2020湖湘杯部分writeup

原创

易咯 于 2020-11-02 23:50:19 发布 926 收藏
文章标签： 信息安全

周末打了湖湘杯，把做题过程记录一下，大家交流学习。

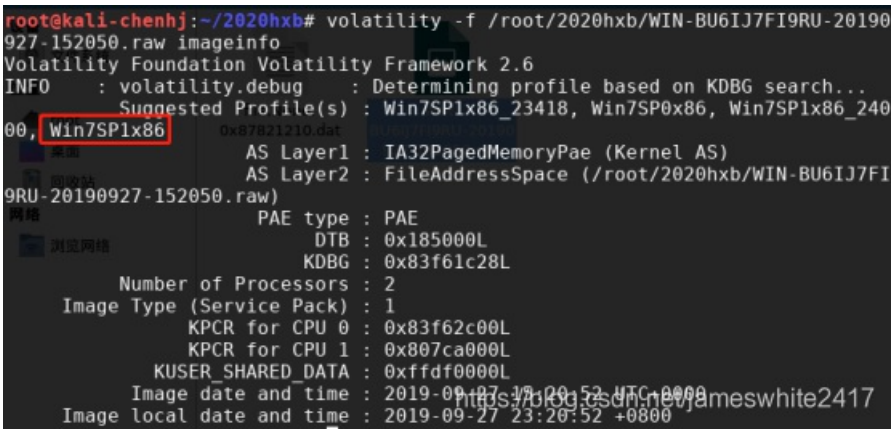
下面的链接里有题目，可以下来看看。

https://download.csdn.net/download/jameswhite2417/13081994

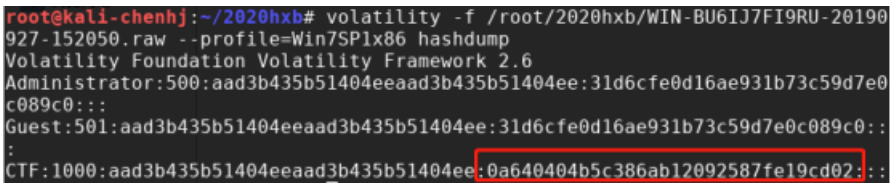
1. misc passwd

下载文件发现是raw，是道内存取证的题目

拷到kail中，先识别配置文件

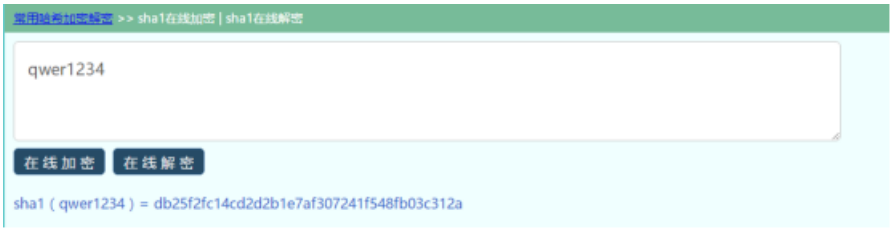


获取密码hash



进行解密

再根据题目用sha1加密得到flag



2.CPYPTO 古典美++

Virginia（维吉尼亚）无密钥解密

1. 破解秘钥长度。

Python代码：

```
#coding=utf-8#-*- coding:utf-8 -*-def c_alpha(cipher):   # 去掉非字母后的密文
    cipher_alpha = ''
    for i in range(len(cipher)):
        if (cipher[i].isalpha()):
            cipher_alpha += cipher[i]
    return cipher_alpha
# 计算cipher的重合指数def count_CI(cipher):
    N = [0.0 for i in range(26)]
    cipher = c_alpha(cipher)
    L = len(cipher)
    if cipher == '':
        return 0
    else:
        for i in range(L):     #计算所有字母的频数，存在数组N当中
            if (cipher[i].islower()):
                 N[ord(cipher[i]) - ord('a')] += 1
            else:
                 N[ord(cipher[i]) - ord('A')] += 1
    CI_1 = 0
    for i in range(26):
        CI_1 += ((N[i] / L) * ((N[i]-1) / (L-1)))
    return CI_1
# 计算秘钥长度为 key_len 的重合指数def count_key_len_CI(cipher,key_len):
    un_cip = ['' for i in range(key_len)]    # un_cip 是分组
    aver_CI = 0.0
    count = 0
    for i in range(len(cipher_alpha)):
        z = i % key_len
        un_cip[z] += cipher_alpha[i]
    for i in range(key_len):
        un_cip[i]= count_CI(un_cip[i])
        aver_CI += un_cip[i]
    aver_CI = aver_CI/len(un_cip)
    return aver_CI
## 找出最可能的前十个秘钥长度def pre_10(cipher):
    M = [(1,count_CI(cipher))]+[(0,0.0) for i in range(49)]
    for i in range(2,50):
        M[i] = (i,abs(0.065 - count_key_len_CI(cipher,i)))
    M = sorted(M,key = lambda x:x[1])   #按照数组第二个元素排序
    for i in range(1,10):
        print (M[i])

F = [0.0651738, 0.0124248, 0.0217339,0.0349835, 0.1041442, 0.0197881,0.0158610, 0.0492888, 0.0558094,0.0009
cipher = 'SZWLVSRVVZICMUOJYIIZBSVSSITFSWHPCWCFPVPFXJMWRVJICVRGTCFLHPRJKJKSRVWYFUSEWHFXLHFOSFLYPFXXYFPOEGXFX

cipher_alpha = c_alpha(cipher)print u"秘钥长度为:"
pre_10(cipher)
```

```
D:\>Python pojie2.py
秘钥长度为:
(28, 0.0001719213326356328)
(35, 0.0003919413919414027 6)
(49, 0.0013821255657990306)
(7, 0.001631618600747689)
(21, 0.001720021341870076)
(14, 0.0017714273724505347)
(42, 0.0020857111988300675)
(46, 0.019786422578184602)
(41, 0.020489585098206767)
```

得出的结果排名靠前的都是7的倍数，我们可以猜测秘钥长度为7

2.将密文分成N组，逐个破解秘钥。

Python代码：

```
# 猜测单个秘钥得到的重合指数def count_CI2(cipher,n):     # n 代表我们猜测的秘钥，也即偏移量
    N = [0.0 for i in range(26)]
    cipher = c_alpha(cipher)
    L = len(cipher)
    for i in range(L):     #计算所有字母的频数，存在数组N当中
        if (cipher[i].islower()):
            N[(ord(cipher[i]) - ord('a') - n)%26] += 1
        else:
            N[(ord(cipher[i]) - ord('A') - n)%26] += 1
    CI_2 = 0
    for i in range(26):
        CI_2 += ((N[i] / L) * F[i])
    return CI_2
def one_key(cipher,key_len):
    un_cip = ['' for i in range(key_len)]
    cipher_alpha = c_alpha(cipher)
    for i in range(len(cipher_alpha)):     # 完成分组工作
        z = i % key_len
        un_cip[z] += cipher_alpha[i]
    for i in range(key_len):
        print (i)
        pre_5_key(un_cip[i])     ####这里应该将7个分组的秘钥猜测全部打印出来
## 找出前5个最可能的单个秘钥def pre_5_key(cipher):
    M = [(0,0.0) for i in range(26)]
    for i in range(26):
        M[i] = (chr(ord('a')+i),abs(0.065 - count_CI2(cipher,i)))
    M = sorted(M,key = lambda x:x[1])   #按照数组第二个元素排序

    for i in range(10):
        print (M[i])

key_len = 7   #输入猜测的秘钥长度
one_key(cipher,key_len)
```

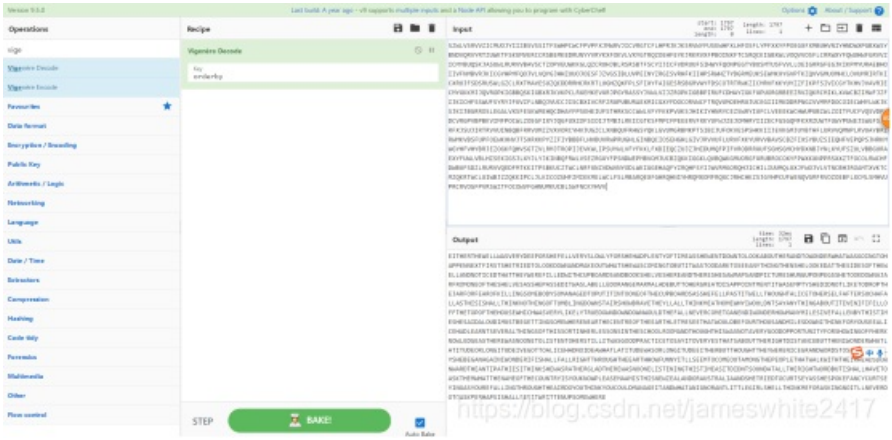
结果

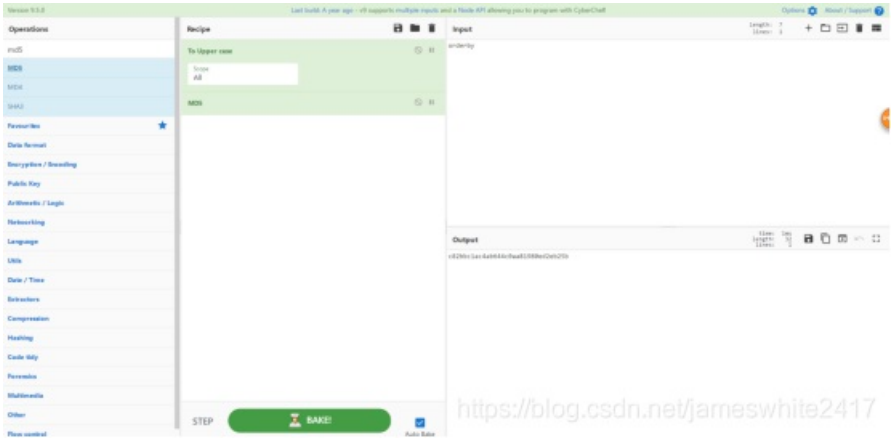
```
0
('o', 0.009511250972762647)
('k', 0.027313619066147862)
('z', 0.02792477821011674)
('b', 0.02967103618677043)
('v', 0.031046701945525286)
('s', 0.03105647859922179)
('d', 0.03138896887159534)
('n', 0.03189639416342412)
('h', 0.032357768482490265)
('c', 0.0334486805474708)
1
('r', 0.011346537354085605)
('n', 0.025409001556420223)
('y', 0.028436771206225675)
('g', 0.029109717120622575)
('c', 0.02978173346303503)
('v', 0.031182320622568087)
('d', 0.03203501634241245)
('e', 0.03293099455252918)
('s', 0.033072902723735406)
('k', 0.03313551206225682)
2
('d', 0.015040221789883255)
('o', 0.02811225564202334)
('n', 0.030048985992217905)
('h', 0.03036853696498054)
('c', 0.03119001517509728 5)
('s', 0.03135425330739299)
('q', 0.031464307782101165)
('z', 0.032152371206225674)
('e', 0.032237097276264594)
('k', 0.03236236770428016)
3
('e', 0.011480157198443579)
('t', 0.027079648249027234)
('p', 0.027742080155642022)
('a', 0.0291769953307393)
('i', 0.02999987587548638)
('x', 0.032120345914396886)
('f', 0.032406759922179)
('d', 0.032586386381322975)
('s', 0.03286390428015564)
('l', 0.03292779105058365)
4
('r', 0.010977754474708168)
('g', 0.02569435136186771 4)
('c', 0.02712001634241245)
('v', 0.029939174708171208)
('n', 0.030012027237354084)
('s', 0.03160716964980546)
('k', 0.031863145914396894)
('e', 0.03186979027237354)
('f', 0.03202659105058366)
('u', 0.03316422217898833)
```

```
5
('b', 0.01242810937500001)
('q', 0.02563354609375)
('m', 0.02898603320312499)
('f', 0.030174262890625005)
('x', 0.030305431640624998)
('u', 0.030315683593750004)
('i', 0.031945382812500006)
('o', 0.03240398515625)
('p', 0.03337633671874999)
('c', 0.033817656640625006)
6
('y', 0.012740912109375002)
('n', 0.026852539453125004)
('c', 0.028177584765624993)
('u', 0.028504367578125002)
('m', 0.029793353906250007)
('j', 0.031394379296875)
('f', 0.03152140312500001)
('r', 0.031999003515625006)
('l', 0.032193663671875004)
('z', 0.033881770312500004)
```

得出的秘钥会按照可能性进行排序，排在第一位的字符取出得到orderby

验证一下





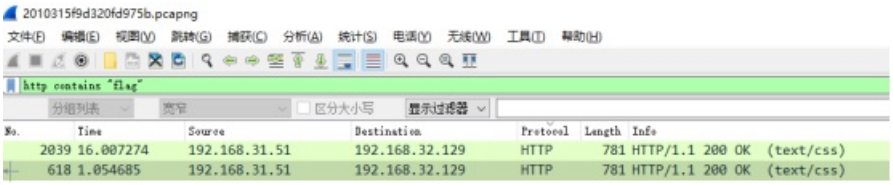
解密的结果最后几个单词明显有意义

按题目要求将秘钥大写，用md5加密得到flag

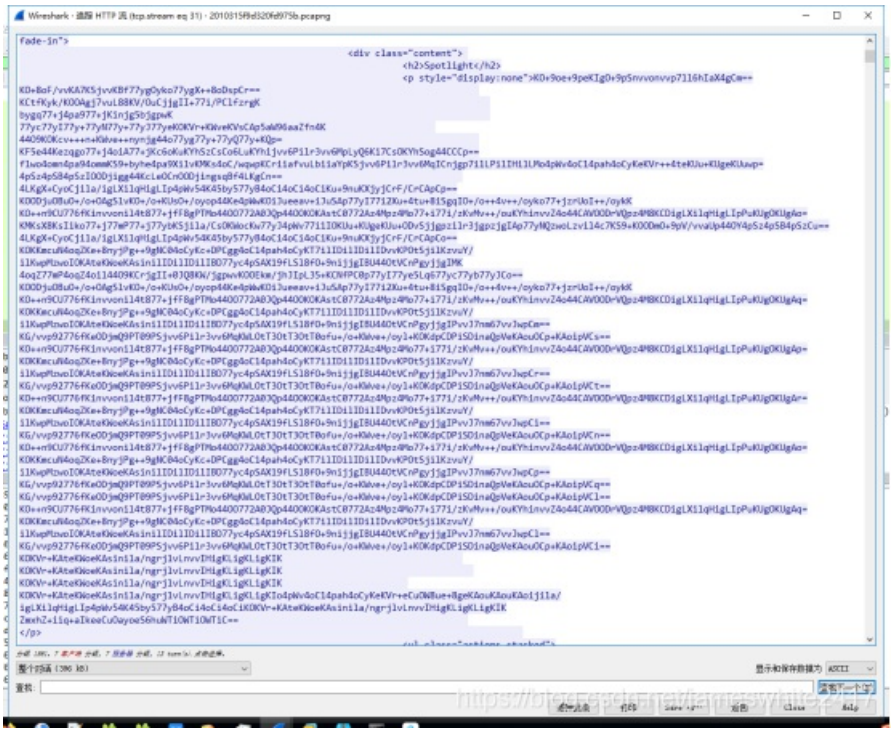
参考：python实现维吉尼亚秘钥破解 - 简书 https://www.jianshu.com/p/23e3dcb3f0e9

3.未解出Misc 颜文字之谜

过滤http中含有flag的内容



追踪http流



Base64解码

AAEncode解码无果 ……

4.未解出Misc 虚实之间

Binwalk一下

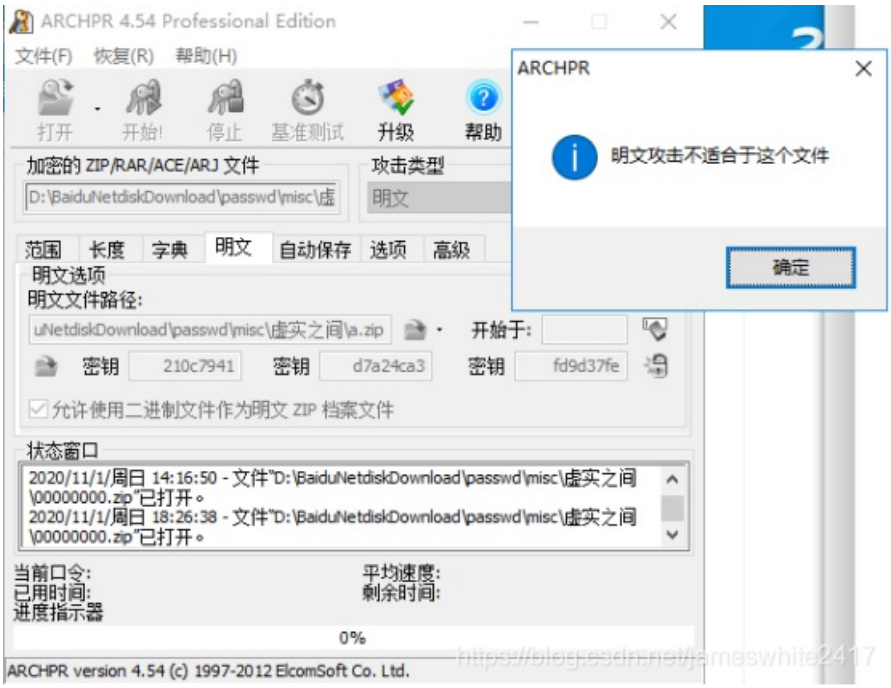
发现有两个zip包

Foremost分离出来

用360压缩打开，有个没加密的副本，把内容拷出来，保存到本地。

用好压打开有3个加密的文件。文件名也是mingwen-副本，还有个正本，crc32一样。应该是明文攻击。把刚才保存到本地的txt压缩一下，对比一下crc32一样，这样我们就有了明文文件。

用archpr进行明文攻击，可是报错了……



5. 未解出 web

文件包含 有过滤 做不动……

47.111.104.99:52601/index.php?file=phpinfo

Directive	Local Value	Master Value
last_modified	0	0
xbithack	0	0

Apache Environment

Variable	Value
HTTP_HOST	47.111.104.99:52601
HTTP_USER_AGENT	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:82.0) Gecko/20100101 Firefox/82.0
HTTP_ACCEPT	text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
HTTP_ACCEPT_LANGUAGE	zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
HTTP_ACCEPT_ENCODING	gzip, deflate
HTTP_CONNECTION	keep-alive
HTTP_UPGRADE_INSECURE_REQUESTS	1
PATH	/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
SERVER_SIGNATURE	<address>Apache/2.4.38 (Debian) Server at 47.111.104.99 Port 52601</address>
SERVER_SOFTWARE	Apache/2.4.38 (Debian)
SERVER_NAME	47.111.104.99
SERVER_ADDR	172.18.0.211
SERVER_PORT	52601
REMOTE_ADDR	120.229.79.131
DOCUMENT_ROOT	/var/www/html
REQUEST_SCHEME	http
CONTEXT_PREFIX	no value
CONTEXT_DOCUMENT_ROOT	/var/www/html
SERVER_ADMIN	webmaster@localhost
SCRIPT_FILENAME	/var/www/html/index.php
REMOTE_PORT	16946
GATEWAY_INTERFACE	CGI/1.1
SERVER_PROTOCOL	HTTP/1.1
REQUEST_METHOD	GET
QUERY_STRING	file=phpinfo
REQUEST_URI	/index.php?file=phpinfo
SCRIPT_NAME	/index.php