2019西湖论剑——一小小部分writeup

原创

地址ch3nye.top 于 2019-04-07 22:38:17 发布 1862 收藏 2
文章标签： ctf writeup

本文链接：https://blog.csdn.net/qq_41420747/article/details/89076214

Misc: 奇怪的TTL

题目描述：

我们截获了一些IP数据报，发现报文头中的TTL值特别可疑，怀疑是通信方嵌入了数据到TTL，我们将这些TTL值提取了出来，你能看出什么端倪吗？

通过不断尝试找到一种正确的思路：

> **在TTL字段中隐藏**
>
> ---
>
> IP报文在路由间穿梭的时候每经过一个路由，TTL就会减1，当TTL为0的时候，该报文就会被丢弃。TTL所占的位数是8位，也就是0-255的范围，但是在大多数情况下通常只需要经过很小的跳数就能完成报文的转发，远远比上限255小得多，所以我们可以用TTL值的前两位来进行传输隐藏数据。
>
> 如：须传送H字符，只需把H字符换成二进制，每两位为一组，每次填充到TTL字段的开头两位并把剩下的6位设置为1（xx111111），这样发4个IP报文即可传送1个字节。
>
> from http://drops.the404.me/1981.html?tdsourcetag=s_pctim_aiomsg

写个python脚本跑一下

```
f = open("ttl.txt")
flag = ""
count = 0
tmp = ""

for i in f.readlines():

    count += 1
    if count == 5:
        print(tmp)
        flag += chr(int(tmp, 2))
        tmp = ""
        count = 1

    s = i.split('=')
    str = s[1]
    num = bin(int(str))
    num = num.split('b')[1].zfill(8)
    tmp += num[0:2]

print(flag)
print(flag.__len__()) # 73843
```

得到解密后的信息是：ffd8ffe10……

很明显是jpeg图片16进制头

找了个java程序把16进制字符串转换为图片

tips: 将python脚本跑出来的字符串保存到txt中，需要注意字符串尾是FFD,需要加一个9 (FFD9是jpg文件结尾标志)

```
import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.InputStream;
import java.io.InputStreamReader;

/**
 * 十六进制转成图片
 */
public class Hex2Image
{
    public static void main(String[] args) throws Exception
    {
        Hex2Image to = new Hex2Image();
        InputStream is = new FileInputStream("D:/111.txt");
        InputStreamReader isr = new InputStreamReader(is);
        BufferedReader br = new BufferedReader(isr);
        String str = null;
        StringBuilder sb = new StringBuilder();
        while ((str = br.readLine()) != null)
        {
            System.out.println(str);
            sb.append(str);
        }
```

```
        }
        to.saveToImgFile(sb.toString().toUpperCase(), "D:/bbb.jpg");
    }

    public void saveToImgFile(String src, String output)
    {
        if (src == null || src.length() == 0)
        {
            return;
        }
        try
        {
            FileOutputStream out = new FileOutputStream(new File(output));
            byte[] bytes = src.getBytes();
            for (int i = 0; i < bytes.length; i += 2)
            {
                out.write(charToInt(bytes[i]) * 16 + charToInt(bytes[i + 1]));
            }
            out.close();
        }
        catch (Exception e)
        {
            e.printStackTrace();
        }
    }

    private int charToInt(byte ch)
    {
        int val = 0;
        if (ch >= 0x30 && ch <= 0x39)
        {
            val = ch - 0x30;
        }
        else if (ch >= 0x41 && ch <= 0x46)
        {
            val = ch - 0x41 + 10;
        }
        return val;
    }
}
```

得到

这只是二维码的一部分，winhex修改宽高，并不行，但是看到里面存在好几个图片exif开头,所以使用foremost分离，

得到了6张二维码图片，拼接到一起扫描

得到

key:AutomaticKey cipher:fftu{2028mb39927wn1f96o6e12z03j58002p}
很明显自动密钥密码在线工具解密

得到

```
flagabdfdeabee
```

再把对应位置的数字加上就行了

flag(2028[illegible]df1d96e[illegible]8002e}

Misc：哈夫曼

这个题目很简单没什么弯，就是普通的huffman编码，学过数据结构的都会，就不多说了。

画出哈夫曼树后 根据flag{}这几个字符再调整一下，就可以拿哈夫曼树解码了。

tips：这里面d和5这两个字符比较坑，如果不对就相互替换一下。