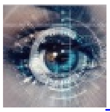
XCTF进阶区Crypto之flag_in_your_hand write up

原创

KogRow 于 2020-07-02 15:35:30 发布 516 收藏 2
分类专栏： CTF Crypto 文章标签： crypto ctf

本文链接：https://blog.csdn.net/shuaicenglou3032/article/details/107085207
版权



CTF 同时被 2 个专栏收录
59 篇文章 4 订阅
订阅专栏

Crypto
11 篇文章 0 订阅
订阅专栏

下载下来查看html源码和js源码：

```
<html>
 <head>
  <title>Flag in your Hand</title>
  <style type="text/css">
   body {
    padding-left: 30%;
   }

   #flag {
    font-family: Garamond, serif;
    font-size: 36px;
   }

   #flagtitle {
    font-family: Garamond, serif;
    font-size: 24px;
   }

   .rightflag {
    color: green;
   }

   .wrongflag {
    color: red;
   }
  </style>
  <script src="script-min.js"></script>
  <script type="text/javascript">
   var ic = false;
   var fg = "";

   function getFlag() {
    var token = document.getElementById("secToken").value;
    ic = checkToken(token);
    fg = bm(token);
    showFlag()
   }

   function showFlag() {
    var t = document.getElementById("flagTitle");
    var f = document.getElementById("flag");
    t.innerText = !!ic ? "You got the flag below!!" : "Wrong!";
    t.className = !!ic ? "rightflag" : "wrongflag";
    f.innerText = fg;
   }
  </script>
 </head>
 <body>
  <h1>Flag in your Hand</h1>
  <p>Type in some token to get the flag.</p>
  <p>Tips: Flag is in your hand.</p>
  <div>
   <p>
    <span>Token:</span>
    <span><input type="text" id="secToken"/></span>
   </p>
   <p>
    <input type="button" value="Get flag!" onclick="getFlag()" />
   </p>
  </div>
  <div>
   <p id="flagTitle"></p>
   <p id="flag"></p>
  </div>
 </body>
</html>
```

```
function hm(s) {
    return rh(rstr(str2rstr_utf8(s)));
}
function bm(s) {
    return rb(rstr(str2rstr_utf8(s)));
}
function rstr(s) {
    return binl2rstr(binl(rstr2binl(s), s.length * 8));
}
function checkToken(s) {
    return s === "FAKE-TOKEN";
}
function rh(ip) {
    try {
        hc
    } catch (e) {
        hc = 0;
    }
    var ht = hc ? "0123456789ABCDEF" : "0123456789abcdef";
    var op = "";
    var x;
    for (var i = 0; i < ip.length; i++) {
        x = ip.charCodeAt(i);
        op += ht.charAt((x >>> 4) & 0x0F) + ht.charAt(x & 0x0F);
    }
    return op;
```

```
}
function rb(ip) {
    try {
        bp
    } catch (e) {
        bp = '';
    }
    var b = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/";
    var op = "";
    var len = ip.length;
    for (var i = 0; i < len; i += 3) {
        var t = (ip.charCodeAt(i) << 16) | (i + 1 < len ? ip.charCodeAt(i + 1) << 8 : 0) | (i + 2 < len ? i
        for (var j = 0; j < 4; j++) {
            if (i * 8 + j * 6 > ip.length * 8)
                op += bp;
            else
                op += b.charAt((t >>> 6 * (3 - j)) & 0x3F);
        }
    }
    return op;
}
function ck(s) {
    try {
        ic
    } catch (e) {
        return;
    }
    var a = [118, 104, 102, 120, 117, 108, 119, 124, 48,123,101,120];
    if (s.length == a.length) {
        for (i = 0; i < s.length; i++) {
            if (a[i] - s.charCodeAt(i) != 3)
                return ic = false;
        }
        return ic = true;
    }
    return ic = false;
}
function str2rstr_utf8(input) {
    var output = "";
    var i = -1;
    var x, y;
    while (++i < input.length) {
        x = input.charCodeAt(i);
        y = i + 1 < input.length ? input.charCodeAt(i + 1) : 0;
        if (0xD800 <= x && x <= 0xDBFF && 0xDC00 <= y && y <= 0xDFFF) {
            x = 0x10000 + ((x & 0x03FF) << 10) + (y & 0x03FF);
            i++;
        }
        if (x <= 0x7F)
            output += String.fromCharCode(x);
        else if (x <= 0x7FF)
            output += String.fromCharCode(0xC0 | ((x >>> 6) & 0x1F), 0x80 | (x & 0x3F));
        else if (x <= 0xFFFF)
            output += String.fromCharCode(0xE0 | ((x >>> 12) & 0x0F), 0x80 | ((x >>> 6) & 0x3F), 0x80 | (x
        else if (x <= 0x1FFFFF)
            output += String.fromCharCode(0xF0 | ((x >>> 18) & 0x07), 0x80 | ((x >>> 12) & 0x3F), 0x80 | ((
    }
    return output;
}
```

```
function rstr2binl(input) {
    console.log(20>>2)
    var output = Array(input.length >> 2);
    for (var i = 0; i < output.length; i++)
        output[i] = 0;
    for (var i = 0; i < input.length * 8; i += 8)
        output[i >> 5] |= (input.charCodeAt(i / 8) & 0xFF) << (i % 32);
    return output;
}
function binl2rstr(i) {
    var o = "";
    for (var j = 0; j < i.length * 32; j += 8)
        o += String.fromCharCode((i[j >> 5] >>> (j % 32)) & 0xFF);
    return o;
}
function binl(x, len) {
    s = binl2rstr(x);
    x[len >> 5] |= 0x80 << ((len) % 32);
    x[(((len + 64) >>> 9) << 4) + 14] = len;
    var a = 1732584193;
    var b = -271733879;
    var c = -1732584194;
    var d = 271733878;
    for (var i = 0; i < x.length; i += 16) {
        var olda = a;
        var oldb = b;
        var oldc = c;
        var oldd = d;
        a = ff(a, b, c, d, x[i + 0], 7, -680876936);
        d = ff(d, a, b, c, x[i + 1], 12, -389564586);
        c = ff(c, d, a, b, x[i + 2], 17, 606105819);
        b = ff(b, c, d, a, x[i + 3], 22, -1044525330);
        a = ff(a, b, c, d, x[i + 4], 7, -176418897);
        d = ff(d, a, b, c, x[i + 5], 12, 1200080426);
        c = ff(c, d, a, b, x[i + 6], 17, -1473231341);
        b = ff(b, c, d, a, x[i + 7], 22, -45705983);
        a = ff(a, b, c, d, x[i + 8], 7, 1770035416);
        d = ff(d, a, b, c, x[i + 9], 12, -1958414417);
        c = ff(c, d, a, b, x[i + 10], 17, -42063);
        b = ff(b, c, d, a, x[i + 11], 22, -1990404162);
        a = ff(a, b, c, d, x[i + 12], 7, 1804603682);
        d = ff(d, a, b, c, x[i + 13], 12, -40341101);
        c = ff(c, d, a, b, x[i + 14], 17, -1502002290);
        b = ff(b, c, d, a, x[i + 15], 22, 1236535329);
        ck(s);
        a = gg(a, b, c, d, x[i + 1], 5, -165796510);
        d = gg(d, a, b, c, x[i + 6], 9, -1069501632);
        c = gg(c, d, a, b, x[i + 11], 14, 643717713);
        b = gg(b, c, d, a, x[i + 0], 20, -373897302);
        a = gg(a, b, c, d, x[i + 5], 5, -701558691);
        d = gg(d, a, b, c, x[i + 10], 9, 38016083);
        c = gg(c, d, a, b, x[i + 15], 14, -660478335);
        b = gg(b, c, d, a, x[i + 4], 20, -405537848);
        a = gg(a, b, c, d, x[i + 9], 5, 568446438);
        d = gg(d, a, b, c, x[i + 14], 9, -1019803690);
        c = gg(c, d, a, b, x[i + 3], 14, -187363961);
        b = gg(b, c, d, a, x[i + 8], 20, 1163531501);
        a = gg(a, b, c, d, x[i + 13], 5, -1444681467);
        d = gg(d, a, b, c, x[i + 2], 9, -51403784);
        c = gg(c, d, a, b, x[i + 7], 14, 1735328473);
        c = gg(c, d, a, b, x[i + 7], 14, 1735328473);
        b = gg(b, c, d, a, x[i + 12], 20, -1926607734);
        a = hh(a, b, c, d, x[i + 5], 4, -378558);
        d = hh(d, a, b, c, x[i + 8], 11, -2022574463);
        c = hh(c, d, a, b, x[i + 11], 16, 1839030562);
        b = hh(b, c, d, a, x[i + 14], 23, -35309556);
        a = hh(a, b, c, d, x[i + 1], 4, -1530992060);
        d = hh(d, a, b, c, x[i + 4], 11, 1272893353);
        c = hh(c, d, a, b, x[i + 7], 16, -155497632);
        b = hh(b, c, d, a, x[i + 10], 23, -1094730640);
        a = hh(a, b, c, d, x[i + 13], 4, 681279174);
        d = hh(d, a, b, c, x[i + 0], 11, -358537222);
        c = hh(c, d, a, b, x[i + 3], 16, -722521979);
        b = hh(b, c, d, a, x[i + 6], 23, 76029189);
        a = hh(a, b, c, d, x[i + 9], 4, -640364487);
        d = hh(d, a, b, c, x[i + 12], 11, -421815835);
        c = hh(c, d, a, b, x[i + 15], 16, 530742520);
        b = hh(b, c, d, a, x[i + 2], 23, -995338651);
        a = ii(a, b, c, d, x[i + 0], 6, -198630844);
        d = ii(d, a, b, c, x[i + 7], 10, 1126891415);
        c = ii(c, d, a, b, x[i + 14], 15, -1416354905);
        b = ii(b, c, d, a, x[i + 5], 21, -57434055);
        a = ii(a, b, c, d, x[i + 12], 6, 1700485571);
        d = ii(d, a, b, c, x[i + 3], 10, -1894986606);
        c = ii(c, d, a, b, x[i + 10], 15, -1051523);
        b = ii(b, c, d, a, x[i + 1], 21, -2054922799);
        a = ii(a, b, c, d, x[i + 8], 6, 1873313359);
        d = ii(d, a, b, c, x[i + 15], 10, -30611744);
        c = ii(c, d, a, b, x[i + 6], 15, -1560198380);
        b = ii(b, c, d, a, x[i + 13], 21, 1309151649);
        a = ii(a, b, c, d, x[i + 4], 6, -145523070);
        d = ii(d, a, b, c, x[i + 11], 10, -1120210379);
        c = ii(c, d, a, b, x[i + 2], 15, 718787259);
        b = ii(b, c, d, a, x[i + 9], 21, -343485551);
        a = sa(a, olda);
        b = sa(b, oldb);
        c = sa(c, oldc);
        d = sa(d, oldd);
    }
    return Array(a, b, c, d);
}
function cmn(q, a, b, x, s, t) {
    return sa(br(sa(sa(a, q), sa(x, t)), s), b);
}
function ff(a, b, c, d, x, s, t) {
    return cmn((b & c) | ((~b) & d), a, b, x, s, t);
}
function gg(a, b, c, d, x, s, t) {
    return cmn((b & d) | (c & (~d)), a, b, x, s, t);
}
function hh(a, b, c, d, x, s, t) {
    return cmn(b ^ c ^ d, a, b, x, s, t);
}
function ii(a, b, c, d, x, s, t) {
    return cmn(c ^ (b | (~d)), a, b, x, s, t);
}
function sa(x, y) {
    var lsw = (x & 0xFFFF) + (y & 0xFFFF);
    var msw = (x >> 16) + (y >> 16) + (lsw >> 16);
    return (msw << 16) | (lsw & 0xFFFF);
```

```
}
function br(n, c) {
    return (n << c) | (n >>> (32 - c));
}
```

明面上的

function checkToken(s) {

  return s === "FAKE-TOKEN";

}

就是个障眼法。

接着看bm(s)这个方法：

str2rstr_utf8是一个将字符串转为utf8格式的字符串，不影响。继续往下看这个函数：

```
function binl(x, len) {
    s = binl2rstr(x);
    x[len >> 5] |= 0x80 << ((len) % 32);
    x[(((len + 64) >>> 9) << 4) + 14] = len;
    var a = 1732584193;
    var b = -271733879;
    var c = -1732584194;
    var d = 271733878;
    for (var i = 0; i < x.length; i += 16) {
        var olda = a;
        var oldb = b;
        var oldc = c;
        var oldd = d;
        a = ff(a, b, c, d, x[i + 0], 7, -680876936);
        d = ff(d, a, b, c, x[i + 1], 12, -389564586);
        c = ff(c, d, a, b, x[i + 2], 17, 606105819);
        b = ff(b, c, d, a, x[i + 3], 22, -1044525330);
        a = ff(a, b, c, d, x[i + 4], 7, -176418897);
        d = ff(d, a, b, c, x[i + 5], 12, 1200080426);
        c = ff(c, d, a, b, x[i + 6], 17, -1473231341);
        b = ff(b, c, d, a, x[i + 7], 22, -45705983);
        a = ff(a, b, c, d, x[i + 8], 7, 1770035416);
        d = ff(d, a, b, c, x[i + 9], 12, -1958414417);
        c = ff(c, d, a, b, x[i + 10], 17, -42063);
        b = ff(b, c, d, a, x[i + 11], 22, -1990404162);
        a = ff(a, b, c, d, x[i + 12], 7, 1804603682);
        d = ff(d, a, b, c, x[i + 13], 12, -40341101);
        c = ff(c, d, a, b, x[i + 14], 17, -1502002290);
        b = ff(b, c, d, a, x[i + 15], 22, 1236535329);
        ck(s);
        a = gg(a, b, c, d, x[i + 1], 5, -165796510);
        d = gg(d, a, b, c, x[i + 6], 9, -1069501632);
        c = gg(c, d, a, b, x[i + 11], 14, 643717713);
        b = gg(b, c, d, a, x[i + 0], 20, -373897302);
        a = gg(a, b, c, d, x[i + 5], 5, -701558691);
        d = gg(d, a, b, c, x[i + 10], 9, 38016083);
        c = gg(c, d, a, b, x[i + 15], 14, -660478335);
        b = gg(b, c, d, a, x[i + 4], 20, -405537848);
        a = gg(a, b, c, d, x[i + 9], 5, 568446438);
        d = gg(d, a, b, c, x[i + 14], 9, -1019803690);
        d = gg(d, a, b, c, x[i + 14], 9, -1019803690);
        c = gg(c, d, a, b, x[i + 3], 14, -187363961);
        b = gg(b, c, d, a, x[i + 8], 20, 1163531501);
        a = gg(a, b, c, d, x[i + 13], 5, -1444681467);
        d = gg(d, a, b, c, x[i + 2], 9, -51403784);
        c = gg(c, d, a, b, x[i + 7], 14, 1735328473);
        b = gg(b, c, d, a, x[i + 12], 20, -1926607734);
        a = hh(a, b, c, d, x[i + 5], 4, -378558);
        d = hh(d, a, b, c, x[i + 8], 11, -2022574463);
        c = hh(c, d, a, b, x[i + 11], 16, 1839030562);
        b = hh(b, c, d, a, x[i + 14], 23, -35309556);
        a = hh(a, b, c, d, x[i + 1], 4, -1530992060);
        d = hh(d, a, b, c, x[i + 4], 11, 1272893353);
        c = hh(c, d, a, b, x[i + 7], 16, -155497632);
        b = hh(b, c, d, a, x[i + 10], 23, -1094730640);
        a = hh(a, b, c, d, x[i + 13], 4, 681279174);
        d = hh(d, a, b, c, x[i + 0], 11, -358537222);
        c = hh(c, d, a, b, x[i + 3], 16, -722521979);
        b = hh(b, c, d, a, x[i + 6], 23, 76029189);
        a = hh(a, b, c, d, x[i + 9], 4, -640364487);
        d = hh(d, a, b, c, x[i + 12], 11, -421815835);
        c = hh(c, d, a, b, x[i + 15], 16, 530742520);
        b = hh(b, c, d, a, x[i + 2], 23, -995338651);
        a = ii(a, b, c, d, x[i + 0], 6, -198630844);
        d = ii(d, a, b, c, x[i + 7], 10, 1126891415);
        c = ii(c, d, a, b, x[i + 14], 15, -1416354905);
        b = ii(b, c, d, a, x[i + 5], 21, -57434055);
        a = ii(a, b, c, d, x[i + 12], 6, 1700485571);
        d = ii(d, a, b, c, x[i + 3], 10, -1894986606);
        c = ii(c, d, a, b, x[i + 10], 15, -1051523);
        b = ii(b, c, d, a, x[i + 1], 21, -2054922799);
        a = ii(a, b, c, d, x[i + 8], 6, 1873313359);
        d = ii(d, a, b, c, x[i + 15], 10, -30611744);
        c = ii(c, d, a, b, x[i + 6], 15, -1560198380);
        b = ii(b, c, d, a, x[i + 13], 21, 1309151649);
        a = ii(a, b, c, d, x[i + 4], 6, -145523070);
        d = ii(d, a, b, c, x[i + 11], 10, -1120210379);
        c = ii(c, d, a, b, x[i + 2], 15, 718787259);
        b = ii(b, c, d, a, x[i + 9], 21, -343485551);
        a = sa(a, olda);
        b = sa(b, oldb);
        c = sa(c, oldc);
        d = sa(d, oldd);
    }
    return Array(a, b, c, d);
}
```

这道题的坑点就在这里了，里面有个ck()函数，偷偷更改了ic的布尔值。

根据这个ck()函数就能还原出正确的token。

来看这个ck()函数：

```
function ck(s) {
    try {
        ic
    } catch (e) {
        return;
    }
    var a = [118, 104, 102, 120, 117, 108, 119, 124, 48,123,101,120];
    if (s.length == a.length) {
        for (i = 0; i < s.length; i++) {
            if (a[i] - s.charCodeAt(i) != 3)
                return ic = false;
        }
        return ic = true;
    }
    return ic = false;
}
```

为了判断这个函数的输入是什么，我们在binl()函数里的ck(s)前加一行console.log(s)，看看输出，发现是输入的token。

确定输入之后开始分析ck(s):

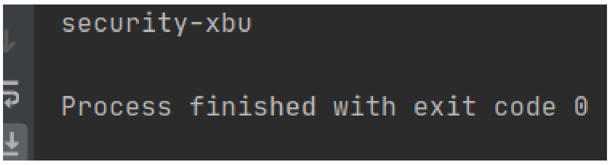
这个函数很简单，读取s的每个字符，假如a[i]-s[i]!=3则返回false

因此，数组a代表的ascii码-3就是s的值。

写一个python3脚本输出正确的token:

```
def main():
    a = [118, 104, 102, 120, 117, 108, 119, 124, 48,123,101,120];
    s = ""
    for i in range(0,len(a)):
        s+=chr(a[i]-3)
    print(s)
if __name__ == '__main__':
    main()
```

成功拿到结果：



```
security-xbu

Process finished with exit code 0
```