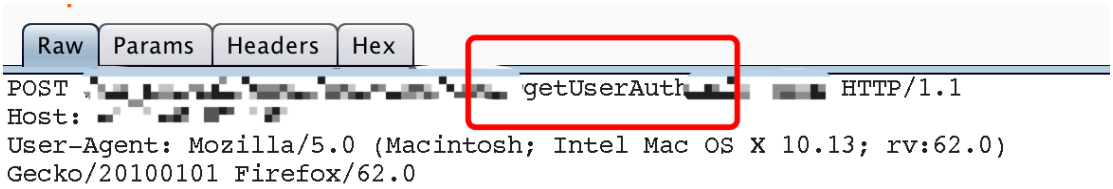
Fuzz 大法之挖掘潜在的逻辑越权

Author：Vulkey_Chen
Blog：gh0st.cn
小密圈：Web 安全 SzS

对一个网站做测试的时候发现了这样一条请求：



这条请求返回的信息是个人信息（用户 ID、手机号、密码）

{"responseData":{"userid":"用户 id","login":"用户名","password":"密码","mobilenum":"手机号","mobileisbound":"01","email":null}}

一开始的想法是变为 GET 请求（可行），然后增加 JSONP 劫持的回调参数。。。（失败）

之前也有人问我怎么去做参数字典的收集：

A. 注意网站参数的命名方式

```
CATALOGID=48&PAGE_INDEX=1&PAGE_COUNT=10&DICTIONARYID=&REQUESTTYPE=0&ITEMTYPE=0&ISSAMETYPE=1
```

大写、英文

B. 返回变参数（注意值都为 B 用户 也就是你需要准备两个用户）

上面所述的返回信息中包含了很多“参数”，可生成如下：

userid=B 用户 id
login=B 用户名
password=B 用户密码
mobilenum=B 用户手机号
email=B 用户邮箱

C. 整合

A 规则+B 收集=C 整合

最后变成如下的字典：

USERID=B 用户 id
LOGIN=B 用户名
PASSWORD=B 用户密码
MOBILENUM=B 用户手机号
EMAIL=B 用户邮箱

然后 Burp Intruder 模块开启，导入字典(这里将参数设在 POST 请求正文)，Start Fuzz：

Payload	Status	Error	Timeout	Length
LOGIN=	200	☐	☐	580
USERID=	500	☐	☐	9951
PASSWORD=	500	☐	☐	10132
MOBILENUM=	500	☐	☐	10132
EMAIL=	500	☐	☐	10132

测试结果发现使用 LOGIN 参数可以成功的从 A 用户的个人信息越权获取到 B 用户的个人信息 ~