

ISCC2014 Basic（基础关）Writeup

转载

adr5970 于 2017-07-21 14:39:00 发布 92 收藏

文章标签: [数据结构与算法](#) [操作系统](#) [python](#)

原文链接: <http://www.cnblogs.com/ljbguanli/p/7217575.html>

版权

第一次参加ISCC（泪奔），整个过程下来还是学到了不少，接触了不少曾经没有遇到过的技术。

以下给出Basic的简要题解，若有不正确，还望大牛们指出 :-)

0x00 兵者诡道 Score: 50

兵者，诡道也。故能而世之不能，用而示之不用，近而示之远，远而示之近。

孙子在下命令中为防敌军窃取情报玩了一个小把戏。你能找出隐藏在这个文件里的password么？

思路：进入关卡页面后什么都没有，仅仅有个Tips：“hello,guess where is the key???”。果断在firebug里一阵狂搜，在Http响应头中找到flag。



```
GET index.php 200 OK script.iscc.org.cn 33 B 202.204.80.166:8

响应头信息 原始头信息
Connection close
Content-Type text/html
Date Fri, 02 May 2014 14:11:29 GMT
Key Welcome-to-ISCC
Server nginx/1.4.6 (Ubuntu)
Via 1.0 www.isclab.org (squid/3.1.10)
X-Cache MISS from www.isclab.org
X-Cache-Lookup MISS from www.isclab.org:80
X-Powered-By PHP/5.5.9-1ubuntu4

请求头信息 原始头信息
Accept text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding gzip, deflate
Accept-Language zh-cn,zh;q=0.8,en-us;q=0.5,en;q=0.3
Connection keep-alive
Host script.iscc.org.cn
Referer http://www.iscc.org.cn/challenges/2014/basic/basic01/
User-Agent Mozilla/5.0 (Windows NT 6.1; WOW64; rv:28.0) Gecko/20100101 Firefox/28.0
```

flag: Welcome-to-ISCC

0x01 知己知彼 Score: 50

知己知彼者，百战不殆；不知彼而知己，一胜一负；不知彼不知己，每战必殆。孙子手下将士截获了敌军命令密文4545 424545454542 454542 42 424542 424545，你能解密成明文，做到知己知彼吗？

思路：依据密文格式，能够看出总体就两个数45和42。相应ascii字符为“-”和“*”，顿时想到摩斯password。将相应的摩斯password“-- .-- --. --. --”解密为“MWGGERK”。提交答案，提示错误。再解密，经过简单处理，通过移位得到明文“ISCCANS”。改为小写提交。成功。

flag: isccans

0x02 正则指令 Score: 50

不知军之不能够进而为之进，不知军之不能够退而谓之退，是谓縻军。

正则表达式是“行军作战指挥命令”的一项“准则”，可帮助将领降低错误的下达指令的概率。请分析这段正则表达式，看看它透露了什么信息？正则表达式例如以下

```
\bw{3}(?<x>\.)[xyz](?<3>o)(?
```

```
<2>u)t\k<2>[bc][de]\k<x>c\3m\watch\?
```

```
v\=5x1vNTjbwcs&list=PL3ZQ5CpNulQm1cXMJ5M6tX3O5vyXnCYFd\b
```

提示:标题即flag

思路: 作为小菜的我。这么复杂的正则表示看着就头大。可是明显能够看出是个url。通过google正则关键部分 (PL3ZQ5CpNulQm1cXMJ5M6tX3O5vyXnCYFd) 能够知道该正则是一个Youtube视频的地址-<http://www.youtube.com/watch?v=5x1vNTjbwcs&list=PL3ZQ5CpNulQm1cXMJ5M6tX3O5vyXnCYFd>

v=5x1vNTjbwcs&list=PL3ZQ5CpNulQm1cXMJ5M6tX3O5vyXnCYFd, 于是打开该地址, 提交视频标题就可以。

(ps: 小菜我开始做此题的时候, 没有提示, 在提交标题时多了个空格, 搞得半天不懂这题要提交什么---, 甚是郁闷)



The screenshot shows a YouTube video player interface. The video is from ABC Action News and is titled "Chile hit by an 8.2 magnitude earthquake". The video player shows a news anchor and a reporter on the left, and a car on the right. The video player controls show a progress bar at 0:04 / 0:46. Below the video player, the title "Chile hit by an 8.2 magnitude earthquake" is displayed in red text. The video is from ABC Action News, which has 19,368 videos and 14,772 subscribers. The video has 3,058 views and 1 like.

flag: Chilehit by an 8.2 magnitude earthquake

0x03 搜索情报 Score: 50

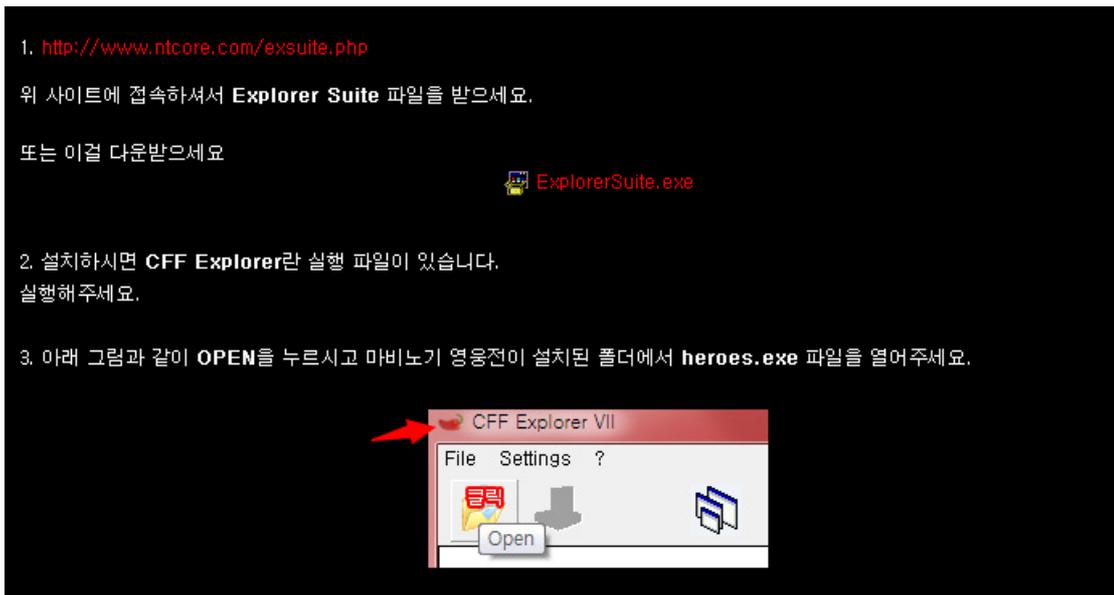
孙子曰：“是故智者之虑，必杂于利害。杂于利而务可信也，杂于害而患可解也。

”

如今有《Windows exploitation in 2013》。文章里面的一款PE工具貌似挺强大的，你能收集到情报，找到这款软件的名字吗？

<http://www.welivesecurity.com/2014/02/11/windows-exploitation-in-2013/>

思路：题目给了网址，果断打开，发现是英文（--我等屌丝怎么看得懂），泛读一顿后，发现文章底部有个 Process Explorer Tool，提交之。错误。然后发现文章中有张图片中有个软件非常可疑没有名字，于是各种 google、baidu 中。发现了以下这个思密达网址-<http://sekainonaka.tistory.com/167>。果断入之。发现里面讲的就是那软件-CFF Explorer，成功过之。



flag: CFFExplorer

0x04 指令暗战 Score: 50

孙子曰：“凡军之所欲击，城之所与攻，人之所欲杀。必先知其守将，左右。谒者，门者，舍人之姓名。令吾间必索知之。”我们派到敌方内部间谍拿到了敌军作战的指令，可是还不够明白，请你将消息确切转化为仔细的机器码，并提交（不需空格）。

MOV AX,CS

MOV DS,AX

MOV ES,AX

MOV CX,0020H

MOV DX,1004H

MOV BX,000CH

MOV AX,2300H

提示:提交时用大写就可以

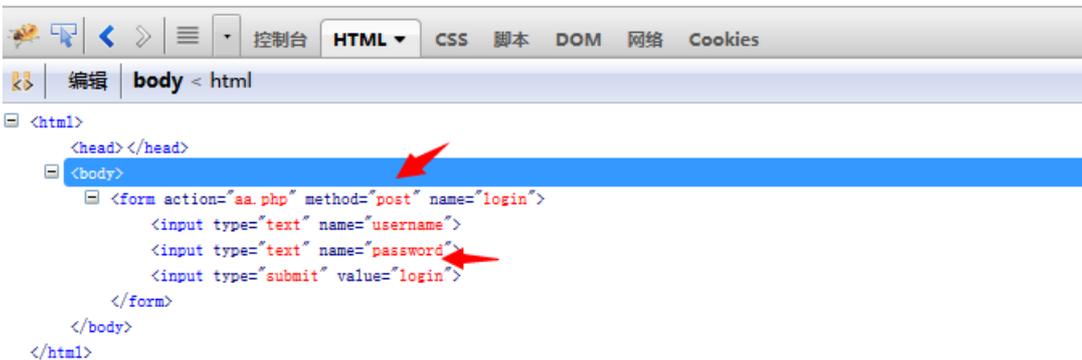
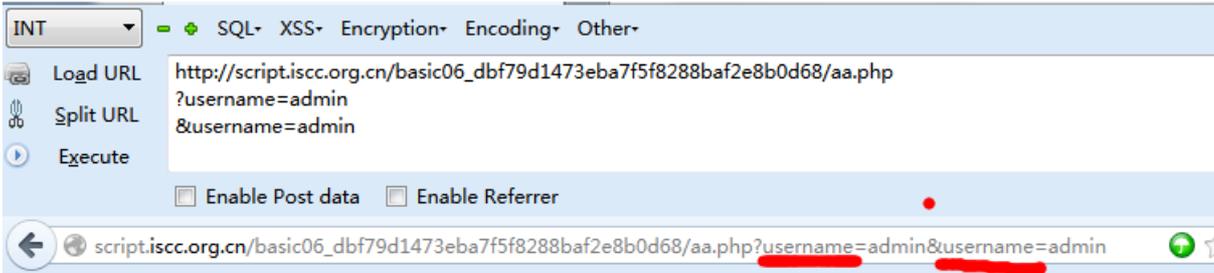
思路：此题一个主要的shellcode转换问题，直接将汇编代码相应的机器码（十六进制表示）写出就可以，通过不论什么方式都可，小菜我是通过直接在debug里面-a输入汇编代码看的-|，大牛勿喷。（ps：此题一开始无提示，看到题目中都全大写。我们答案肯定也得大写啦）

flag: 8CC88ED88EC0B92000BA0410BB0C00B80023

0x05 巧入军营 Score: 50

故曰：“知己知彼。胜乃不殆。知天知地，胜乃不穷。”请运用基础的“技巧”登陆此敌军页面。拿到flag。

思路：进入关卡页面后，发现是个畸形的登陆页面，随便输入后，发现两个input框变量名一样-同为username。So，改一个name为password。输入“admin admin”提交，无果。改请求method为POST后，再次提交。成功拿到flag。



key: 4qrPccxPe9

flag: 4qrPccxPe9

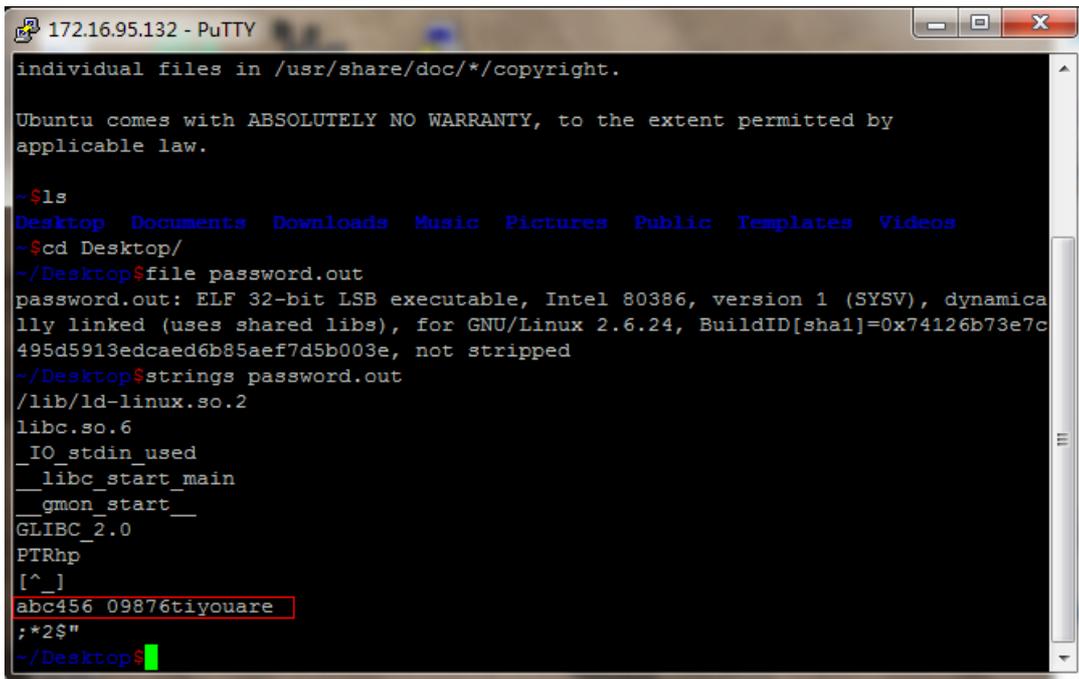
0x06 知兵之将 Score: 50

孙子曰：“故知兵之将。生民之司命。国家安危之主也。”

行军打仗最基础的是令行禁止。但知兵之将却要下达最正确的指令给他所带领的军队。

请用最基础的linux命令。获得flag，证明你是知兵之将。

思路：down下文件后，二话不说，用file命令查一下，发现是elf文件-32位（ps：.out文件一般为*unix系统下由gcc/g++自己主动编译连接生成的可执行文件），执行一下发现没有输出，于是用strings命令搜索下该程序中有没有什么字符串，果然！让我发现了秘密。



```
172.16.95.132 - PuTTY
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

~$ls
Desktop Documents Downloads Music Pictures Public Templates Videos
~$cd Desktop/
~/Desktop$file password.out
password.out: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically
linked (uses shared libs), for GNU/Linux 2.6.24, BuildID[sha1]=0x74126b73e7c
495d5913edcaed6b85aef7d5b003e, not stripped
~/Desktop$strings password.out
/lib/ld-linux.so.2
libc.so.6
_IO_stdin_used
__libc_start_main
__gmon_start__
GLIBC_2.0
PTRhp
[^_]
abc456_09876tiyouare
;*2$"
~/Desktop$
```

flag: abc456_09876tiyouare

0x07 虚实密文 Score: 50

孙子曰：“故策之而知得失之计，作之而知动静之理，行之而知死生之地。角之而知有余不足之处。

”

截获敌军情报是掌握敌军动向的一个好方法，或许这样的加密方法可能不多见。可是仅仅要能看破密文中的虚虚实实的消息，就能拿到我们所希望的信息。



FEAR CAN HOLD U PRISONER, HOPE CAN SET U FREE.

思路：文件down下来之后，发现是一张png，于是各种分析无果后，发现图片中的那句话穿插着字母正体和斜体。网上搜索一下。发现是培根password（ps：大家自行搜索普及一下）。于是依据培根password的加密方法，正体为a，斜体为b，转化成例如以下编码：aaba bba aaaa a baaaabaa, aaab bab bab a babb。再将其每5个分为一组：aabab|baaaa|aabaa|aabaa|aaabb|abbab|ababb。通过查培根编码表能够得到明文：freedom。

flag: freedom

0x08 经之五事索其情 Score: 50

孙子曰：“故经之以五事。校之以计而索其情：一曰道，二曰天。三曰地，四曰将，五曰法。”

你是否熟悉RSA算法？敌军正在用RSA算法加密，可是防范不周，被我军获取了部分信息。请解密密文是981， $w = 13$ ， $n = 2537$ 。分解式的一个因子是43的明文。

思路：题目说了是RSA加密（ps：大二学了password学。当时做题的时候哈都不知道-|），对着加解密公式，将变量带入就可以，此题难度较低。最后算得结果-702。（参考文

章：http://www.ruanyifeng.com/blog/2013/07/rsa_algorithm_part_two.html）

flag: 704

0x09 趁虚而入 Score: 100

孙子曰：“攻其不备，出其不意。此兵家之胜，不可先传也。”我方能不能借助敌人的指挥系统来获得信息，混入敌方内部绝对是一个不错的方法。如今因为对方的疏忽。我们能够如今已有“握手口令”（握手包）。请你帮助我方统帅搞定password吧！

思路：依据题目描写叙述，down下来的附件中包括有一个handshake。利用软件EWSA进行跑包就可以（ps：此题password较弱，不须要自行加入字典。使用软件默认就可以），跑出password: zzzzzzzz。



flag: zzzzzzzz

0x0a 出其不意 Score: 100

孙子曰：“出其所不趋，趁其所不意。”

敌方竟然把WIFI设成了ChinaUnicom，还真是出乎了我方的意料。但终于还是被我方发现了，而且我们发现他们用WEP加密,既然如此，那么就破解了他吧！从数据包中找到某人用ISCC账号通过该Wifi登录www.bitunion.org的password。

思路：题目给了两个.pkt的网络流量包，依据题意解题步骤应该是：首先破解wep的password，然后从网络流量中找到该人用ISCC账号登陆www.bitunion.org的password。

解题步骤。先将两个.pkt包转成.pcap格式，然后使用aircrack-ng 破解weppassword，再使用airdecap解密两个加密了的数据包。在里面搜索http协议的包，就可以找到password。

```
C:\Windows\system32\cmd.exe - aircrack-ng *.pcap

C:\Users\RickGray\Desktop>aircrack-ng *.pcap
Opening crackme1.pcap
Opening crackme2.pcap
Read 105205 packets.

# BSSID          ESSID          Encryption
1  28:10:7B:50:36:62  WEP (89349 IUs)
2  74:E5:43:DB:7E:88  WEP (4 IUs)

Index number of target network ? 1_
```

```
C:\Windows\system32\cmd.exe

AirCrack-ng 1.2 beta3

[00:00:01] Tested 26219 keys (got 89349 IUs)

KB  depth  byte(vote)
0   0/ 1    32<126720> 37<102656> 54<101888> 63<101632> 03<100608>
1   0/ 1    30<124160> A2<105216> 3E<102144> 81<101888> 69<99840>
2   0/ 1    31<121088> 58<102144> 09<100864> 36<99584> 44<98816>
3   0/ 1    34<110592> 1A<100608> 58<99328> F5<99328> D6<98816>
4   0/ 1    49<119552> 85<106752> 34<101888> D3<101376> 71<101376>
5   0/ 1    73<130048> AC<99840> 8A<99584> 69<99584> 00<99584>
6   0/ 1    63<108032> D8<100608> 37<100352> 93<99328> 24<99328>
7   0/ 1    43<113920> B3<102144> E6<101120> 6D<100096> 3C<100096>
8   0/ 1    77<119040> 82<106240> 2D<102912> C8<100608> CE<99840>
9   0/ 1    69<114688> B3<106496> CE<101376> B4<101120> AB<101120>
10  7/ 1    82<97280> E1<96768> F7<96000> 38<95744> 3C<95744>
11  0/ 1    42<102912> 7B<100864> 2B<100864> BF<100352> F8<100352>
12  0/ 1    59<106772> F1<102968> D8<99992> E3<99248> 71<99044>

KEY FOUND! [ 32:30:31:34:49:73:63:43:77:69:66:69:59 ] (ASCII: 2014IscCwif iY
)

Decrypted correctly: 100%

C:\Users\RickGray\Desktop>
```

```

C:\Windows\system32\cmd.exe

C:\Users\RickGray\Desktop>airdecap-ng -l -w 32303134497363437769666959 crackme1.pcap
Total number of packets read          90065
Total number of WEP data packets      77039
Total number of WPA data packets      37
Number of plaintext data packets      0
Number of decrypted WEP packets      76353
Number of corrupted WEP packets      686
Number of decrypted WPA packets      0

C:\Users\RickGray\Desktop>airdecap-ng -l -w 32303134497363437769666959 crackme2.pcap
Total number of packets read          15140
Total number of WEP data packets      12771
Total number of WPA data packets      5
Number of plaintext data packets      0
Number of decrypted WEP packets      12679
Number of corrupted WEP packets      92
Number of decrypted WPA packets      0

C:\Users\RickGray\Desktop>

```

crackme1-dec.pcap [Wireshark 1.10.8 (v1.10.8-2-g52a5244 from master-110)]

Filter: http.request.method == "POST"

No.	Time	Source	Destination	Protocol	Length	Info
16	3.077994	192.168.0.100	10.1.10.253	HTTP	1052	POST /logging.php?action=login HTTP/1.1 (application/x-www-form-urlencoded)
202	19.752248	192.168.0.100	192.168.0.1	HTTP/XML	1213	POST /soap.cgi?service=WANIPConn1 HTTP/1.1
215	22.876196	192.168.0.100	192.168.0.1	HTTP/XML	717	POST /soap.cgi?service=WANIPConn1 HTTP/1.1
224	22.976507	192.168.0.100	192.168.0.1	HTTP/XML	395	POST /soap.cgi?service=WANIPConn1 HTTP/1.1
267	35.317260	192.168.0.100	119.147.146.126	HTTP	321	POST /spp/?t=10933189 HTTP/1.0 (application/octet-stream)
2653	54.606456	192.168.0.100	192.168.0.1	HTTP/XML	1213	POST /soap.cgi?service=WANIPConn1 HTTP/1.1
2675	54.751415	192.168.0.100	192.168.0.1	HTTP/XML	717	POST /soap.cgi?service=WANIPConn1 HTTP/1.1
2686	54.864116	192.168.0.100	192.168.0.1	HTTP/XML	395	POST /soap.cgi?service=WANIPConn1 HTTP/1.1
2894	55.972496	192.168.0.100	192.168.0.1	HTTP/XML	1213	POST /soap.cgi?service=WANIPConn1 HTTP/1.1

crackme1-dec.pcap [Wireshark 1.10.8 (v1.10.8-2-g52a5244 from master-110)]

Filter: tcp.stream eq 1

Follow TCP Stream

Stream Content

```

POST /logging.php?action=login HTTP/1.1
Host: www.bitunion.org
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:27.0) Gecko/20100101 Firefox/27.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-cn,zh;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.bitunion.org/logging.php?action=login&referer=%2Fhome.php%3F
Cookie: __utma=17892390.700436135.1395996045.1396092464.1396100761.4;
__utmz=17892390.1396100761.4.3.utmcsr=bitren.com|utmccn=(referral)|utmcid=referral|
utmctt=/; cookietime=2592001; _cookietime=0; sid=ggDzowjN; bu_styleid=10;
__utmb=17892390.13.10.1396100761; __utmc=17892390; lastvisit=1396100286
Connection: keep-alive
content-type: application/x-www-form-urlencoded
Content-Length: 172

referer=%2Fhome.php%
3F&username=ISCC&password=Thisiskey&verify=37110&verifyimgid=c2beb91f4245281d3a5253eb31
fac24c&styleid=&cookietime=0&logInSubmit=%BB%E1%D4%B1%B5%7%2%BCHTTP/1.1 200 OK
Server: nginx/0.7.65
Date: Sat, 29 Mar 2014 14:20:50 GMT
Content-Type: text/html
Connection: keep-alive
X-Powered-By: PHP/5.3.2-1ubuntu4.22
Set-Cookie: _discuz_user=deleted; expires=Fri, 29-Mar-2013 14:20:49 GMT
Set-Cookie: _discuz_pw=deleted; expires=Fri, 29-Mar-2013 14:20:49 GMT
Set-Cookie: _discuz_userid=deleted; expires=Fri, 29-Mar-2013 14:20:49 GMT
Set-Cookie: sid=ggDzowjN
Set-Cookie: bu_styleid=10
Set-Cookie: _cookietime=0; expires=Sun, 29-Mar-2015 14:20:50 GMT
Set-Cookie: _discuz_user=ISCC
Set-Cookie: bu_styleid=10

```

Entire conversation (17790 bytes)

Find Save As Print ASCII EBCDIC Hex Dump C Arrays Raw

Filter Out This Stream Close

flag: Thisiskey

0x0b 择人任势 Score: 100

孙子曰：“故善战者。求之于势。不责于人，故能择人而任势。

”现有一vbs文件，你能从中找出理想的flag吗？

思路：down下题目附件，是一个vbs脚本（ps：vbs没有学过！！

），用编辑器打开，发现格式很糟糕，经过代码调整和分析。能够知道，他使用excute()函数运行了关键语句，将它的str逆序过来观察后。得到例如以下代码：

```
dim tbl,pwd,err,str,i,x
tbl=split("56|117|149|186|125|5|37|205|230|121|184|173|82|98|237|6|222|192|141|132|131|53|133|118|188|143|1
pwda=split("94|45|144|52|118|22|46|88|-39|-37|38|127|-11|-45", "|")
pwdb=split("157|24|6|107|251|35|94|67|136|199|12|34|97|202|188|31", "|")

err="password错误！ "
ok="你输入的password就是KEY。 "
x=0:a=0:b=0
str=inputbox("查看KEY请输入password","", "")

if (len(str)=14) then
for i=0 to 13
if Int(asc(mid(str,14-i,1))+pwda(i))=Int(tbl(i+pwdb(i))) then
x=x+1
exit for
end if
next
if x=14 then
msgbox ok
end if
else
msgbox err
end if
```

能够看到。匹配部分在for循环那里，我们改一下程序，使它不进行比较，直接输出tbl(i+pwdb(i))-pwda(i)就可以，得到数据序列：xxx，将其逆序就是flag了。

flag: vB5_5cR1pT.Vb\$

0x0c 庙算多寡，胜负定矣 Score: 100

孙子曰：“夫未战而庙算胜者。得算多也。未战而庙算不胜者，得算少也。

”

在一次作战中，你获得了对方用于加密的文件，和密文“+% =keky+% =jjnx”这或许关系到敌人的下一步作战行动，你是否能将此密文通过从加密文件里获得信息。把密文给解密出来？

思路：down下附件。是个exe程序，执行之，发现是个简单的加密程序。果断载进IDA分析。加载IDA后，发现该程序能够成功的F5，于是分析他的代码，发现了关键的加密部分（简单说明：每次读取待加密文件里的一个字符，经过处理，映射到还有一个字符上），分析清楚后，小菜我用python写了个简单的爆破程序，去爆明文，得到明文“&#a606D&#a5579”，可是提交发现不正确，细致看了下明文发现606D是十六进制，于是把a改为x后。在html中能够成功显示-“恭喜”，但题目仅仅需提交“恭ᗋ”就可以。

The screenshot shows the IDA Pro interface with the 'Functions window' on the left and the 'Pseudocode-B' window on the right. The 'Functions window' lists various system and runtime functions, with 'main' highlighted. The 'Pseudocode-B' window displays the following C code:

```

v10 = fopen("TempFile.pyq", "wb+");
if ( v10 )
{
    while ( !sub_401B00(v11) )
    {
        u9 = fgetc(v11);
        if ( u9 != -1 && u9 )
        {
            if ( u9 <= 47 || u9 > 96 )
            {
                if ( u9 > 46 )
                    u9 -= u9 % 61;
                else
                    u9 += u9 % 11;
            }
            else
            {
                u9 += 53;
            }
            fputc(u9, v10);
        }
    }
    fclose(v10);
    fclose(v11);
    sprintf(&u6, "del %s", &u7);
}

```

At the bottom of the window, it indicates 'Line 6 of 39' and 'main:43'.

爆破程序例如以下:

```

code = '+%=keky+=jjnx'
codelist = []
for i in code:
    codelist.append(ord(i))

result = ''
offset = 0
codedic = []

for i in codelist:
    codedic.append({offset: []})
    for k in range(256):
        x = k
        if x <= 47 or x > 96:
            if x > 46:
                x -= x % 61
            else:
                x += x % 11
        else:
            x += 53

        if x == i:
            result += chr(k)
            #print 'Found the %s nume: %s' % (offset, k)
            codedic[offset][offset].append(chr(k))
            break
    offset += 1

print 'S: %s' % code
print 'M: %s' % result

```

flag: 恭ᗋ

转载于:<https://www.cnblogs.com/ljbguanli/p/7217575.html>