XCTF的simple_js

原创

小 白 渣 于 2019-09-27 17:33:08 发布 763 收藏

分类专栏： 十六进制转换文本字符 python脚本 js代码 文章标签： xctf web简单题

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/qq_45552960/article/details/101547025

版权

十六进制转换文本字符 同时被 3 个专栏收录

1 篇文章 0 订阅

订阅专栏

python脚本

1 篇文章 0 订阅

订阅专栏

js代码

1 篇文章 0 订阅

订阅专栏

打开之后，直接查看源代码，发现一串JS代码

```
function dechiffre(pass_enc){
        var pass = "70,65,85,88,32,80,65,83,83,87,79,82,68,32,72,65,72,65";
        var tab  = pass_enc.split(',');
                var tab2 = pass.split(',');var i,j,k,l=0,m,n,o,p = "";i = 0;j = tab.length;
                        k = j + (l) + (n=0);
                        n = tab2.length;
                        for(i = (o=0); i < (k = j = n); i++ ){o = tab[i-l];p += String.fromCharCode((o = tab2[i]
));
                                if(i == 5)break;}
                        for(i = (o=0); i < (k = j = n); i++ ){
                        o = tab[i-l];
                                if(i > 5 && i < k-1)
                                        p += String.fromCharCode((o = tab2[i]));
                        }
        p += String.fromCharCode(tab2[17]);
        pass = p;return pass;
    }
    String["fromCharCode"](dechiffre("\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x
2c\x31\x31\x34\x2c\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\x39\x2c\x35\x30"));

    h = window.prompt('Enter password');
    alert( dechiffre(h) );
```

要做这道题就得看懂JS代码表达的意思，首先来了解代码中出现的一些方法

split() 方法用于把一个字符串分割成字符串数组。
fromCharCode() 可接受一个指定的 Unicode 值，然后返回一个字符串。
prompt()方法用于显示可提示用户进行输入的对话框。

在本例中，我们将根据 Unicode 来输出 "HELLO" 和 "ABC"：

```
<script type="text/javascript">

document.write(String.fromCharCode(72,69,76,76,79))
document.write("<br />")
document.write(String.fromCharCode(65,66,67))

</script>
```

以上代码的输出：

```
HELLO
ABC
```

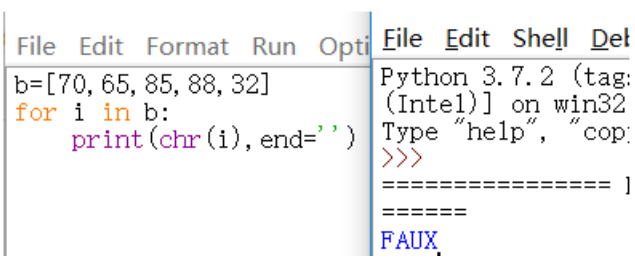
https://blog.csdn.net/qq_43431158
https://blog.csdn.net/qq_45552960

了解之后，便可以观察代码了

```
function dechiffre(pass_enc){
    var pass = "70,65,85,88,32,80,65,83,83,87,79,82,68,32,72,65,72,65";
    var tab  = pass_enc.split(',');
            var tab2 = pass.split(',');var i,j,k,l=0,m,n,o,p = "";i = 0;j = tab.length;
                    k = j + (1) + (n=0);
                    n = tab2.length;
                    for(i = (o=0); i < (k = j = n); i++ ){o = tab[i-1];p += String.fromCharCod
                            if(i == 5)break;}
                    for(i = (o=0); i < (k = j = n); i++ ){
                    o = tab[i-1];
                            if(i > 5 && i < k-1)
                                    p += String.fromCharCode((o = tab2[i]));
                    }
    p += String.fromCharCode(tab2[17]);
    pass = p;return pass;
}
```

https://blog.csdn.net/qq_43431158
https://blog.csdn.net/qq_45552960

这一部分代码分为两个循环，但是仔细观察的话，第一个循环是将前五个Unicode 值，然后返回一个字符串，写一个简单的python脚本验证下



>>>

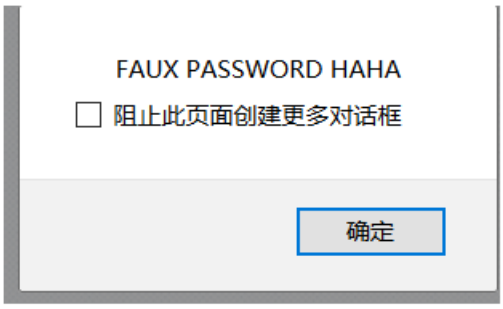
https://blog.csdn.net/qq_45552960

后一个循环功能是一样的，只不过是把后面的Unicode 值，返回成字符串。

```
b=[80, 65, 83, 83, 87, 79, 82, 68, 32, 72, 65, 72, 65]
for i in b:
    print(chr(i), end='')
```

```
File  Edit  Shell  Debug  Options
Python 3.7.2 (tags/v3.7.2:9a3f
(Intel)] on win32
Type "help", "copyright", "cre
>>>
================ RESTART: C:\l
======
FAUX
>>>
================ RESTART: C:\l
======
PASSWORD HAHA
>>>
```

https://blog.csdn.net/qq_43431158

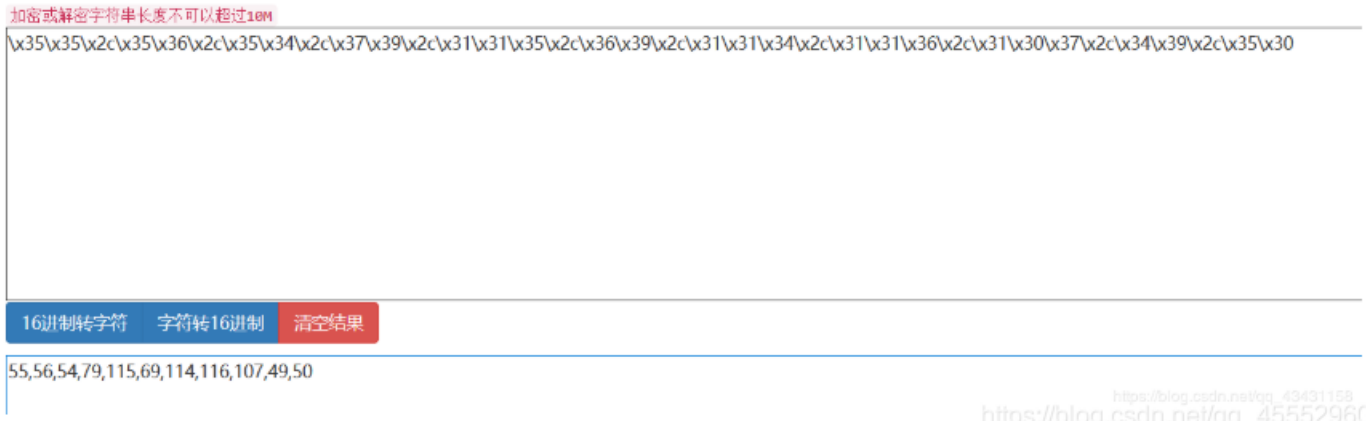


https://blog.csdn.net/qq_45552960

返回的pass为

即不管我们输入什么，最终显示都是这个，所以这个是假的密码，那么真正的密码应该就是

十六进制转化为文本字符



https://blog.csdn.net/qq_43431158
https://blog.csdn.net/qq_45552960

再用简单的python跑一下，即可得出flag

```
File  Edit  Format  Run  Options  Window  Help
b=[55, 56, 54, 79, 115, 69, 114, 116, 107, 49, 50]
for i in b:
    print(chr(i), end='')
```

```
File  Edit  Shell  Del
Python 3.7.2 (tag
(Intel)] on win32
Type "help", "cop
>>>
================
======
```

题虽然不是太难，但是可以学习到很多知识，接下来得快点学习python以写脚本用，方便做题。

https://blog.csdn.net/qq_45552960