

# CTF训练STEGA题目-数独

原创

灵梦归希 于 2018-05-12 23:42:48 发布 4182 收藏 1

分类专栏: [ctf writeup](#) 文章标签: [ctf writeup](#) [数独](#) [stega](#) [数独隐写](#)

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/qq\\_30167299/article/details/80295867](https://blog.csdn.net/qq_30167299/article/details/80295867)

版权



ctf同时被 2 个专栏收录

3 篇文章 0 订阅

订阅专栏



writeup

3 篇文章 0 订阅

订阅专栏

看到这名字, 还以为要解数独。。。

下载文件, 解压出来好多数独...



1.png



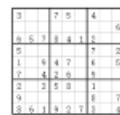
2.png



3.png



4.png



5.png



6.png



7.png



8.png



9.png



10.png



11.png



12.png



13.png



14.png



15.png



16.png



17.png



18.png



19.png



20.png



21.png



22.png



23.png



24.png



25.png

想起那句话, 没有无缘无故的信息。

那么可以推测, flag应该藏在全部的数独中, 每个数独仅仅包含一部分信息。

但? 这是普通的数独, 能有什么信息? 最多就是数独排列不同。

忽然发现, 会不会是那些数字组成的形状, 细看1.png,5.png,21.png.

这不就是二维码的定位形状吗? 这些数独图片可以组成二维码。

需要将图片1.png,5.png,21.png重命名成:5.png,21.png,1.png(重命名失败? 肯定是你的原因)

下面, 手残喽。。。

使用文本形式记录数独信息, 0代表没有数字, 1代表有数字

我这里给出我的结果:保存文件为2.txt

```
45 45
11111101 010101000 101000001 111110000 101111111
100000101 100111101 010011101 100011001 001000001
101110101 110011111 010011111 101000101 001011101
101110101 101100010 001010000 011110001 101011101
101110100 011100100 001111101 111111011 101011101
100000101 100100000 011000100 001110100 001000001
111111101 010101010 101010101 010101011 101111111
000000000 011001101 001000110 100110011 100000000
110011100 100100001 111111100 100101000 000101111

101001001 011111111 101110101 011110101 101001100
100000111 100100100 000110001 101001101 010001010
001100010 011010001 010011000 100000010 110010000
010110101 010001111 110100011 101001110 101101111
100011000 100011100 111011101 101100101 101110001
001100110 100000000 010010000 111100101 101011010
101000001 011010111 110011011 111101001 110100011
110111110 111011001 101100010 100001110 000100000
110101000 010101000 011101101 101110101 101001100

010011111 110001011 111010001 000011011 101101100
011001011 001010101 100011110 101001100 001010010
010111111 111101011 111111101 101101111 111111100
011110001 100000100 001000101 000100100 100011110
111110101 110011100 111010110 100110100 101010010
110010001 011101011 101000111 100000011 100010000
101011111 011100111 101111111 100001010 111110010
110100011 000111000 100111101 101111101 000100010
111101111 110001001 000011010 110001111 110111110

011001010 101000110 010100010 001000101 101010001
011101110 101101101 100100001 101101000 111101001
110110001 001101100 010101101 111110100 101100110
000011100 111000000 000100001 010101111 100010010
111010010 011110011 101110010 100001011 111010010
101001100 010111111 110100000 100001010 101010100
000010011 001001101 110101001 111100101 111101101
000010111 101110001 101011000 001000101 110100110
011110011 010100010 100000011 011000001 110010000

100110100 100001101 111111101 100101110 111110011
000000001 111110101 101000101 011100100 100011010
111111100 011111011 011010101 101110011 101011110
100000101 110101101 101000111 110010001 100010001
101110101 011100001 111111101 101001000 111111011
101110100 110111101 101000001 001101100 011101101
101110100 000011101 100001101 010110010 010010001
100000101 011001011 111011001 011000011 010110000
111111101 010101001 111011110 101101110 000101101
```

我是使用c++完成图像显示(使用opencv)

如果你也是c++，可以看下

```
#include <iostream>
#include <core/core.hpp>
#include <highgui/highgui.hpp>
#pragma comment(lib,"opencv_world331.lib")
#pragma comment(lib,"opencv_world331d.lib")
using namespace std;
using namespace cv;
int main(){
    int size = 4;
    FILE *fp = fopen("2.txt", "r");
    int w, h;
    fscanf(fp,"%d%d", &h, &w);
    Mat image(h*size,w*size,16);
    if (!image.data) {
        std::cout << "无法打开图像文件" << std::endl;
        system("pause");
        return -1;
    }
    cout << "height:" << image.rows << endl;
    cout << "weight:" << image.cols << endl;
    cout << "type:" << image.type() << endl;
    Vec3b white(255,255,255);
    Vec3b black(0, 0, 0);

    for (int i = 0; i < image.rows; i+=size) {
        for (int j = 0; j < image.cols; j+=size) {
            Vec3b flag = white;
            int f;
            fscanf(fp, "%1d", &f);
            if (f == 0) flag = white;
            else flag = black;
            for (int kx = i; kx < image.rows&&kx<i+size; kx++) {
                for (int ky = j; ky < image.cols&&ky<j+size; ky++) {
                    image.at<Vec3b>(kx, ky) = flag;
                }
            }
        }
    }
    fclose(fp);
    system("pause");
    namedWindow("hello world", CV_WINDOW_AUTOSIZE);
    imshow("hello world", image);
    waitKey();
}
```

得到图像(二维码)，使用工具扫一下。



得到结果:

```
Vm0xd1NtUXlWa1pPVldoVFIUSINjRIJVGtOamJGWNlWMjFHMUxV1ZqTldNakZIWVcxS1lxTnNhRmhoTVZwe\
```

这很好办，多次base64解码就行。(算下，要解7次)

得到flag:flag{y0ud1any1s1}

AC。。。