

# upload（难度：中）

发现上传的图片并导出

upload.pcap

文件(F) 编辑(E) 视图(V) 转换(C) 捕获(C) 分析(A) 统计(S) 电话(T) 无线(W) 工具(I) 帮助(H)

应用显示过滤器: ... <ctrl- />

No.	Time	Source	Destination	Protocol	Length	Info
4337	41.951768	127.0.0.1	127.0.0.1	TCP	84	52951 → 8080 [ACK] Seq=1 Ack=1 Win=2619648 Len=0
4338	41.952019	127.0.0.1	127.0.0.1	HTTP	534	GET http://127.0.0.1/upload/flag.jpg HTTP/1.1
4339	41.952054	127.0.0.1	127.0.0.1	TCP	84	8080 → 52951 [ACK] Seq=1 Ack=451 Win=2619648 Len=0
4340	41.952121	127.0.0.1	127.0.0.1	TCP	85	52379 → 52378 [PSH, ACK] Seq=30 Ack=1 Win=65535 Len=1
4341	41.952148	127.0.0.1	127.0.0.1	TCP	84	52378 → 52379 [ACK] Seq=1 Ack=31 Win=63060 Len=0
4342	41.952196	127.0.0.1	127.0.0.1	TCP	85	52379 → 52378 [PSH, ACK] Seq=31 Ack=1 Win=65535 Len=1
4343	41.952224	127.0.0.1	127.0.0.1	TCP	84	52378 → 52379 [ACK] Seq=1 Ack=32 Win=63059 Len=0
4344	41.963990	127.0.0.1	127.0.0.1	TCP	108	52952 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM=1
4345	41.964037	127.0.0.1	127.0.0.1	TCP	108	80 → 52952 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM=1
4346	41.964079	127.0.0.1	127.0.0.1	TCP	84	52952 → 80 [ACK] Seq=1 Ack=1 Win=2619648 Len=0
4347	41.964151	127.0.0.1	127.0.0.1	HTTP	513	GET /upload/flag.jpg HTTP/1.1
4348	41.964180	127.0.0.1	127.0.0.1	TCP	84	80 → 52952 [ACK] Seq=1 Ack=430 Win=2619648 Len=0
4349	41.964766	127.0.0.1	127.0.0.1	HTTP	1108	HTTP/1.1 200 OK (image/jpeg)
4350	41.964799	127.0.0.1	127.0.0.1	TCP	84	52952 → 80 [ACK] Seq=430 Ack=1025 Win=2618624 Len=0
4351	41.964915	127.0.0.1	127.0.0.1	TCP	84	80 → 52952 [FIN, ACK] Seq=1025 Ack=430 Win=2619648 Len=0
4352	41.964942	127.0.0.1	127.0.0.1	TCP	84	52952 → 80 [ACK] Seq=430 Ack=1026 Win=2618624 Len=0

> Frame 4349: 1108 bytes on wire (8864 bits), 1068 bytes captured (8544 bits)  
> Null/Loopback  
> Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1  
> Transmission Control Protocol, Src Port: 80, Dst Port: 52952, Seq: 1, Ack: 430, Len: 1024  
> Hypertext Transfer Protocol  
> Media Type

0000 02 00 00 00 45 00 04 28 9f cd 40 00 00 06 00 00 ...E...{ ..@.....  
0010 7f 00 00 01 7f 00 00 01 00 50 ce d8 e9 9b c6 7b .....P.....{  
0020 91 fc 95 4f 50 18 27 f9 77 02 00 00 48 54 54 50 ...OP...u...HTTP  
0030 2f 31 2e 31 20 32 30 30 20 4f 4b 0d 0a 44 61 74 /1.1 200 OK--Dat

Wireshark · 导出 · HTTP 对象列表

分组	主机名	内容类型	大小	文件名
2586	127.0.0.1	multipart/form-data	1026 bytes	upload.php
4259	127.0.0.1	multipart/form-data	1008 bytes	upload.php
4261	127.0.0.1	text/html	883 bytes	upload.php
4265	127.0.0.1	text/html	883 bytes	upload.php
4349	127.0.0.1	image/jpeg	722 bytes	flag.jpg
4353	127.0.0.1	image/jpeg	722 bytes	flag.jpg

文本过滤器:

Save Save All Close Help

图片实际上是一个压缩包，改文件后缀名。

MEY INDEX	flagjpg																	ANS	ASCII
	Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
	00000000	50	4B	03	04	14	00	01	00	00	00	7A	64	BF	50	63	8E	PK	zd&Pc
	00000010	F1	1B	12	00	00	06	00	00	00	05	00	00	00	31	2E	FF	%	1.
	00000020	74	78	74	78	08	24	73	23	5A	AB	FF	67	1E	48	7B	FD	txtx \$S#Z<yg H\y	
	00000030	4E	00	51	52	55	50	4B	03	04	14	00	01	00	00	00	85	N QRUPK	I
	00000040	64	BF	50	AA	4F	21	05	12	00	00	00	06	00	00	00	05	d&P&0!	
	00000050	00	00	00	32	2E	74	78	74	67	9F	C5	AE	45	1E	FE	EC	2.txtg!A&E pi	
	00000060	5C	54	18	7B	A1	35	67	FF	14	E1	50	4B	03	04	14	00	\T {iSgy &PK	
	00000070	01	00	00	00	95	64	BF	50	A5	3D	6C	A9	12	00	00	00	!d&P%~1@	
	00000080	06	00	00	00	05	00	00	00	33	2E	74	78	74	10	4C	96	3.txt L!	
	00000090	54	49	CF	B5	55	03	DB	D7	6A	24	27	75	8E	54	C6	50	TIiPu Uxj\$~u!t&P	
	000000A0	4B	03	04	14	00	01	00	00	00	B5	64	BF	50	D3	B5	7C	K	md&P&u
	000000B0	A5	12	00	00	00	06	00	00	00	05	00	00	00	34	2E	74	%	4.t
	000000C0	78	74	1E	29	32	54	52	91	D4	3A	FF	E0	15	DE	A4	99	xt )2TR~O:ya ~x!	
	000000D0	FB	9F	9C	78	50	4B	03	04	14	00	01	00	00	00	CA	64	u!lxPK	Ed
	000000E0	BF	50	D5	AC	E9	61	12	00	00	00	06	00	00	00	05	00	zP&~éa	
	000000F0	00	00	35	2E	74	78	74	67	08	12	EA	13	7E	0F	84	23	5.txtg é ~ !#	
	00000100	09	49	1E	97	A6	C8	A1	AA	08	50	4B	01	02	3F	00	14	I ! É!a PK ?	
	00000110	00	01	00	00	00	7A	64	BF	50	63	8E	F1	1B	12	00	00	zd&Pc!ñ	
	00000120	00	06	00	00	00	05	00	24	00	00	00	00	00	00	00	20	\$	
	00000130	00	00	00	00	00	00	00	31	2E	74	78	74	0A	00	20	00	1.txt	
	00000140	00	00	00	00	01	00	18	00	0A	57	4E	F6	04	37	D6	01	WNö 7ö	
	00000150	0C	43	E7	F6	04	37	D6	01	40	1C	30	05	00	37	D6	01	Cçö 7ö @ 0 7ö	
	00000160	50	4B	01	02	3F	00	14	00	01	00	00	00	85	64	BF	50	PK ?	!d&P

根据压缩包文件大小猜测 crc 爆破

```
G:\CRC32\crc32-master>python crc32.py reverse 0x1bf18e63
4 bytes: {0x3b, 0x24, 0xe9, 0x6a}
verification checksum: 0x1bf18e63 (OK)
alternative: 7worVu (OK)
alternative: 8x1sqb (OK)
alternative: ARXx7G (OK)
alternative: DwZXDq (OK)
alternative: Ie_WS7 (OK)
alternative: MaBVRT (OK)
alternative: R_wEiX (OK)
alternative: S_6trA (OK)
alternative: TF1JXj (OK)
alternative: W7XXwf (OK)
alternative: XTutT5 (OK)
alternative: Y8GhKh (OK)
alternative: can_U_ (OK)
alternative: ijlhn2 (OK)
alternative: jwvQCz (OK)
alternative: mnqoiQ (OK)
alternative: oRtq0L (OK)
alternative: uICBxv (OK)
alternative: w8kaLc (OK)
alternative: y7tQpm (OK)
```

```
G:\CRC32\crc32-master>python crc32.py reverse 0x05214faa
4 bytes: {0x4c, 0x63, 0x4f, 0x32}
verification checksum: 0x05214faa (OK)
alternative: 22Bt1V (OK)
alternative: 5fhwvu (OK)
alternative: 66_u05 (OK)
alternative: 8tmxa3 (OK)
alternative: KtXRs7 (OK)
alternative: OpESrT (OK)
alternative: QN1qRA (OK)
alternative: VW60xj (OK)
alternative: ZErqt5 (OK)
alternative: apiZu_ (OK)
alternative: dhz6n1 (OK)
alternative: fH0t68 (OK)
alternative: find_t (OK)
alternative: hfqlcz (OK)
alternative: if0exc (OK)
alternative: oc96HE (OK)
alternative: rOkZFH (OK)
alternative: wXDCXv (OK)
alternative: zJAHO0 (OK)
```

```
G:\CRC32\crc32-master>python crc32.py reverse 0xa96c3da5
4 bytes: {0xe1, 0x0f, 0x49, 0xb4}
verification checksum: 0xa96c3da5 (OK)
alternative: V8R_sl (OK)
alternative: fjyYPk (OK)
alternative: gj8hKr (OK)
alternative: gvw4Jf (OK)
alternative: hefile (OK)
alternative: jYcw5x (OK)
alternative: t6u4yq (OK)
alternative: v7afH4 (OK)
alternative: yTLJkg (OK)
```

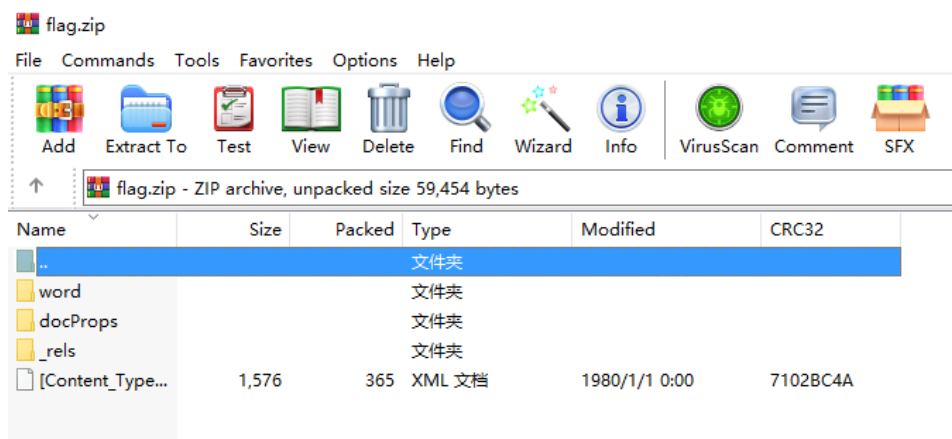
```
G:\CRC32\crc32-master>python crc32.py reverse 0xa57cb5d3
4 bytes: {0xa1, 0xa3, 0x2a, 0xb8}
verification checksum: 0xa57cb5d3 (OK)
alternative: 8olqa3 (OK)
alternative: NwJ7hY (OK)
alternative: OkDZrT (OK)
alternative: PUqIIX (OK)
alternative: QU0xRA (OK)
alternative: VL7Fxf (OK)
alternative: WLvwcs (OK)
alternative: Yb7W61 (OK)
alternative: akhSu (OK)
alternative: from_t (OK)
alternative: nd6KKH (OK)
alternative: odwcIQ (OK)
alternative: u2mmlc (OK)
alternative: wCENXv (OK)
```

```
G:\CRC32\crc32-master>python crc32.py reverse 0x61e9acd5
4 bytes: {0x23, 0x3e, 0x0c, 0x05}
verification checksum: 0x61e9acd5 (OK)
alternative: 5zHv4V (OK)
alternative: 7FMhmK (OK)
alternative: 9IRXQE (OK)
alternative: EF9sdV (OK)
alternative: F7PaKZ (OK)
alternative: H80QwT (OK)
alternative: JIgrCA (OK)
alternative: O1eR0w (OK)
alternative: RnU_Rf (OK)
alternative: UwRaxM (OK)
alternative: cPLEna (OK)
alternative: f8cXp_ (OK)
alternative: jFTKxD (OK)
alternative: 13aiM2 (OK)
alternative: m_SuRo (OK)
alternative: raffic (OK)
alternative: tdo5YE (OK)
alternative: uxaXCH (OK)
```

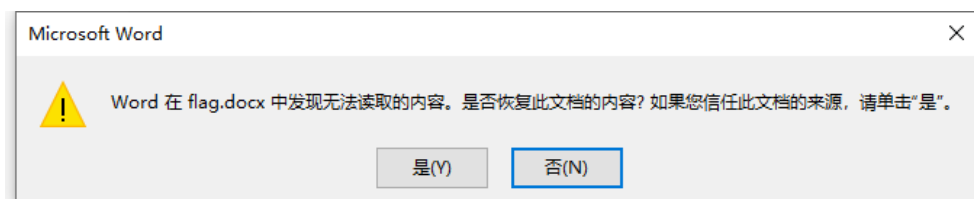
flag: Dozerctf{can\_U\_find\_thefilefrom\_traffic}

## word（难度：简单）

打开文件发现是个 word 文档



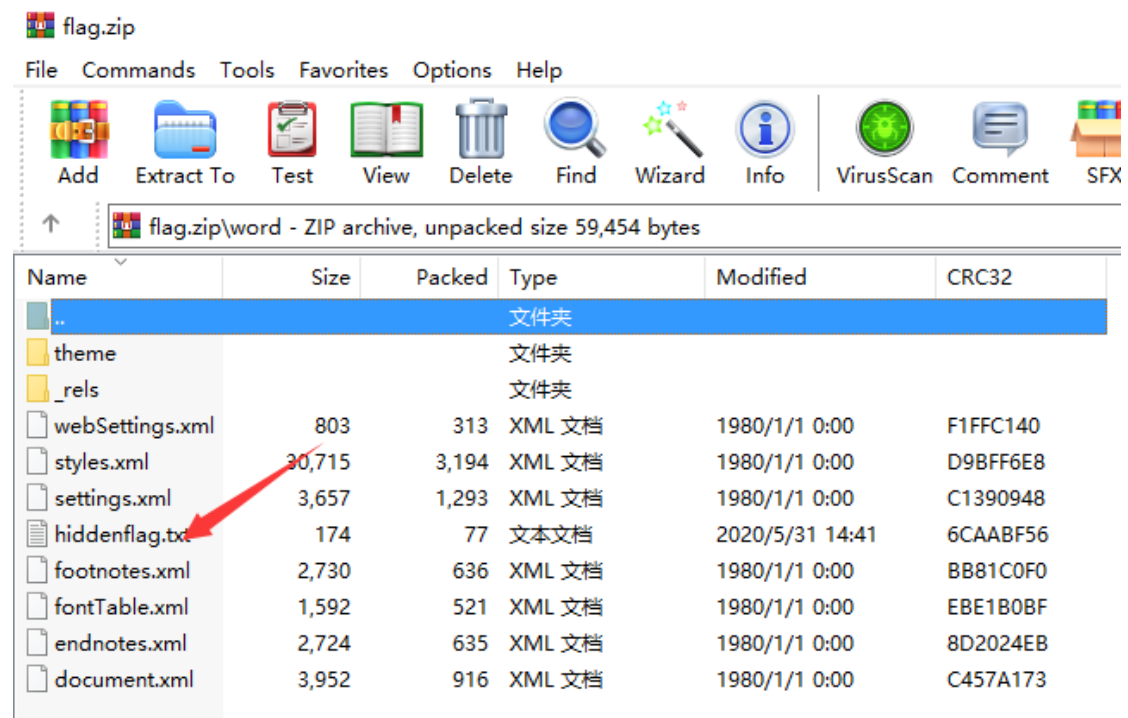
改后缀后打开发现隐藏文字且有报错



flag is here

it is not the flag you want

再次仔细检查 zip 发现 flag 文件



Unicode 解码

&#58;&#122;&#101;&#14;&#99;&#116;&#102;&#123;&#68;&#111;&#95;&#85;&#95;&#117;&#115;&#101;&#77;&#83;&#87;&#79;&#82;&#68;&#95;&#111;&#114;&#95;&#87;&#80;&#83;&#63;&#125;

Dozerctf{Do\_U\_useMSWORD\_or\_WPS?}

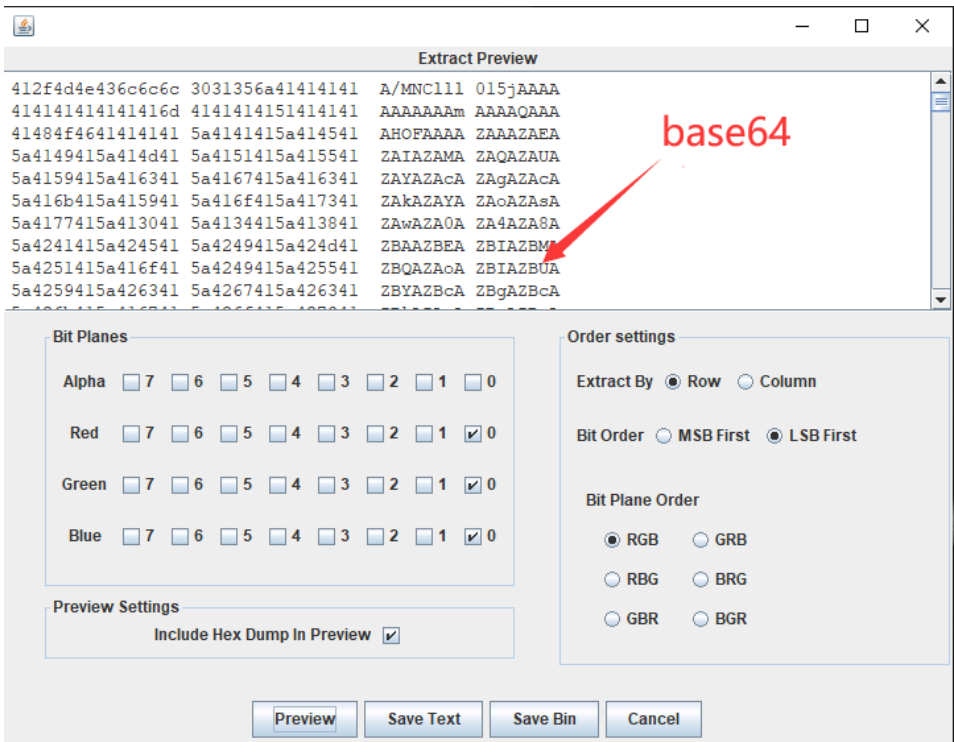
flag: Dozerctf{Do\_U\_useMSWORD\_or\_WPS?}

# py 吗（难度：中）

修改图片高度

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI	ASCII
00000000	89	50	4E	47	0D	0A	1A	0A	00	00	00	0D	49	48	44	52	!PNG	IHDR
00000010	00	00	01	8E	00	00	01	8E	08	06	00	00	00	80	04	E4		ä
00000020	7C	00	01	00	00	49	44	41	54	78	9C	C4	FD	57	93	2D		IDATx ÄyW -
00000030	59	96	DF	89	FD	B6	70	75	64	E8	B8	22	45	55	56	96	Y B y pudè,"EUV	
00000040	68	74	37	AA	A1	05	39	63	80	0D	6C	40	33	7E	02	BE	ht7 i 9c  1@3~ %	

lsb 隐写



导出的 base64 根据提示猜测为 py 或 pyc 文件

```
1 import base64
2
3 fin=open("py.txt","r")
4 fout=open('2.pyc',"wb")
5 base64.decode(fin,fout)
6 fin.close()
7 fout.close()
```

将 pyc 文件反编译得到加密脚本

请选择pyc文件进行解密。支持所有Python版本

未选择任何文件

```
39     '97',
40     '125',
41     '105',
42     '73']
43
44 def encode():
45     flag = 'Dozerctf{python_is_the_best_language!}'
46     ciphertext = []
47     for i in range(len(flag)):
48         s = chr(i ^ ord(flag[i]))
49         if i % 2 == 0:
50             s = ord(s) + 5
51         else:
52             s = ord(s) - 5
53         ciphertext.append(str(s))
54
55     print ciphertext[::-1]
```

写出对应揭秘脚本得到 flag

C:\Users\lenovo\Desktop\e.py - Sublime Text

File Edit Selection Find View Goto Tools Project Preferences Help

```
1 ciphertext=['83', '10', '65', '74', '59', '90', '115',
2 '117', '119', '117', '63', '115', '101', '130', '112',
3 '78', '107', '129', '98', '82', '93', '126', '75', '101',
4 '93', '105', '122', '120', '116', '120', '92', '119', '97',
5 '123', '97', '125', '105', '73']
6
7
8 def decode():
9     i=0
10    flag=''
11    for str in ciphertext[::-1]:
12        s=int(str)
13
14        if i%2 == 0:
15            s=s-5
16        else:
17            s=s+5
18
19        flag_bit=i^s
20
21        flag=flag+chr(flag_bit)
22        i=i+1
23
24    print flag
25    decode()
```

Line 6, Column 1 Spaces: 4 Python

flag: Dozerctf{python\_is\_the\_best\_language!}