

图片隐写题思路

原创

stars77 于 2019-05-31 17:00:32 发布 1445 收藏 6

分类专栏: [CTF](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/weixin_42040292/article/details/90719439

版权



[CTF 专栏收录该内容](#)

4 篇文章 0 订阅

订阅专栏

图片隐写题思路

做CTF杂项的时候会碰到让下载图片, 从图片里面找flag, 这里介绍两个软件binwalk和foremost, 它可以查看下载的文件中是否包含其它文件。

0X01 binwalk相关命令

打开kali, 直接在终端输入binwalk (前提是kali安装了这个软件, 如果没有需要自己搜一下具体安装方法)。打开软件后, 使用命令“binwalk -e 文件路径”就可以分离所需要分离的文件了。

这里以bugku一道题为例, 下载后来得到如下图片:

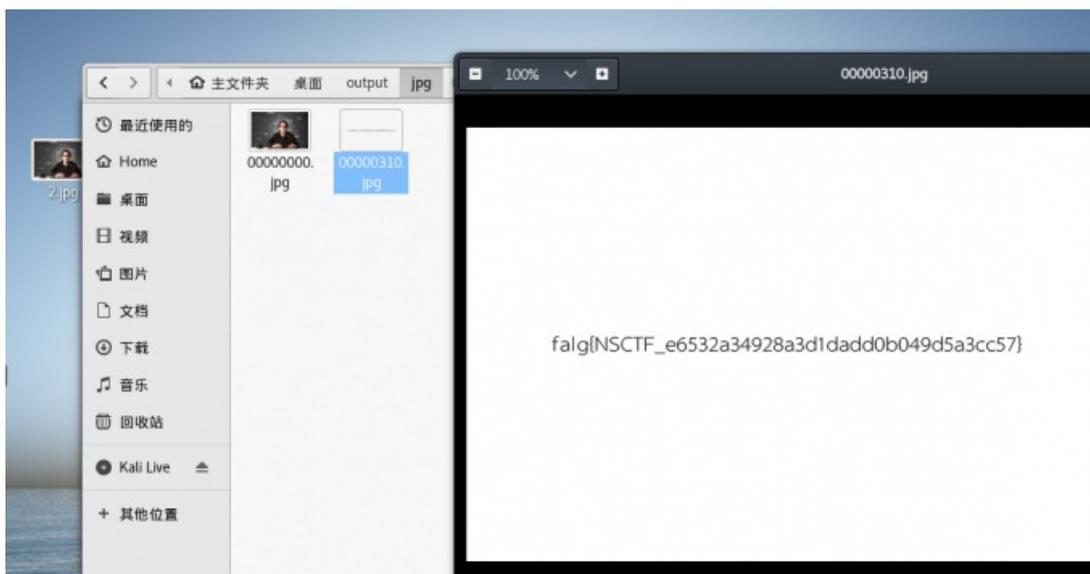


将这张图片放到桌面并用binwalk打开它

```
root@kali: ~/桌面
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
root@kali:~# cd 桌面
root@kali:~/桌面# binwalk 2.jpg
```

```
root@kali: ~/桌面
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
root@kali:~# cd 桌面
root@kali:~/桌面# binwalk 2.jpg
DECIMAL      HEXADECIMAL  DESCRIPTION
-----
0            0x0         JPEG image data, EXIF standard
12           0xC        TIFF image data, big-endian, offset of first image directory: 8
13017        0x32D9     Unix path: /www.w3.org/1999/02/22-rdf-syntax-ns#> <rdf:Description rdf:about="
" xmlns:photoshop="http://ns.adobe.com/photoshop/1.0/" xmlns
158792       0x26C48    JPEG image data, JFIF standard 1.02
158822       0x26C66    TIFF image data, big-endian, offset of first image directory: 8
159124       0x26D94    JPEG image data, JFIF standard 1.02
162196       0x27994    JPEG image data, JFIF standard 1.02
164186       0x2815A    Unix path: /www.w3.org/1999/02/22-rdf-syntax-ns#> <rdf:Description rdf:about="
" xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:xap="htt
168370       0x291B2    Copyright string: "Copyright (c) 1998 Hewlett-Packard Company"
```

这里可以看到这张图片里包含了两个图片文件，159124和162196，用命令“binwalk -e 文件路径”将其分离随后得到另一张含有flag的图片



0X02 foremost 相关命令

foremost的命令是“foremost -T 文件名”，然后会在桌面有一个输出文件夹，里面就会有分离出来的含有flag的图片。

0X03 "dd" 相关命令

使用方法如下

```
root@kali:~/桌面# dd if=2.jpg of=2-1.jpg skip=158792 bs=1
记录了27689+0 的读入
记录了27689+0 的写出
27689 bytes (28 kB, 27 KiB) copied, 0.0785581 s, 352 kB/s
```

相关的其它命令如下

dd命令使用详解

1.命令简介

dd 的主要选项:

指定数字的地方若以下列字符结尾乘以相应的数字:

b=512, c=1, k=1024, w=2, xm=number m

if=file #输入文件名, 缺省为标准输入。

of=file #输出文件名, 缺省为标准输出。

ibs=bytes #一次读入 bytes 个字节(即一个块大小为 bytes 个字节)。

obs=bytes #一次写 bytes 个字节(即一个块大小为 bytes 个字节)。

bs=bytes #同时设置读写块的大小为 bytes, 可代替 ibs 和 obs。

cbs=bytes #一次转换 bytes 个字节, 即转换缓冲区大小。

skip=blocks #从输入文件开头跳过 blocks 个块后再开始复制。

seek=blocks #从输出文件开头跳过 blocks 个块后再开始复制。(通常只有当输出文件是磁盘或磁带时才有效)。

count=blocks #仅拷贝 blocks 个块, 块大小等于 ibs 指定的字节数。

conv=conversion[,conversion...] #用指定的参数转换文件。