

LinkedIn: [Kelvin Kimotho](#)

CYBER TALENTS

Challenge Name: Image Catch

 **Category:** Digital Forensics

 **Level:** medium

 **Tries:** 1565 Times

 **Solved:** 660 Times

Difficulty Level 



Rating 



Challenge Description

Capture the flag from the supplied image

Solution

I downloaded the given image file using **wget** a command-line tool for file download. Then used the **file** command line tool to confirm the file type of the downloaded file which confirmed that the file downloaded was a jpeg.

```
kali@kali: ~/Desktop
File Actions Edit View Help

(kali@kali)-[~/Desktop]
└─$ wget https://hubchallenges.s3.eu-west-1.amazonaws.com/forensics/challenge.jpg
--2025-02-20 08:11:08-- https://hubchallenges.s3.eu-west-1.amazonaws.com/forensics/challenge.jpg
Resolving hubchallenges.s3.eu-west-1.amazonaws.com (hubchallenges.s3.eu-west-1.amazonaws.com) ... 3.5.68.163
Connecting to hubchallenges.s3.eu-west-1.amazonaws.com (hubchallenges.s3.eu-west-1.amazonaws.com)|3.5.68.163|:443 ... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7746628 (7.4M) [image/jpeg]
Saving to: 'challenge.jpg'

challenge.jpg 100%[=====] 7.39M 702KB/s in 18s

2025-02-20 08:11:27 (432 KB/s) - 'challenge.jpg' saved [7746628/7746628]

(kali@kali)-[~/Desktop]
└─$ ls
challenge.jpg

(kali@kali)-[~/Desktop]
└─$ file challenge.jpg
challenge.jpg: JPEG image data, JFIF standard 1.01, aspect ratio, density 1x1, segment length 16, progressive, precision 8, 660x440, components 3

(kali@kali)-[~/Desktop]
└─$
```

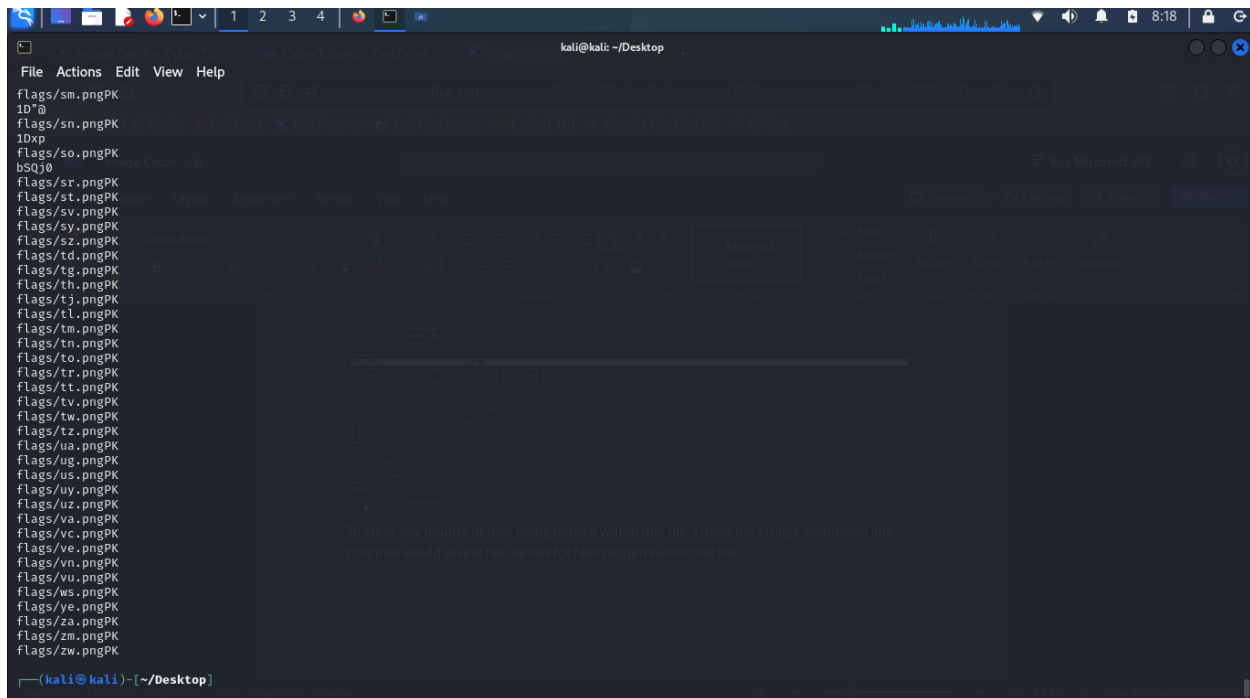
I then used the **exiftool** command line tool to examine the file's meta data but found no flag but the file size was suspicious ” **File Size : 7.7 MB**” . I felt like 7.7mb for an image was too much and there were possibilities that something else was being hidden in that file.

```
(kali@kali)-[~/Desktop]
└─$ exiftool challenge.jpg
ExifTool Version Number      : 12.76
File Name                    : challenge.jpg
Directory                    : .
File Size                    : 7.7 MB
File Modification Date/Time  : 2025:02:20 13:11:05+00:00
File Access Date/Time       : 2025:02:20 08:11:27+00:00
File Inode Change Date/Time  : 2025:02:20 08:11:27+00:00
File Permissions             : -rw-rw-r--
File Type                    : JPEG
File Type Extension          : jpg
MIME Type                    : image/jpeg
JFIF Version                 : 1.01
Resolution Unit              : None
X Resolution                  : 1
Y Resolution                  : 1
Image Width                  : 660
Image Height                  : 440
Encoding Process              : Progressive DCT, Huffman coding
Bits Per Sample               : 8
Color Components              : 3
Y Cb Cr Sub Sampling         : YCbCr4:2:0 (2 2)
Image Size                   : 660x440
Megapixels                   : 0.290

(kali@kali)-[~/Desktop]
└─$
```

To clear any doubts of files being hidden within this file, I used the **strings** command line tool that would reveal file names for files hidden within this file.

```
(kali@kali)-[~/Desktop]
$ strings challenge.jpg
JFIF
!###
6)6")
-----
VtON
p0svk
Wn){
p3#^
```



The strings command confirmed to me that there was a directory named **flags** which contained **.png** images which was hidden within this file. I went ahead and used **binwalk** a tool for searching a given binary image for embedded files.

```

7117459 0x6C9A93 Zip archive data, at least v2.0 to extract, compressed size: 16615, uncompressed size: 26211, name: flags/tz.png
7134116 0x6CDBA4 Zip archive data, at least v2.0 to extract, compressed size: 375, uncompressed size: 9233, name: flags/ua.png
7134533 0x6CDD45 Zip archive data, at least v2.0 to extract, compressed size: 35655, uncompressed size: 44363, name: flags/ug.png
7170230 0x6D68B6 Zip archive data, at least v1.0 to extract, compressed size: 40463, uncompressed size: 40463, name: flags/us.png
7210735 0x6E06EF Zip archive data, at least v2.0 to extract, compressed size: 76519, uncompressed size: 86535, name: flags/uy.png
7287296 0x6F3200 Zip archive data, at least v2.0 to extract, compressed size: 16047, uncompressed size: 22562, name: flags/uz.png
7303385 0x6F70D9 Zip archive data, at least v2.0 to extract, compressed size: 203953, uncompressed size: 220666, name: flags/va.png
7507380 0x7280B4 Zip archive data, at least v2.0 to extract, compressed size: 9502, uncompressed size: 17191, name: flags/vc.png
7516924 0x7282FC Zip archive data, at least v2.0 to extract, compressed size: 24042, uncompressed size: 32327, name: flags/vc.png
7541008 0x731110 Zip archive data, at least v2.0 to extract, compressed size: 18188, uncompressed size: 24678, name: flags/vn.png
7559238 0x735846 Zip archive data, at least v2.0 to extract, compressed size: 54698, uncompressed size: 62874, name: flags/vu.png
7613978 0x742E1A Zip archive data, at least v2.0 to extract, compressed size: 12944, uncompressed size: 18833, name: flags/ws.png
7626964 0x7460D4 Zip archive data, at least v2.0 to extract, compressed size: 372, uncompressed size: 9298, name: flags/ye.png
7627378 0x746272 Zip archive data, at least v2.0 to extract, compressed size: 17796, uncompressed size: 25484, name: flags/za.png
7645216 0x74A820 Zip archive data, at least v2.0 to extract, compressed size: 49454, uncompressed size: 58631, name: flags/zm.png
7694712 0x756978 Zip archive data, at least v2.0 to extract, compressed size: 40432, uncompressed size: 45472, name: flags/zw.png
7746606 0x76342E End of Zip archive, footer length: 22

```

```

kali@kali:~/Desktop
$

```

```

kali@kali:~/Desktop
$ binwalk -M challenge.jpg

```

DECIMAL	HEXADECIMAL	DESCRIPTION
0	0x0	JPEG image data, JFIF standard 1.01
40816	0x9F70	Zip archive data, at least v2.0 to extract, name: flags/
40852	0x9F94	Zip archive data, at least v2.0 to extract, compressed size: 129551, uncompressed size: 139360, name: flags/ad.png
170445	0x299CD	Zip archive data, at least v2.0 to extract, compressed size: 403, uncompressed size: 7024, name: flags/ae.png
170890	0x2988A	Zip archive data, at least v2.0 to extract, compressed size: 167209, uncompressed size: 176623, name: flags/af.png
338141	0x528DD	Zip archive data, at least v2.0 to extract, compressed size: 32936, uncompressed size: 38656, name: flags/ag.png
371110	0x5494F	Zip archive data, at least v2.0 to extract, compressed size: 70594, uncompressed size: 78027, name: flags/al.png
441755	0x68D9B	Zip archive data, at least v2.0 to extract, compressed size: 368, uncompressed size: 6992, name: flags/am.png
442165	0x6B3F5	Zip archive data, at least v2.0 to extract, compressed size: 40806, uncompressed size: 46827, name: flags/ao.png
482213	0x758A5	Zip archive data, at least v2.0 to extract, compressed size: 80727, uncompressed size: 89682, name: flags/ar.png
562982	0x89726	Zip archive data, at least v2.0 to extract, compressed size: 373, uncompressed size: 9300, name: flags/at.png
563397	0x898C5	Zip archive data, at least v2.0 to extract, compressed size: 16020, uncompressed size: 16351, name: flags/au.png
579459	0x8D783	Zip archive data, at least v2.0 to extract, compressed size: 12490, uncompressed size: 18524, name: flags/az.png
591991	0x90877	Zip archive data, at least v2.0 to extract, compressed size: 31149, uncompressed size: 39398, name: flags/ba.png
623182	0x9824E	Zip archive data, at least v2.0 to extract, compressed size: 17094, uncompressed size: 24654, name: flags/bb.png
640318	0x9C53E	Zip archive data, at least v2.0 to extract, compressed size: 16842, uncompressed size: 22014, name: flags/bd.png
657202	0xA0732	Zip archive data, at least v2.0 to extract, compressed size: 396, uncompressed size: 11934, name: flags/be.png
657640	0xA0BE8	Zip archive data, at least v2.0 to extract, compressed size: 9803, uncompressed size: 17746, name: flags/bf.png
667485	0xA2F5D	Zip archive data, at least v2.0 to extract, compressed size: 347, uncompressed size: 8317, name: flags/bg.png
667074	0xA3E2	Zip archive data, at least v2.0 to extract, compressed size: 5492, uncompressed size: 11548, name: flags/bi.png
673408	0xA4680	Zip archive data, at least v2.0 to extract, compressed size: 34840, uncompressed size: 41664, name: flags/bi.png
708290	0xACEC2	Zip archive data, at least v2.0 to extract, compressed size: 388, uncompressed size: 9247, name: flags/bj.png
708720	0xAD070	Zip archive data, at least v2.0 to extract, compressed size: 97959, uncompressed size: 103110, name: flags/bn.png
806721	0xC4F41	Zip archive data, at least v2.0 to extract, compressed size: 378, uncompressed size: 9514, name: flags/bo.png
807141	0xC50E5	Zip archive data, at least v2.0 to extract, compressed size: 78952, uncompressed size: 87003, name: flags/br.png
886135	0xD8577	Zip archive data, at least v2.0 to extract, compressed size: 13831, uncompressed size: 18699, name: flags/bs.png
900008	0xDBA8	Zip archive data, at least v2.0 to extract, compressed size: 252591, uncompressed size: 286121, name: flags/bt.png
1152641	0x119681	Zip archive data, at least v2.0 to extract, compressed size: 429, uncompressed size: 9319, name: flags/bw.png

The embedded files were extracted and i navigated to the flags folder.

```

kali@kali:~/Desktop
$ ls
challenge.jpg  _challenge.jpg.extracted

kali@kali:~/Desktop
$ cd _challenge.jpg.extracted

kali@kali:~/Desktop/_challenge.jpg.extracted
$ ls
9F70.zip  flags

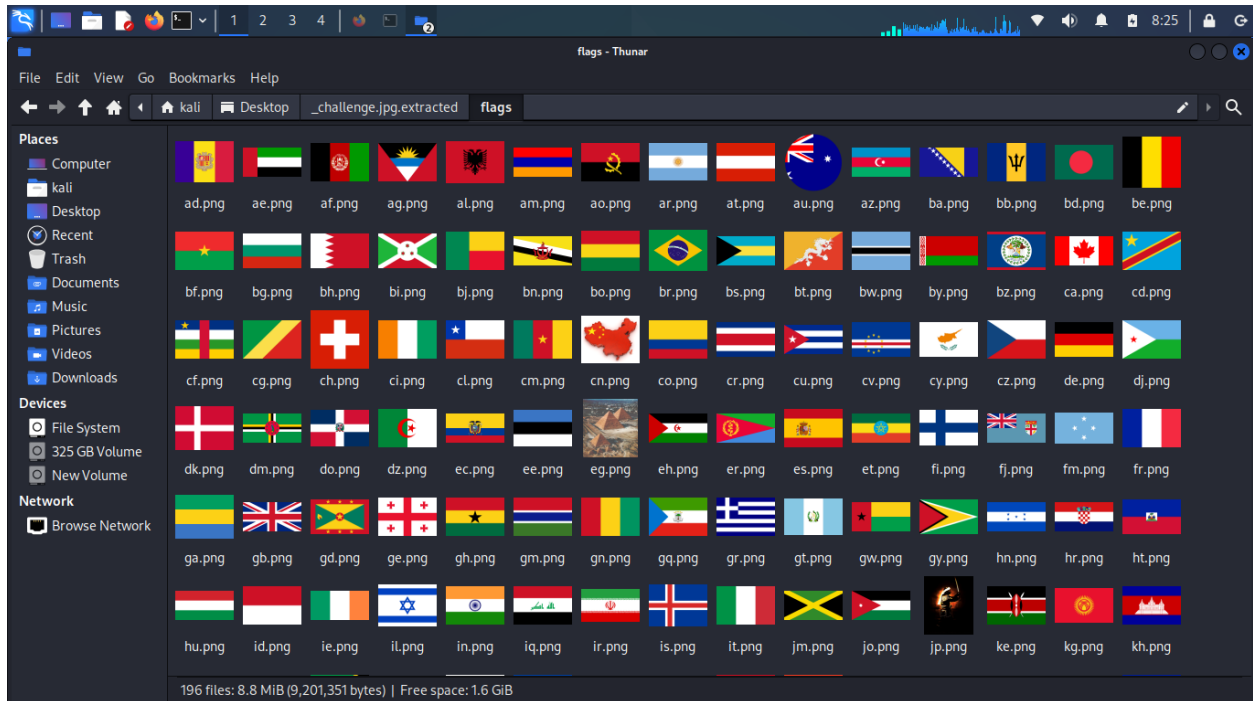
kali@kali:~/Desktop/_challenge.jpg.extracted
$ cd flags

kali@kali:~/Desktop/_challenge.jpg.extracted/flags
$ ls
ad.png  az.png  bn.png  cf.png  cv.png  ee.png  ga.png  gw.png  iq.png  ki.png  lc.png  mc.png  mt.png  ni.png  ph.png  rw.png  sm.png  th.png  tz.png  vu.png
ae.png  ba.png  bo.png  cg.png  cy.png  eg.png  gb.png  gy.png  ir.png  km.png  li.png  md.png  mu.png  nl.png  pk.png  sa.png  sn.png  tj.png  ua.png  ws.png
af.png  bb.png  br.png  ch.png  cz.png  eh.png  gd.png  hn.png  is.png  kn.png  lk.png  me.png  mv.png  no.png  pl.png  sb.png  so.png  tl.png  ug.png  ye.png
ag.png  bd.png  bs.png  ci.png  de.png  er.png  ge.png  hr.png  it.png  kp.png  lr.png  mg.png  mw.png  np.png  pt.png  sc.png  sr.png  tm.png  us.png  za.png
al.png  be.png  bt.png  cl.png  dj.png  es.png  gh.png  ht.png  jm.png  kr.png  ls.png  mh.png  mx.png  nr.png  pw.png  sd.png  st.png  tn.png  uy.png  zw.png
am.png  bf.png  bw.png  cm.png  dk.png  et.png  gm.png  hu.png  jo.png  ks.png  lt.png  mk.png  my.png  nz.png  py.png  se.png  sv.png  to.png  uz.png  zw.png
ao.png  bg.png  by.png  cn.png  dm.png  fi.png  gn.png  id.png  jp.png  kw.png  lu.png  ml.png  mz.png  om.png  qa.png  sg.png  sy.png  tr.png  va.png
ar.png  bh.png  bz.png  co.png  do.png  fj.png  gq.png  ie.png  ke.png  kz.png  lv.png  mm.png  na.png  pa.png  ro.png  si.png  sz.png  tt.png  vc.png
at.png  bi.png  ca.png  cr.png  dz.png  fm.png  gr.png  il.png  kg.png  la.png  ly.png  mn.png  ne.png  pe.png  rs.png  sk.png  td.png  tv.png  ve.png
au.png  bj.png  cd.png  cu.png  ec.png  fr.png  gt.png  in.png  kh.png  lb.png  ma.png  mr.png  ng.png  pg.png  ru.png  sl.png  tg.png  tw.png  vn.png

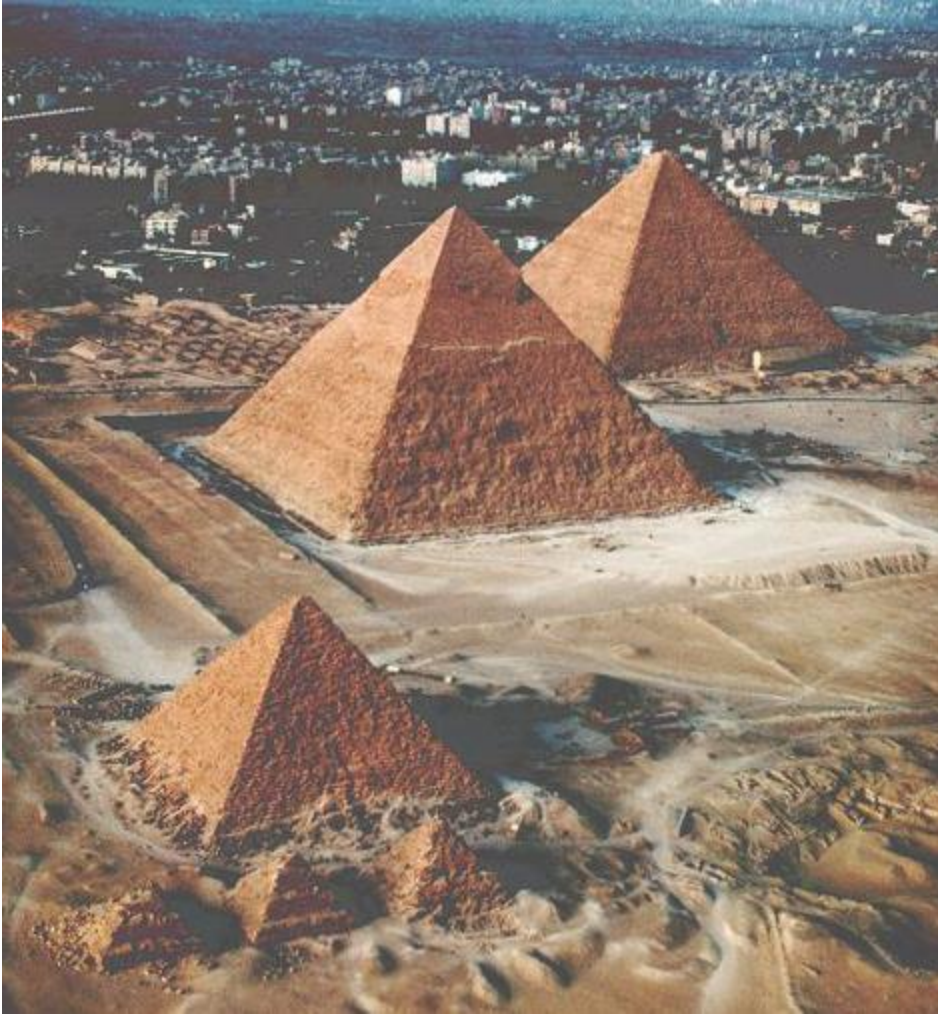
kali@kali:~/Desktop/_challenge.jpg.extracted/flags
$

```

The image files were many, I went ahead and opened the files to understand what the images were.



The images were flags for different countries. But on looking closely i noticed two images which were not flags **eg.png**.



and [jp.png](#).



Looking closely, jp.png had some writings “6C76648BEDF6B6C5A9D4B564FC68868D” and that was the flag.