

# XCTF 混合编码 WP

原创

lqvir 于 2020-12-13 20:21:53 发布 188 收藏

文章标签: [base64](#) [编码学](#)

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/weixin\\_46227016/article/details/111144475](https://blog.csdn.net/weixin_46227016/article/details/111144475)

版权

混合编码

## 过程

下载附件后得到一段很长的字符串

```
JiM3NjsmlzEyMjsmlzY5OyYjMTlwOyYjNzk7JiM4MzsmzU2OyYjMTlwOyYjNzc7JiM2ODsmlzY5OyYjMTE4OyYjNzc7JiM4NDsmlzY1OyYjNTI7JiM3NjsmlzEyMjsmlzEwNzsmzUzOyYjNzY7JiMxMjI7JiM2OTsmlzEyMDsmlz3OyYjODM7JiM1NjsmlzEyMDsmlz3OyYjNjg7JiMxMDc7JiMxMTg7JiM3Nzsmzlg0OyYjNjU7JiMxMjA7JiM3NjsmlzEyMjsmlzY5OyYjMTlwOyYjNzg7JiMxMDU7JiM1NjsmlzEyMDsmlz3OyYjODQ7JiM2OTsmlzExODsmlz3OyYjODQ7JiM5OTsmlzExODsmlz3OyYjODQ7JiM2OTsmlzUwOyYjNzY7JiMxMjI7JiM2OTsmlzEyMDsmlz4OyYjMTA1OyYjNTY7JiM1Mzsmz4OyYjMTlxOyYjNTY7JiM1Mzsmz5OyYjODM7JiM1NjsmlzEyMDsmlz3OyYjNjg7JiM5OTsmlzExODsmlz3OyYjODQ7JiM5OTsmlzExODsmlz3OyYjODQ7JiM2OTsmlzExOTsmlz2OyYjMTlyOyYjNjk7JiMxMTk7JiM3NzsmzY3OyYjNTY7JiMxMjA7JiM3NzsmzY4OyYjNjU7JiMxMTg7JiM3Nzsmzlg0OyYjNjU7JiMxMjA7JiM3NjsmlzEyMjsmlzY5OyYjMTE5OyYjNzc7JiMxMDU7JiM1NjsmlzEyMDsmlz3OyYjNjg7JiM2OTsmlzExODsmlz3OyYjODQ7JiM2OTsmlzExOTsmlz2OyYjMTlyOyYjMTA3OyYjNTM7JiM3NjsmlzEyMjsmlzY5OyYjMTE5OyYjNzc7JiM4MzsmzU2OyYjMTlwOyYjNzc7JiM4NDsmlzEwNzsmzExODsmlz3OyYjODQ7JiM2OTsmlzEyMDsmlz2OyYjMTlyOyYjNjk7JiMxMjA7JiM3ODsmlzY3OyYjNTY7JiMxMjA7JiM3NzsmzY4OyYjMTAzOyYjMTE4OyYjNzc7JiM4NDsmlzY1OyYjMTE5Ow==
```

发现最后为==, 说明很有可能采用了base64编码, 于是使用python进行base64解码, 得到新的字符串

```
&#x76;&#x122;&#x69;&#x120;&#x79;&#x83;&#x56;&#x120;&#x77;&#x68;&#x69;&#x118;&#x77;&#x84;&#x65;&#x52;&#x76;&#x122;&#x107;&#x53;&#x76;&#x122;&#x69;&#x120;&#x77;&#x83;&#x56;&#x120;&#x77;&#x68;&#x107;&#x118;&#x77;&#x84;&#x65;&#x120;&#x76;&#x122;&#x69;&#x120;&#x78;&#x105;&#x56;&#x120;&#x77;&#x84;&#x69;&#x118;&#x79;&#x84;&#x99;&#x118;&#x77;&#x84;&#x69;&#x50;&#x76;&#x122;&#x69;&#x120;&#x78;&#x105;&#x56;&#x53;&#x78;&#x121;&#x56;&#x53;&#x79;&#x83;&#x56;&#x120;&#x77;&#x68;&#x99;&#x118;&#x79;&#x84;&#x99;&#x118;&#x77;&#x84;&#x69;&#x119;&#x76;&#x122;&#x69;&#x19;&#x77;&#x67;&#x56;&#x120;&#x77;&#x68;&#x65;&#x118;&#x77;&#x84;&#x65;&#x120;&#x76;&#x122;&#x69;&#x119;&#x77;&#x105;&#x56;&#x120;&#x77;&#x68;&#x69;&#x118;&#x77;&#x84;&#x69;&#x119;&#x6;&#x122;&#x107;&#x53;&#x76;&#x122;&#x69;&#x119;&#x77;&#x83;&#x56;&#x120;&#x77;&#x84;&#x107;&#x118;&#x77;&#x84;&#x69;&#x120;&#x76;&#x122;&#x69;&#x120;&#x78;&#x67;&#x56;&#x120;&#x77;&#x68;&#x103;&#x118;&#x77;&#x84;&#x65;&#x119;
```

查询这种编码方式得到为Webdings

对照表

[https://blog.csdn.net/weixin\\_33989780/article/details/85868511?utm\\_medium=distribute.pc\\_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-4.control&depth\\_1-utm\\_source=distribute.pc\\_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-4.control](https://blog.csdn.net/weixin_33989780/article/details/85868511?utm_medium=distribute.pc_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-4.control&depth_1-utm_source=distribute.pc_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-4.control)

使用python将将这种编码方式转化为对应字母, 发现仍然是乱码

```
LzExOS8xMDEvMTA4Lzk5LzExMS8xMDkvMTAxLzExNi8xMTEvOTcvMTE2LzExNi85Ny85OS8xMDcvOTcvMTEwLzEwMC8xMDAvMTAxLzEwMi8xMDEvMTEwLzEwMS8xMTkvMTEwLzExNC8xMDgvMTAw
```

发现不符合MD5加密规则, 首先尝试base64, 发现可以得到有意义的字符串, 得到答案。

```
import base64
s=r'JiM3NjZmIzEyMjZmIzY5OyYjMTIwOyYjNzZk7JiM4MzsmIzU2OyYjMTIwOyYjNzc7JiM2ODsmIzY5OyYjMTE4OyYjNzc7JiM4NDsmIzY1OyYj
NTI7JiM3NjZmIzEyMjZmIzEwNzsmIzUzOyYjNzY7JiMxMjI7JiM2OTsmIzEyMDsmIzc3OyYjODM7JiM1NjZmIzEyMDsmIzc3OyYjNjg7JiMxMdc7
JiMxMTg7JiM3NzsmIzg0OyYjNjU7JiMxMjA7JiM3NjZmIzEyMjZmIzY5OyYjMTIwOyYjNzg7JiMxMDU7JiM1NjZmIzEyMDsmIzc3OyYjODQ7JiM2
OTsmIzExODsmIzc5OyYjODQ7JiM5OTsmIzExODsmIzc3OyYjODQ7JiM2OTsmIzUwOyYjNzY7JiMxMjI7JiM2OTsmIzEyMDsmIzc4OyYjMTA1OyYj
NTY7JiM1MzsmIzc4OyYjMTIxOyYjNTY7JiM1MzsmIzc5OyYjODM7JiM1NjZmIzEyMDsmIzc3OyYjNjg7JiM5OTsmIzExODsmIzc5OyYjODQ7JiM5
OTsmIzExODsmIzc3OyYjODQ7JiM2OTsmIzExOTsmIzc2OyYjMTIyOyYjNjk7JiMxMTk7JiM3NzsmIzY3OyYjNTY7JiMxMjA7JiM3NzsmIzY4OyYj
NjU7JiMxMTg7JiM3NzsmIzg0OyYjNjU7JiMxMjA7JiM3NjZmIzEyMjZmIzY5OyYjMTE5OyYjNzc7JiMxMDU7JiM1NjZmIzEyMDsmIzc3OyYjNjg7
JiM2OTsmIzExODsmIzc3OyYjODQ7JiM2OTsmIzExOTsmIzc2OyYjMTIyOyYjMTA3OyYjNTM7JiM3NjZmIzEyMjZmIzY5OyYjMTE5OyYjNzc7JiM4
MzsmIzU2OyYjMTIwOyYjNzc7JiM4NDsmIzEwNzsmIzExODsmIzc3OyYjODQ7JiM2OTsmIzEyMDsmIzc2OyYjMTIyOyYjNjk7JiMxMjA7JiM3ODsm
IzY3OyYjNTY7JiMxMjA7JiM3NzsmIzY4OyYjMTAzOyYjMTE4OyYjNzc7JiM4NDsmIzY1OyYjMTE5Ow=='
st=base64.b64decode(s).decode("utf-8")
l=list(st.split(';'))
l.pop(len(l)-1)
ans=[]
for i in l:
    ans.append(chr(int(i[2:])))

print(l)
t=''
for i in ans:
    t+=i
print(t)
st=base64.b64decode(t).decode("utf-8")
t=list(st.split(r'/'))
t.pop(0)
for i in t:
    print(chr(int(i)),end='')
```