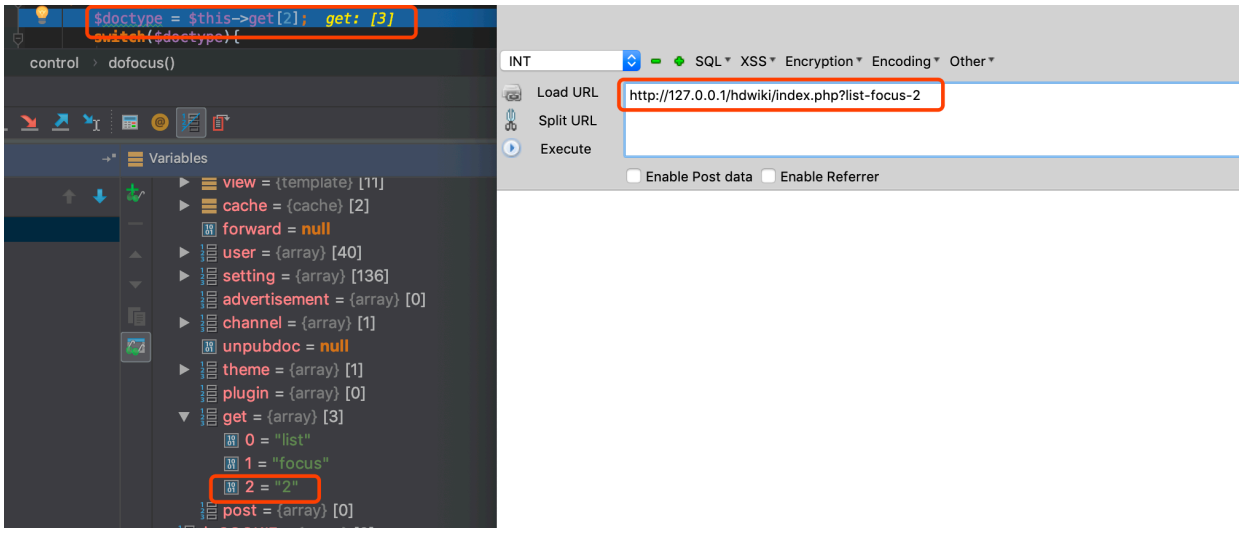
漏洞分析

这里选择 **HDwiki sql注入漏洞** 作为分析例子，由于在 **declare(strict_types=1)** 强类型声明下，通过 **array_walk** 进行调用依旧有弱类型的问题，所以这次也选择一个弱类型导致的漏洞进行分析。

首先漏洞的位置是在 **hdwiki/control/list.php#dofocus** ，我们看看相关代码：

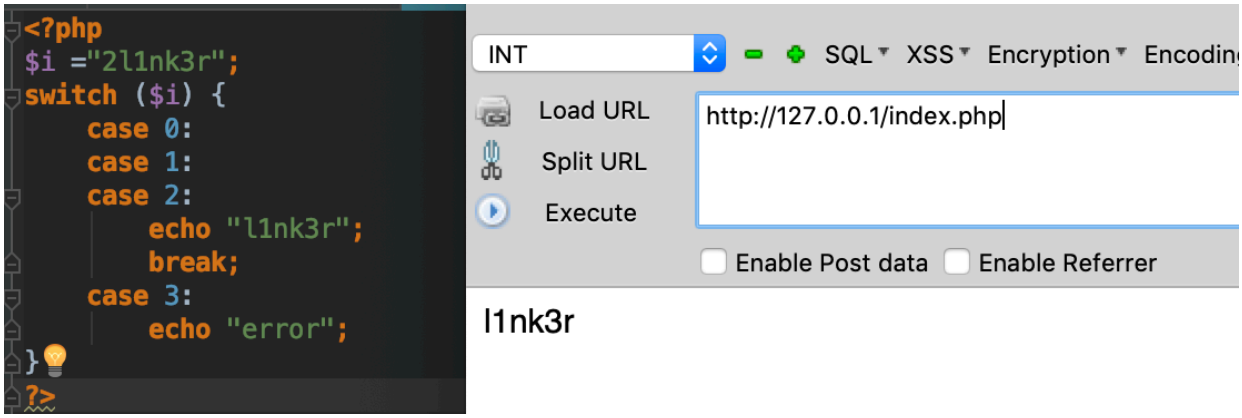
```
1     function dofocus(){
2         $doctype = $this->get[2];
3         switch($doctype){
4             case 2:
5                 $type = 'hot';
6                 $navtitle = $this->view->lang['hotDoc'];
7                 break;
8             case 3:
9                 $type = 'champion';
10                $navtitle = $this->view->lang['wonderDoc'];
11                break;
12            default:
13                $doctype = 1;
14                $navtitle = $this->view->lang['focusDoc'];
15                $type = 'focus';
16        }
17        $url = 'list-focus-'.$doctype;
18        $page = max(1, intval($this->get[3]));
19        $start_limit = ($page - 1) * $this->setting['list_focus'];
20        $total=100;
21        $num=10;
22        $count=$this->db->fetch_total('focus',"type=$doctype");
```

首先代码第二行 **$doctype** 变量从 **get[2]** 中获取数据，我们动态调试一下，可以看到 **get[2]** 中的数据实际上就是 **list-focus-2** 进行分割之后的 **2** ，然后第三行 **switch** 语句根据 **$doctype** 变量的值进行选择。



我们可以看一下这个 **switch** ， 当 **case等于2或者3** 时候， **$doctype** 变量不进行重新赋值，否则 **$doctype** 变量在 **第13行** 重新赋值为 **1** ，然后 **第22行** 调用 **fetch_total** 方法进入数据库查询。这里进入数据库查询的时候 **type= $doctype** 。我们前面说过了 **$doctype** 变量可控，如果构造sql注入payload，会进入到 **default** 中。而在 **default** 中却会对 **$doctype** 重新赋值为1。

这里有个小trick，switch会将case参数转换为int类型，我们看个例子。



我们可以看到 **$i=2l1nk3r** ，由于弱类型的类型转换，最后输出是 **l1nk3r** 。

那么假设我们传入 **list-focus-2 and 1=1** ，那么经过 **get[2]** 处理之后的数据是 **2 and 1=1** ，由于 **switch** 的弱类型问题，进入 **case2** 这个选择，而 **case2** 不会对 **$doctype** 变量进行重新赋值，然后在 **第22行** 调用 **fetch_total** 方法的时候，会把 **2 and 1=1** 带入到数据库进行查询。

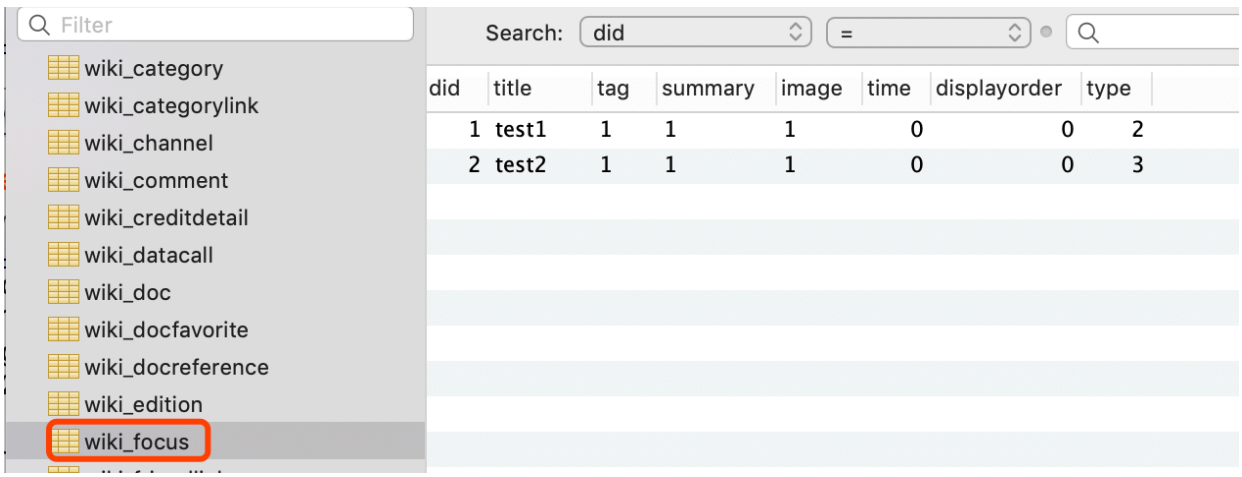
我们开启mysql的log监控，传入 **list-focus-2 and 1=1** 看看日志如下：

```
2018-12-22T09:38:01.354308Z   1423 Query  SELECT COUNT(*) num FROM wiki_focus WHERE type=2 and 1=1
2018-12-22T09:38:01.355574Z   1423 Query  SELECT * FROM wiki_focus WHERE 1=1  AND `type`=2 and 1=1 ORDER BY displayorder ASC,time DESC
LIMIT 0,10
2018-12-22T09:38:01.358460Z   1423 Quit
2018-12-22T09:41:24.850406Z   1424 Connect    root@localhost on  using Socket
2018-12-22T09:41:24.850688Z   1424 Query  SET character_set_connection=utf8, character_set_results=utf8, character_set_client=binary
2018-12-22T09:41:24.850841Z   1424 Query  SET sql_mode=''
```

很明显这里就存在SQL注入了。

漏洞验证

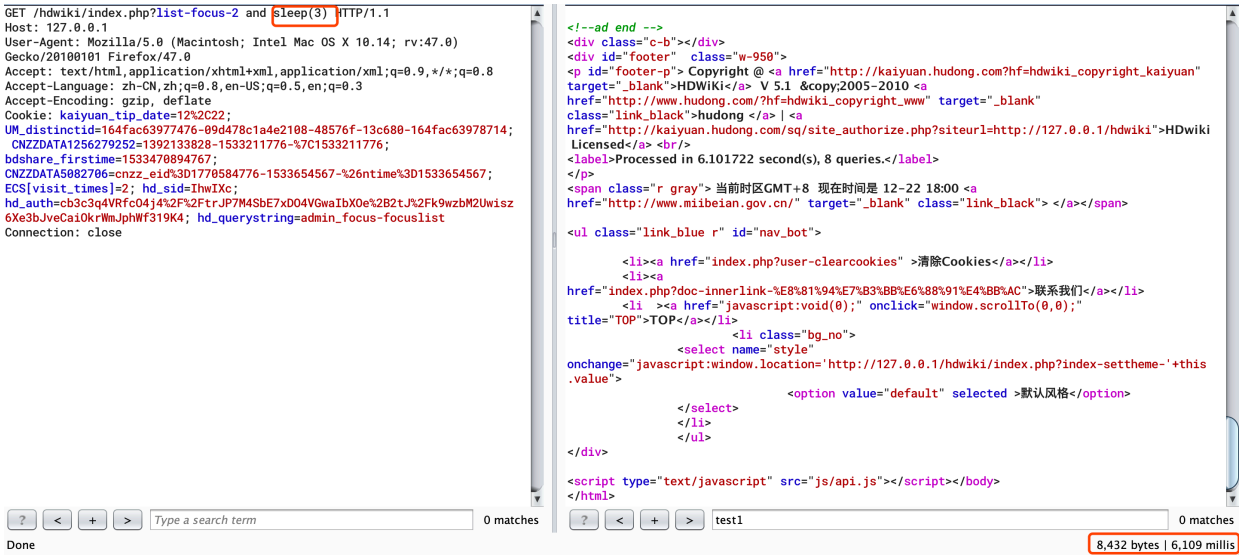
漏洞验证的时候有几个坑点，首先 **wiki_focus** 这个表刚安装的时候没有数据，所以需要手工写入一下数据。



另外在 **hdwiki/model/hdwiki.class.php#checksecurity** 中，有着关于get方式的过滤。

```
    function checksecurity() {
        $check_array = array (
            'get' => array( 'cast', 'exec', 'insert', 'select', 'delete', 'update',
                            'execute', 'from', 'declare', 'varchar', 'script',
                            'iframe', ';', '0x', '<', '>', '\\', '%27', '%22', '(', ')' ),
        );
```

这里我把get改成post，然后输入下图中的payload，最后确认存在注入。



修复建议

由于前端页面中针对 **list-focus** 写死了。

```
 class="l w-230">
<div class="p-b10 m-t10 sidebar">
<ul>
    <li><a href="index.php?list-focus-2"  class="on"><img src="style/default/gl_space/hot.gif" />热门词条</a></li>
    <li><a href="index.php?list-focus-3"  ><img src="style/default/gl_space/champion.gif" />精彩词条</a></li>
    <li><a href="index.php?list-focus-1"  ><img src="style/default/gl_space/bang_tj.gif" />推荐词条</a></li>
    <li><a href="index.php?list-recentchange"  ><img src="style/default/gl_space/bang_zjgx.gif"/>最近更新</a></li>
    <li><a href="index.php?list-weekuserlist"  ><img src="style/default/gl_space/bang_szgxb.gif"/>上周贡献榜</a></li>
    <li><a href="index.php?list-allcredit"  ><img  src="style/default/gl_space/bang_z.gif" />总贡献榜</a></li>
    <li><a href="index.php?list-popularity" ><img src="style/default/gl_space/bang_rq.gif"/>人气指数</a></li>
</ul>
```

所以我们可以在 **case 2** 和 **case 3** 中重新将 **$doctype** 变量进行赋值。

```
switch($doctype){
    case 2:
        $doctype = 2;                       //修复代码
        $type = 'hot';
        $navtitle = $this->view->lang['hotDoc'];
        break;
    case 3:
        $doctype = 3;                       //修复代码
        $type = 'champion';
        $navtitle = $this->view->lang['wonderDoc'];
```

```
        break;
    default:
        $doctype = 1;
        $navtitle = $this->view->lang['focusDoc'];
        $type = 'focus';
}
```

最后成功解决了这个问题。