

# python图像隐写技术\_Python3 图片隐写术

[weixin\\_39837124](#) 于 2020-12-10 13:27:25 发布 264 收藏

文章标签: [python图像隐写技术](#)

先看看下面两张图,有什么差别。

看不出来吧?其实右图中隐藏了一段文字,只是通过一般手段无法察觉,需要编码才能破解,这就用到了一种黑科技——隐写术。隐写术是一门关于信息隐藏的技巧与科学,所谓信息隐藏指的是不让除预期的接收者之外的任何人知晓信息的传递事件或者信息的内容。

这是维基百科中对隐写术的简介。

隐写技术是信息隐藏技术中很重要的一种技术,它以图像、文本、音频、视频等数字媒体作为载体,将秘密信息嵌入到原始载体,并以不引起第三方注意的方式在公共信道上传输。如果说加密技术是保护消息的内容,那么数字隐写技术就是隐藏消息的存在。

作为不可信网络环境中的一种新型保密通信手段,隐写术在学术界和国防安全领域受到了普遍关注。

通过这个教程带你走入隐写术的大门。

教程是由FrostSigh发布在实验楼,完整教程及在线练习地址: [Python3 图片隐写术](#)

项目名称:

项目简介:

通过Python3实现将关键信息隐藏在图片的效果,主要目的是为了不让预期接收者以外的人知晓传递的内容。与电视剧中使用特殊墨水传递信息一样,表面看就是一张什么都没写的白纸,实则暗藏着重要信息。

项目效果图:

可以看到“施法”前后的图片肉眼看不出区别,然而图片却真实的隐藏了一些数据在里面。

隐藏的文字

涉及知识点: Pillow 模块

最低有效位

lambda 表达式递归

UTF-8 编码

实验原理:

还是引用 wikipedia 的解释：载体文件(cover file)相对隐秘文件的大小(指数据含量，以比特计)越大，隐藏后者就越加容易。因为这个原因，数字图像(包含有大量的数据)在因特网和其他传媒上被广泛用于隐藏消息。这种方法使用的广泛程度无从查考。例如：一个24位的位图中的每个像素的三个颜色分量(红，绿和蓝)各使用8个比特来表示。如果我们只考虑蓝色的话，就是说有 $2^8$ 种不同的数值来表示深浅不同的蓝色。而像11111111和11111110这两个值所表示的蓝色，人眼几乎无法区分。因此，这个最低有效位就可以用来存储颜色之外的信息，而且在某种程度上几乎是检测不到的。如果对红色和绿色进行同样的操作，就可以在差不多三个像素中存储一个字节的信息。更正式一点地说，使隐写的信息难以探测的，也就是保证“有效载荷”(需要被隐蔽的信号)对“载体”(即原始的信号)的调制对载体的影响看起来(理想状况下甚至在统计上)可以忽略。这就是说，这种改变应该无法与载体中的噪声加以区别。(从信息论的观点来看，这就是说信道的容量必须大于传输“表面上”的信号的需求。这就叫做信道的冗余。对于一幅数字图像，这种冗余可能是成像单元的噪声；对于数字音频，可能是录音或者放大设备所产生的噪声。任何有着模拟放大级的系统都会有所谓的热噪声(或称“1/f”噪声)，这可以用作掩饰。另外，有损压缩技术(如JPEG)会在解压后的数据中引入一些误差，利用这些误差作隐写术用途也是可能的。)隐写术也可以用作数字水印，这里一条消息(往往只是一个标识符)被隐藏到一幅图像中，使得其来源能够被跟踪或校验。

总而言之，这个教程便是利用图片四个颜色分量(rgba)的最低有效位(英语：Least Significant Bit, Lsb)来隐藏信息(本实验隐藏的是文字)

### 实现步骤

这个项目教程，通过4个步骤来教你完全了解图片的隐写术原理：安装开发所需要的包：实验中会用到 Pillow，所以需要安装 Pillow 依赖包；

编码：将隐藏信息编码到图片中；

解码：将接收到的图片进行解码，识别所隐藏的信息；

测试：项目的最后，当时是需要进行测试咯，看看是否可以准确的识别出图片所隐藏的信息；

关于项目的项目开发文档、代码、步骤等，[点击这里](#)即可进行查看啦。

作者：实验楼

链接：<https://www.jianshu.com/p/ac2c1d384be4>