Steghide

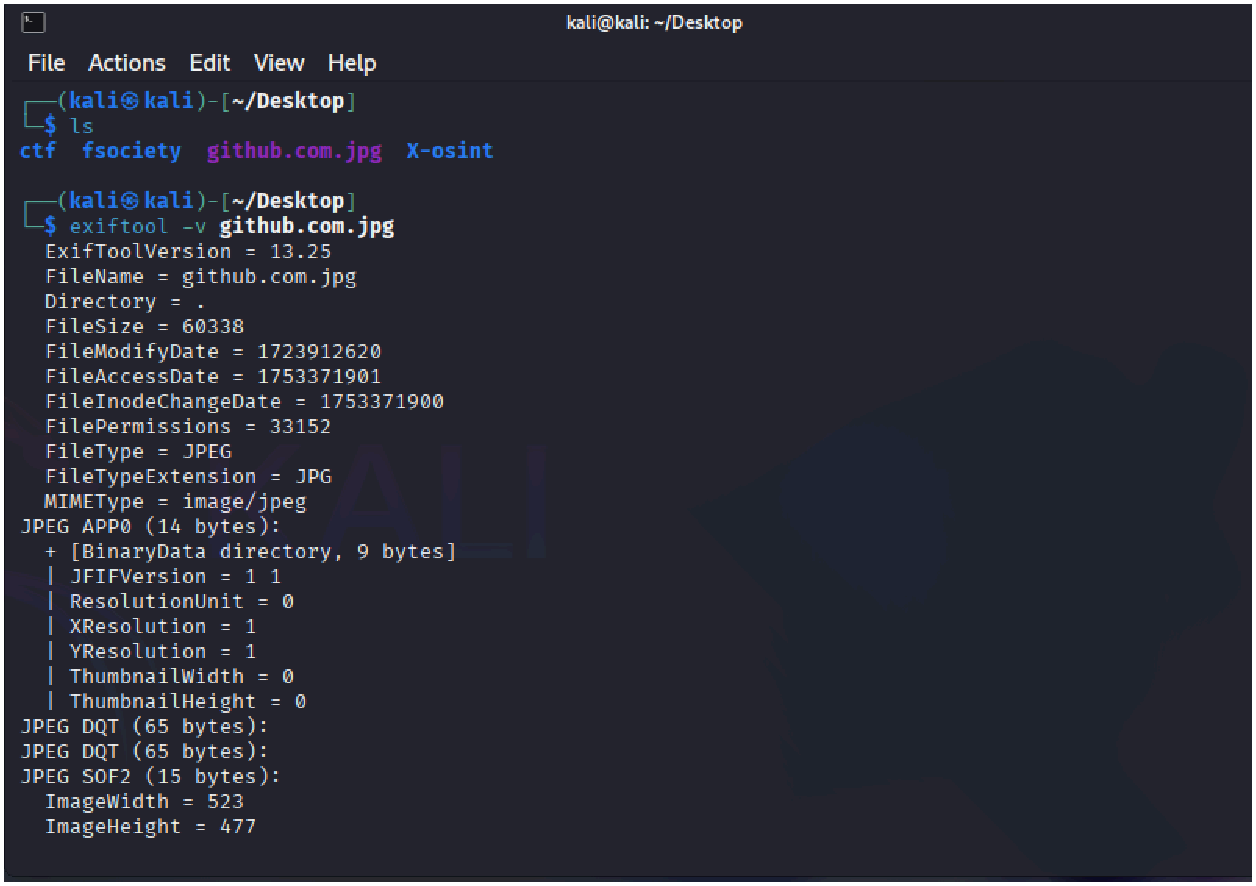
Tags

This demo is part of my OSINT and steganography learning series. It shows how metadata can be used to hide information in plain sight. A beginner-friendly demo showing how to hide and extract messages from images using ExifTool on Kali Linux.

1- have a photo

GUESS WHAT?
NO FLAG HERE
imgflip.com

2- Run `exiftool` in verbose mode to inspect the metadata

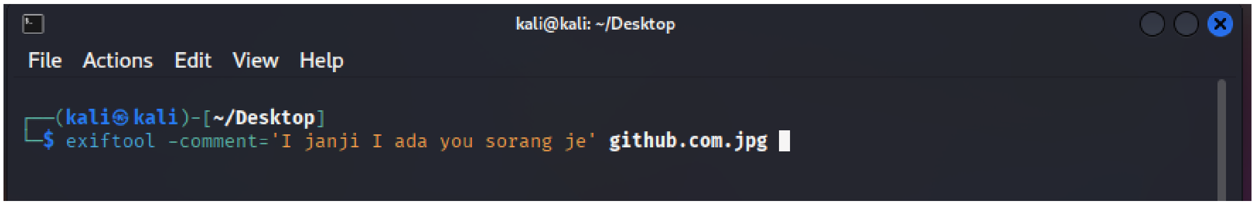


```
┌──(kali㉿kali)-[~/Desktop]
└─$ ls
ctf  fsociety  github.com.jpg  X-osint

┌──(kali㉿kali)-[~/Desktop]
└─$ exiftool -v github.com.jpg
  ExifToolVersion = 13.25
  FileName = github.com.jpg
  Directory = .
  FileSize = 60338
  FileModifyDate = 1723912620
  FileAccessDate = 1753371901
  FileInodeChangeDate = 1753371900
  FilePermissions = 33152
  FileType = JPEG
  FileTypeExtension = JPG
  MIMEType = image/jpeg
JPEG APP0 (14 bytes):
  + [BinaryData directory, 9 bytes]
  | JFIFVersion = 1 1
  | ResolutionUnit = 0
  | XResolution = 1
  | YResolution = 1
  | ThumbnailWidth = 0
  | ThumbnailHeight = 0
JPEG DQT (65 bytes):
JPEG DQT (65 bytes):
JPEG SOF2 (15 bytes):
  ImageWidth = 523
  ImageHeight = 477
```

then add comment in the photo using this command

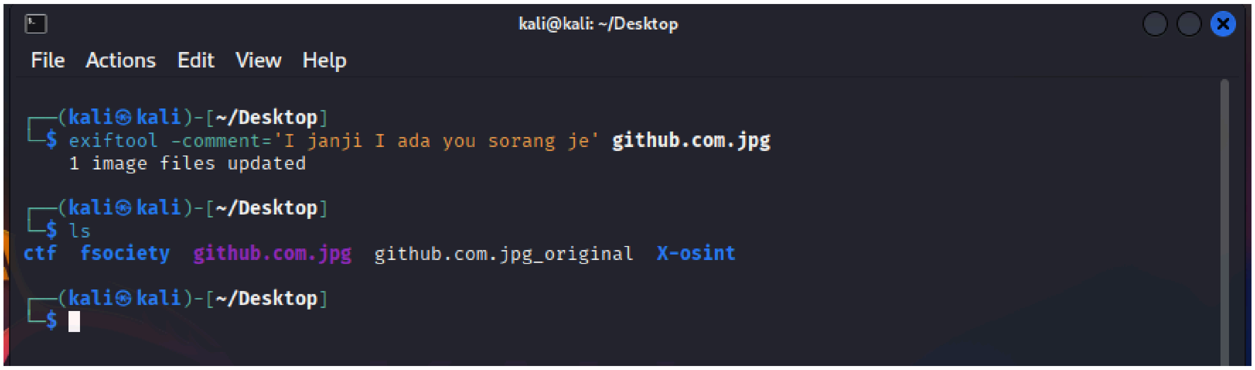
```
└─$ exiftool -comment='I janji I ada you sorang je' github.com.jpg
```

exiftool - comment='Your secret message' <photo name and ext>



```
┌──(kali㉿kali)-[~/Desktop]
└─$ exiftool -comment='I janji I ada you sorang je' github.com.jpg
```

ExifTool creates a backup of the original image automatically.



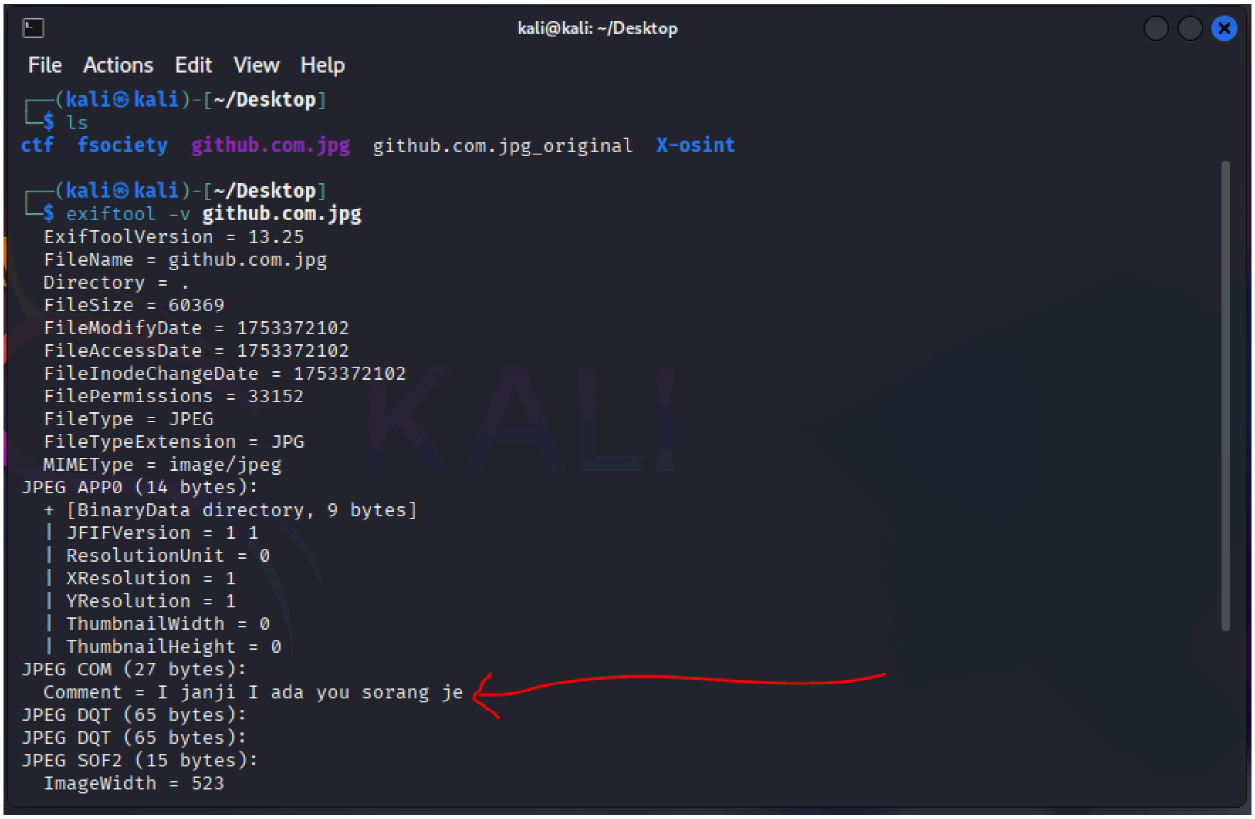
```
┌──(kali㉿kali)-[~/Desktop]
└─$ exiftool -comment='I janji I ada you sorang je' github.com.jpg
    1 image files updated

┌──(kali㉿kali)-[~/Desktop]
└─$ ls
ctf  fsociety  github.com.jpg  github.com.jpg_original  X-osint

┌──(kali㉿kali)-[~/Desktop]
└─$
```

you can see the original one and the modified one

Verify

to check the modified one message just use this command

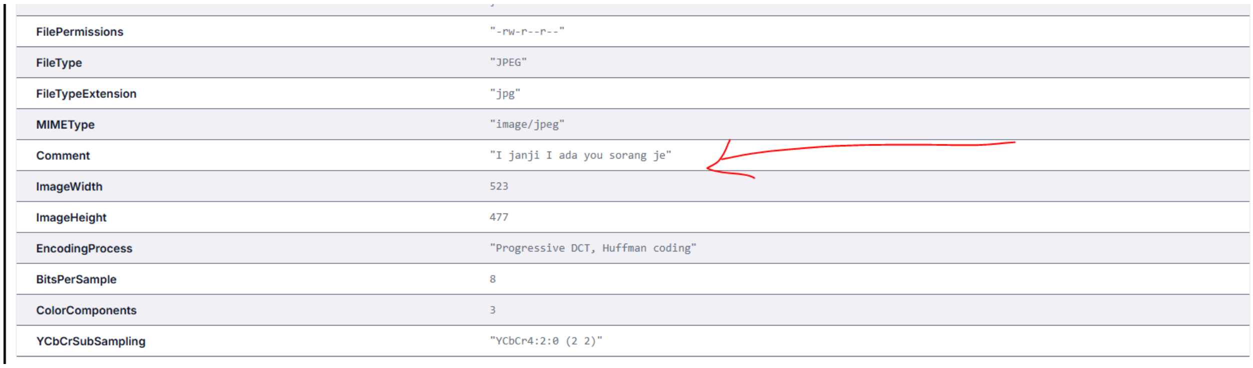
```
exiftool -v <purple file>
```



```
┌──(kali㉿kali)-[~/Desktop]
└─$ ls
ctf  fsociety  github.com.jpg  github.com.jpg_original  X-osint

┌──(kali㉿kali)-[~/Desktop]
└─$ exiftool -v github.com.jpg
  ExifToolVersion = 13.25
  FileName = github.com.jpg
  Directory = .
  FileSize = 60369
  FileModifyDate = 1753372102
  FileAccessDate = 1753372102
  FileInodeChangeDate = 1753372102
  FilePermissions = 33152
  FileType = JPEG
  FileTypeExtension = JPG
  MIMEType = image/jpeg
JPEG APP0 (14 bytes):
  + [BinaryData directory, 9 bytes]
  | JFIFVersion = 1 1
  | ResolutionUnit = 0
  | XResolution = 1
  | YResolution = 1
  | ThumbnailWidth = 0
  | ThumbnailHeight = 0
JPEG COM (27 bytes):
  Comment = I janji I ada you sorang je
JPEG DQT (65 bytes):
JPEG DQT (65 bytes):
JPEG SOF2 (15 bytes):
  ImageWidth = 523
```

or you can use online exiftool platform

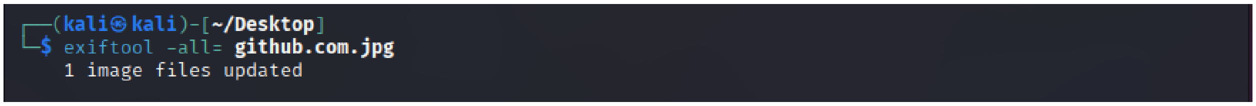
https://exifmeta.com/



FilePermissions	"-rw-r--r--"
FileType	"JPEG"
FileTypeExtension	"jpg"
MIMEType	"image/jpeg"
Comment	"I janji I ada you sorang je"
ImageWidth	523
ImageHeight	477
EncodingProcess	"Progressive DCT, Huffman coding"
BitsPerSample	8
ColorComponents	3
YCbCrSubSampling	"YCbCr4:2:0 (2 2)"

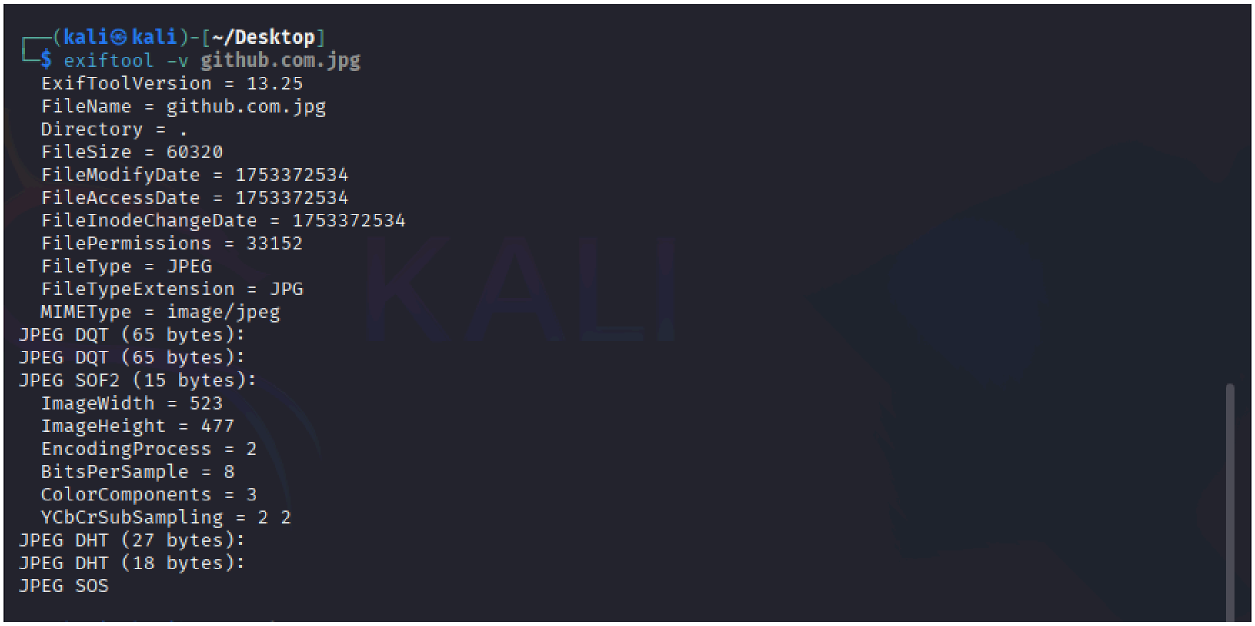
To remove metadata / hidden message use this command

```
exiftool -all=filename
```



```
┌──(kali㉿kali)-[~/Desktop]
└─$ exiftool -all= github.com.jpg
    1 image files updated
```

to verify just use the exift command back



```
┌──(kali㉿kali)-[~/Desktop]
└─$ exiftool -v github.com.jpg
  ExifToolVersion = 13.25
  FileName = github.com.jpg
  Directory = .
  FileSize = 60320
  FileModifyDate = 1753372534
  FileAccessDate = 1753372534
  FileInodeChangeDate = 1753372534
  FilePermissions = 33152
  FileType = JPEG
  FileTypeExtension = JPG
  MIMEType = image/jpeg
JPEG DQT (65 bytes):
JPEG DQT (65 bytes):
JPEG SOF2 (15 bytes):
  ImageWidth = 523
  ImageHeight = 477
  EncodingProcess = 2
  BitsPerSample = 8
  ColorComponents = 3
  YCbCrSubSampling = 2 2
JPEG DHT (27 bytes):
JPEG DHT (18 bytes):
JPEG SOS
```

the message is removed