Misc 总结 ----隐写术之图片隐写(四)

M1n3 / 2017-12-29 23:03:00 / 浏览数 12038 安全技术 CTF 顶(1) 踩(0)

# 实验简介

隐写术是关于信息隐藏,即不让计划的接收者之外的任何人知道信息的传递事件(而不只是信息的内容)的一门技巧与科学。英文写作Steganography,而本套教程内容将

# 实验内容

本次图片隐写实验包括四大部分

- 一、附加式的图片隐写
- 二、基于文件结构的图片隐写
- 三、基于LSB原理的图片隐写
- 四、基于DCT域的JPG图片隐写
- 五、数字水印的隐写
- 六、图片容差的隐写

## 第四部分 基于DCT域的JPG图片隐写

- 操作机: Windows XP
  - 实验工具:
    - Stegdetect
    - Jphide
    - Outguess

#### 背景知识

JPEG图像格式使用离散余弦变换 ( Discrete Cosine

Transform, DCT)函数来压缩图像,而这个图像压缩方法的核心是:通过识别每个8×8像素块中相邻像素中的重复像素来减少显示图像所需的位数,并使用近似估算法降低Compression)技术,但一般不会影响图像的视觉效果。

在这个隐写家族中,常见的隐写方法有JSteg、JPHide、Outguess、F5等等 Stegdetect

实现JPEG图像Jphide隐写算法工具有多个,比如由Neils

Provos开发通过统计分析技术评估JPEG文件的DCT频率系数的隐写工具Stegdetect,它可以检测到通过JSteg、JPHide、OutGuess、Invisible

Secrets、F5、appendX和Camouflage等这些隐写工具隐藏的信息,并且还具有基于字典暴力破解密码方法提取通过Jphide、outguess和jsteg-shell方式嵌入的隐藏信息。 JPHS

一款JPEG图像的信息隐藏软件JPHS,它是由Allan

Latham开发设计实现在Windows和Linux系统平台针对有损压缩JPEG文件进行信息加密隐藏和探测提取的工具。软件里面主要包含了两个程序JPHIDE和JPSEEK,

JPHIDE程序主要是实现将信息文件加密隐藏到JPEG图像功能,而JPSEEK程序主要实现从用JPHIDE程序加密隐藏得到的JPEG图像探测提取信息文件,Windows版本的JPHS Outquess

Outgusee算法是Niels Provos针对Jsteg算法的缺陷提出的一种方法:

- 嵌入过程不修改ECT系数值为0,1的DCT系数,利用为随机数发生器产生间隔以决定下一个要嵌入的DCT系数的位置
- 纠正过程消除对效应的出现

对应的,也有针对该算法的隐写工具,名字也叫Outguess。

## **JPHS**

### 实验

- ■■■■■■■JPhide.jpg
- ----
- **BE**Stegdetect**BEBBBBBBBBB**
- **INCOME SET OF STREET OF**
- **BESSESS**flag**E**flag**(Good\_you\_got\_it)**

Stegdetect的指令介绍

- -q
- -n DINTERSON -n DI
- -d **| | | | | | | |**
- -t
- j **IIIIIIIIIIII** jsteq**IIII**
- o
- p **IIIIIIIIIIIII**jphide**IIIII**
- -V **-V**

Stegdetect 1131113

首先,在cmd中运行,Stegdetect,对目标图片进行检测

发现结果显示是jphide隐写的可能性很大。

接着,我们使用工具JPHS提取信息

这是一款针对Jphide算法的隐写工具,正如对症下药,我们也得用对工具,才能更好的解决问题。

首先,在使用JPHS打开图片,点击Seek功能,紧接着会弹出一个密码的输入框,我们这里默认为空口令,直接点击OK,将提取出来的信息保存为flag.txt。

### Outquess

### 实验

- -----
- ■■■■■■Outguess.jpg
- ----
- **Stegdetect**
- **IIIIIIIIIII**flag**I**flag**[**Good\_you\_got\_it]

第一步,在cmd中运行,Stegdetect,对目标图片进行检测

从结果可以得知,这里的隐写方式是,Outguess。

第二步,使用Outguess工具提取隐写信息

点击Enter key功能,输入密码,这里我隐写的时候,使用的密码是123456 点击Load image 加载目标图片 点击Extract file功能,将提取出来的信息保存成flag.txt文件。 打开保存后的文件,flag是flag{Outguess}

#### 小结:

从上面的实验来看,JPG图片常用的隐写方式一般也是DCT域的隐写了,不过一般在CTF赛场上,这种题目是可以直接用Stegdetect所检测出来的。

### 思考

- 1. 尝试其他DCT域隐写,如F5
- 2. 自己尝试使用JPHS的Hide功能,实现一个隐写。

下面进行实验Part 5 数字水印隐写

# 第五部分 数字水印隐写

- 操作机: Windows XP
  - 实验工具:
    - Matlab

# 背景知识

#### 数字水印

数字水印(digital watermark)技术,是指在数字化的数据内容中嵌入不明显的记号。 特征是,被嵌入的记号通常是不可见或不可察的,但是可以通过计算操作检测或者提取。 盲水印与傅里叶变换 这里介绍的盲水印是以知乎某答主的频域添加盲水印的文章为基础,在2016HCTF的也出了一个隐写题目,也是以频域为背景的。 盲水印,是指人感知不到的水印,包括看不到或听不见(没错,数字盲水印也能够用于音频)。其主要应用于音像作品、数字图书等,目的是,在不破坏原始作品的情况下

对图像进行傅里叶变换,起始是一个二维离散傅里叶变换,图像的频率是指图像灰度变换的强烈程度,将二维图像由空间域变为频域后,图像上的每个点的值都变成了复数

## 频域盲水印隐写

#### 实验

- ----
- ■■■■■■■ori.jpg■res,png
- \_

```
有必要提以下的是,如果用mathlab生成的盲水印隐写是不需要原图的,这里我能力有限,只能做到需要原图才能提取水印。
```

```
python decode.py --original <original image file> --image <image file> --result <result file>
# coding=utf-8
import cv2
import numpy as np
import random
import os
from argparse import ArgumentParser
ALPHA = 5
def build_parser():
   parser = ArgumentParser()
   parser.add_argument('--original', dest='ori', required=True)
  parser.add_argument('--image', dest='img', required=True)
  parser.add_argument('--result', dest='res', required=True)
  parser.add_argument('--alpha', dest='alpha', default=ALPHA)
   return parser
def main():
   parser = build_parser()
   options = parser.parse_args()
   ori = options.ori
   img = options.img
   res = options.res
   alpha = options.alpha
   if not os.path.isfile(ori):
       parser.error("original image %s does not exist." % ori)
   if not os.path.isfile(img):
       parser.error("image %s does not exist." % img)
   decode(ori, img, res, alpha)
def decode(ori_path, img_path, res_path, alpha):
   ori = cv2.imread(ori_path)
   img = cv2.imread(img_path)
   ori_f = np.fft.fft2(ori)
   img_f = np.fft.fft2(img)
   height, width = ori.shape[0], ori.shape[1]
   watermark = (ori_f - img_f) / alpha
   watermark = np.real(watermark)
   res = np.zeros(watermark.shape)
   random.seed(height + width)
   x = range(height / 2)
   y = range(width)
   random.shuffle(x)
   random.shuffle(y)
   for i in range(height / 2):
       for j in range(width):
           res[x[i]][y[j]] = watermark[i][j]
   cv2.imwrite(res_path, res, [int(cv2.IMWRITE_JPEG_QUALITY), 100])
```

```
if __name__ == '__main__':
    main()
```

#### 提取盲水印

original 是输入原图 ,image是之后跟的是加入了水印的图 ,result是保存水印图片。

#### 结果

如果是像HCTF那样的隐写题,只需要有mathlab这个强大的工具,再运用提取盲水印的代码,是不需要原图的、代码如下。

```
imageA = imread('3.bmp','bmp');
fftA = fft2(imageA);
imshow(fftshift(fftA))
imshow(fft(rgb2gray(imread('shimakaze.bmp'))), [1,2]);
```

## 思考

- 1. 请查阅关于空域盲水印的资料
- 2. 试着对频域盲水印攻击,如截屏、倒转等操作,再进行提取水印,看看水印是否被破坏。

下面进行实验Part 6 图片容差隐写

# 第六部分 图片容差隐写

- 操作机: Windows XP
  - 实验工具:
    - · Beyond Compare

## 背景知识

#### 容差

容差,在选取颜色时所设置的选取范围,容差越大,选取的范围也越大,其数值是在0-255之间。

### 容差比较的隐写

beyond compare

beyond compare是一款很适合用来对图片进行比较的工具,就图片而言,它支持容差、范围、混合等模式。

- 实验:
- -----
- ■■■■■■ pokemon\_1.jpg■pokemon\_2.jpg
- BellBeyond Compare
- ----
- **IIIIIIIIIIIII**flag

打开工具,选择图片比较,导入pokemon\_1.jpg,pokemon\_2.jpg

选择容差模式,并调整容差大小

小结:

如果在CTF赛场中,就隐写这一部分,出题人给于两张或者多张图片,一般都是需要对图片的内容进行比较的。

# 思考

1. Stegsolve 也有图片的比较的功能,是否能完成这个隐写?如果不可以为什么?

DCT.zip (0.074 MB) <u>下载附件</u>

点击收藏 | 3 关注 | 3

上一篇: Pentest Wiki Part... 下一篇: 机器学习之安全数据集

1. 8 条回复



<u>老锥</u> 2018-01-23 16:59:40

支持

0 回复Ta



<u>187\*\*\*\*5199</u> 2018-05-28 19:20:30

请问图片容差隐写的那两张图的资源有吗?

0 回复Ta



<u>1815837370479554</u> 2018-05-29 13:59:01

支持

0 回复Ta



small\*\*\*\*@live.c 2018-07-11 15:25:42

666

0 回复Ta



wuq\*\*\*\*@126.com 2018-07-21 08:32:46

支持

0 回复Ta



mr.bingo 2018-07-27 00:01:36

太厉害了!!!

0 回复Ta



<u>zhi\*\*\*\*</u> 2018-10-30 16:42:28

学习了

0 回复Ta



robert\*\*\*\* 2019-10-02 23:29:34

好好学习了

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板