第一届东软杯网络CTF竞赛-DNUICTF部分wp

原创

whathay 于 2021-12-06 19:58:57 发布 2938 收藏 1
分类专栏： ctf比赛wp 文章标签： 网络 安全 web安全

本文链接：https://blog.csdn.net/weixin_52829570/article/details/121754672

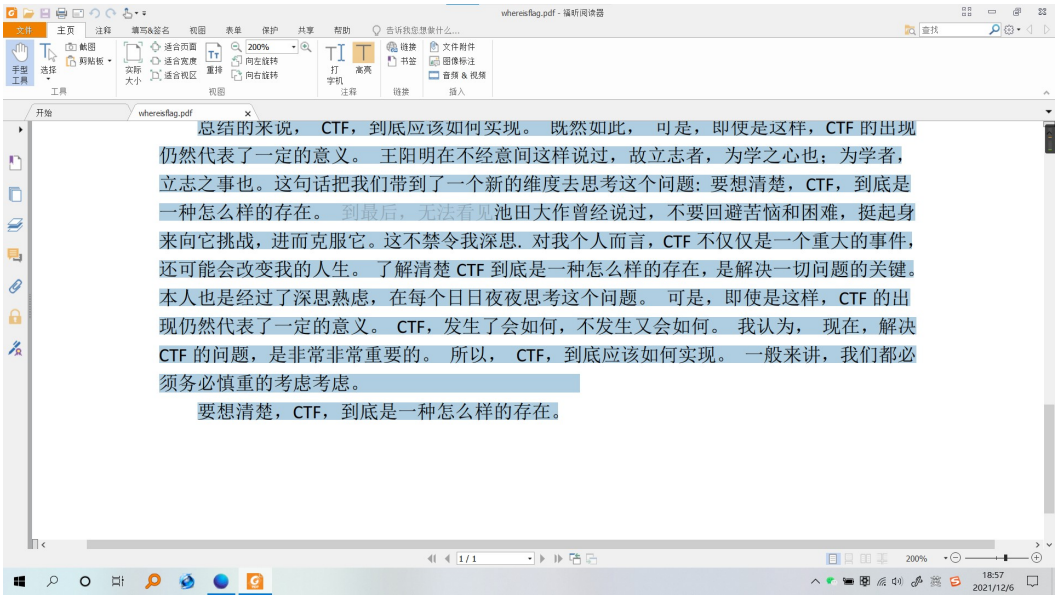
MISC

[签到]签到

直接提交flag即可

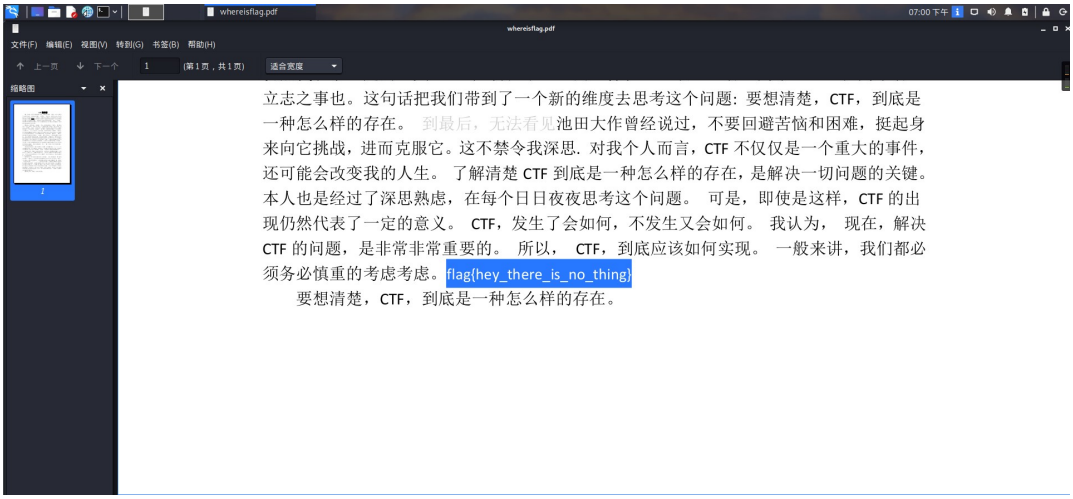
flag:flag{Dnui_ctf_2021_s1gn_in}

[萌新]在哪呢



ctrl+a全选发现有个空白的地方被选中了

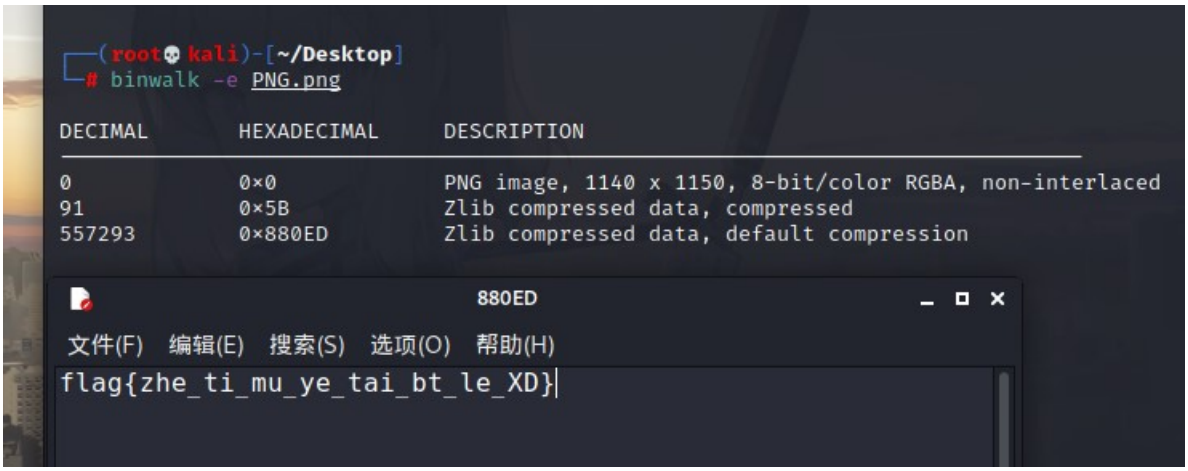
在kali打开选中，拿到flag



flag:flag{hey_there_is_no_thing}

只是个PNG，别想太多了.png

binwalk命令查看png图片:binwalk -e PNG.png，发现flag



flag:flag{zhe_ti_mu_ye_tai_bt_le_XD}

压缩包压缩包压缩包压缩包

zip压缩包套娃,压缩包的密码是下一个压缩包的文件名，python脚本解套：

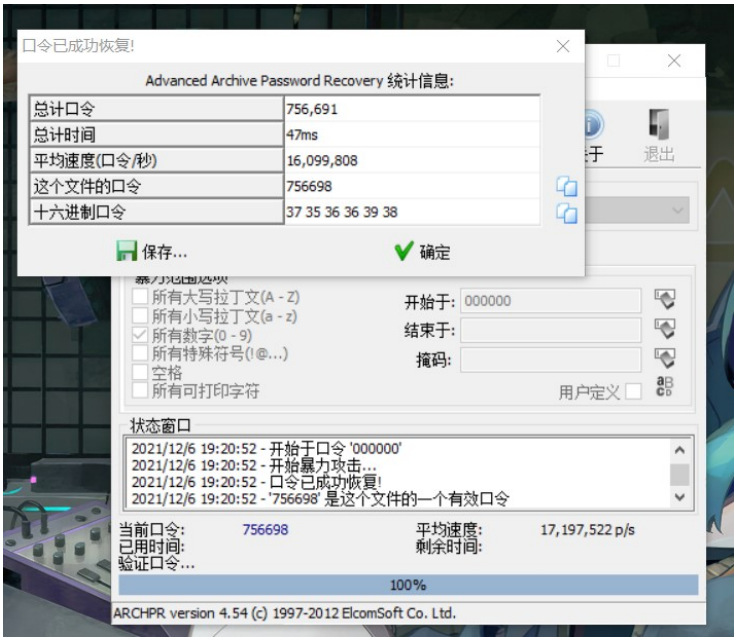
```
import zipfile

dir = "D:\\Desktop\\test1111\\" # 文件路径

n = 0

def flag():
    i = "23898.zip" # 文件名
    for x in range(10):
        # i[i:] 从i开始取后面的字符串，后面不填默认取全部
        # i[:i] 从i开始取前面的字符串，前面不填默认取全部
        s = i[:i.find('.')]
        print(x,i)
        zpf = zipfile.ZipFile(dir + s + ".zip")
        # print(zpf)
        zip_list = zpf.namelist()
        # print(zip_list)
        for f in zip_list:
            # split('.')[0] 取.前面的字符串
            # split('.')[1] 取.后面的字符串
            pwd = f.split('.')[0]
            print(pwd,'成功')
            zpf.extract(f,dir,bytes(pwd.encode("utf-8")))
            i = str(f)
flag()
```

解压到最后得到23333.zip,打开发现注释提示密码6位数,ARCHPR爆破得756698



解压后打开文件搜索flag拿到flag

flag:flag{Unz1p_i5_So_C00l##}

easysteg

一张缺了一个定位符的二维码png图片(补码后发现不补也能扫出来)，用微信扫出内容 某种常见的隐写

用010editor 16进制查看，发现尾部有zip压缩包，分离处理，解压出一张png图片

比赛的时候用stegsolve和zsteg看了各个通道，用了各种方法都没找出来

比赛完看了别的师傅的wp用stegpy一把嗦出来了。。。

(痛失千分题，哭死，这道做出来应该能进前80)

flag:flag{Do_U_Kn0w_Ste9py??}

CRYPTO

[签到]键盘侠

键盘密码，按照字母顺序在键位一顿比划,拿到flag

flag:flag{CLCKOUTHK}

[萌新]素数

队友写的脚本

```
import random
def rabin_miller(num):
    s = num - 1
    t = 0
    while s % 2 == 0:
        s = s // 2
        t += 1

    for trials in range(5):
        a = random.randrange(2, num - 1)
        v = pow(a, s, num)
        if v != 1:
            i = 0
            while v != (num - 1):
                if i == t - 1:
                    return False
                else:
                    i = i + 1
                    v = (v ** 2) % num
    return True


def is_prime(num):
    # 排除0,1和负数
    if num < 2:
        return False

    # 创建小素数的列表,可以大幅加快速度
    # 如果是小素数,那么直接返回true
    small_primes = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83,
    if num in small_primes:
        return True

    # 如果大数是这些小素数的倍数,那么就是合数,返回false
    for prime in small_primes:
        if num % prime == 0:
            return False

    # 如果这样没有分辨出来,就一定是大整数,那么就调用rabin算法
    return rabin_miller(num)


# 得到大整数,默认位数为1024
def get_prime(key_size=1024):
    while True:
        num = random.randrange(2**(key_size-1), 2**key_size)
        if is_prime(num):
            print(num)
            return num
a=1027
while 1:
    get_prime(a+1)
    if a>1037:
        break
```

REVERSE

[签到]signin

拉进IDA，alt+t搜索flag拿到flag

```
                ;org 404000h
; char Format[]
Format          db 'flag in outher function',0
                                        ; DATA XREF: main+14↑o
; char aFlagReverse1sV[]
aFlagReverse1sV db 'flag{REVERSE_1s_Very_3asy!}',0
                                        ; DATA XREF: getflag(void)+8↑o
                align 20h
```

flag:flag{REVERSE_1s_Very_3asy!}

WEB

[签到] flag

页面不断随机输出flag各位置对应字符

```
一共20位，第7位是u
一共20位，第17位是W
一共20位，第6位是t
一共20位，第0位是Z
一共20位，第4位是Z
一共20位，第17位是W
一共20位，第6位是t
一共20位，第12位是b
一共20位，第13位是G
一共20位，第12位是b
一共20位，第8位是c
一共20位，第11位是f
一共20位，第1位是m
一共20位，第7位是u
一共20位，第19位是9
一共20位，第0位是Z
一共20位，第11位是f
一共20位，第11位是f
一共20位，第8位是c
一共20位，第2位是又(小写)
```

搜集拼接后发现是个base64，解码得到flag

flag:flag{nss_login}

最终排名

142	szsec_飞起来	529