

隐写术分析

原创

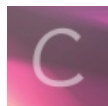
lmn_ 于 2022-03-01 10:10:15 发布 1093 收藏

分类专栏: [密码学](#) [网络安全技术](#) 文章标签: [安全](#) [web安全](#)

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_43211186/article/details/123199021

版权



[密码学](#) 同时被 2 个专栏收录

18 篇文章 0 订阅

订阅专栏



[网络安全技术](#)

16 篇文章 1 订阅

订阅专栏

文章目录

[0x01 隐写术概要](#)

[0x02 特征](#)

[0x03 隐写术应用](#)

[0x04 题目解析思路](#)

0x01 隐写术概要

首先看一下官方定义, “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。”

简单说就是将一种东西隐藏在另一种东西中, 在现实中例如图片版权问题, 就可以看似运用隐写技术将所有者的id或身份验证添加到图片中, 也可以应用于核实身份

在CTF的Misc中, 经常会看到关于隐写的题目

0x02 特征

鲁棒性(robustness)

鲁棒来自于“Robust”，也可以理解为健壮性，指不因某种改动而导致隐藏信息丢失的能力。比如传输时的噪音或滤波操作、重采样、有损编码压缩等

不可检测性(undetectability)

指隐蔽载体与原始载体具有一致的特性，如具有一致的统计噪声分布等，以便使非法拦截者无法判断是否有隐蔽信息

透明性(invisibility)

利用人类视觉系统或人类听觉系统属性，经过一系列隐藏处理，使目标数据没有明显的降质现象，而隐藏的数据却无法人为地看见或听见

安全性(security)

指隐藏算法有较强的抗攻击能力，即它必须能够承受一定程度的人为攻击，而使隐藏信息不会被破坏

0x03 隐写术应用

1. 可以隐藏互联网上涉及敏感信息的交谈内容
2. 在发送方发送的内容中添加水印等识别所有者的信息，属于不可抵赖性
3. 可以在数据传输中保证数据的完整性

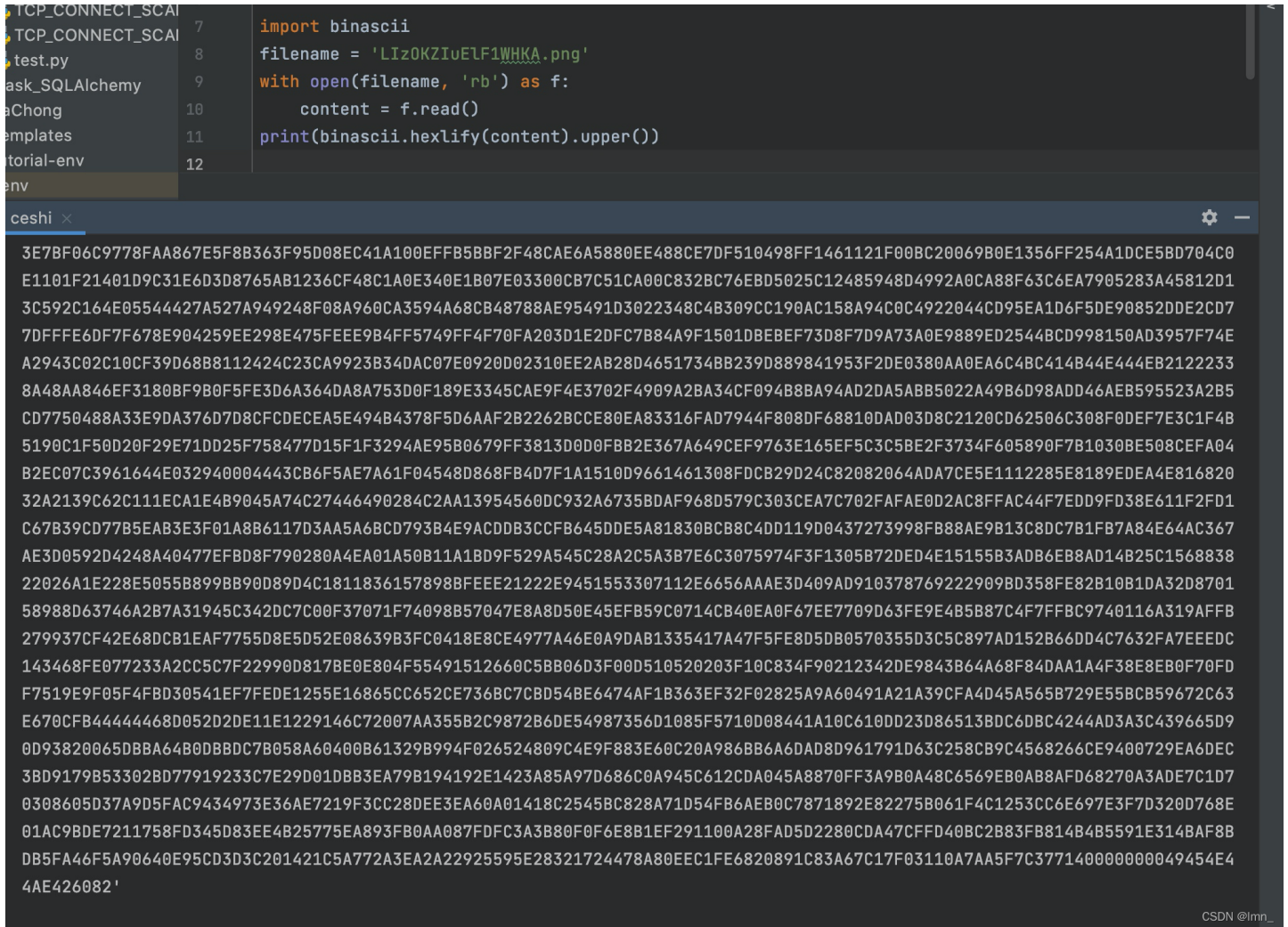
0x04 题目解析思路

使用binwalk的话，使用kali，“binwalk 图片名”

binwalk中的参数

-e, --extract 自动提取已知的文件类型
-D, --dd=<type[:ext[:cmd]]> 提取签名(正则表达式)，给文件一个扩展名，然后执行
-M, --matryoshka 递归扫描提取的文件
-d, --depth= 限制matyoshka递归深度(默认值:8级深度)
-C, --directory= 解压文件/文件夹到自定义目录(默认:当前工作目录)
-j, --size= 限制每个提取文件的大小
-n, --count= 限制提取文件的数量
-r, --rm 解压后删除刻录的文件
-z, --carve 从文件中提取数据，但不要执行提取实用程序
-V, --subdirs 提取到由偏移量命名的子目录中

图片类隐型经常会将一部分数据隐藏在一张图片中，判断思路可以，Jpg格式的文件在16进制中的表示是以 ff d9 两个字节结尾，可以通过判断末尾来判断是否为真正的图片文件，有时也可以通过文件的大小稍加判断



```
import binascii
filename = 'LIzOKZiUElF1WHKA.png'
with open(filename, 'rb') as f:
    content = f.read()
print(binascii.hexlify(content).upper())
```

也可以使用最低有效为
最低有效位 (Least Significant Bit)
我们常说的三原色是红、黄、蓝

LSB隐写就是修改了像数中的最低的1bit, png图片是一种无损压缩的位图片形格式, png图片中的图像像数一般是由RGB三原色（红绿蓝）组成, 每一种颜色占用8位, 取值范围为0x00~0xFF, 即有256种颜色, 一共包含了256的3次方的颜色, 即16777216种颜色。而人类的眼睛可以区分约1000万种不同的颜色, 这就意味着人类的眼睛无法区分余下的颜色大约有6777216种

Color (Green)	Base 10	Binary	Change
	238	11101110	+3
	235	11101011	(base)
	232	11101000	-3

CSDN @Imn_

十进制的235表示的是绿色，我们修改了在二进制中的最低位，但是颜色看起来依旧没有变化，我们就可以修改最低位中的信息，实现信息的隐写

? | score: 2 | [3.46](#) [5.49](#) [5.34](#) | [Solved By 1954 People](#) | 72679 views | since Dec 26, 2010 - 23:10:22

Training: LSB (Stegano, Image, Training)

Training: LSB - The least significant bit

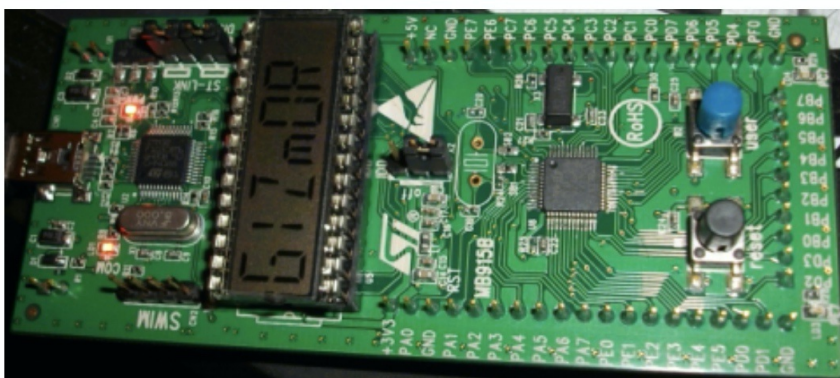
This is a very simple, yet spectacular trick.
Can you see the hidden solution?

Note: The solution is bound to your session ID and consists of twelve letters.

Hidden Hint: <http://wechall.blogspot.com/2007/11/steganabara-explained.html>

Thanks go out to buttmonkey for creating the image :)

The carrier image



Your solution for Training: LSB

Answer

Submit

© 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021 and 2022 by [Gizmore](#)

CSDN @Imn_

点开链接提示了工具



We Chall

I Chall. You Chall. We Chall!!!

Wednesday, 14 November 2007

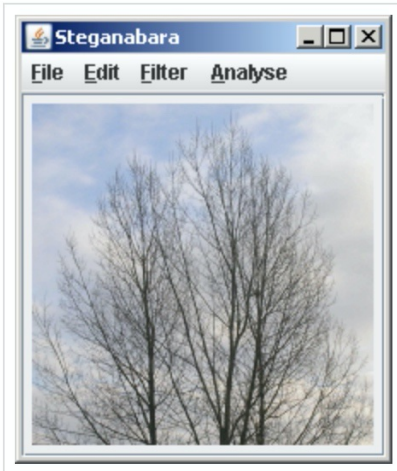
Steganabara explained

Hey guys,

Finally I defeated my laziness and created a project page for Steganabara. You can access it from [here](#).

Also included an explanation of the key features: Bit Mask Filter, Color Map Filter, Color Explorer, Histogram and Color Table.

And now a demonstration of the Bit Mask Filter, applying on [StenographyOriginal.png](#), from the [wikipedia page about steganography](#)




Contributors

-  [Gizmore](#)
-  [Sphinx](#)
-  [loki](#)
-  [ozehka](#)
-  [quangntenemy](#)

Blog Archive

- ▶ 2014 (1)
- ▶ 2008 (8)
- ▼ 2007 (15)
 - ▶ December (3)
 - ▼ November (6)
 - [Smash The Stack!](#)
 - [platonic defection](#)
 - [Steganabara explained](#)
 - [TheBlacksheep](#)
 - [rankk.org](#)
 - [Mod-x](#)
 - ▶ October (6)

Label Cloud

3564020356 [challenge site](#)
[hackquest](#) [rankk](#) [electrica](#)
[theblacksheep](#) [slyfx](#) [stegano](#)
[challenge](#) [ma's](#) [reversing](#) [omega](#)
[project euler](#) [steganabara](#) [@Imn](#)

不过有一个更好用的被大家公认的工具



找到答案



这是另一道其他思路的题

?

score: 3 | 5.15 5.45 6.48 | Solved By 311 People | 59285 views | since Nov 10, 2008 - 20:19:00

Simply Red (Stegano, Image)

Find the sentence hidden in me or I'll have to destroy you.

"No sacrifice is too great in the service of freedom."

Your solution for Simply Red

Answer

Submit

© 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021 and 2022 by anto

CSDN @imn_

题目意思：
找出隐藏在我身上的那句话，否则我将不得不摧毁你。
“在为自由服务的过程中，任何牺牲都不算大。”
既然从根本开始做，那就假设我不知道这是谁，既然是饮用那就是他说出的话，搜索一下

Animated character biography

Optimus Prime (formerly known as **Orion Pax**) is constantly, if not always, depicted as having strong **moral character**, excellent **leadership**, and sound **decision-making** skills, and possesses brilliant **military tactics**, powerful **martial arts**, and advanced extraterrestrial weaponry. Optimus Prime has a strong sense of honor and justice, being dedicated to building peaceful and mutually beneficial co-existence with humans, the protection of life and liberty of all sentient species.^[3] As the current **Matrix of Leadership** bearer, Optimus Prime is the **de facto** leader of the **Autobots**, a faction of a **transforming** species of **synthetic intelligence** from the planet Cybertron. The Autobots are constantly waging **civil war** against a rival faction of transforming robots called **Decepticons**. According to **Bob Budiansky**, co-writer of the Transformers series, **Dennis O'Neil** was responsible for his name.

Optimus Prime is usually depicted as being a member of an ancient Transformers race called the Dynasty of Primes, often receiving the title "The Last Prime" in many stories, in which he is depicted as being the last of the Primes. In the *Transformers: Covenant of Primus*, it was established that Optimus Prime was the last born of the original Thirteen Transformers created by the creator Primus. It was his unique spark and his inspiring reassurance that "All are One" that allowed the Primes to rally and succeed in their battle against the Chaos Bringer **Unicron**. When tragedy at last ended the era of the Primes and brought forth the new race of lesser descendant Transformers he alone chose to be reborn in the Well of All Sparks as one of them, that he might know them and their needs more completely. All memory of his past life gone, he took the name "Orion Pax" and sought his way like any other robot on the new world becoming Optimus Prime once more when receiving the **Matrix of Leadership** when Cybertron faced a new enemy in his former friend, **Megatron** and his army of followers, the **Decepticons**. This brings a Great War to their planet of Cybertron. Optimus's origins and personality can vary depending on which "universe" he's seen in. This origin is the most consistent between the various incarnations. Further differences are listed in the respective sections below.

Generation 1

In-universe information

Affiliation	Autobot/Maximal
Japanese name	Inochi, Convoy
Sub-group	Action Masters, Autorollers, Combat Heroes, Deluxe Vehicles, Go-Bots, Masterpiece, Primes, Powermasters, Voyagers
Function	Autobot Leader, Supreme Commander, Chief Commander
Rank	10
Partner	Roller, Hi-Q
Motto	"Freedom is the right of all sentient beings." (Generation 1) "No sacrifice is too great in the service of freedom." (Classics)
Alternate modes	Freightliner Cab-over-engine Class 8 truck, Cybertron truck, COBRA Sentry & Missile System tank, 1920s style truck, Peterbilt Truck, Mid-90s Peterbilt 4964EX Tanker Truck, Lamborghini Diablo , Dump truck, Dodge Ram SRT-10 , Nissan GT-R , Bat

我们知道原来它叫 Optimus Prime

第一眼竟然认成了 Option Prime

分析一定与素数有关，并且是红色为底色，之前有一道题是提图片的HSB，既然是红色，那就分析像素的R为素数尝试一下

在搜索代码的时候发现已经有大佬写出了这个代码

地址：<https://www.tuziang.com/index.php/combat/1985.html>

```
from PIL import Image
from math import sqrt

def is_prime(n):
    if n==1:
        return False
    for i in range(2, int(sqrt(n) + 1)):
        if n % i == 0:
            return False
    return True

img = Image.open("op.png")
width = img.size[0]
height = img.size[1]

for x in range(width):
    for y in range(height):
        r,g,b = img.getpixel((x,y))
        if is_prime(r):
            continue
        else:
            img.putpixel((x,y),(255,255,255))

img.show()
```



```

In [1]: from PIL import Image
        from math import sqrt

        def is_prime(n):
            if n==1:
                return False
            for i in range(2, int(sqrt(n) + 1)):
                if n % i == 0:
                    return False
            return True

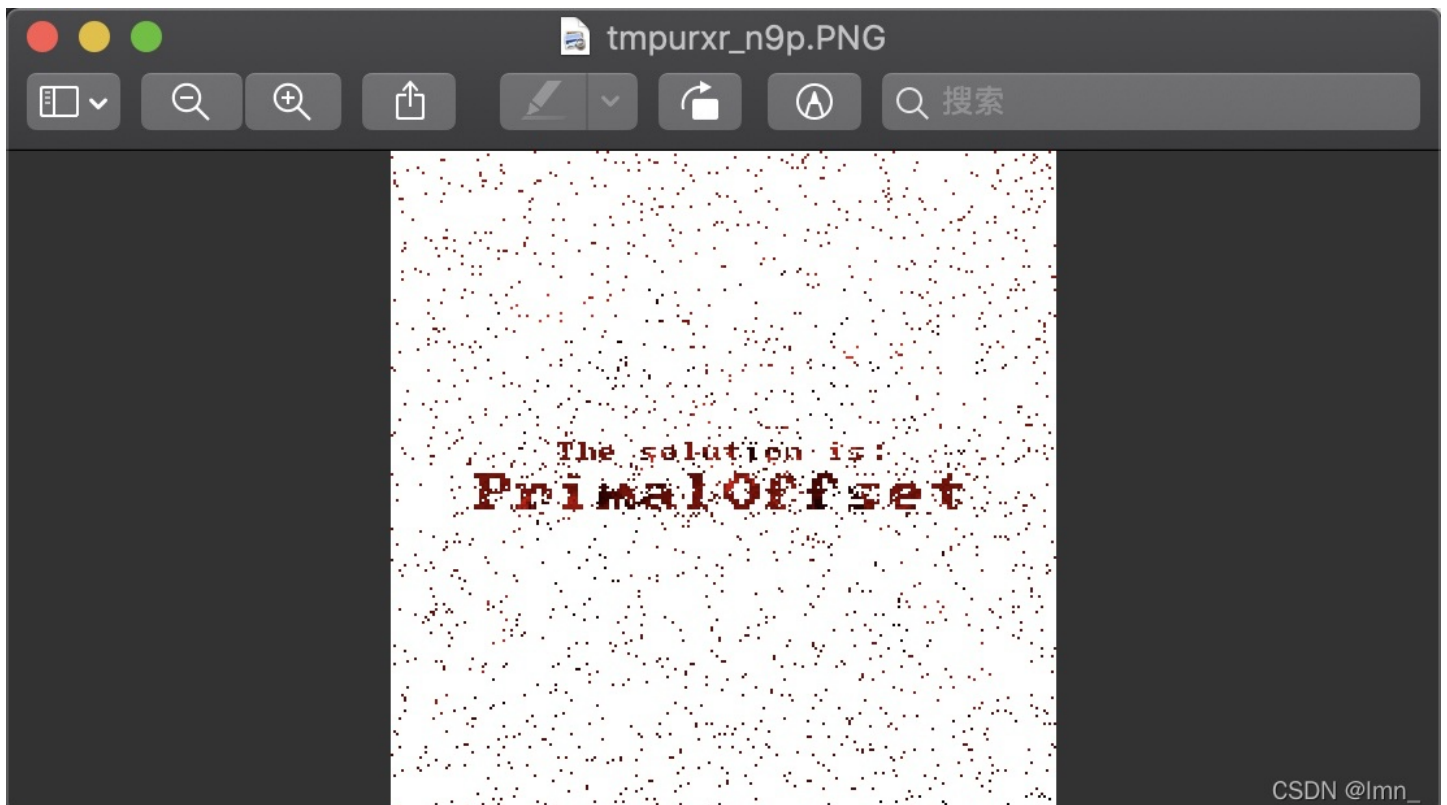
        img = Image.open("op.png")
        width = img.size[0]
        height = img.size[1]

        for x in range(width):
            for y in range(height):
                r,g,b = img.getpixel((x,y))
                if is_prime(r):
                    continue
                else:
                    img.putpixel((x,y),(255,255,255))

        img.show()

```

CSDN @Imn_



CSDN @Imn_

参考文章

<https://zhuanlan.zhihu.com/p/301815967>

[https://blog.csdn.net/superdont/article/details/1396172?](https://blog.csdn.net/superdont/article/details/1396172?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522164605916216780271951936%2522%252C%2522scm%2522%253A%25220140713.130102334...%2522%257D&request_id=164605916216780271951936&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2_alltop_ulrmf_defaultdefault-3-1396172.pc_search_insert_ulrmf&utm_term=%E4%BF%A1%E6%81%AF%E9%9A%90%E8%97%8F&spm=1018.2226.3001.4187)

[ops_request_misc=%257B%2522request%255Fid%2522%253A%2522164605916216780271951936%2522%252C%2522scm%2522%253A%25220140713.130102334...](https://blog.csdn.net/superdont/article/details/1396172?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522164605916216780271951936%2522%252C%2522scm%2522%253A%25220140713.130102334...%2522%257D&request_id=164605916216780271951936&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2_alltop_ulrmf_defaultdefault-3-1396172.pc_search_insert_ulrmf&utm_term=%E4%BF%A1%E6%81%AF%E9%9A%90%E8%97%8F&spm=1018.2226.3001.4187)

[%2522%257D&request_id=164605916216780271951936&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2_alltop_ulrmf_defaultdefault-3-](https://blog.csdn.net/superdont/article/details/1396172?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522164605916216780271951936%2522%252C%2522scm%2522%253A%25220140713.130102334...%2522%257D&request_id=164605916216780271951936&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2_alltop_ulrmf_defaultdefault-3-1396172.pc_search_insert_ulrmf&utm_term=%E4%BF%A1%E6%81%AF%E9%9A%90%E8%97%8F&spm=1018.2226.3001.4187)

[1396172.pc_search_insert_ulrmf&utm_term=%E4%BF%A1%E6%81%AF%E9%9A%90%E8%97%8F&spm=1018.2226.3001.4187](https://blog.csdn.net/superdont/article/details/1396172?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522164605916216780271951936%2522%252C%2522scm%2522%253A%25220140713.130102334...%2522%257D&request_id=164605916216780271951936&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2_alltop_ulrmf_defaultdefault-3-1396172.pc_search_insert_ulrmf&utm_term=%E4%BF%A1%E6%81%AF%E9%9A%90%E8%97%8F&spm=1018.2226.3001.4187)

https://blog.csdn.net/weixin_34075268/article/details/88744599