ECShop前台无限制SQL注入

1685587406164124 / 2017-12-28 19:37:00 / 浏览数 3857 安全技术 漏洞分析 顶(0) 踩(0)

Author:Joseph

这里我需要说明一下echsop具备全局的一个安全转义
全局文件:/includes/init.php

```
if (!get_magic_quotes_gpc())
{
   if (!empty($_GET))
   {
       $_GET  = addslashes_deep($_GET);
   }
   if (!empty($_POST))
   {
       $_POST = addslashes_deep($_POST);
   }

   $_COOKIE   = addslashes_deep($_COOKIE);
   $_REQUEST  = addslashes_deep($_REQUEST);
}
```

当未开启gpc后进入addslashes_deep函数内容,php在高版本后gpc处于默认关闭状态

```
function addslashes_deep($value)
{
   if (empty($value))
   {
       return $value;
   }
   else
   {
       return is_array($value) ? array_map('addslashes_deep', $value) : addslashes($value);
   }
}
```

addslashes_deep当string时直接进行安全转义，数组时递归转义，这便是ecshop的全局安全。
漏洞文件:user.php

```
$post_data = json_decode(str_replace('\\','',$_POST['JSON']),1);
   //■■■■■■■■■■■■
   if((!isset($_SESSION['v_code']) || $_SESSION['v_code']!='true') && !isset($post_data['no_need_vcode']) ){
       make_json_result('v_code fail');exit;
   }
   $_SESSION['v_code'] = 'false';
   $mobile = $post_data['mobile'] ? $post_data['mobile'] : false;
   $is_send = 'fail';
```

POST获取JSON值并进行json_decode而json_decode具备绕过全局转义以及gpc的特点，也就是会吞并反斜杠。当我们提交'时会被转义成\'而这里再经过一次json_decode

```
if((!isset($_SESSION['v_code']) || $_SESSION['v_code']!='true') && !isset($post_data['no_need_vcode']) ){
       make_json_result('v_code fail');exit;
   }
```

判断SESSION的v_code值是否为true与no_need_vcode是否定义，因为这里用的是&&所以我们满足其中一个条件即可绕过判断，post_data是由我们json_decode后赋值的

```
$_SESSION['v_code'] = 'false';
   $mobile = $post_data['mobile'] ? $post_data['mobile'] : false;
   $is_send = 'fail';
   if($mobile){
       // ■■■■■■■■■■■■■■■■■■
       if (isset($post_data['action']) && $post_data['action'] == 'sms_get_password') {
           $is_reg = $user->check_user($mobile);
           if (!$is_reg) {
               make_json_result($_LANG['phone_number_reg_check_fail']);
               exit();
```

```
        }
    }
```

将post_data中的mobile赋予$mobile并一下判断action是否指定为sms_get_password方法，当指定时则进入check_user方法

```
function check_user($username, $password = null)
   {
       if ($this->charset != 'UTF8')
       {
           $post_username = ecs_iconv('UTF8', $this->charset, $username);
       }
       else
       {
           $post_username = $username;
       }

       if ($password === null)
       {
           $sql = "SELECT " . $this->field_id .
                  " FROM " . $this->table($this->user_table).
                  " WHERE " . $this->field_name . "='" . $post_username . "'";

           return $this->db->getOne($sql);
       }
```

username为我们提交的mobile内容,password未带入所以这里为NULL故进入password===null条件中执行

```
$sql = "SELECT " . $this->field_id .
                  " FROM " . $this->table($this->user_table).
                  " WHERE " . $this->field_name . "='" . $post_username . "'";
```

期间未再经过转义所以导致存在注入但ecshop在新版本后存在安全防护
安全防护文件:/includes/safety.php

```
<?php
$url_arr=array(
'xss'=>"\\=\\+\\/v(?:8|9|\\+|\\/)|\\%0acontent\\-(?:id|location|type|transfer\\-encoding)",
);
$args_arr=array(
'xss'=>"[\\'\\\"\\;\\*\\<\\>].*\\bon[a-zA-Z]{3,15}[\\s\\r\\n\\v\\f]*\\=|\\b(?:expression)\\(|\\<script[\\s\\\\\\\/]|\\b(?:eval|
'sql'=>"(EXTRACTVALUE|EXISTS|UPDATEXML)\\b.+?(select|concat)|[^\\{\\s]{1}(\\s|\\b)+(?:select\\b|update\\b|insert(?:(\\/\\*.*?\
'other'=>"\\.\\.[\\\\\\/].*\\%00([^0-9a-fA-F]|$)|%00[\\'\\\"\\.]");
if( !function_exists('filterData') ){
function filterData(&$data,$type){
   $data and filterArray($data,$type);
}
```

以上为安全防护的正则内容,一看就输入可以绕过的,因为们有json_decode所以可以打乱关键字导致绕过，至于硬刚绕过，菜鸟不会。

上一篇：聊聊CSRF漏洞攻防----久等的暴漫 下一篇：PHP安全新闻早八点-高级持续渗透...

1. 3 条回复

風見鶏 2017-12-29 08:31:25

最新版？

0 回复Ta

 hades 2017-12-29 09:19:19

@風見鶏 测试貌似3.6的版本

0 回复Ta

 刘德华 2017-12-29 13:53:04

在今年年初下的ecshop 3.6 竟然没有这段代码。。

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板