常见的隐写工具的使用

z4yn:) 于 2020-09-28 15:33:35 发布 965 收藏 4
分类专栏： Try Hack Me靶机 文章标签： 黑盒测试 信息安全 安全漏洞

本文链接：https://blog.csdn.net/qq_44101248/article/details/108850686
版权

Try Hack Me靶机 专栏收录该内容
44 篇文章 1 订阅
订阅专栏

隐写术

常见的隐写工具的使用

---

steghide

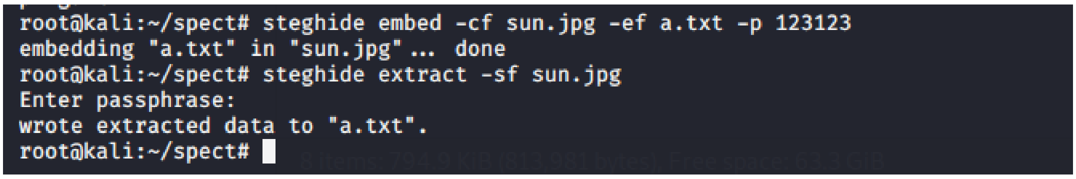
隐藏文件

steghide embed -cf [载体] -ef [隐藏的文件] -p [设置密码]

例：steghide embed -cf sun.jpg -ef a.txt -p 123123

提取文件

steghide extract -sf sun.jpg



```
root@kali:~/spect# steghide embed -cf sun.jpg -ef a.txt -p 123123
embedding "a.txt" in "sun.jpg"... done
root@kali:~/spect# steghide extract -sf sun.jpg
Enter passphrase:
wrote extracted data to "a.txt".
root@kali:~/spect# 
```

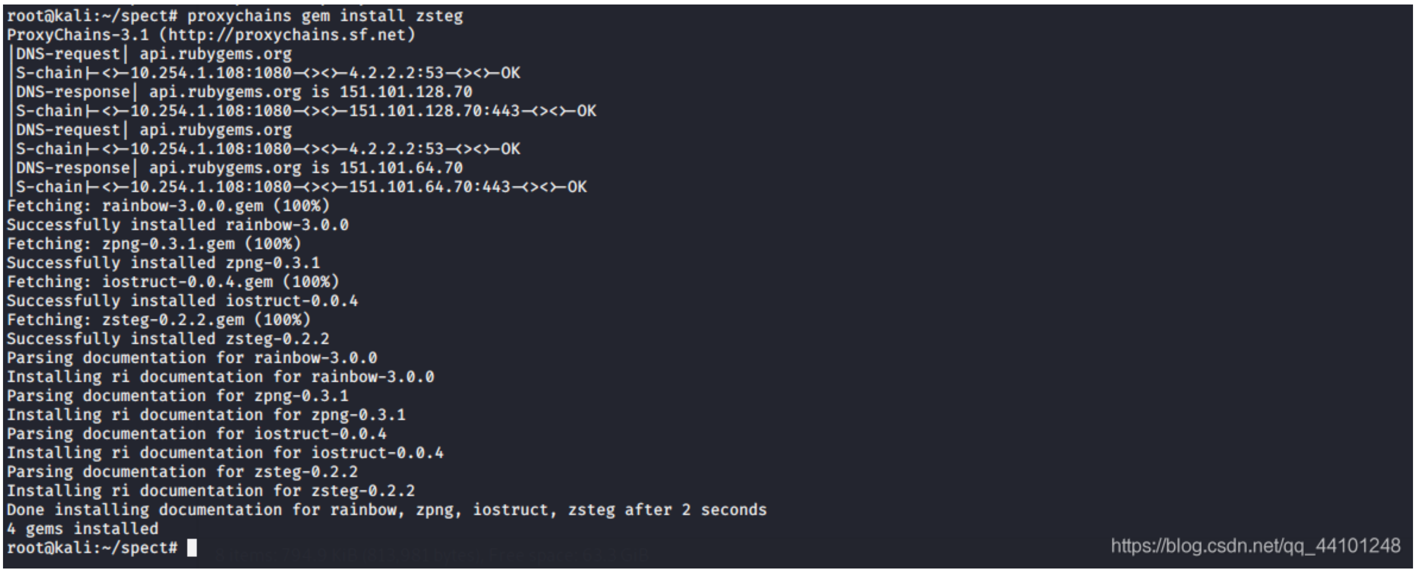
steghide不支持png格式的隐写，zsteg支持png。

---

zsteg

用gem安装zsteg，没有小飞机的用国内的源

proxychains代理一挂，30秒安装完了。

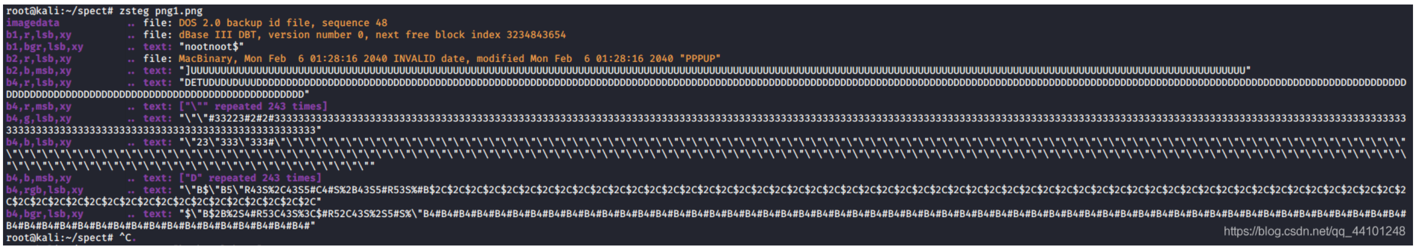


```
root@kali:~/spect# proxychains gem install zsteg
ProxyChains-3.1 (http://proxychains.sf.net)
|DNS-request| api.rubygems.org
|S-chain|-<>-10.254.1.108:1080-<><>-4.2.2.2:53-<><>-OK
|DNS-response| api.rubygems.org is 151.101.128.70
|S-chain|-<>-10.254.1.108:1080-<><>-151.101.128.70:443-<><>-OK
|DNS-request| api.rubygems.org
|S-chain|-<>-10.254.1.108:1080-<><>-4.2.2.2:53-<><>-OK
|DNS-response| api.rubygems.org is 151.101.64.70
|S-chain|-<>-10.254.1.108:1080-<><>-151.101.64.70:443-<><>-OK
Fetching: rainbow-3.0.0.gem (100%)
Successfully installed rainbow-3.0.0
Fetching: zpng-0.3.1.gem (100%)
Successfully installed zpng-0.3.1
Fetching: iostruct-0.0.4.gem (100%)
Successfully installed iostruct-0.0.4
Fetching: zsteg-0.2.2.gem (100%)
Successfully installed zsteg-0.2.2
Parsing documentation for rainbow-3.0.0
Installing ri documentation for rainbow-3.0.0
Parsing documentation for zpng-0.3.1
Installing ri documentation for zpng-0.3.1
Parsing documentation for iostruct-0.0.4
Installing ri documentation for iostruct-0.0.4
Parsing documentation for zsteg-0.2.2
Installing ri documentation for zsteg-0.2.2
Done installing documentation for rainbow, zpng, iostruct, zsteg after 2 seconds
4 gems installed
root@kali:~/spect# 
```

一般直接zsteg加图面png

zsteg png1.png

也可以-E指定特定payload



```
root@kali:~/spect# zsteg png1.png
imagedata           .. file: DOS 2.0 backup id file, sequence 48
b1,r,lsb,xy         .. file: dBase III DBT, version number 0, next free block index 3234843654
b1,bgr,lsb,xy       .. text: "nootnoot$"
b2,r,lsb,xy         .. file: MacBinary, Mon Feb  6 01:28:16 2040 INVALID date, modified Mon Feb  6 01:28:16 2040 "PPPUP"
b2,b,msb,xy         .. text: "]UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU"
b4,r,lsb,xy         .. text: "DETUDUDUDUUDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD"
b4,r,msb,xy         .. text: ["\"" repeated 243 times]
b4,g,lsb,xy         .. text: "\"\"#33223#2#2#333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333
333333333333333333333333333333333333333333333333333"
b4,b,lsb,xy         .. text: "\"23\"333\"333#\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"
\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\
"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\"\""
b4,b,msb,xy         .. text: ["D" repeated 243 times]
b4,rgb,lsb,xy       .. text: "\"B$\"B5\"R43S%2C43S5#C4#S%2B43S5#R53S%#B$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2
C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C$2C"
b4,bgr,lsb,xy       .. text: "$\"B$2B%2S4#R53C43S%3C$#R52C43S%2S5#S%\"B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#
B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#B4#"
root@kali:~/spect# ^C
```

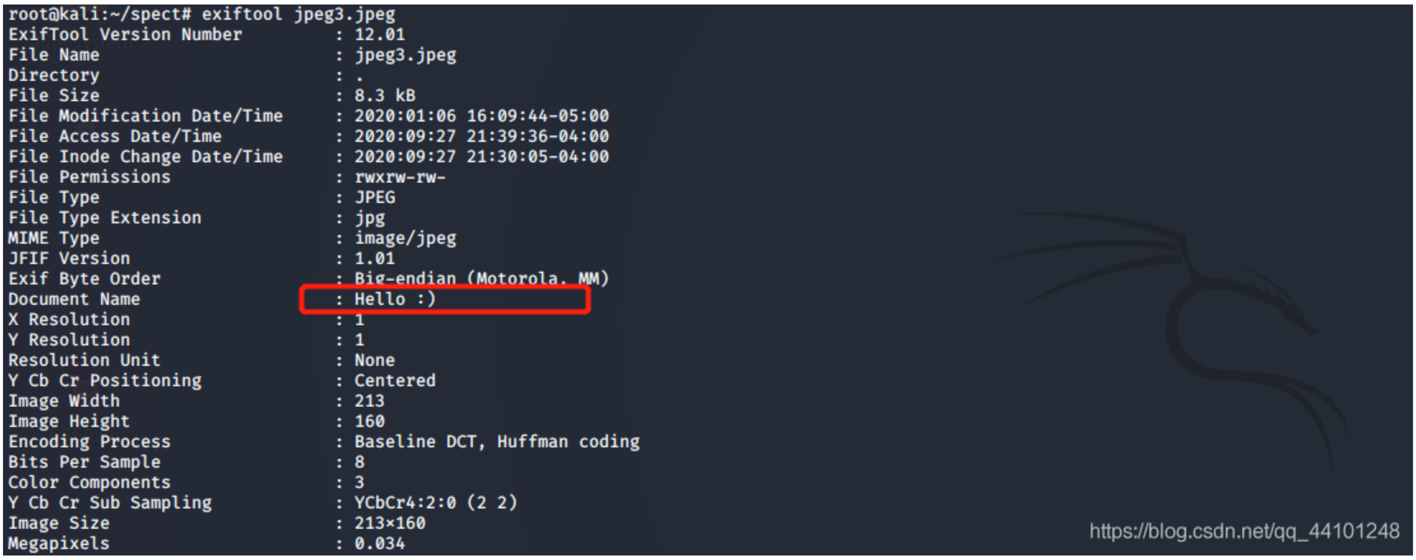
zsteg也支持BMP格式，但主要用于png

---

exiftool

exiftool是图片信息查看工具，有一些题目会把通关信息隐藏在图片的信息里

exiftool 图片名



```
root@kali:~/spect# exiftool jpeg3.jpeg
ExifTool Version Number         : 12.01
File Name                       : jpeg3.jpeg
Directory                       : .
File Size                       : 8.3 kB
File Modification Date/Time     : 2020:01:06 16:09:44-05:00
File Access Date/Time           : 2020:09:27 21:39:36-04:00
File Inode Change Date/Time     : 2020:09:27 21:30:05-04:00
File Permissions                : rwxrw-rw-
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Exif Byte Order                 : Big-endian (Motorola, MM)
Document Name                   : Hello :)
X Resolution                    : 1
Y Resolution                    : 1
Resolution Unit                 : None
Y Cb Cr Positioning             : Centered
Image Width                     : 213
Image Height                    : 160
Encoding Process                : Baseline DCT, Huffman coding
Bits Per Sample                 : 8
Color Components                : 3
Y Cb Cr Sub Sampling            : YCbCr4:2:0 (2 2)
Image Size                      : 213x160
Megapixels                      : 0.034
```

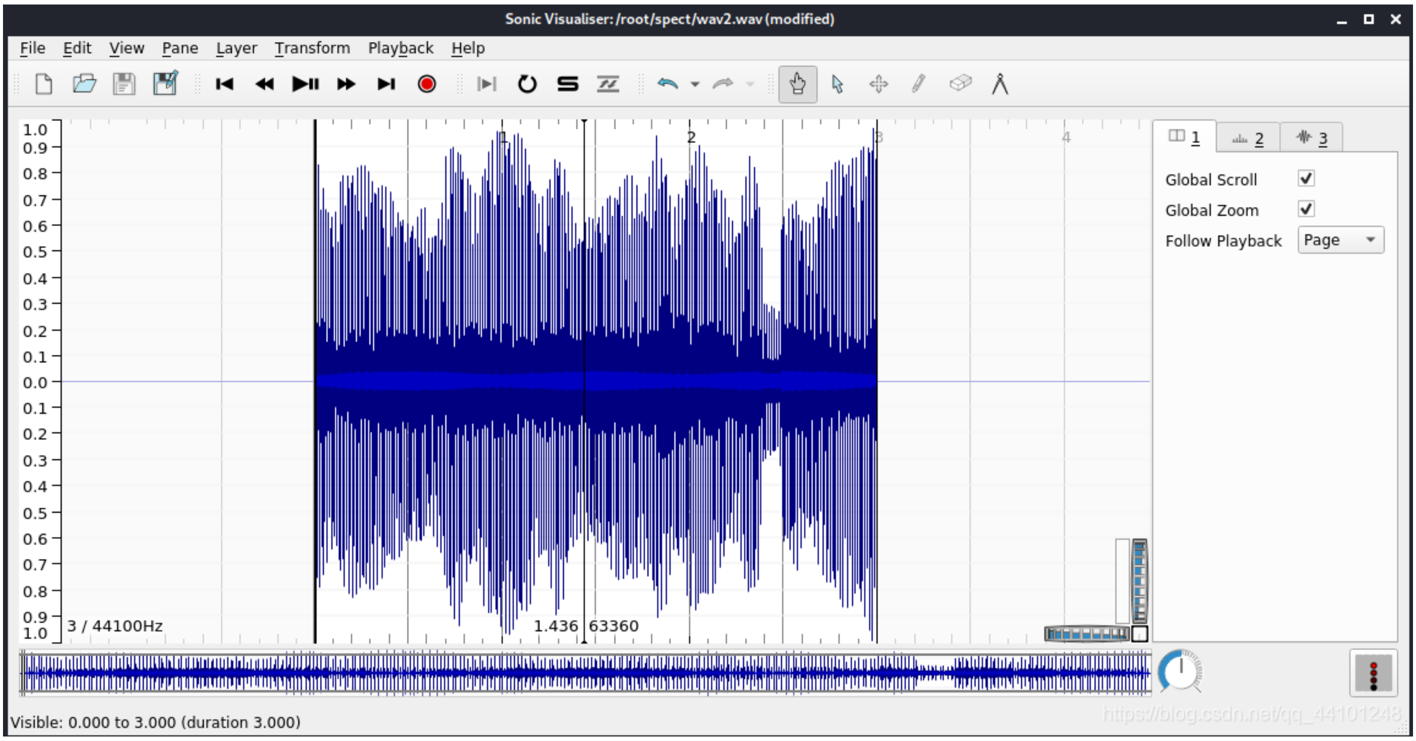
Stegoveritas

Stegoveritas几乎支持所有类型的文件提取，其它工具提取不到可以用它尝试，说不定有意想不到的收获。

提取文件

stegoveritas -steghide jpeg2.jpeg

提取出来后可以到results目录下查看

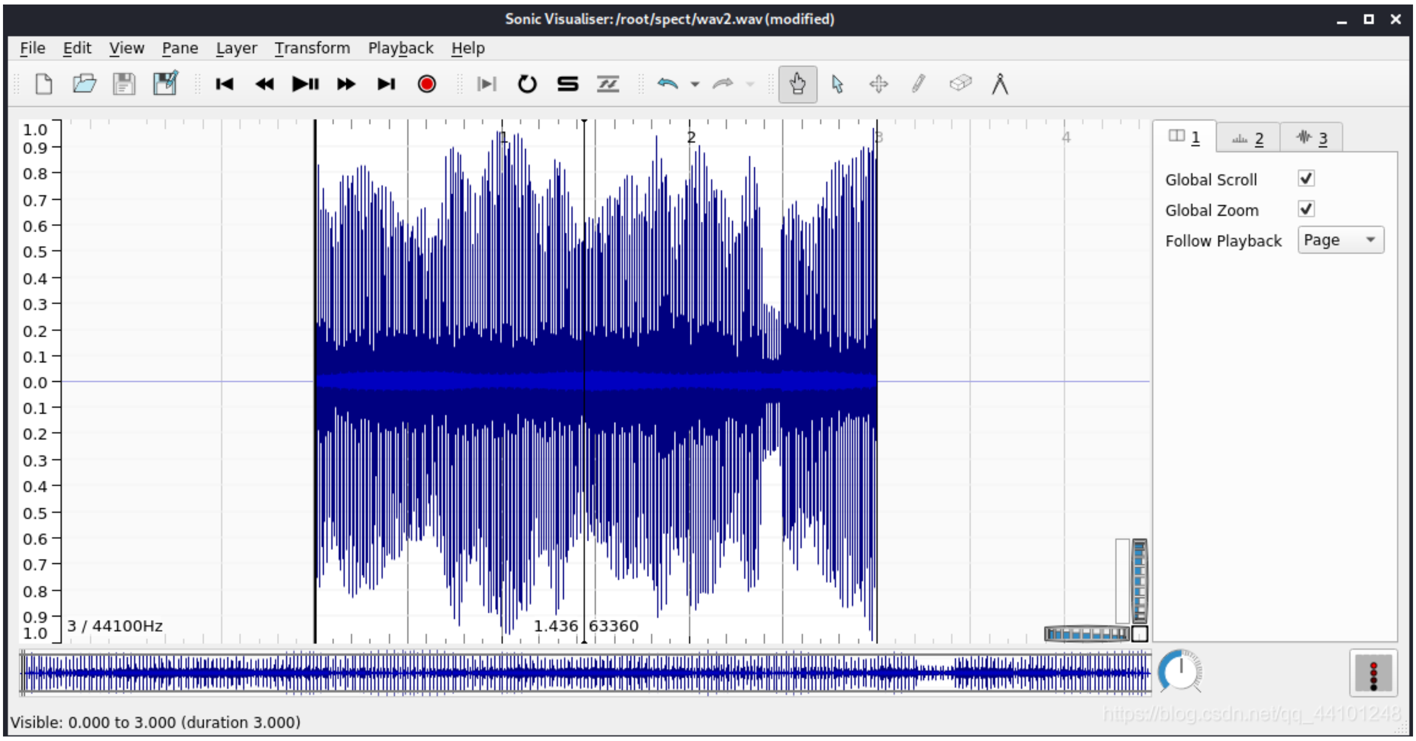

Sonic Visualiser: /root/spect/wav2.wav (modified)
File Edit View Pane Layer Transform Playback Help
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
3 / 44100Hz
1.436
63360
1
2
3
4
Global Scroll
Global Zoom
Follow Playback
Page
Visible: 0.000 to 3.000 (duration 3.000)
https://blog.csdn.net/qq_44101248

Sonic Visualizer

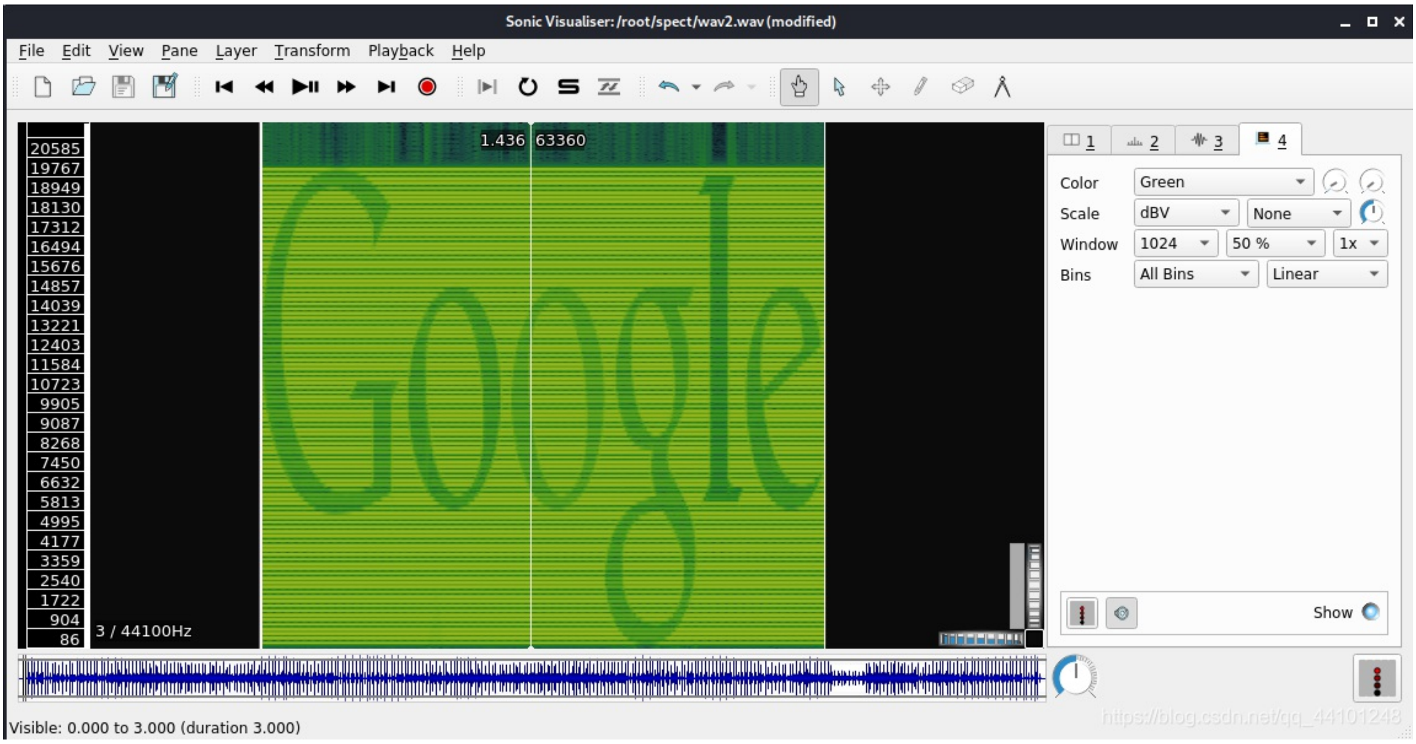
声谱隐写术是把信息放在音频文件里。需要通过某些工具才能看到，例如Sonic Visualizer

下载地址：https://www.sonicvisualiser.org/download.html

比如这个音频看似没什么异常


Sonic Visualiser: /root/spect/wav2.wav (modified)
File Edit View Pane Layer Transform Playback Help
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
3 / 44100Hz
1.436 63360
Global Scroll
Global Zoom
Follow Playback
Page
Visible: 0.000 to 3.000 (duration 3.000)
https://blog.csdn.net/qq_44101248

用Layer->Add Spectrogram增加光谱就能看到一个Google

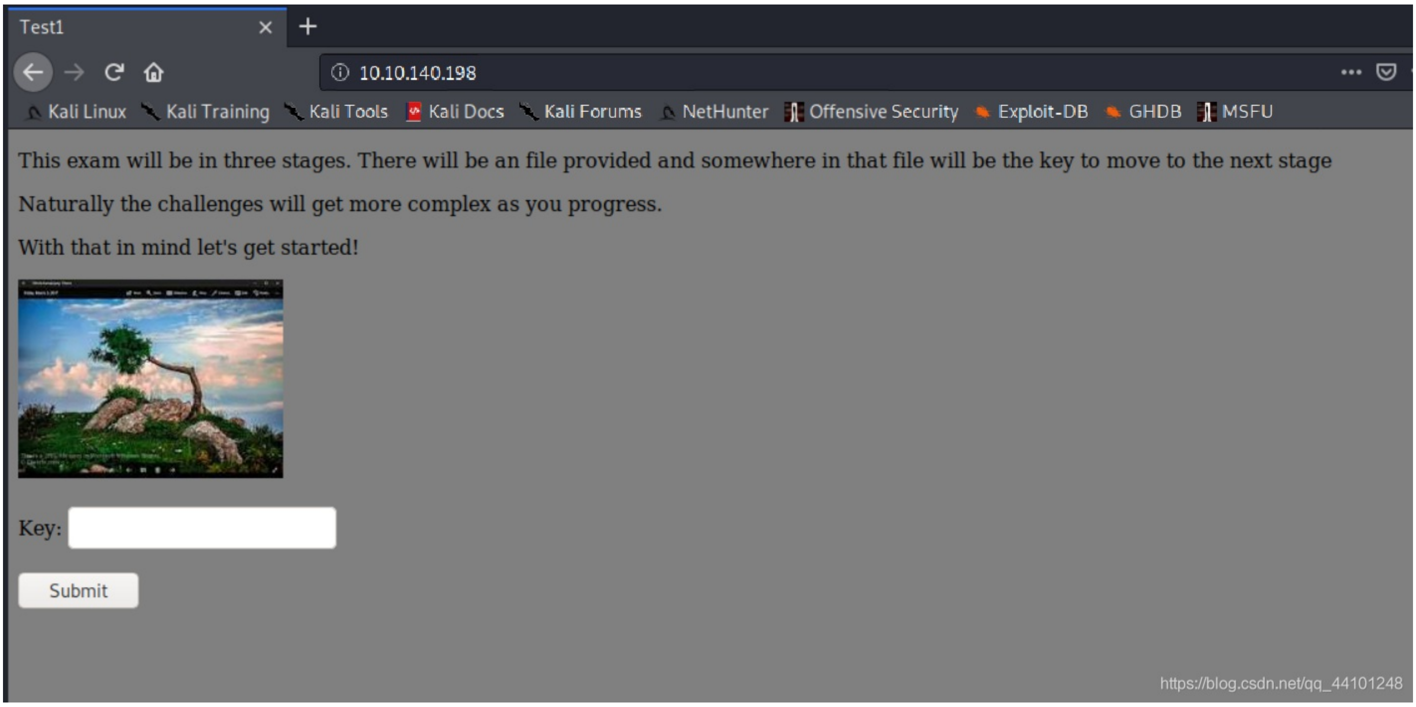


考试

一共有三关

TEST 1

登录进来就是图片要你输入Key，图片不能另存为，要复制链接用wget下载



This exam will be in three stages. There will be an file provided and somewhere in that file will be the key to move to the next stage

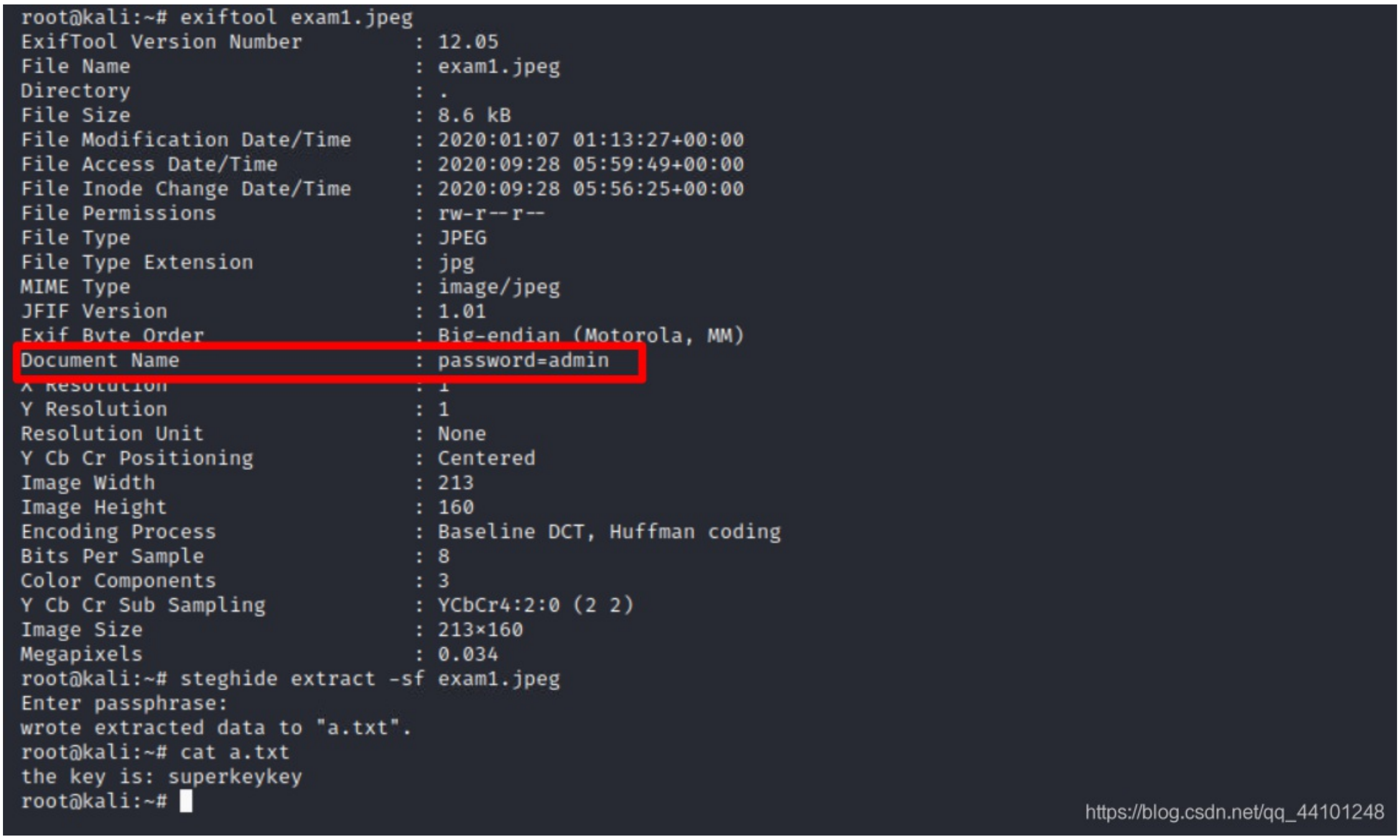
Naturally the challenges will get more complex as you progress.

With that in mind let's get started!

Key:

用exiftool查看一下图片信息得到一个password=admin

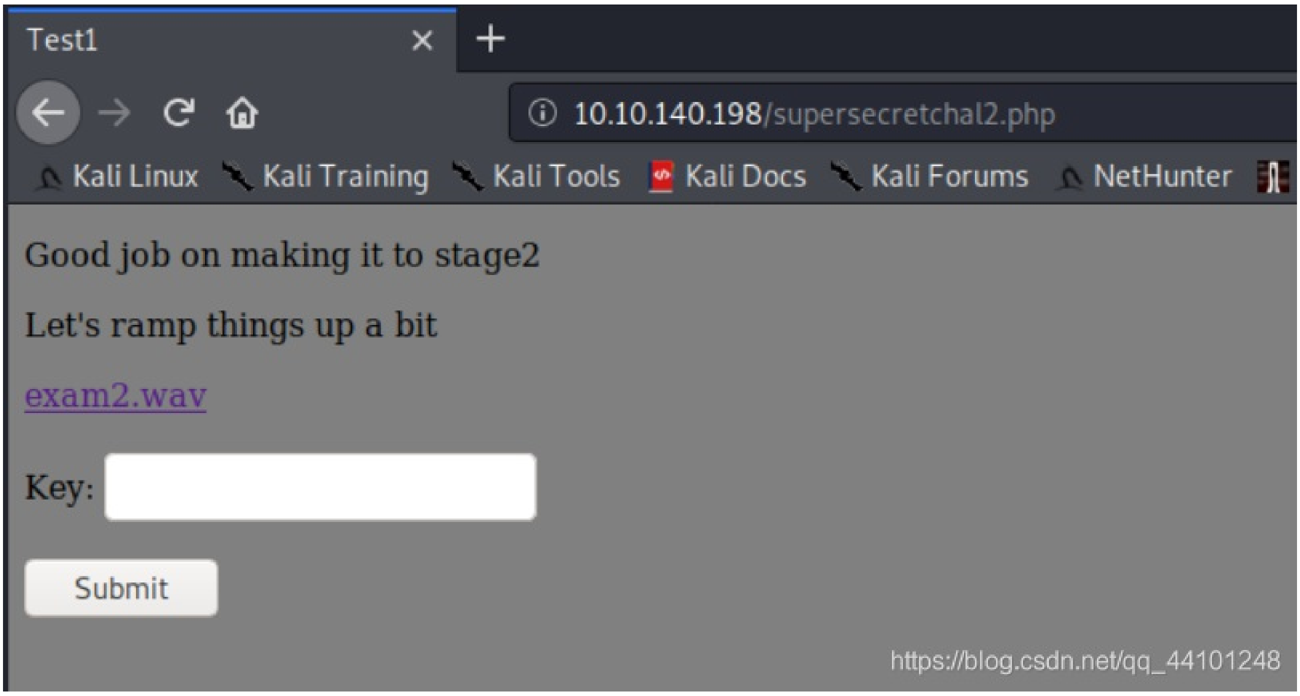
在用steghide分离得到a.txt，里面就是key



```
root@kali:~# exiftool exam1.jpeg
ExifTool Version Number         : 12.05
File Name                       : exam1.jpeg
Directory                       : .
File Size                       : 8.6 kB
File Modification Date/Time     : 2020:01:07 01:13:27+00:00
File Access Date/Time           : 2020:09:28 05:59:49+00:00
File Inode Change Date/Time     : 2020:09:28 05:56:25+00:00
File Permissions                : rw-r--r--
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Exif Byte Order                 : Big-endian (Motorola, MM)
Document Name                   : password=admin
X Resolution                    : 1
Y Resolution                    : 1
Resolution Unit                 : None
Y Cb Cr Positioning             : Centered
Image Width                     : 213
Image Height                    : 160
Encoding Process                : Baseline DCT, Huffman coding
Bits Per Sample                 : 8
Color Components                : 3
Y Cb Cr Sub Sampling            : YCbCr4:2:0 (2 2)
Image Size                      : 213x160
Megapixels                      : 0.034
root@kali:~# steghide extract -sf exam1.jpeg
Enter passphrase:
wrote extracted data to "a.txt".
root@kali:~# cat a.txt
the key is: superkeykey
root@kali:~# 
```

TEST 2

第二个进来就是一个音频文件



Good job on making it to stage2

Let's ramp things up a bit

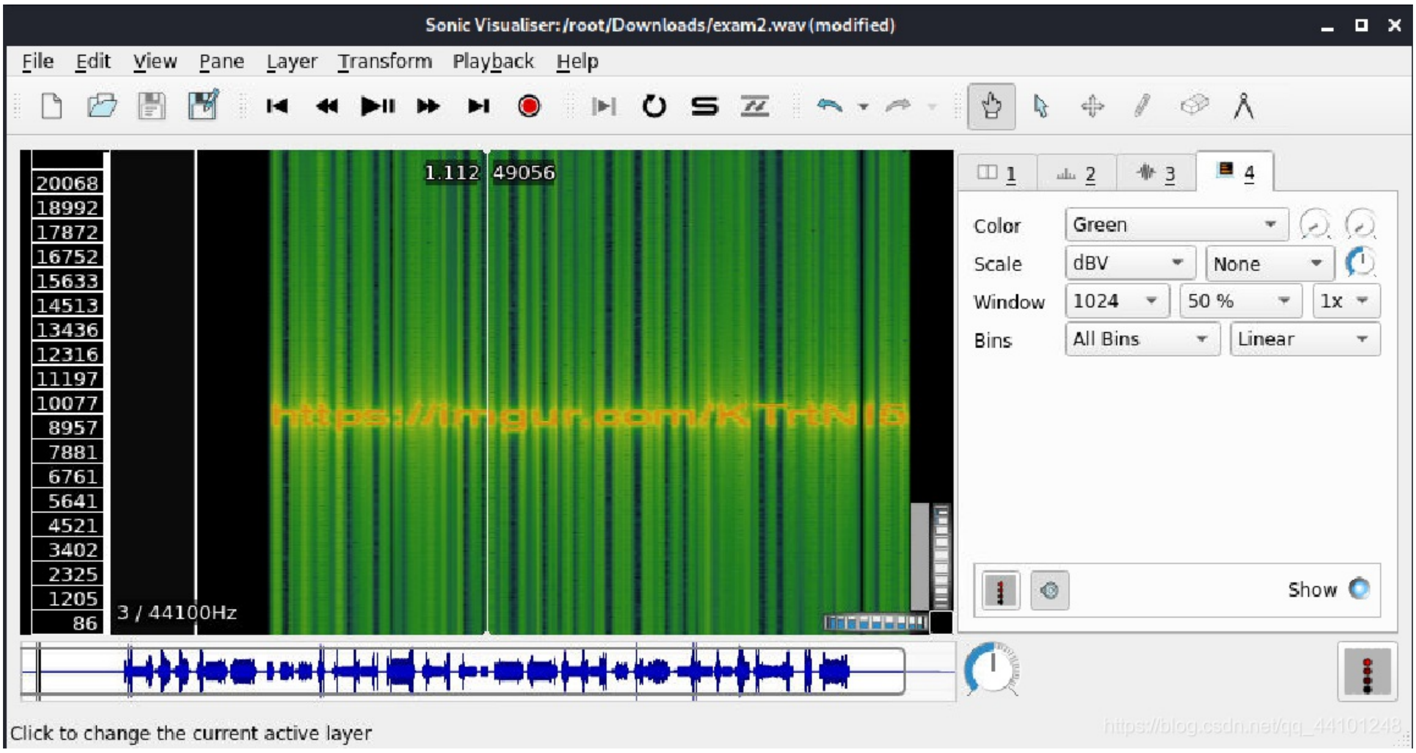
exam2.wav

Key:

Submit

下载下来用Sonic Visualizer光谱查看得到一个URL

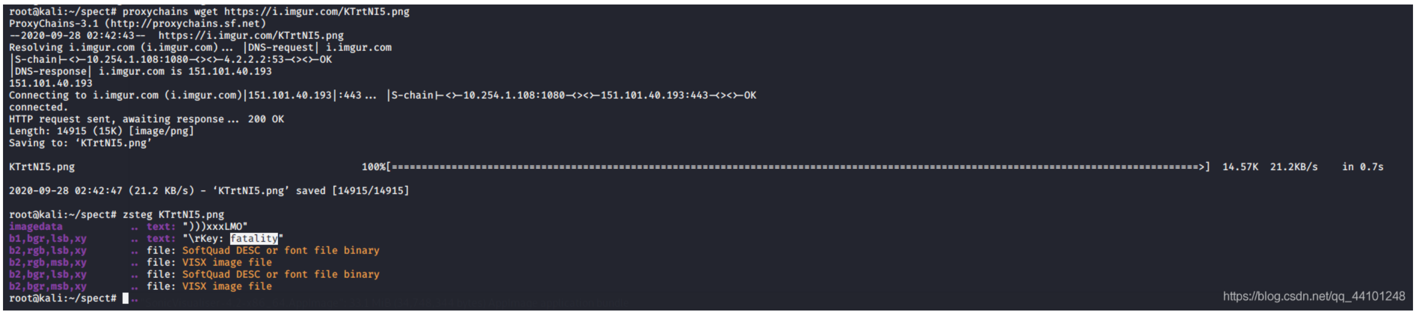
https://imgur.com/KTrtNI5



Color Green
Scale dBV None
Window 1024 50 % 1x
Bins All Bins Linear

Click to change the current active layer

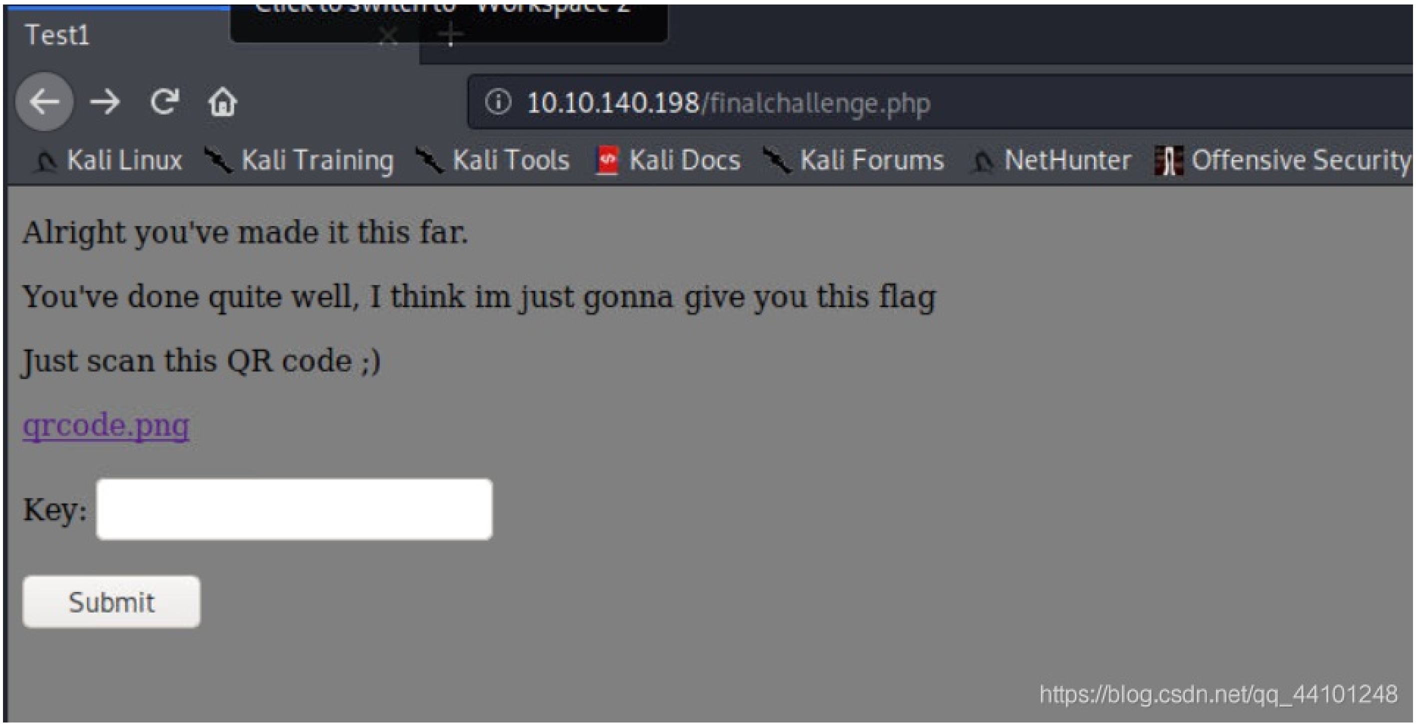
还是用wget下载得到一个png图片，用zsteg获取里面第二个Key



```
root@kali:~/spect# proxychains wget https://i.imgur.com/KTrtNI5.png
ProxyChains-3.1 (http://proxychains.sf.net)
--2020-09-28 02:42:43--  https://i.imgur.com/KTrtNI5.png
Resolving i.imgur.com (i.imgur.com)...  |DNS-request| i.imgur.com
|S-chain|-<>-10.254.1.108:1080-<><>-4.2.2.2:53-<><>-OK
|DNS-response| i.imgur.com is 151.101.40.193
151.101.40.193
Connecting to i.imgur.com (i.imgur.com)|151.101.40.193|:443...  |S-chain|-<>-10.254.1.108:1080-<><>-151.101.40.193:443-<><>-OK
connected.
HTTP request sent, awaiting response... 200 OK
Length: 14915 (15K) [image/png]
Saving to: ‘KTrtNI5.png’

KTrtNI5.png                          100%[==================================================================================================================>]  14.57K  21.2KB/s    in 0.7s

2020-09-28 02:42:47 (21.2 KB/s) - ‘KTrtNI5.png’ saved [14915/14915]

root@kali:~/spect# zsteg KTrtNI5.png
imagedata           .. text: ")))xxxLMO"
b1,bgr,lsb,xy       .. text: "\rKey: fatality"
b2,rgb,lsb,xy       .. file: SoftQuad DESC or font file binary
b2,rgb,msb,xy       .. file: VISX image file
b2,bgr,lsb,xy       .. file: SoftQuad DESC or font file binary
b2,bgr,msb,xy       .. file: VISX image file
root@kali:~/spect# █..
```

TEST 3

第三关是一张二维码的图片



Alright you've made it this far.

You've done quite well, I think im just gonna give you this flag

Just scan this QR code ;)

qrcode.png

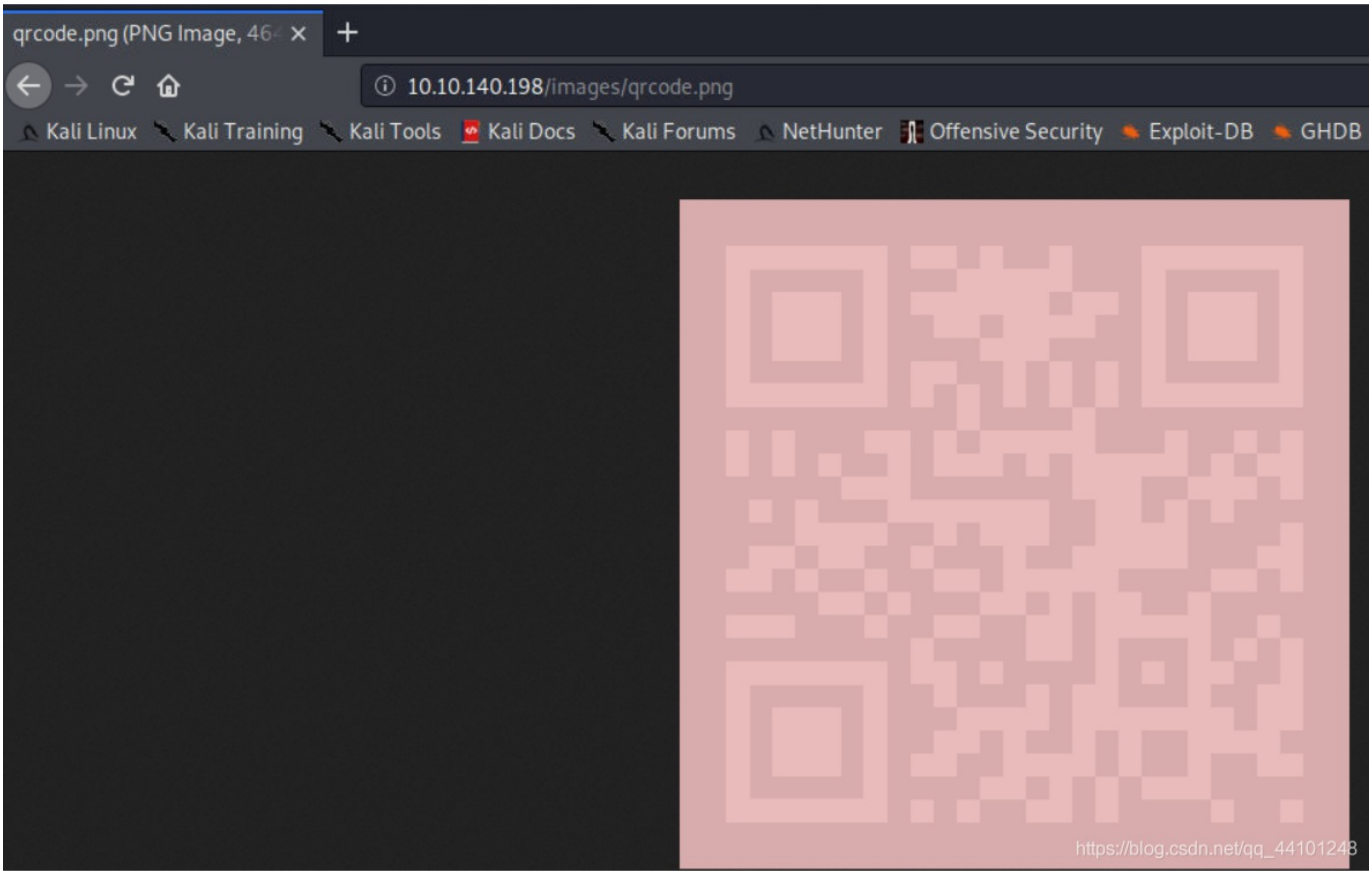
Key:

Submit

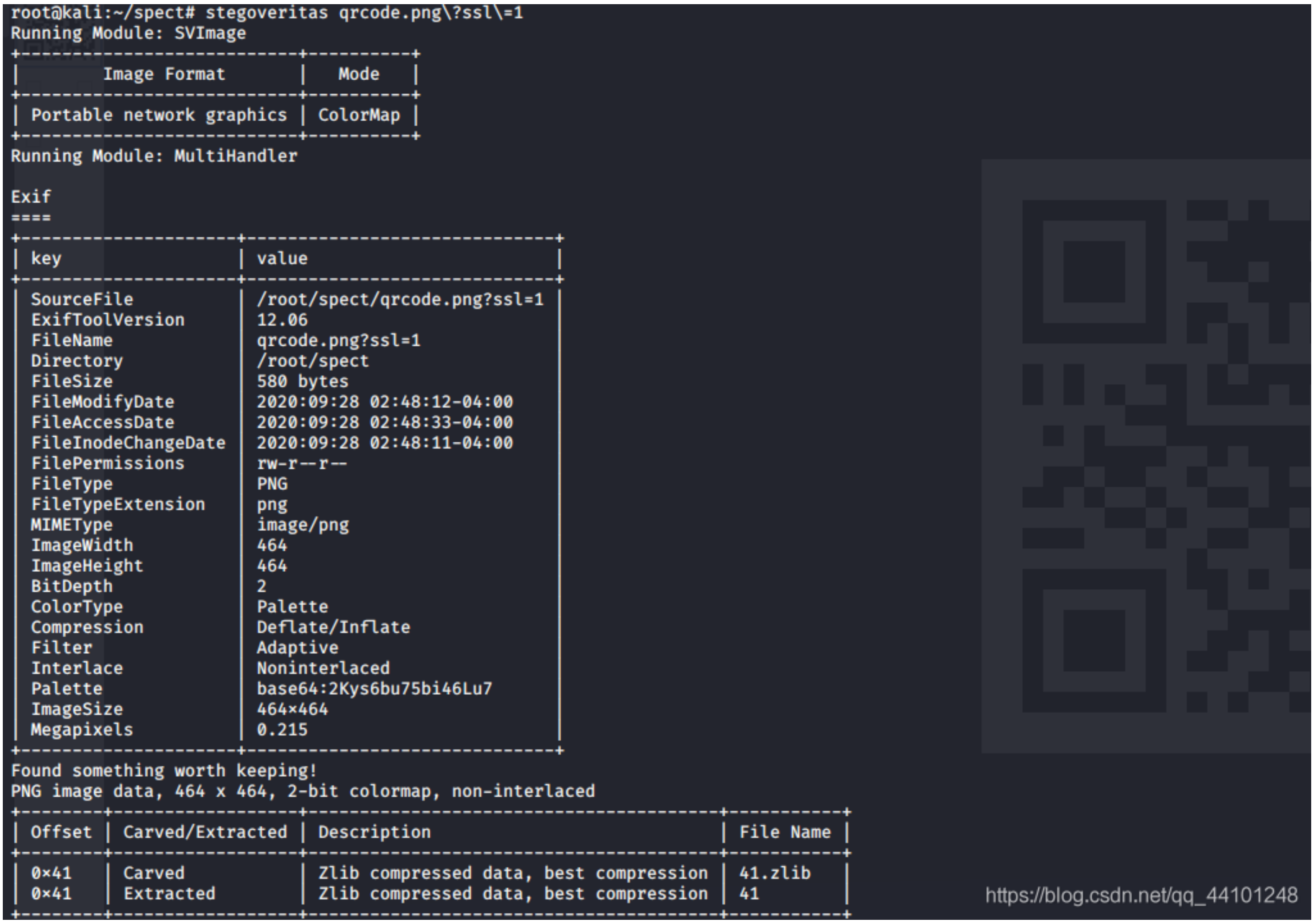
https://blog.csdn.net/qq_44101248

第一时间掏出手机扫了一下，扫不了

好像还有最后一个工具没有使用了，Stegoveritas



用Stegoveritas 分析一下图片会在当前目录下生成一个results文件夹

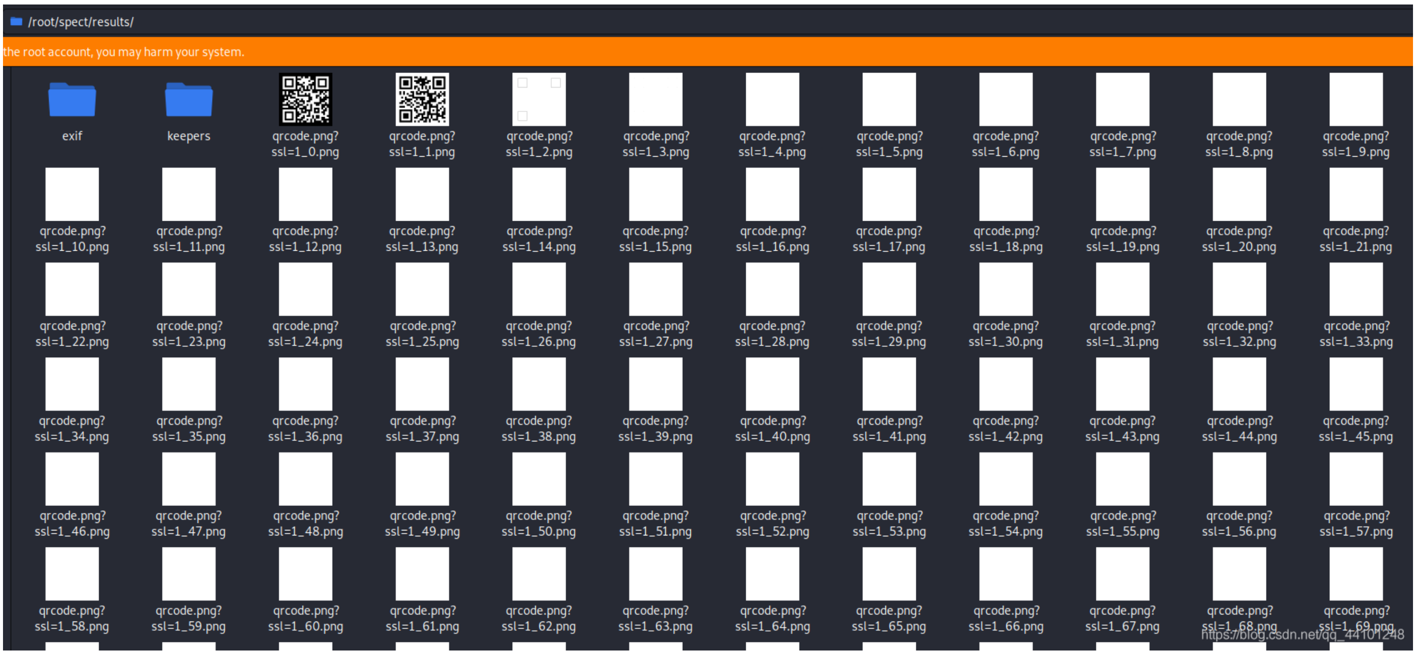


```
root@kali:~/spect# stegoveritas qrcode.png\?ssl\=1
Running Module: SVImage
+----------------------------+-----------+
|        Image Format        |   Mode    |
+----------------------------+-----------+
| Portable network graphics | ColorMap |
+----------------------------+-----------+
Running Module: MultiHandler

Exif
====
+---------------------+----------------------------+
| key                 | value                      |
+---------------------+----------------------------+
| SourceFile          | /root/spect/qrcode.png?ssl=1 |
| ExifToolVersion     | 12.06                      |
| FileName            | qrcode.png?ssl=1           |
| Directory           | /root/spect                |
| FileSize            | 580 bytes                  |
| FileModifyDate      | 2020:09:28 02:48:12-04:00  |
| FileAccessDate      | 2020:09:28 02:48:33-04:00  |
| FileInodeChangeDate | 2020:09:28 02:48:11-04:00  |
| FilePermissions     | rw-r--r--                  |
| FileType            | PNG                        |
| FileTypeExtension   | png                        |
| MIMEType            | image/png                  |
| ImageWidth          | 464                        |
| ImageHeight         | 464                        |
| BitDepth            | 2                          |
| ColorType           | Palette                    |
| Compression         | Deflate/Inflate            |
| Filter              | Adaptive                   |
| Interlace           | Noninterlaced              |
| Palette             | base64:2Kys6bu75bi46Lu7    |
| ImageSize           | 464x464                    |
| Megapixels          | 0.215                      |
+---------------------+----------------------------+
Found something worth keeping!
PNG image data, 464 x 464, 2-bit colormap, non-interlaced
+--------+-----------------+------------------------------------------+-----------+
| Offset | Carved/Extracted | Description                              | File Name |
+--------+-----------------+------------------------------------------+-----------+
| 0x41   | Carved          | Zlib compressed data, best compression | 41.zlib   |
| 0x41   | Extracted       | Zlib compressed data, best compression | 41        |
+--------+-----------------+------------------------------------------+-----------+
```

里面有各种调完色后的图片，Stegoveritas有调色的功能

因为二维码需要强烈的颜色差，我们看到的二维码都是黑白色的。

用手机浏览器扫描一下黑白的二维码就能得到key（扫描后看URL）



/root/spect/results/

the root account, you may harm your system.

exif
keepers
qrcode.png?ssl=1_0.png
qrcode.png?ssl=1_1.png
qrcode.png?ssl=1_2.png
qrcode.png?ssl=1_3.png
qrcode.png?ssl=1_4.png
qrcode.png?ssl=1_5.png
qrcode.png?ssl=1_6.png
qrcode.png?ssl=1_7.png
qrcode.png?ssl=1_8.png
qrcode.png?ssl=1_9.png
qrcode.png?ssl=1_10.png
qrcode.png?ssl=1_11.png
qrcode.png?ssl=1_12.png
qrcode.png?ssl=1_13.png
qrcode.png?ssl=1_14.png
qrcode.png?ssl=1_15.png
qrcode.png?ssl=1_16.png
qrcode.png?ssl=1_17.png
qrcode.png?ssl=1_18.png
qrcode.png?ssl=1_19.png
qrcode.png?ssl=1_20.png
qrcode.png?ssl=1_21.png
qrcode.png?ssl=1_22.png
qrcode.png?ssl=1_23.png
qrcode.png?ssl=1_24.png
qrcode.png?ssl=1_25.png
qrcode.png?ssl=1_26.png
qrcode.png?ssl=1_27.png
qrcode.png?ssl=1_28.png
qrcode.png?ssl=1_29.png
qrcode.png?ssl=1_30.png
qrcode.png?ssl=1_31.png
qrcode.png?ssl=1_32.png
qrcode.png?ssl=1_33.png
qrcode.png?ssl=1_34.png
qrcode.png?ssl=1_35.png
qrcode.png?ssl=1_36.png
qrcode.png?ssl=1_37.png
qrcode.png?ssl=1_38.png
qrcode.png?ssl=1_39.png
qrcode.png?ssl=1_40.png
qrcode.png?ssl=1_41.png
qrcode.png?ssl=1_42.png
qrcode.png?ssl=1_43.png
qrcode.png?ssl=1_44.png
qrcode.png?ssl=1_45.png
qrcode.png?ssl=1_46.png
qrcode.png?ssl=1_47.png
qrcode.png?ssl=1_48.png
qrcode.png?ssl=1_49.png
qrcode.png?ssl=1_50.png
qrcode.png?ssl=1_51.png
qrcode.png?ssl=1_52.png
qrcode.png?ssl=1_53.png
qrcode.png?ssl=1_54.png
qrcode.png?ssl=1_55.png
qrcode.png?ssl=1_56.png
qrcode.png?ssl=1_57.png
qrcode.png?ssl=1_58.png
qrcode.png?ssl=1_59.png
qrcode.png?ssl=1_60.png
qrcode.png?ssl=1_61.png
qrcode.png?ssl=1_62.png
qrcode.png?ssl=1_63.png
qrcode.png?ssl=1_64.png
qrcode.png?ssl=1_65.png
qrcode.png?ssl=1_66.png
qrcode.png?ssl=1_67.png
qrcode.png?ssl=1_68.png
qrcode.png?ssl=1_69.png