h:	65	64	76	4D	79	4E	34	76	34	51	4B	69	46	75	6E	77		edvMyN4v4QKiFunw								
1960h:	3D	ЗD	0D	A0														==								
																	Ļ									
																					http	s://blo	og.cs	dn.ne	t/Bok	_choy

看见尾端有两个==,以为是base64加密,但是试了下并没有得到结果,后来得到大佬的提示,看到题目,可以试一试aes加密,正好之前拼图得到了一串字符ISCC 将ISCC作为密匙解密得到新的字符串 于是再次尝试解密,得到flag

加密/解密	AES加密/解密	DES加密/解密	RC4加密/解密	Rabbit加密/解密	TripleDes加密/解密	MD5加/解密	Base64加/解密	Hash加/解密	JS 加密	JS 解密	
U2FsdGVkX1 HZriDDV0EQ	8OvTUIZubDnmvk /qt38Tw73tbQ==	2ISAkb8Jt4Zv6UW	pE7Xb43f8uzeFRU	KGMo6QaaNF	ISCC 密码是可选项,也就是可比 《 解密 加密,	U2F 87q Uau	sdGVkX19QwGkcg[yZL5kAf1fmAH4Oe1 edvMyN4v4QKiFunv	D0fTjZxgijRzQOGŁ 3lu435bfRBuZgHp v==	vCWALh4sRDe	c2w6xsY/ux53Vuj/AMZBE IONIR3t0+Oa8yG/tOKJM)J N
				在线项目	管理 - 使用Wrike	轻松搞定				https://blogtsdn.net/	Bok_cho



flag{DugUpADiamondADeepDarkMine}

解码链接https://www.sojson.com/encrypt_aes.html

a_good_idea

a_good_idea 🔟 🛚 🖶	圭Writeup由admin提供
难度系数:	
题目来源: 2019_NJUPT_CTF	
题目描述: 汤姆有个好主意	
题目场景: 暂无	
题目附件: 附件1	
	https://blog.csdn.net/Bok_choy

下载后解压得到misc文件夹,打开发现是一张Tom的表情包,用记事本打开发现最后藏了两张png图片

🥘 a_very_good_idea.jpg - 记事本					— C	- ×	
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)							
?*0S?0儦□3E +z8?4?鋴a??bl? □-	·??k4□?品爭箮8? L F垺鰣□ば	卑B沖h鰣';★痽臀鮠塌	高1A?鄧%FP?e`軲?	' 懊):xF□鶌□□7□玟	kcF 鍨鷿	^	•
-? ╡7┏炳□0Ү?讇???涮<祴:[J厡□牘瓶	育?硂擗?Pk?Ev? 孉蛿:?URkl绯	簙缟蓍?So&锝	X?鵽a忮以还 摚	縉? 0 眦貜泱O?齵n`	蓐□5^祳玓	IV偬oEJ	
'城忄o4€邬 榰ev '姨g祘虉朰鉻	好l艨 f鲳□浸'13踗設隔□赸P??	?NG+ 3疓柵牎?R排	蔡 QQ?搾?gJ墄5	W洵垀;?恂`耮?□籡			
ы□~b? 殧鋣R4盌愅狵軩口P鉆?骪^雪	鲠全?8?@臚□□f 弔□j茭鐈?#カ	嵐?B鉰汽 爔腛qGL/)質謚稄o釖醄□D?	熇□4?劫% 鰚[x X:+	讌??恀∨?	?By]□f?	
`众□蜨?~							
\$用4]?9片□□?錭a@#N&2ub蘥刐<倔	> 晌繷O徬䲠0K瑘 r□9f(?	□[壄]蹞矺□M□C嚼	塩Fst梂@谘?2□	3?)亹0?桠□Fe{?jJa	8曜□妕x?□	▶葬□悏	
₩用 瘇鐗P}?^z+□??o ??M澬n烸?	35 w`?濱崸r逴Z鮋佁\生??O	tRK#0k殪錁墷墄♠i	<u>⊾_□芈絿M□</u> e缨	眎v? f {6姮S?/?Db?ロ]?籆RfSF?的	単?虳繬	
O忏sZ?o綞□Q?Z~S坍轷} г □pV%繴贁	□v 汤樿霢忘U?T名?剂	舍?芨黛骅友塛篌	?鮪u蜉 互塛`le牾	谔叟蓷K硂姒?,糬\z;>	文瘲o??	□莎毓鎭	
?L彎ra貞J?惋♠^??II鲚 �����Vн.幀	∖瀨□讨g攮碳Y泏?髗i 苘藟椐	鳔熫□ ~殡劤□-??	? t阊卦簧蒞?蟝'	尬N 0踣	刑,鈡廩騳	鑠	
?鶇Vo勠l牮炾阯鮴i}3忇□(□婶K禵 I f1	膯IF€橶礹Ra%♠苮□□食 f糛3#	菍蒞癕冶u汥嶱蕡帚	?8□j□?□紺治箙撦	呃wo?(?€櫄ラ鞆?酣	?b_缑R	尫1R~	
/葉^肰 撤ks緼袞□尥途灇~1	1 文銒焴搊祧□撞諆陴?襟lr鲥	疑乎S?€偬 炽of芪	誤權n運C 攜=	?凂X,怯&爄莟Q馘Zt	?r摸?lG?悚	₹q7弸□	
-茠/醯禳□?Q□^iCO屃?硋{黏莏j鳒祆	所?P饥N軾R?□?□?□?5廸笄籏	摧					
%域K笗遣€侃.崦N珛梪□?賲□瓓D纮□□	D孼 兖0纣Q??6t匨:€s? 腟譱	o?蛅M硊kjv?杏務G	w ?G綥陜璟?f4□H	l夋瞍^/??@+??減<	?a嫣 7/□)	鸑l錝亞	
k□N#0鉰啌ゾ□ Sk枸@a? <r?jj?38?< td=""><td>3煚z(♠迟\$□Q?窅??巆鍽□?禖蒹</td><td>;%Xq&h?</td><th></th><td></td><td></td><td></td><td></td></r?jj?38?<>	3煚z(♠迟\$□Q?窅??巆鍽□?禖蒹	;%Xq&h?					
漥C旃c?□[r?□)e閔 P□棌?雍*?□瑝耣	,)?2壁d?!~□V7迭扽Z舖)瘲滴	§?蠪6*欜w+麪葪[釺	音[B懲D硇鼊B乪EU	賞因?胶)棪:{瘻諟□朝	莺"&Φ,0{□	??□?Z	
[@ ?髏'& 噸□?? 璱颿?p>/眷*.□#檛	??腤 1?竬璿?3Ⅰ閚?□??顾轧′	`?摻斄~鍝L糫勥?滦	s貲q?0肎??□□?□I	??€□♠;0□<甅?箶o?キ	#/1R□噹'!>	·蚂兌	
噵□□兦?铆碌香迪? 钽{摨nM几??根	雪?lv?0洕z□<□□???□[渳□锑7]	N霞炭{51 喵_鶶	呷^ 掼涃?]榶□	W?□`骳o鄠 妆贿-锡	IIII滶p%J?	毝□镭济	
□嚦?□泼?5kO託q 媫O R逰O鶋	?□kO O垃+						
挂*鯗祓漬?圪u舅斟□ 0c稻乩?0c漖楔□	▲P鶏7???读ヴi□襲□□丟蒝z	?e^;?砚□`O盃x	??R□€□偳?	'n?Nu囓?蚦昣5i ?3□	e4稤naaG	蝜R:蛍C	
苪??Au澍?N#罾儽tj燫疚硔观杆撦SAg	?繆飌`@Q:; w盩?5+? c?S?`鎀	p□录A賵鈎怤頿?鮹	盲€□闱ax?瘠邱X	o檑lo? oH弥oe??of	塗?□?\$2茥	? IEN	
²K□ □ □?rO慆B""							
澱□PK□ □ □ ?rO嚏)找? 若□ □ \$	p misc/to.png						
ᄓPK□ □ □?rO?[)愦 卮 □\$	k? misc/to_do.png						
1PK□□ □ □ s□ z?						~	,
<						>	
			第1行, 第1列	100% Unix (LF) https://	/blog ANSI ne	#/Bok_choy	

运行命令提取文件

binwalk -e a_very_good_idea.jpg

	< > < ☆ ±	文件夹	Q = = - • ×
root@kali: ~/se7en/misc	同 最近使田	0	•
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)			









这个信息让我误以为是关于图片高度的隐写,实际上并没有 当我们看见两张相似的图片是,我们可以作比较,看看有什么不同 同样,扔进stegsolve

🛓 S	tegSolve	1.3 by C	—		×
File	Analyse	Help			
Norma	File Form	nat		_	
	Data Ext	ract	-/		
	Storoogr	am Salvar		and the	



a_very_go	ood_idea.jpg
文件名(<u>N</u>):	to_do.png
文件类型(<u>T</u>):	Images 🗸
	打开 取消 https://blugrostinned Bok_choy



然后我们得到了这样一张图 当我们仔细观察这张图时会发现他可能是一张二维码 通过PS调节亮度就可以得到一张二维码 此外还可以用到Beyond Compare 软件,直接得到二维码



扫码得到flag NCTF{m1sc_1s_very_funny!!!}

Training-Stegano-1

Training-Stegano-1	Writeup由欧米伽安全团队 • SUN_提供
难度系数: • + 1.0	
题目来源: 暂无	
题目描述:这是我能想到的最基础的图片	▶ 隐写术
题目场景: 暂无	
题目附件: 附件1	https://blog.csdn.net/Bok_choy

用记事本打开就看见flag啦 steganol

🥘 2e5	e19744c	6449	1292	8eddc88	32f3b0b9.bmp	- 记事本
文件(F)	编辑(E)	格式	(O)	查看(V)	帮助(H)	
BMf	6 (0	Look what the hex-edit revealed: passwd:steganol

https://blog.csdn.net/Bok_choy

can_has_stdio?



下载后解压,用记事本打开



奇怪的加密 直接brainfuck就可以啦 brainfuck传送地址

将全文复制,选择brainfuck to text

115-3115-3115-311	50115501155	205-205-	2018-2018-2	NS-XNS-XNX	NX-XIX-X	XXIX	XIX-XIX-XI
	Home	Projects	Services	Personal	Shop		Search
	All the	hard work	(like actu	ally underst	tanding	how th	ose langua
	7 un erre i						in angua
	done by	Daniel Lo	orch and hi	s Brainfuck	interpret	er in PF	<u>1P</u>
	+++ +++ ++++ +++ ++++ +++ ++++ +++++ +++++ +++++ +++++ +++++ +++++ ++++++ +++++++ +++++++++++++++++++++++++++++++++++	 	++<<<<<<<<<>>>>>>> ++ +<<<<<<<<<<>>>>>>> ++ +<<<<<<<<<>>>>>> ++ +<<<<<<<<<>>>>>> ++ >>>>> +<<<<<<<<<<>>>>>>>>> +<<<<<<<<>>>>>>>>>>>> +<<<>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	• + •		
	Text to Ook	Text to short C	Ook! Ook! to Te	xt		\$	
	Text to Diali	Drainid					
	1			Sa Sherry		24 (11) 	

		· · · · · · · · · · · · · · · · · · ·
		1.000
		1.10
		1
Text to Ook!	Text to short Ook! Ook! to Text	
Toxt to Proin	uck Brainfuck to Taxt	

flag{esolangs_for_fun_and_profit}

János-the-Ripper

János-the-Ripper	123 最佳Writeup由sins7 • giun提供
难度系数: • ★ 1.0	
题目来源: tinyctf-2014	
题目描述: 暂无	
题目场景: 暂无	
题目附件: 附件1	https://blog.csdn.net/Bok_choy

记事本打开发现有flag.txt藏在最后面

			- 🗆 X
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) PK□□□□□ DD營W' □ □ flag.txt □□K?□頿	□儭W圵尶A狝?鰠兀 趕??棄D:PK□ □ □ □ □ □	DD響w' 。 。 。	flag.txtPK
<	유도 4 년 - 유도 4 년 1	1000/ Windows //PDI	
	第1行,第1列	100% Windows (CRLF	hod WRI usr. Box_cuoA

将文件提取出来

binwalk -e misc100





打开后有个文件夹,里面的flag.txt文件打开是空的,真正的flag在压缩包里,但是压缩包是经过加密的



"0.zip"需要密码

密码(P):



pted at



此时需要用到压缩包解密工具,题目是一种解密工具,但是我虚拟机没有安装,所以直接拷贝出来用其他工具,我用的是 ziperello,然后按照他的步骤依次来就是了

这里附加其他博主的关于John the Ripper使用的博文 传送门~

👔 Ziperello				_	
ziperello zip password	recovery tool		帮助) X Ŧ	退出
加密的 ZIP 压缩包			打开		
步骤 1	压缩包中加密的文件列表:				
1. 打开 ZIP 压缩包。Ziperello 将读 取并显示其中加密的文件。 2. 根据提示选择相应的文件及操 作。 3. 点击右侧箭头进入下一步	文件名	大小	加密方式		
C BAER	步骤 1 / 4:	选择ZIP文件			NEXT 💓
Ziperello ver. 2.1 版权所有 (C) 2008 FDRLab					

记得将后面的flag文件点上

👔 Ziperello				_	
ziperello zip password rec	overy tool		帮助	关于	退出
加密的 ZIP 压缩包 E:\GooleDown\0.zp 步骤 1 1. 打开 ZIP 压缩包。Zperello 将读 取并显示其中加密的文件。 2. 根据提示选择相应的文件及操 作。 3. 点击右侧箭头进入下一步	读取 zip 压缩包 包r 文f Lag 加密的文件 读取状态 1009 Ok	× 1 读取完成	 如何复 两点规 1.加密 2.密码 	送加密文件 ? 则: 济式相同 }相同	

C BAEK	步骤 1 / 4: 选择ZIP文件		
Ziperello ver. 2.1	版权所有 (C) 2008 FDRLab	https://blog.	esdmnei/Bok_enoy

这里有三种破解方式,可以根据自己的条件选择,因为没有什么关于密码的信息,所以我们直接暴力破解吧

7 Ziperello		- 🗆 X
ziperella zip pass	D word recovery	tool 表于 退出
──搜索模式		
○ 暴力破解		请您任选一种密码恢复方式 1. 暴力破解将尝试各种可能的密码组合直至找到正确密码
○ 字典破解		 2. 字典破解将尝试字典中所有单词的各种组合方式以解密 3. 基于模版的破解是一种特殊的暴力破解形式,依照用户定义的模版来生成密码组合
○ 基于模版的破解	*A5***RD	参阅帮助文件可获取详情 如果您稍后需要返回至此页面可点击左侧的箭头
		步骤 2 / 4: 选择搜索方式 NEXT >>>>
Ziperello ver. 2.1	版权所有 (C) 20	008 FDRLab https://biogresdmine//Bok_dhoy

7 Ziperello		_	
ziperello zip password recovery tool	帮助	关于	退出
字符类型: 固定字符集 _			<i>6</i>
字符集 ▼ 数字(0-9) □ 特殊符号(!@) ▼ 小写字母(a-z) □ 空格 ▼ 大写字母(A-Z) □ 所有印刷字符			
步骤 3			
1. 选择字符集类型: 固定或自设。			
定义可能存在于密码中的字符集。			
3. 设置最小及最大密码长度。			
4. 输入起始密码(非必需)			
Ziperello ver. 2.1 版权所有 (C) 2008 FDRLab		https://blog.cs	umnet/Bok_choy



猜测密码中存在的字符集,然后勾选,设置好密码长度即可

当前? 当前? 当前ì	密码长度 密码 速度 当前速度	步骤 4 准备就绪,请点击[开始] 按钮 注意:搜索进度条(%)及剩余时间字段显示的信息与当前的密码效验长度相关。破解 AES 算法加密的密码可能 耗时按长。
逝去时间: 00:00:00	0% 剩余时间:00:00:(开始 停止	
点击 [开始] 按钮开始破解密码		
Ziperello ver. 2.1 版机	步骤 4 / 4: 破解密码 7所有 (C) 2008 FDRLab	.Go NEXT >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

7 Ziperello					_	
ziperello zip passw	vord recovery tool			帮助	关于	退出
逝去时间: 00:00:00 10:56:07: 密码: "fish ".时间:	当前密码长度 当前密码 当前速度 16% 开始 1s	4 信息 × 密码: fish 确定	准备注意的耗时	;就绪,请点击 [开 ;: 搜索进度条 (% 密码效验长度相; 按长。	步骤 4 「始] 按钮 6) 及剩余时间字段 关。破解 AES 算法	显示的信息与当 加密的密码可能
		步骤 4 / 4: 破解密码.Go				NEXT
Ziperello ver. 2.1	版权所有 (C) 2008 FDRI	Lab			https://blog.cs	Imnel/Bok_choy

密码出来了! 密码是fish, 用密码解压后得到flag

🧻 flag.txt - 记事本	_	×
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)		
lag{ev3n::y0u::bru7us?!}		^

					\sim
<					>
	第1行, 第1列	100% Wi	ndows (CRLF)	UTF-8 et/Bok_	shoy
L					

flag{ev3n::y0u::bru7us?!}

Test-flag-please-ignore

Test-flag-please-ignore	最佳Writeup由B301 • dals提供
难度系数: ◆★1.0	
题目来源: tinyctf-2014	
题目描述:暂无	
题目场景: 暂无	
题目附件: 附件1	
	https://blog.csdn.net/Bok_choy

下载解压记事本打开

/////////////////////////////////////		
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)		
666c61677b68656c6c6f5f776f72	6c647d	

一来就base64解密,果然不可能这么简单o(T---T)o,仔细观察发现字符串没有出现 f 之后的字母了,所以应该是16进制加密 16进制解码传送门

加密或解密字符串长度不可以超过10M

					=
					-
16进制转字符	字符转16进制	测试用例	清空结果	复制结果	





下载解压打开是一张斑马的图片,斑马身上是条形码,条形码经过处理不可以扫出来 尝试过抠图,太难了,对小白十分不友好不友好不友好 大佬推荐了一个在线扫码工具,巨好用!! 3Q~ 传送门



https://blog.csdn.net/Bok_choy





Free Online Barcode Reader

To get such results using ClearImage SDK use TBR Code 103.

If your **business** application needs barcode recognition capabilities, email your technical questions to support@inliteresearch.com email your sales inquiries to sales@inliteresearch.com

	रभेर स्वार्थन स्वार्थन स्व			Neur File
File:	斑马斑马.pn	g		New File
Pages:	1		Barcodes	1
Barcode: 1	of 1	Type: Code39		Page 1 of 1
Length: 1	6	Rotation: none		
Module: 1	.6pix R	ectangle: {X=71,Y=93,Wid	h=410.Height=119}	
			1	Sec.
FLAG IS IE	INSHINE			B B

Barcode Reader Software Development Kit (SDK). Decode barcodes in C#, VB, Java, C\C++, Delphi, PHP and other languages.

Get ClearImage SDK

Barcode Director. Barcode scanner application renames, sorts and splits documents using barcode values.

Get Barcode Director

Barcode Reader Web Server with RESTful API. Client SDKs for JavaScript, .NET (C# or VB), Java, Node.js, PHP, Python or Ruby.

Web API Test Server

This site offers free limited demonstration. See terms of

FLAG IS TENSHINE 直接读取太简单了吧爽歪歪~~~

reverseMe



下载打开是一张翻转倒立的flag图片

 $f1ag\,\{4f7548f93c7bef1dc6a0542cf04e796e\}$

https://blog.csdn.net/Bok_choy

直接编辑就可以啦



Hear-with-your-Eyes



🥘 bf87ed29ac5a46d0aa433880dac5e6d8 - 记事本	- 🗆 X
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) sound.wav	0000644 0005772 0005772 00006565314 12411300030 012305
	~
<	>

看见sound.wav再结合题目就知道是音频隐写啦

在虚拟机中提取文件得到sound.wav的文件,然后用工具Audacity打开,选择频谱图得到flag

文件(E) 編編(E) 査看(M) 虚琐和(M) 逸琐末(E) 帮助(H) → 日 库 × ○ 在此处键入内容 ▼ <u> </u>	→ sound × × 文件(F) 編集(E) 透輝(S) 視題(N) 撤還(T) 生成(G) 效果(C) 分析(A) 工具(O) 解助(H) × × · 0 工 ★ / 5 → - i4 → 48 → 42 → · 点由开始监视 14 → - i8 → - i2 → 6 0 · × 0	-
□ 只約許書期, □ shert-kaka1 □ Ubuntu □ kali Linux □ 共享的違拟机 ② 大学(广) 编辑(E) 查看(V) 搜索(S) rootg(kal1:-# cd se7en rootg(kal1:-# se7en# bimatk -e		
Image: Construction of the system of the	 番執上部(0) 番執大部(回) 番執大部(回) 番執大部(回) 番執大部(回) 香枝(a) 本改約(直到)(意代) 透形(V) 透形(A) 受洗(B) 透形(V) 委戒書(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) · 或者型(5) ·	
	近形額色(W) 検式(F) 水 水(F本(E)) 水(F本(E)) 水(F本(E)) 水(F本(E)) 水(F本(E)) 水(F本(E)) 水(F本(E)) (00 时 00 分 00,000 秋▼ 00 时 00 分 00,000 秋▼ (00 H 00 分 00,000 秋▼ (00 H 00 分 00,000 H▼ (00 H 00 H 00,000 H▼ (00 H 00,000 H 00,000 H 00,000 H 00,000 H▼ (00 H 00,000 H 00,0000 H 00,0000 H 00,000 H 00,000 H 00,0000 H 00,0000 H 00,0000 H 00,000 H 00,0000 H 00,0000 H 00,0000 H 00,0000 H 00,000 H 00,0000 H 00,0000,0000 H 00,00000 H 00,0000 H 00,0000 H 00,0000 H 00,0000 H 00	
要將输入走向到该虚拟机,请在虚拟机内部单击或按 Ctrl+G。	다. 아 등 다. 아 등 다. 11:33 ^ ⓒ <i>@</i> 아 등 thes <mark>thesthese constants of the thesthese constants of the thest</mark>) Noy

🚗 sou	nd												_	C	ב	×
文件(F)	编辑(E)	选择(S)	视图(V)	擂录(N)	轨道(T)	生成(G)	效果(C)	分析(A)	工具(O)	帮助(H)						
			14			I-	*/	↓ 左 方	-54	-48 -42	——- 点击开始监视 34—	-18	-12	-6	0	

			•	★ ● ^左	-54 -48 -42	-36 -30	-24 -18	-12 -6	0
. •	•••••	• • • • • • • •	% 🖸			898	▶ <u></u>	9	+
MME	~ Ų :	麦克风 (Realtek	(R) Audio)	~ 2 (立体声) 录制声道 🛛 🗸	あ 声器 (Rea	ltek(R) Audio)		~
▼ 1.0	0 ₁₁ 0	1.0 2	0 3.0	4.0	5.0 6.0	7.0	8.0	9.0	10.0
× sound ▼ 8 野斎 独要 二 → + 芝 右 立体戸. 44100 Hz 32位 浮点 0 8		; e53	536676	57578	bd4da	10898	a8e2(d767	^
		;: e53	536676	57578	bd4da	10898	a8e2(d767	
	¢						ļ		~
项目采样率(Hz) 44100 ~	吸附到 关闭 ~	音频位置 00 时 00	分 00.000 秒	选区的起点和线 ▼ 00 时 00	^{&} 点 分 00.000 秒	▼ 00 时 00	分 00.000	~ D 秒▼	
已停止.							https://	blog.csdn.net	Bok_choy

What-is-this



同样用记事本打开看到jpg

e66ea8344f034964ba0b3cb9879996ff - 记事本	_		×
pic2.jpg	0000644 0005772 0005772 00000217270 1240333261	1 011775	C ^

▶□□□□□□□□□ ? □□□□□□ ? □ □□□□□ · □□□□ ? 4□ □ □ □□□□□ □□□□□□□□□□	
5壶□碬□M泄 ?嘳魘潪来'}桮?鶙?鏃O'纱旲1μ燮E貏賹I?鼡?★ 栵K竻谜R)?no	
v<璯哄?j? Le玶尤澵bu刌賄?l铈ta€穩殀狱?7蛆?犱唘帎□証鋜w缙T倻]□蛚呭f4□N?MT□t?芈? a铙唆<瓆□泠?r褐>毪[?驻#6厍猲z祃窫潕	
??€霿欄□瓍靱瀇 鮘璙9 ?Ar龐馱廞渠瘷しFW^?E??p締 螠l尘习踷□糸 靯? ?顇??刕r蘭+n狒t?匮?锞?堾宝〖?□5櫿蝥t秤7]秩暻U非? 蔛獴??缬	
重謱鷜Cl3告睉k仉 溁o 悈蠴?7櫔?9s *螥w,S┏)喘S轾□q~,ペ僳TU?]羡 叩酫蓑x??滙n赪鯲□哑6?7+屗喀□耛=?□貘螈)孉 閠鹡曝ユ鹭.:	
??穔^□? ~'?"縦co8??N?绕	
循3汗 樤?雇C抱q泮l蹄??n?坨4瑮[} WL5*摎W猈槳駐???? ,衶v9? 鰵?齓癘i鲣1} 醮旞屧鐅?峣廳Zh摃i 斨蝗仗>[♠?ブ缈k皁 ~蒉┿仏卒□LW~浚	
□ 襑~御?顷?軬 P8縚?惀 h(蝢S□m	
旟?鯸^啼戄 濢?琑搴? □y□29?[R O裉镮睨?元 □=勸嘩H#襺□嘒□籝Y# Q妤?y鶿??炜蕬G □]?G蒕j欹?> \^?劤{kS□?A弣NA	
実院?F⑷嶾□%駁 □?r?'□ 縥=n妔軷1闽 绱+□脔酅OF錟]F?H 騰鍏\? 壿} 4媪□饉]貗黍>*涠铠鶶彣 诇指p>=9沓<:殎撐錽m2□vy灌 □□S	
:□nnネ e?紽RF)0?kk菧蓷柒?J躻#?/T鞣豪? 9□? 磪4甚、?煬 5:抝□蝘□v渴朡+甇? U託□ II Q?娛nOh玕颹?n8J錤?=柚捰 O墷晏?6翬? 影	
?□d f?□価?聢=歫啍香岅萔+M:耔m@ 枇#?Z莫Y?]?舿阈隮?謪澌希咐褟{鸢□时<澌=3闥Rt雾o;@闣□1?n釼gD?冁牾G萍乐A]葭%K>	
??Q□GQM嚭F 庹餹銼Y犰r靓y□溲斛慄憬॥k?郇魝□□昤m8?訔o??*h靋嘲VUw 鎐龘况痗?淆D{挤 GB阼★h纊腱?晢0-舶閭c □c9?□□?绯赸	
臺q止酸w徙貢b?=P(砬B!	
开G?)?鯀□鈪?%虾摇f颱V?□嬘伪la蠭絋姃秴/5"Y□蕦H□槼×〃恝rD"□u蟛?瓘躣鯏<兪踅o?舷bN枷b KS WKWQ" 馆珎?鸴c褖?vii)秜□iiF?}	
昇伏X?"n ?h□楝安??9涵锬譢h`W閗t杰N濯?胗 蟚又薓4卄?総稟XF鯨榃p□倍璳葬]魄 酷喛r鎯蝨?颥跧O□圧膥;犍l嬷K□懶 `鲝o嚜6□+浵□{4h	
>悜? 攚'"O?5鶍?□裃?哀压音}" <痆包 p畝	
□?舝:緯U翗潨?C`?栂WOo阳 d闶E眆茨/<鞑??介2x飬(捥?□)/槢	
第1行,第1列 100% Unix (LF) ^{_https://blog} /ANSI ^{nel/Bok} _ch	

提取文件



跟上面那个Tom差不多啦,两张相似的图片我们就去作比较

🛓 S	tegSolve	1.3 by Cae	_		×
File	Analyse	Help			
Norm	File Form	nat	antarranno	7-00-101-00-00-	ESCONTRE I
	Data Ext	ract			
	Stereogr	am Solver			
	Frame B	rowser			
	Image Co	ombiner			



[_ pic1.jpg pic2.jpg	
	bic2.jpg	
:	mages 🔻	
	打开取消	
	< >	oy.



MISCall



没发现什么信息,也不知道文件是什么类型的 在虚拟机跑一下

file 文件名

得到文件类型是一个压缩包,将文件类型改为bzip2



解压压缩包

tar xvf 文件

root@kali tar xvf 321.bzip2	
ctf/	
ctf/flag.txt	
ctf/.git/	
ctf/.git/description	
ctf/.git/refs/	
ctf/.git/refs/heads/	
ctf/.git/refs/heads/master	
ctf/.git/refs/stash	
ctf/.git/refs/tags/	
ctf/.git/ORIG_HEAD	
ctf/.git/logs/	
ctf/.git/logs/refs/	
ctf/.git/logs/refs/heads/	
ctf/.git/logs/refs/heads/master	
ctf/.git/logs/refs/stash	https://blog.csdp.pet/Bok_cboy
ctf/.git/logs/HEAD	https://blog.coun.net/bok_cho





但是打开后什么也没有……然后没有然后了 观察目录,发现都有git文件目录,参考大佬的wp,得知考察 git文件泄露、git stash命令 https://blog.csdn.net/qq_46927150/article/details/105880794 查看git记录

git log



查看修改列表

git stash list

tog	
<pre>root@kali: /ctf# git log</pre>	
<pre>commit bea99b953bef6cc2f98ab59b10822bc42afe5abc (HEAD ->)</pre>	master)
Author: Linus Torvalds <torvalds@klaava.helsinki.fi></torvalds@klaava.helsinki.fi>	
Date: Thu Jul 24 21:16:59 2014 +0200	
Initial commit	
<pre>root@kali: /ctf# git stash list</pre>	
<pre>stash@{0}: WIP on master: bea99b9 Initial commit</pre>	
<pre>root@kali:/ctf#</pre>	

校验列表的存储文件,发现有文件改动s.py

git stash show

<pre>root@kali /ctf# git log</pre>
<pre>commit bea99b953bef6cc2f98ab59b10822bc42afe5abc (HEAD -> master)</pre>
Author: Linus Torvalds <torvalds@klaava.helsinki.fi></torvalds@klaava.helsinki.fi>
Date: Thu Jul 24 21:16:59 2014 +0200
Initial commit
root(kali
Toolgkati.
<pre>stash@{0}: WIP on master: bea99b9 Initial commit</pre>
<pre>root@kali /ctf# git stash show</pre>
flag.txt 25 ++++++++++++++++++++++++++++++++++
s.py 4 ++++
<pre>2 files changed, 28 insertions(+), 1 deletion(-)</pre>
root@kali /ctf# https://blog.csdn.net/Bok_choy

将s.py导出来运行



python s.py



NCN4dd992213ae6b76f27d7340f0dde1222888df4d3

Reverse-it



这里记事本不好用啦,打开010editor,查看文件头文件尾,对文件头文件尾熟悉的人可以看书华点啦~

	Q	1	-2	3	4	Ş	6	1	8	9	Ą	B	Ċ	Ď	Ē	F	0123456789ABCDEF
h:	9D	FF	70	0D	В6	DA	FC	93	72	63	28	22	22	BD	D2	18	.ÿp.¶Úü"rc(""½Ò.
h:	В4	25	D4	7B	8C	00	A1	AB	60	9E	Е7	07	BB	04	31	56	%Ô{Œ.;«`žç.».1V
h:	7B	5A	Dб	84	BB	D4	в9	7E	00	A 0	AB	6D	36	CD	31	C9	{ZÖ"»Ô¹~. «m6Í1É
h:	Α4	D1	01	2D	58	72	Α9	в5	ЗA	CF	C1	04	1B	51	47	00	¤ÑXr©µ:ÏÁQG.

1D60h:	A3	03	03	A3	13	03	03	23	64	00	14	46	F6	26	56	02	££.	#d	Föð	εV.
1D70h:	05	86	Fб	47	Fб	37	86	Fб	07	02	54	C6	56	D6	56	Eб	.töGö	ö7†ö.	.TÆVĊ	ŚVæ
1D80h:	47	37	10	00	00	00	84	00	00	00	10	00	00	00	84	00	G7			
1D90h:	00	00	00	00	00	00	A0	00	00	00	10	00	00	00	40	00				.0.
1DA0h:	96	78	C8	00	00	00	41	00	00	00	20	00	23	10	27	00	-xÈ	.A	#.	
1DB0h:	00	00	91	00	00	00	20	00	13	10	00	00	20	00	10	00				
1DC0h:	00	00	30	00	82	10	Аб	00	00	00	10	00	00	00	50	00	0.,			Ρ.
1 01	- 1	10	00	0.0	00	0.0	10	00	00	~ ~	50	00		10	0.0	0.0			_	

TDDOU:	ът	TO	20	00	00	00	TO	00	00	00	20	00	AT	TO	00	00	Ι.αΕ
1DE0h:	10	00	10	00	00	00	30	00	21	10	70	00	80	00	00	00	0.!.p.€
1DF0h:	A2	00	D4	D4	00	00	66	96	87	54	2D	00	1E	FF	00	00	¢.ÔÔf-‡Tÿ
1E00h;	84	00	84	00	10	10	10	00	64	94	64	A 4	01	00	0E	FF	"."d″d¤ÿ
1E10h:	8D	FF															.ÿ
L																	
																	https://blog.csdp.pet/Bok.chov
																	https://blog.osdii.ite/box_olog



转载 tcrefreshing	ybreeze 2018-08-15 14:04:38 💿 1547 🌪 收藏	2
JPEG (jpg),	文件头: FFD8FF	文件尾: FF D9
PNG (png),	文件头: 89504E47	文件尾: /
GIF (gif),	文件头: 47494638	文件尾: 0
Archive (zip),	文件头: 504B0304	https://blog.c文件尾词50.4/

再结合题目,倒过来,就很明显是16进制给倒着输入了,我们需要把他正回来才可以得到正常的图片

辑え	ל: ל	-六进	İ制(F	I) ~	运	行脚	本 ~	运	行模	板~	/					
																F 0123456789ABCDEF
n:	D3	46	96	02	22	FB	BB	FE	22	D3	Eб	96	76	56	26 02	02 ÓF−."û»þ"Óæ−vV&.
n:	47	56	в6	36	16	07	87	F3	C3		F2	03	E2	13	F2 07	07 GV16‡óÃ.ò.â.ò.
n:	16	87	F2	D6	Fб	36	E2	56	26	Fб	46	16	E2	37	E6 F2	F2 .‡òöö6âV&öF.â7æò
n:	F2	A3	07	47	47	86	09	90	1E	FF	00	00	00	00	A1 00	00 ò£.GG†ÿ;.
n:			10				40				8C				10 00	00@.0.02
n:	00	00	40	00	20	A 0	00	00	10	00	10	00	00	00	30 00	00@0.
n:	10					23	03	A3	13	43	A3	03	13	02	23 23	230#.£.C£##
n:	A 3	03	03	A3	13	03	03	23	64	00	14	46	Fб	26	56 02	02 ££#dFö&V.
n:	05	86	Fб	47	Fб	37	86	F6	07	02	54	C6	56	D6	56 E6	E6 .tögö7töTÆVÖVæ
n:	47	37	10	00	00	00	84		00	00	10	00	00	00	84 00	00 G7,
n:											10				40 00	00
n:	96	78	C8	00	00		41				20	00	23	10	27 00	00 -xÈA#.'.
n:			91						13	10					10 00	
n:	00	00	30	00	82	10	Aб	00	00	00	10	00	00	00	50 00	000.,.¦P.
n:	В1	10	26				10						A 1	10		00 ±.&P.
n:	10	00	10	00			30		21	10	70	00	80	00	00 00	000.!.p.€.
n:	A 2		D4	D4			66	96	87	54	2D		1E	FF		00 ¢.ôôf-≠エÿ 🤒 导出十六进制 🛛 🕹 🖌 🖌 🕹
n:	84		84		10	10	10		64	94	64	Α4	01		OE FF	FF "."
n:	8D	FF														· ý 导出文件(F):\1 /1 / >>
																导出类型 (T): 十六进制文本 (*. txt) ▼
																毎行ウ芽粉(P) 14
																- 🐨 选项 (Q)
																https://blog.csdn.net/Bok_ch

然后再倒序输出得到正确的16进制





输出的文件在010editor中导入16进制得到图片

SECCON{6in_tex7}

https://blog.csdn.net/Bok_choy

将他翻转得到flag啦

something_in_image



放进虚拟机运行命令

strings 文件|grep Flag

Flag.txtt.swx	
.Flag.txt.swpe	
.Flag.txt.swx	
.Flag.txt.swpe	
Flag.txtt.swx	
.Flag.txt.swpe	
Flag.txtt.swx	
.Flag.txt.swpe	
loit framework swx	
.Flag.txt.swpe	
.Flag.txt.swx	
.Flag.txt.swpe	
Flag.txtt.swx	
Flag.txt	
Flag.txt	
ridy.ixi	
Flog()	
Flag/vc/nl@fvic2k1+7Tl	
/mnt/test/Flag tyt	
Flag{}	
Flag{yc4pl0fvjs2k1t7T}	https://blog.csdn.net/Bok_choy

Flag{yc4pl0fvjs2k1t7T}

打野

打野 Left Writeup由admin提供
难度系数: •★★2.0
题目来源: 强网杯2019
题目描述: 菜你了解CTF圈的实时动态么? flag格式qwxf{}
题目场景:暂无
题目附件: 附件1
https://blog.csdn.net/Bok_choy

虚拟机跑命令没得到什么信息

root@kali	¥ binwalk -e	瞅啥.bmp
DECIMAL	HEXADECIMAL	DESCRIPTION
0 681325 ,I/-J/-J0.K0.H GF`HGaKJd	0x0 0xA656D <1/L1/L1/L1/L53P6	PC bitmap, Windows 3.x format,, 1224 x 689 x 24 Unix path: /M0.L1/M20N42P53Q0.K0.K0.K/-J/-J/-J/-J. 54Q64Q75R86S97T97T97T64Q75R86S:8U<:W> <y@>[@>[GF`GF`</y@>
684991	0xA73BF	Unix path: /M1/M0.L0.L0.L1/M31042P0.K0.K/-J.,I.,I
+H-+H-+H-+H-+H DC]DC]ED^	H-+H.,I.,I/-J/-J/	/-J31N31N42042053P64Q75R75R53P64Q86S:8U<:W> <y?=z@> </y?=z@>
1390195	0x153673	Unix path: /C/0D/0D/0D67K78L9:N; <p=>R?@TABVBCWGJYH</p=>
KZJM\LO^MP_NQ` 58G8;J:=L	NQ`NQ`KN]IL[HKZC	GJYFIXCFU@CR N<?N; M9 <k8;j69h69h58g58g 2a03b14c25d<="" td=""></k8;j69h69h58g58g>
1404862	0x156FBE	Unix path: /B20C31D-/A-/A-/A/1C13E24F35G46H57I7
9K: <n>@RACUCEW -0?-0?.1@</n>	NDFXDGUDGUEHVFIWC	GJXGJXFIWFIWFIWCFT?BP M:=K9<J69G47E03B/2A.1@-0?-0?</td
1415857	0x159AB1	Unix path: /D./D89N56K34I45J67L67L34I/0E.0B.0B.0B/
1C14C36E58F690 56@56@31D	58;I8;I9=H:>I=AL@	@DOCGREITHISFGQFGQHISKLVKLVFGQBCMABL@AK=>H; <f89c67#< td=""></f89c67#<>
1419881 968979:9:<:<=	0x15AA69 :=><>?=?@99?>>DFF	Unix path: /-/0-/0-/0-/0+/0691/123134245463578668968

stegsovle跑一下无果 运行该命令得到flag

zsteg 瞅啥.bmp

root@kali.	# zsteg 瞅啥.bmp
[?] 2 bytes of	
extradata:0	["\x00" repeated 2 times]
imagedata	text: ["\r" repeated 18 times]
b1,lsb,bY	<pre> <wbstego ,="" data="\x1Ef\xDE\x9</pre></td></tr><tr><td>E\xF6\xAE\xFA\x</td><td>CE\x86\x9E" even="false" ext="\x00\x8E\xEE" size="120,"></wbstego></pre>
bl,msb,bY	<pre> text: "qwxf{you_say_chick_beautiful?}"</pre>
b2,msb,bY	text: "i2,C8&k0."
b2,r,lsb,xY	text: "0000090000000000000000000000000000000
b2,g,msb,xY	text: ["U" repeated 22 times]
b2,b,lsb,xY	text: ["U" repeated 10 times]
b3,g,msb,xY	text: "V9XDR\\d@"
b4,r,lsb,xY	file: TIM image, Pixel at (4353,4112) Size=12850x8754
b4,g,lsb,xY	<pre> text: "3\"\"\"\"3###33##3#UDUEEEEEDDUETEDEDDUEEDTEEEUT#</pre>
i	
b4,g,msb,xY	<pre> text: "\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\</pre>
DD\"\"\"\"DDDDD	DDDDDDD*LD"
b4,b,lsb,xY	text: "gfffffvwgwfgwwfw" https://blog.codp.pot/Rok.chov
root@kali.	#

qwxf{you_say_chick_beautiful?}

倒立屋



root@kali:	¥ zsteg 倒立屋.png
imagedata	text: "\t\t\t\r\r\r\r\r\r"
bl,rgb,lsb,xy	text: "IsCc_2019"
b2,r,msb,xy	text: "t^y\t_{!i0"
b2,g,msb,xy	text: "UUUUUU`\rUUUU"
b2,b,msb,xy	text: "UUUUUU`\rUUUU"
b2,rgb,msb,xy	text: "jZ]?0]k0"
b4,r,lsb,xy	text: "#UwcDS#z"
b4,r,msb,xy	text: ["f" repeated 8 times]
b4,g,lsb,xy	text: "w17ffd2T3EB"
b4,g,msb,xy	text: "wwwwwwwwww3{"
b4,b,lsb,xy	text: "ffffwwww"
b4,b,msb,xy	<pre> text: "fffffffffffff","j"</pre>
b4,rgb,lsb,xy	text: "iVugVUUU6" https://blog.oodp.pot/Pok.obov
b4,bgr,lsb,xy	text: "YevWUUUV" https://biog.csdh.nevBok_choy

因为是倒立,所以要反过来,flag是9102_cCsl

2017_Dating_in_Singapore



直接上链接= =https://blog.csdn.net/qq_42016346/article/details/104234416

simple_transfer



用binwalk查看一下

root@kali	<pre># binwalk -e</pre>	111.pcap
DECIMAL	HEXADECIMAL	DESCRIPTION
339380 339454 340171 6380104 6385002	0x52DB4 0x52DFE 0x530CB 0x615A48 0x616D6A	PDF document, version: "1.5" Zlib compressed data, default compression Zlib compressed data, default compression Zlib compressed data, default compression Zlib compressed data, default compression

用foremost将PDF提取出来

HITB{b3d0e380e9c39352c667307d010775ca}

https://blog.csdn.net/Bok_choy

Erik-Baleog-and-Olaf

Erik-Baleog-and-Olaf **177** 最佳Writeup由HeliantHuS提供



第一步打开记事本,尾端有个链接,尝试进入却打不开

□□霽2?豯 #tEXthin: http://i.imgur.com/22k	Urzm.pngn >	IEND瓸`	?	
	第1行, 第1列	100%	Windows (CRLF)	1

查看文件类型

旦省人们大空	
oot@kali:~# oot@kali: # file stego100 tego100: PNG image data, 640 x 480, 8-bit/color RGB, m	non-interlaced
	< ☆ ☆ 全 ☆ 主文件夹 s en →
	◎ 最近使用
	企 主目录 7yue stego100
	- 単直
	□ 视频
	1 图片
	□ 文档
	④ 下载
	♬ 音乐
	💿 回收站 https://blog.csdn.net/Bok_choy

是png类型,于是后缀改为png,用工具打开,看到了一个二维码

🛓 StegSolve 1.3 by Caesum	_	×
File Analyse Help		
Blue plane 1		
an a start a st		



二维码扫码扫不上,看大佬说要变化通道找到合适的通道才能扫码,也有说找个清晰一点的二维码重新查看然后扫码,我太菜了 没找到......

查看隐写,还是这个链接,但是同样在虚拟机中也进不去

root@kali:	# zsteg stego100
meta hint	<pre> text: "http://i.imgur.com/22kUrzm.png"</pre>

出题人应该是想打开链接下载原图然后就可以把两个图作比较最后得到flag,可惜链接打不开了 复制一个flag: flag{#justdiffit}

hit-the-core

hit-the-core 101 最佳Writeup由sins7 • giun提供
难度系数: •★★2.0
题目来源: alexctf-2017
题目描述: 暂无
题目场景: 暂无
题目附件: 附件1
https://blog.csdn.net/Bok_choy

🦲 8de	eb5f0c2d	d841438	07b6175	5f58d6f3f.	core - 记事本

 \times

_

文件(F)	编辑(E)	格式(0) 查	看(V) 帮助	ታ(H)															
																			^
																		_	_
																			•
\]]脜	L峬鋟	Z?A瞿□□□	它 ?L?	塃銱?f?	H??9	? L?N	? 凓□	tR1纼?□	劒?4	? D	9#u窵□!	휇予H=	蓺;□喝	妻 1補	┋- 蕷;兟	;□H=	蓺;w	颒□□H埠	G⊐à
_		0/	~						~ ~	•	(-	(-	• -	_	_	^		<u> </u>	
U		70	0						2 :	0	(⊔	(⊔	o L		L	A			
3 6 36			<u>ሐንሐንሐን</u> ሐ			A A A	51 5 1517			<u>تھ تھ ت</u>	- (- (- (-)	***	<u>ፍናዱና</u> ዱና	<u>تەركەن</u>	<u></u>	*****	53 6 36		37
92929	e e e e e e e e e e e e e e e e e e e	<u>w</u> www		<u> </u>	l a la la la la	RR R R R			<u>R</u> RR	<u>n</u> na n		RR R R	C C C C C C C C C C C C C C C C C C C	<u> <u>n</u></u>				<u>nnnn</u>	
观观	观观观炎	观观观观	观观观视	观观观视	观观炎	观观观	观观观	观观观	观观观	烫烫烫	观观观观	观观观	观观观视	观观	观观观说		で 次 次	观观观炎	观
烫烫	反须须须	烫烫烫烫烫	页须须须	烫烫烫烫烫	烫烫烫烫	烫烫烫烫	应 须须须	反须须须	烫烫烫	烫烫烫	反须须须	烫烫烫	反须须须	烫烫	应 须须须	烫烫烫烫	杨汤汤	烫烫烫烫	烫
_			_																
		▲□?_ ≭	睘□							終□		0					紞□		~
<		▲□? ≭	≅ □							斜□		C					紞□		~

用010editor打开

/ 14	• •			44 / 1												
	1	2		4	5			8			B		D			0123456789ABCDEF
: (ĊC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (C	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	111111111111111
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (C	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (C	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (C	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (C	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (C	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (C	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	
: (CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	ÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌÌ
: (C	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	
: (C	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	00	2E	73	68	ÌÌÌÌÌÌÌÌÌÌÌsh
	74	72	74	61	62	00	6E	6F	74	65	30	00	6C	бF	61	strtab.note0.loa
	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	d
) (00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	https://blog.ende.end/Dataster
	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	nttps://blog.csdn.net/Bok_choy

无果

然后在kali里面查看是否有特殊的字符串

用string命令可以查看 UH



在众多字符串中看见这个有点像flag的,然后观察字符串,每五个字母为一个小组的话每个小组第四个都是大写字母,且遇到了 第四个为{,然后我们都以这样的规律,每五组的第四个符号提取出来,得到flag ALEXCTF{K33P_7H3_g00D_w0rk_up}

glance-50

glance-50 10 最佳Writeup由Kyrie • Kyr	rieKiki提供
难度系数: •★★2.0	
题目来源: mma-ctf-2nd-2016	
题目描述:暂无	
题目场景: 暂无	
题目附件: 附件1	
	https://blog.csdn.net/Bok_choy

https://blog.csdn.net/zz_Caleb/article/details/89490494



一个动态图片分解网站,直接出flag https://tu.sioe.cn/gj/fenjie/

Ditf



下载附件后提取出来,有一个png图片和一个压缩包,图片没有附带其他文件,也没有隐写,压缩包是加密的,尝试着暴力破解 没有破解出来。后来还是回到了图片本身,图片应该藏有压缩包密码,改写图片的高度将图片的高度调大一点,可以查看是否藏

Ò	1	2	3	4	5	Ģ	7	8	9	Ä	B	Ċ	Ď	Ę	Ę	0123456789ABCDEF
89	50	4E	47	0D	A0	1A	A0	00	00	00	0D	49	48	44	52	%PNGIHDR
00	00	03	9E	00	00	A 0	3C	08	02	00	00	00	38	16	5A	ž < 8.Z
34	00	00	00	09	70	48	59	73	00	00	0B	13	00	00	0B	4pHYs
13	01	00	9A	9C	18	00	00	06	D4	69	54	58	74	58	4D	šœÔiTXtXM
4C	3A	63	6F	6D	2E	61	64	6F	62	65	2E	78	6D	70	00	L:com.adobe.xmp.
00	00	00	00	3C	3F	78	70	61	63	6B	65	74	20	62	65	xpacket be</td
67	69	6E	ЗD	22	EF	BB	BF	22	20	69	64	3D	22	57	35	gin="ï»;" id="W5
4D	30	4D	70	43	65	68	69	48	7A	72	65	53	7A	4E	54	MOMpCehiHzreSzNT
63	7A	6B	63	39	64	22	3F	3E	20	3C	78	ЗA	78	6D	70	czkc9d"?> <x:xmp< td=""></x:xmp<>
6D	65	74	61	20	78	6D	6C	6E	73	ЗA	78	3D	22	61	64	meta xmlns:x="ad
6F	62	65	ЗA	6E	73	3A	6D	65	74	61	2F	22	20	78	ЗA	obe:ns:meta/" x:
78	6D	70	74	6B	ЗD	22	41	64	6F	62	65	20	58	4D	50	xmptk="Adobe XMP
20	43	бF	72	65	20	35	2E	36	2D	63	31	34	32	20	37	Core 5.6-c142 7
39	2E	31	36	30	39	32	34	2C	20	32	30	31	37	2F	30	9.160924, 2017/0
37	2F	31	33	2D	30	31	ЗA	30	36	ЗA	33	39	20	20	20	7/13-01:06:39
20	20	20	20	20	22	3E	20	3C	72	64	66	ЗA	52	44	46	"> <rdf:rdf< td=""></rdf:rdf<>
20	78	6D	6C	6E	73	3A	72	64	66	3D	22	68	74	74	70	xmlns:rdf="http
3A	2F	2F	77	77	77	2E	77	33	2E	6F	72	67	2F	31	39	https://bhog.cogn.net/Bdk9_choy
20	20	25	20	22	25	22	22	20	72	61	66	20	72	70	65	00/02/22 rdf cur



然后分析流量,尝试搜索flag,无果,搜索ctf,无果,搜索png(此处查看了其他大佬的wp),

分組字号流 ◇ (室存 → □ 区分大小写) 字符車 ◇ pmg No. Time Source Destination Protoc Length Info ^ No. Time Source Destination Protoc Length Info ^	查找	取
No. The Source Destination Protoclength Info		
7 30 400 433 300 431 1 403 400 31 50 TCD 4450 00 - 33307 [ACK] can 04040 Ack 733 Usin 34300 Lan 4404 [TCD comment of a nearcombled DDU]		
$7 20.409 123.200.131.1 192.108.31.59 TCP 1458.80 \rightarrow 33307 [ACK] Seq=81946 ACK=723 WIN=31366 Len=1464 [TCP segment of a reassembled PD0]$		
7 20.409 123.206.131.1 192.168.31.59 TCP 1458.80 → 33307 [ACK] Seq=83350 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.412 123.206.131.1 192.168.31.59 TCP 1458 80 → 33307 [ACK] Seq=84754 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.412 123.206.131.1 192.168.31.59 TCP 1458 80 → 33307 [ACK] Seq=86158 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.412 123.206.131.1 192.168.31.59 TCP 1458 80 → 33307 [ACK] Seq=87562 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.412 123.206.131.1 192.168.31.59 TCP 1458 80 → 33307 [ACK] Seq=88966 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.339 123.206.131.1 192.168.31.59 TCP 1458 80 → 33307 [ACK] Seq=8938 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.412 123.206.131.1 192.168.31.59 TCP 1458 80 → 33307 [ACK] Seq=90370 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.412 123.206.131.1 192.168.31.59 TCP 1458 80 → 33307 [ACK] Seq=91774 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.422 123.206.131.1 192.168.31.59 TCP 1458 80 → 33307 [ACK] Seq=93178 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.422 123.206.131.1 192.168.31.59 TCP 1458 80 → 33307 [ACK] Seq=94582 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.422 123.206.131.1 192.168.31.59 TCP 1458 80 → 33307 [ACK] Seq=95986 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		
7 20.422 123.206.131.1 192.168.31.59 TCP 1458.80 → 33307 [ACK] Seq=97390 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]		

7 20.422	123.206.131.1	192.168.31.59	TCP	1458 80 → 33307 [ACK] Seq=98794 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]	
1 29.454	123.206.131.1	192.168.31.59	TCP	54 80 → 33307 [FIN, ACK] Seq=683127 Ack=723 Win=31360 Len=0	
7 21.629	123.206.131.1	192.168.31.59	TCP	1458 80 → 33307 [PSH, ACK] Seq=275698 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]	
9 24.480	123.206.131.1	192.168.31.59	TCP	1458 80 \rightarrow 33307 [PSH, ACK] Seq=630910 Ack=723 Win=31360 Len=1404 [TCP segment of a reassembled PDU]	
6 20.304	123.206.131.1	192.168.31.59	TCP	66 80 → 33307 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1424 SACK_PERM=1 WS=128	
 6 20.305	192.168.31.59	123.206.131.1	HTTP	432 GET / HTTP/1.1	
6 20 224	402 400 34 50	400 000 404 4	UTTO	DOD CET UNITE ATTA A	

追踪http流,看见有一串特殊字符,明显的base64加密

Wieshark Jew http://www.icp.sueanieq.75/.bit.pcaping	×
<pre>GET / HTTP/1.1 Host: 123.206.131.120 Connection: keep-alive Upgrade-Insecure-Requests: 1 User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.99 Safari/537.36 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8 Accept-Encoding: gzip, deflate Accept-Language: zh-CN,zh;q=0.9</pre>	5
HTTP/1.1 200 OK Date: Sun, 01 Jul 2018 09:46:19 GMT Server: Apache/2.4.18 (Ubuntu) Last-Modified: Sun, 01 Jul 2018 09:45:26 GMT ETag: "c6-56fecf0c66879-gzip" Accept-Ranges: bytes Vary: Accept-Encoding Content-Encoding: gzip Content-Length: 177 Keep-Alive: timeout=5, max=100 Connection: Keep-Alive Content-Type: text/html	
<pre><html> <html> <head> <meta content="text/html; charset=utf-8" http-equiv="Content-Type"/> </head> <body> <img_src=" kiss.png"=""></img_src="> ZmxhZ3tPel80bmRfSGlyMF9sb3YzX0ZvcjN2ZXJ9 </body> </html> </html></pre>	
Hundright Hundr	; ~
整个对话(1256 bytes)	~
查找: 查找下一 滤掉此流 打印 Save as··· 返回	个(N) lp

然后进行解密得到flag

flag{Oz_4nd_Hir0_lov3_For3ver}

ZmxhZ3tPel80bmRfSGlyMF9sb3YzX0ZvcjN2ZXJ9

https://blog.csdn.net/Bok_choy

flag{Oz_4nd_Hir0_lov3_For3ver}

4-1



题目来源: WDCTF-2017	
题目描述: 暂无	
题目场景: 暂无	
题目附件: 附件1	
	https://blog.csdn.net/Bok_choy

用记事本打开看见有zip的压缩包

🥘 画风不一样的喵.png - 记事本	- 🗆 X
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)	
O3 tv?M=?藸ǜ题鶩u涂纯妨E?殈S施2欠版□□CRo僳?鴹]k麋騘(O虐\$9%?0Q8SR牞爯K戀-h	Av瀺ょ?輻B= Fw□厖□赕搊域A?醌賶i采#妝XY*J冪 ^
?3 岊緌3\俊;Pk\ZG訣+8?□咘RT.玻絃,厥藿鸹戮_粗□oOn焀?曙+d檰?8cc湫薸a?J璿曰=(8){X	@Z□8轅□姦䴓钍 楕?缅t<腧Y 煬鐕騇9^□ }繰栲衣
塵?Si甐^\揈□/v?x熇 [嚂, Ye+?颙□M鵸峅矟?A溃媃瓱□諧糧? >a□3w % H?竞??個	弛y"z□激???╤?3B0? 牞 g魎堊 □ WiF膉#?鬖e幾
魩O楓t沁?/}c-、单?F棲 嫭!?倆級Y馂?I励腊?它ro9邳=?兢勇oo01 ,bn@ HDooN?尅Xjw	HgA+^:璎?}6?r{ M♠?KY'□,S?□^賤□>爑忥柩U@牲
□豪块?铱:?&雭□3\s\$? ?#n??迈镦"* 顧?S;n\h簀? 稼N???B□垙Z□□x曭c?□愒銾[爻锟m□□糖	?锿唰瘹kG鍴? 蓺!憊遌奴;□□\$J<□鼐臺?€謄餺嬵^^
□销樗鲫8U□糩?筱□□榫{?□f硨□l霜(鄆貅偷澴;\03暡缱4♠□碯倵"保,??黪h缃Eqc娭鱞嬪!7件@	?h煄巙Ej 嗗蜎?+紵(F/.:挅呤???c7j?□ и _帗 L?騕植
铉?<8+z霮□ □J r皐铚卺嵣?oHI??‖喌a麠M}婞錨f□ #	
握□~;[@??罕芥)P黎錯□JB斜寸^??顭??缡睇m/%□FHk?蹏顑?]A鰱?t禴?验盂[趬郛』泇M.瞍	RS禥得痖2?镓偷? 荜 滣 趈?#q?闾汙问倫□?EB`p峾
?瓂b?Y□0%虮箛l□?%餔 ?桼朕k□ !! ?□+h□@€r粟?qo□nfb鉀8吚攅晽s□传黮博□□?娛b`X	'A僣?苶苮□闗-€{箘g?S逶I 惑 H□壩蚓戌T??€ ?F
s4諁钴d汁□f\鋖峼□?4處(/侎6焞□;箍A萗[?凒踃媤y?s~I靝Yf())躗娹5餫俞mRO?蝐\A 繂9??□	ı:]y@鍳ナ□\$隠晚,d縯~鴀-鴏/枉On霿□聜?D倔爭賜謳
彗sc絶 溜Nh?嚓 曼♠??秖摁b娛殭♠紊?n□/?頀o?□_?j?3啧蒈鉔?h涖漇;n?阴□燲摠y-癫k:□	保Q缤O餟志?!?譖E粪1} q翊(Sn傤鍂yf 脉? 穃xIN)
□?赐牽}?"癄 獽?E摇刾7□ 媪溝饴p□~,嶅)丗吷{??帮□? L@V2 ?拔 秎?□h薜?□□zv殳嫉鼬	"疅駆n□r4□?嬴麀惕??
h伢□愁贵2??p□s㖞'?麐,饬苍漤鷗 z肼婍m m□咺`福g3 □玕羿A~□?匽T屃詈?永 I b?-(??汤P□/_gX□檼踂c61h ?毵?韫H6?梬籷T-M□*□Ow"
玠1黣螪`/撣(?>0阅D悻r 朊□髻嚝□□JZ□y?材蓩?z摹F紶颠?□蒶U豏?睂S欲澱□!!扇溇萄1	'嘋 '7Kb竓ENe□縣 蟩?漭OQ閫N貊灏JiU a??IT□
[瞜?□镋熵□?, 注 層zO 役 蘳□ 苦 葦 L 姎 鳔W?f 奰 嫺 裒-□ 勘□? 旃 U 綷^ 釀] d 艂 賫? a? 静 蚍 g 矰	頁*?浩躲 鮻峒│?□望3□碎皠□ 殪彈洱'?□堛□
茕 (//j 鼎桦]T? 耸灸A9?饮峘嗟SM亶 屉□[^@?霃敢檰?/润□O□晦P8Q怓高;\谯□4	□蒟菎?□鋜\♠攴へ?萜岈?l8-]E7?鑟;鶿\$2□胝鹦(<b< td=""></b<>
L推鯫訜p\x瘙巪偡s碝 ?铸QW□?(貲?PK□ ? □ □ H□'K邟Ym\H□ a□ □ \$ day2.	ong
□ □ 兌□U1'?K?T1'?K?T1'?PK□ ? □ □ ?'Ku?纗? /? □ \$ 侣□ day1.png	
□ □ 2紅?'??h?'?r浣k0'?PK□□ ? □	
PK□?□□?'K讀□填N□\$ tips.txt	
□□齋奈3'?=1?'?=1?'?PK□ ?	
d⊡'K⊡p合?	
⇒ p day2's secret.zip	
	×
× 第1	> 行第1列 100% Windows (CRE的) OLANCENE/Pok-chov-
1.82	

经过提取文档,发现有一个tips

Although - <mark>\Al\A3</mark> -	two	days	doing	the	same	things,	but	day2	has	а	secret	than	day1		
						EL									



打开压缩包,里面有两张图,根据提示,第二张图是经过隐写的,两张图对比,可以看见day2的图文件大许多









一开始看见两个相似的图片,就会进行比较,看看是否有信息,但是这次没有









于是回到第二张图,查看有什么信息然后什么也没有查到…… 后来得知还有一种隐写方式是盲水印(`·_·`) https://blog.csdn.net/qq_43504939/article/details/100673360?utm_medium=distribute.pc_relevant.none-task-blogbaidulandingword-1&spm=1001.2101.3001.4242 wdflag{My_c4t_Ho}

适合作为桌面



打开图片觉得蛮怪异的,第一想法是反色,然后操作一下得到新的图,正常多了嘻嘻~(虽然还不知道有什么用)





之前有关于图片处理的方法都用了之后没发现什么信息,不敢相信,于是再来一遍,终于发现(之前太过于粗心没注意到)



扫描之后得到一段信息

03F30D0A79CB05586300000000000000000000000000000000730D000006400008400005A0000640100532802000006 3000000003000001600000430000073780000064010064020064030064040064050064060064070064030064080064 0900640A00640600640B00640A00640700640800640C00640C00640D00640E00640900640F006716007D00006410007D 0100781E007C0000445D16007D02007C01007400007C0200830100377D0100715500577C0100474864000053281100000 04E6966000000696C00000069610000006967000000697B0000069330000006938000000693500000069370000069300 0000069320000069340000006931000006965000000697D0000074000000028010000074030000063687228030000 00740300000737472740400000666C616774010000069280000000280000000730400000312E7079520300000001 00000730A00000001480106010D0114014E280100000520300000280000002800000028000000028000000073040000003 12E70797408000003C6D6F64756C653E01000000730000000

感觉像是16进制,想到之前有一道题reverse it,做题的过程中尝试过这种方法,于是我们导入16进制

·	<u> </u>						æ -			/ \-	• •	~ ~	۳ ٦	~ ~		CH 🗸	🔪 🔤 🖘 11 📰 🚥 ' 🛄 🚏 🚧 🛵 🌆 🔽 761
启动	16 * ×																
▼ 编辑	为: 十	-六进	<u>ŧ制(</u>)	H) ~	· 运	行脚	本、	∕ ìz	行祷	顿、	×						
	Ò					5	Ģ			9	Ä	B	Ċ	Ď	Ę		0123456789ABCDEF
0000h:	03	F3	0D	A 0	79	СВ	05	58	63	00	00	00	00	00	00	00	.óyË.Xc
0010h:	00	01	00	00	00	40	00	00	00	73	0D	00	00	00	64	00	@sd.
0020h:	00	84	00	00	5A	00	00	64	01	00	53	28	02	00	00	00	."ZdS(
0030h:	63	00	00	00	00	03	00	00	00	16	00	00	00	43	00	00	cC
0040h:	00	73	78	00	00	00	64	01	00	64	02	00	64	03	00	64	.sxddd
0050h:	04	00	64	05	00	64	06	00	64	07	00	64	03	00	64	80	dddd.
0060h:	00	64	09	00	64	A0	00	64	06	00	64	0B	00	64	0A	00	.dddd
0070h:	64	07	00	64	08	00	64	0C	00	64	0C	00	64	0D	00	64	ddddd
0080h:	0E	00	64	09	00	64	0F	00	67	16	00	7D	00	00	64	10	ddg}d.
0090h:	00	7D	01	00	78	1E	00	7C	00	00	44	5D	16	00	7D	02	.}x D]}.
00A0h:	00	7C	01	00	74	00	00	7C	02	00	83	01	00	37	7D	01	. t f7}.

00B0h:	00	71	55	00	57	7C	01	00	47	48	64	00	00	53	28	11	.qU.W GHdS(.
00C0h:	00	00	00	4E	69	66	00	00	00	69	6C	00	00	00	69	61	Nifilia
00D0h:	00	00	00	69	67	00	00	00	69	7B	00	00	00	69	33	00	igi{i3.
00E0h:	00	00	69	38	00	00	00	69	35	00	00	00	69	37	00	00	i8i5i7
00F0h:	00	69	30	00	00	00	69	32	00	00	00	69	34	00	00	00	.i0i2i4
0100h:	69	31	00	00	00	69	65	00	00	00	69	7D	00	00	00	74	i1iei}t
0110h:	00	00	00	00	28	01	00	00	00	74	03	00	00	00	63	68	(tch
0120h:	72	28	03	00	00	00	74	03	00	00	00	73	74	72	74	04	r(tstrt.
0130h:	00	00	00	66	6C	61	67	74	01	00	00	00	69	28	00	00	flagti(
0140h:	00	00	28	00	00	00	00	73	04	00	00	00	31	2E	70	79	(s1.py
0150h:	52	03	00	00	00	01	00	00	00	73	A0	00	00	00	00	01	Rs
0160h:	48	01	06	01	0D	01	14	01	4E	28	01	00	00	00	52	03	HN(R.
0170h:	00	00	00	28	00	00	00	00	28	00	00	00	00	28	00	00	((
0180h:	00	00	73	04	00	00	00	31	2E	70	79	74	08	00	00	00	s1.pyt
0190h:	3C	6D	6F	64	75	6C	65	3E	01	00	00	00	73	00	00	00	<module>s</module>
01A0h:	00																
																	https://blog.csdn.net/Bok_choy

看见文本后面有一个pyt,猜测是py文件,于是保存为py文件,但是但是.....不会做了 后面看大佬wp,说是要保存为pyc文件并解码(oh我是个'□'脑子)

det tlag(): -	
str = [
102,	
108,	
97,	
103,	
123,	
51,	
56,	
97,	
53,	
55,	
48,	
51,	
50,	
48,	
56,	
53,	
52,	
52,	
49,	
101,	
55,	
125]	
flag = ''	
for i in str:	
flag += chr(i)	
print flag	
def flag():	
str = [
102,	
108,	
97,	
103,	
123,	
51,	
56	

	97,	
	53,	
	55,	
	48,	
	51,	
	50,	
	48,	
	56,	
	53,	
	52,	
	52,	
	49,	
	101,	
	55,	
	125]	
fla		
for	i in str:	
	flag += chr(i)	
pri	t flag	

在线反汇编链接https://tool.lu/pyc/

运行之后得到	
flag{38a57032085441e7}	
Process finished with exit code 0	

flag{38a57032085441e7}

心仪的公司

心仪的公司 12 最佳Writeup由admin提供
难度系数: •★★2.0
题目来源: 世安杯
题目描述:暂无
题目场景:暂无
题目附件: 附件1
https://blog.csdn.net/Bok_choy

打开后直接筛选http,然后追踪http流,分析http流(永远在乱搞,从没有分析),在最后发现了一大段信息

🚄 Wireshark · 追踪 HTTP 流 (tcp.stream eq 3) · webshell.pcapng	 	×
X-Forwarded-For: 8.8.8.8		^
Content-Length: 3301		
Content-Type: image/png Last-Modified: Mon, 17 Aug 2015 08:45:31 GMT		

Accept-Ranges: bytes	
ETag: "57be4f13c9d8d01:0"	
Server: Microsoft-IIS/7.5	
X-Powered-By: ASP.NET	
Date: Sat. 29 Oct 2016 01:27:05 GMT	
PNG	
THDR 2 { SRT 0 PLTE)) DCC 3 9 YYY *3* II	
fff k MM V csc a S/// M 3 K	
\cdot	
2	
.y!)B) r #A# Ra:.!C!	
.cC[.]{!	
.IDAThX.w:.4H[.D%0.+B.%.\L.Ty" 3w.wccKK.5V	
·······	
.0VD	
=u2.H.(.A)6`=xbs6'K7Z".<2Y@.@~R.:)73Z./6?.}H8t.L].dT.	
3nN.@#.F.8.h4jC.0jv%u)z0q	
.50.(U.\$z=. L5.Tc.%6R#r .F`.>KTs{xBzVuc	
;.SBC52)bjYOj. <d:~&j6rpd7tg.< td=""><td></td></d:~&j6rpd7tg.<>	
Pr}f.z'."#C'7.)?f+	_
14A.0 =LB.H.N.,.qJiyB@,Q).t[z.ocD.>y.{.gY=K#OV~~& .Y.6.eZy\H(.%) .
\32G6 u.a?v.w:.D.l0^.8}8.0wz.;&KP}0wi.0xvM;.3A4.*x0 .RNg'	× .
分组 5218。7 客戶端 分组,6 服务器 分组,11 turn(s).点击选择。	
整个对话(13 kB)	I ~
查找。	- 个 (N)
	1 1
滤掉此流 打印 Save as… 返回 Close H	elp

然后用010打开,应该是png文件,但是文件头和png文件有出入,png文件头是89 50 4E 47,文件尾也是有出入的,png文件尾是 AE 42 60 82

0	L.	- 4	5	4	J	0	1	0	2	м	ы	6	D	Ľ	Е	0173420102WDCDUL	
2E	50	4E	47	0D	0A	2E	0D	0A	2E	2E	2E	0D	0 A	49	48	.PNGIH	
44	52	2E	2E	2E	2E	2E	2E	2E	32	2E	2E	2E	2E	2E	7B	DR{	
2E	2E	2E	2E	2E	2E	2E	73	42	49	54	2E	2E	2E	2E	2E	sBIT	
4 E	' 2E	2E	2E	2E	2E	50	4C	54	45	2E	2E	2E	2E	2E	2E	0PLTE	
2E	2E	2E	29	29	29	2E	2E	2E	2E	2E	2E	44	43	43	2E)))DCC.	
2E	2E	2E	2E	2E	2E	33	2E	2E	39	2E	2E	2E	2E	59	59	39YY	
59	2E	2E	2E	2E	2E	2E	2A	33	2A	2E	2E	2E	2E	2E	2E	Y*3*	
2E	75	2E	09	0D	0A	2E	2E	7B	2E	.u{.							
2E	2E	2E	2E	2E	2E	66	66	66	2E	2E	2E	2E	6B	2E	4D	fffk.M	
4E	4D	2E	59	2E	2E	2E	2E	73	73	73	2E	2E	2E	2E	65	MM.Yssse	
2E	2E	2E	2E	2E	2E	2E	2E	53	2E	28	3B	28	2E	2E	2E	s.(;(
2E	2E	2E	2E	2E	2E	2E	4D	2E	2E	2E	2E	2E	33	2E	2E	M3	
2E	2E	2E	2E	2E	2E	4B	2E	0D	A0	2E	0D	A 0	2E	2E	2E	K	
2E	79	2E	2E	2E	2E	2E	2E	2E	2E	29	2E	2E	2E	2E	2E	.y))	
2E	2E	2E	3F	2E	2E	2E	2E	2E	69	2E	2E	2E	2E	2E	2E	?i	
2E	33	33	33	2E	2E	0D	A0	2E	2E	2E	2E	70	2E	0D	A 0	.333p	
36	0D	0A	21	54	21	23	2C	23	09	71	09	09	2E	09	2E	6!T!#,#.q	
2E	2E	21	39	21	2E	2E	22	4A	!9!"J								
22	2E	51	2E	2E	32	2E	2E	58	2E	20	6C	20	2E	2E	2E	".Q2X. 1	
2E	2E	09	2E	2E	2E	27	4C	27	2E	09	2E	2E	2E	2E	2E	'L'	
42	0D	0A	2E	2D	2D	2E	В										
2E	2E	2E	7B	2E	2E	2E	2E	2E	2A	2E	21	34	21	2E	2E	{*!4!	
2E	21	21	21	2E	2E	2E	21	5A	21	2E	2E	2E	0D	A0	7A	.!!!!Z!z	
09	2E	2E	2E	2E	2E	2E	2E	6C	2E	2D	2E	2E	0D	A0	4A	JJ	
01	A 0	2E	79	2E	2E	21	2E	2E	2E	2E	29	42	29	09	3F	y!)B).?	
09	23	41	23	09	52	09	2E	2E	2E	2E	64	2E	2E	2E	2E	.#A#.Rd	
09	2E	09	2E	3A	2E	21	63	21	2E	2E	2E	2E	2E	2E	0D	:.!c!	
04	2E	0D	A0	2E	63	2E	2E	43	2E	2E	5B	2E	21	7в	21	cC[.!{!	
2E	2E	2E	2E	2E	2E	2E	2E	2E	68	2E	2E	2E	09	70	48	hpH	https://blog.csdn.net/Bok_chov
5.0	72	2 ਸ	25	25	25	25	25	25	25	25	25	25	75	25	25	Ve	interesting and an interesting and

尝试修改了下文件头和文件尾,失败了......弄了好久看了好多资料,都失败了可能被自己误导了,重新来一次,这次换追踪TCP流

Wireshark · 追踪 TCP 流 (tcp.stream eq 473) · webshell.pcapng	-		×
6CT.~4 S.e)=.U]			~
txioaL EM%7.][]a).0			
2.h	i		
9E S3e x @s ^ xvo Sk o 73g _ s x _ 2 3 w _ (J		
$\{1, 6, =, D^{2}, 4, 6, =, D^{2}, 4, 6, =, 5, 7, 4, 1, 1, 1, 2, 5, 7, 6, 1, 1, 1, 2, 5, 7, 7, 1, 1, 2, 1, 2, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,$			
{ # 7=71 ~ V2 V^ N \> ?'=dv \$ ^F [u"H 77 0 0 0 0 0 0			
$W = *R$ I II h0 1 $zym/(k_0 + 6 + 6 + 2R_0 + 8 yr)$ sc 0SPs · //			
adma (light b) a number of the solid line with a number of the state of the state of the state of the solid line solid li	1.		
	••••		
$(\cdot, \cdot, \cdot$			
	۸ ۸	A /	、
		^ ^	<u>``</u>
	A A	AA	
	AA.		
	•••••		•
	•	۸ ۸	/
		A A	
	Δ Δ	Δ /	`
		ΔΔ.	Δ
	Δ Δ	Δ Δ	·^
	Δ Δ	Δ 1	.
	Α	ΔΔ.	Δ
11400 2 2 11400 3 2 11400 3 2 114		A A	· ·
$\Delta P_{a} \rightarrow 1$ D L $\%$ $Af 7 A 2 eM R s$			
	s •		
3 E [6] KT"] $0 \pm f_{C}YE_{0}$ " [1] $f_{C} \wedge q = q$ ik mut $\sqrt{r} + v$ [6] $k + k^{*} = c(v_{C})$	2	•	
$\langle V \rangle V \wedge 0 \rangle = 100$ V/V V/V/V/V/V/V/V/V/V/V/V/V/V/V/V/V/V/	• 2 •		
μ λ	1	0	
H = D H = A + V + V + V + V > 2 $uk = b + b + b + b + b + b + b + b + b + b$	WC ^	0	
i D myOFG n K V B9k' f 8F 8 ^ R 0.4 V^ R 47 4A 0.K (M H4A			
T n G V		•	
F			
50. (.N`k.llullT.D			
3.1, b	I.v	L.:.	
6i.*.NLsq;.Nicfl4g:{ftop Is Waiting 4 y}			
			~
<u>ガ#1,0000,1 ★/ 例 ガ#,5 /8 か#,1 run(s), 別田延存。</u> 整个对话(7088 hytes)		法国	73 🔺
	F	ᄴᄖ	···· ·
		■14 P —	1.00
· 滤掉此流 打印 Save as… 返回 Cl	ose	He	lp clack

awsl这种题我永远是在靠运气



来康康大佬们的wp

心仪的公司
[目标]
流量分析
[环境]
无
[工具]
wireshark
[分析过程]
ip. addr == 192.168.1.0/24过滤内网ip
192.168.1.111大量访问外网,猜测是192.168.1.108做反向代理攻击
过滤192.168.1.108追踪tcp流得到flag
shell s@. (.Nk.UuUI.Doci.H.mjxt?.].b9.;n5f]\$.6.r0.bD,WMa.B+a <bh&< td=""> <.~8.hgvq;.N.wq;.N.vL;.6iNLsq;.Nicfl4g:{ftop_ls_Waiting_4_y}`</bh&<>

真好,我不会~菜的理直气壮555

使用 linux strings 命令查看	
对{进行匹配	
strings webshell.pcapng grep {	
function settable(tabler if (doing) { if (page) { function mssqlinfo(dbnar !sf{ QiJ{	name,doing,page) { ne) {
fl4g:{ftop_Is_Waiting_4_ !{6S	_y}
Je, {d <mark>root@kali</mark> :/mnt/hgfs/shau	res# strings webshell.pcapng grep {
得到 flag	

待到 flag fl4g:{ftop_ls_Waiting_4_y}

https://blog.csdn.net/Bok_choy

啊这啊这,我怎么没想到55555555