

python隐写_Python中的隐写术 – Stepic

原创

[weixin_39987211](#) 于 2020-12-15 17:01:40 发布 144 收藏

文章标签: [python隐写](#)

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_39987211/article/details/111431978

版权

stepic 0.3在图像中使用最简单的隐写方法.直接从模块引用:

```
def decode_imdata(imdata):
```

```
    """Given a sequence of pixels, returns an iterator of characters
```

```
    encoded in the image"""
```

```
    imdata = iter(imdata)
```

```
    while True:
```

```
        pixels = list(imdata.next()[:3] + imdata.next()[:3] + imdata.next()[:3])
```

```
        byte = 0
```

```
        for c in xrange(7):
```

```
            byte |= pixels[c] & 1
```

```
            byte <<= 1
```

```
            byte |= pixels[7] & 1
```

```
            yield chr(byte)
```

```
        if pixels[-1] & 1:
```

```
            break
```

秘密数据的每个八位字节加上一个标志是否是最后一个字节,都隐藏在三个连续的像素中.更准确地说,stepic使用每个像素的前三个分量(通常是RGB)的最低有效位.看到这个非常丑陋的图表,对于每个组件有4位的RGBA流(D表示数据,E表示流末尾):

```
| pixel 0 | pixel 1 | pixel 2 |
```

```
image viewer sees: | rrrr gggg bbbb aaaa | rrrr gggg bbbb aaaa | rrrr gggg bbbb aaaa |
```

```
stepic sees: | __D __D __D ____ | __D __D __D ____ | __D __D __E ____ |
```

因为这种变化引入的噪声在已经“嘈杂”的图像(第256个)中很小,所以通常无法在视觉上真正检测到这种噪声.这意味着实现了这项技术的目标:数据隐藏在明显的视线中,因为没有人可以将它与自然发生的噪音区分开来.

这很有效.至少,它适用于无损格式,例如PNG.唉,JPG不是无损的,它的压缩很可能会改变至少一个编码位.它足以改变第九位以使该方法变得无用,因为隐藏数据将被截断为单个字节.

JPG图像中的隐写术仍然可以用于多种形式,但你不能仅仅调整解码的像素值.更好(但更复杂)的方法可能是将数据隐藏在压缩器估计的参数中.