

Cette épreuve nous donne un fichier doté d'une extension png. Cependant, l'image n'est pas visible et semble corrompue:

```
chelinka@RUSTYCOMPUTER:~/Downloads$ pngcheck leak.png
leak.png  this is neither a PNG or JNG image nor a MNG stream
ERROR: leak.png
```

En l’observant de plus près, l’en-tête du fichier correspond à un exécutable ELF 64-bit. cependant, l’architecture du binaire semble corrompue aussi. Nous allons donc regarder le fichier avec hexedit pour aviser de l’état de la corruption.

00000000	7F 45 4C 46	02 01 01 00	00 00 00 00	49 48 44 52	00 00 01 C2	00 00 02 A4	08 06 00 00	00 93 5D 3F	.ELF.....IHDR.....]?
00000000	9F 00 00 25	84 7A 54 58	74 52 61 77	20 70 72 6F	66 69 6C 65	20 74 79 70	65 20 65 78	69 66 00 00	...%.ZtXRaw profile type exif...
00000040	78 DA B5 9C	69 92 1C 39	72 85 FF E3	14 3A 02 16	C7 76 1C AC	66 BA 81 8E	AF EF 21 AB	B8 4D CF 8C	x...i.9r.....v.f.....!..M...
00000060	F5 C8 34 36	59 5C CA CC	08 00 EE FE	16 07 A2 DD	F9 9F FF BE	EE BF F8 35	5A 2E CE 72	6D A5 97 E2	..D6Y.....Z..rm.....
00000080	F9 65 DD 7A	1C 7D 83 FC	EF D7 78 7F	0F 6F AE EF	F7 6B 65 AC	AF CE 2F 3F	77 C3 7C 7F	DF E9 0D 89	.e.z]...X.o...k.....2w]....
000000A0	AF E9 F3 42	2B 9F A1 E1	FB E7 5F 1F	FB FE 1A 06	FD E5 5F 2E	D4 D6 D7 0B	F3 F7 17 BA	70 BE C6 FE	..B+.....]....
000000C0	C7 85 3E B7	F5 49 23 D2	F7 FB EB 42	FD EB 42 29	7E 5E 08 5F	17 18 9F 6B	F9 D2 58 FD	75 0A F3 7C	..>..I#.....B.B.)^.....[.u.]...
000000E0	BE 7E 7D FE	B3 0C FC 71	FA CB DA EF	C3 FE 87 7F	57 56 6F 67	EE 93 62 3C	29 24 CF DF	29 C5 CF 00	.....q.....wVog..bc\$)....
00000100	92 FE 44 97	06 2F 05 FE	8E A9 F0 6C	06 5F FD A4	70 77 4E ED	EB 62 2C 8F	5F AD D3 8F	5F 9D 11 5D	..D./.....F.....w.N..b.....
00000120	0D D5 FE F2	4D BF 45 E5	7B A4 76 46	EB FB 38 F7	67 B4 EC 38	BC E9 8C 45	2F 3B BE 5F	E5 CF 5D C8	.....M.E.(.f..W.;g.....E.?......]
00000140	7F BC 90 7E	DC 27 FE FA	F7 6B 5F DF	C3 63 7E 3E	E2 D7 88 FE	F2 7D 8D F9	77 B7 FB E6	CC 2C 86 15	.....~..zgk..?>.....X)..w.....
00000160	96 BA 7C 4D	E8 2A FA EF	3B DE 37 B9	94 65 D0 1C	43 2B BE F2	57 83 89 FA	7E 77 37 37	B2 7A 91 0A	.. M.(.?.7..n.Cn+..C's..~w7.Z..
00000180	DB 2F 3F F9	BD 42 0F 91	70 DD 60 61	87 11 6E 38	EF EB 0A A4	7A B4 78 5C	AC 7C 13 E3	8A E9 FD B0	/?..B..p..a..n8.....z.x].....
000001A0	AC 1A 7B 5C	49 F1 33 FD	0E 37 D6 D4	D3 4E 8D 78	AE 17 76 4F	F1 F7 58 C2	BB 6D F7 CB	BD BB 35 4E	...[I.3..7.....n.x.vk..X..m.....5.....
000001C0	B5 03 6F 8D	81 8B 29 05	FE F6 6F 77	73 7F 0A AF	4A 21 04 FF	B5 F8 A4 05	E3 8A 51 8B	CD 30 14 39	..(o.....o.w?p.]...0.0.0.9.....
000001E0	FD CD DB 88	48 8B 5F 8B	9A DF 02 7F	FF FE F3 97	E2 9A 88 60	D6 2A AB 44	3A 0B 3B 3F	97 98 3F 63	.....H.....*?D.;?..9.....
00000200	44 82 F4 02	9D 78 63 BE	EB A7 06 43	FD 5E 17 60	89 8B 75 66	30 54 86 05	A2 1E 52 0E	25 F8 1A FC	D...xc.....C.....uf0T....R.%..C.....
00000220	0D 81 85 6C	04 68 30 F4	98 2C 4E 22	10 72 8E 9B	41 46 48 54	51 8D 2D EA	D6 7C A4 86	F7 D6 98 23	...l.h0...,"n"....r..AFKtQ....R....#.....

Nous remarquons la présence de blocs png corrects après l'en-tête du fichier. En remplaçant cet en-tête par la signature d'un fichier png `b"\x89PNG\x0d\x0a\x1a\x0a"`, nous retrouvons une image png valide !

```

00000000  89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52 00 00 01 C2 00 00 02 A4 08 06 00 00 00 93 5D 3F .PNG.....IHDR.....]?
00000020  9F 00 00 25 84 7A 54 58 74 52 61 77 20 70 72 6F 66 69 6C 65 20 74 79 70 65 20 65 78 69 66 00 00 ...zTXtRaw profile type exif..
00000040  78 DA B5 9C 69 92 1C 39 72 B5 FF E3 14 3A 02 1F C7 76 1C AC 66 BA 81 8E AF EF 21 AB 68 0D CF 8C x...i.9r.....v.f.....!M..
00000060  F5 C8 44 36 59 C5 CA CC 08 00 EE FE 16 07 A2 DD F9 9F FF BE EF BF 8D 5A 2E CE 72 6D A5 97 E2 .D6V.....Z.....rm..
00000080  F9 65 DD 7A 1C 7C D3 FC E7 D7 78 7F 07 6F EF EF F7 6B C5 AF EF C2 EF 3F 77 C3 7C 7C DF E9 0D 89 .e.z.|...X...o...k.....?w.|....

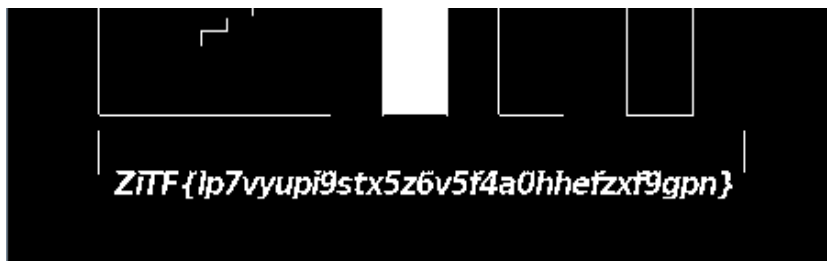
```

```
chelinka@RUSTYCOMPUTER:~/Downloads$ pngcheck -v leak.png
File: leak.png (29190 bytes)
  chunk IHDR at offset 0x0000c, length 13
    450 x 676 image, 32-bit RGB+alpha, non-interlaced
  chunk zTXt at offset 0x00025, length 9604, keyword: Raw profile type exif
  chunk iCCP at offset 0x025b5, length 388
    profile name = ICC profile, compression method = 0 (deflate)
    compressed profile = 375 bytes
  chunk iTXt at offset 0x02745, length 3448, keyword: XML:com.adobe.xmp
    uncompressed, no language tag
    no translated keyword, 3427 bytes of UTF-8 text
  chunk bKGD at offset 0x034c9, length 6
    red = 0x00ff, green = 0x00ff, blue = 0x00ff
  chunk pHYs at offset 0x034db, length 9: 2835x2835 pixels/meter (72 dpi)
  chunk tIME at offset 0x034f0, length 7: 26 Nov 2022 17:16:59 UTC
  chunk IDAT at offset 0x03503, length 8192
    zlib: deflated, 32K window, maximum compression
  chunk IDAT at offset 0x0550f, length 7395
  chunk IEND at offset 0x071fe, length 0
No errors detected in leak.png (10 chunks, 97.6% compression).
chelinka@RUSTYCOMPUTER:~/Downloads$
```

C'est tout réparé !

Nous pouvons désormais afficher l'image:

En remarquant que le bas de la ligne noire en-dessous du ZitF est coupée par endroit, nous pouvons utiliser stegsolve afin de voir le texte presque invisible caché ici:



Nous avons bien notre flag !

`ZiTf{lp7vyupi9stx5z6v5f4a0hhefzxf9gpn}`