New issue

Jump to bottom

74cmsSE Storage cross site scripting vulnerability(XSS) #1

Open YLoiK opened this issue on Sep 23 · 0 comments

YLoiK commented on Sep 23 • edited Owner

Vulnerability Name: Storage cross site scripting vulnerability(XSS)

Date of Discovery: 23/9/2022

Product version： 74cmsSEv3.12.0 DownloadLink : https://www.74cms.com/download/detail/89.html

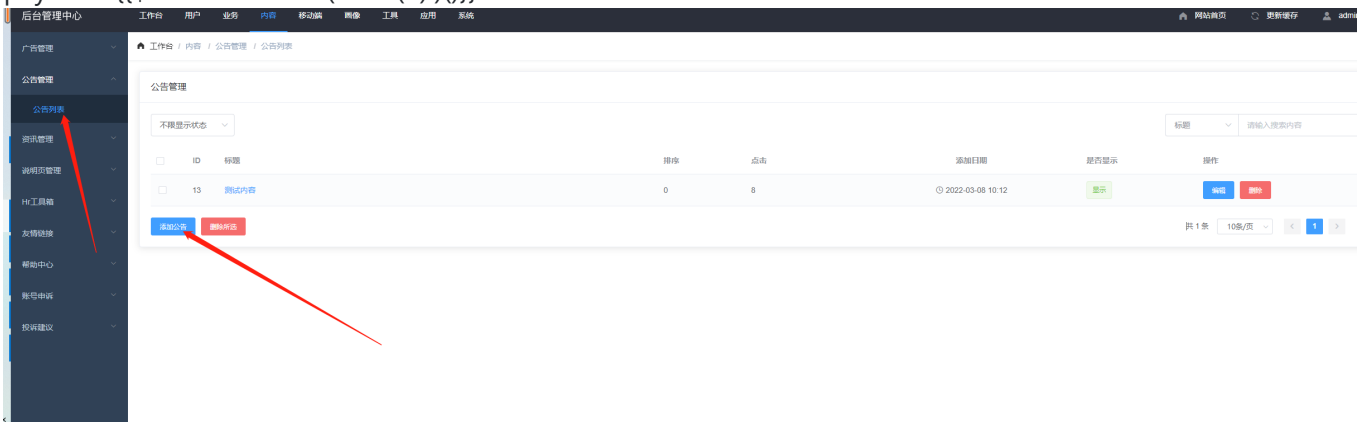
Author: xxhzz

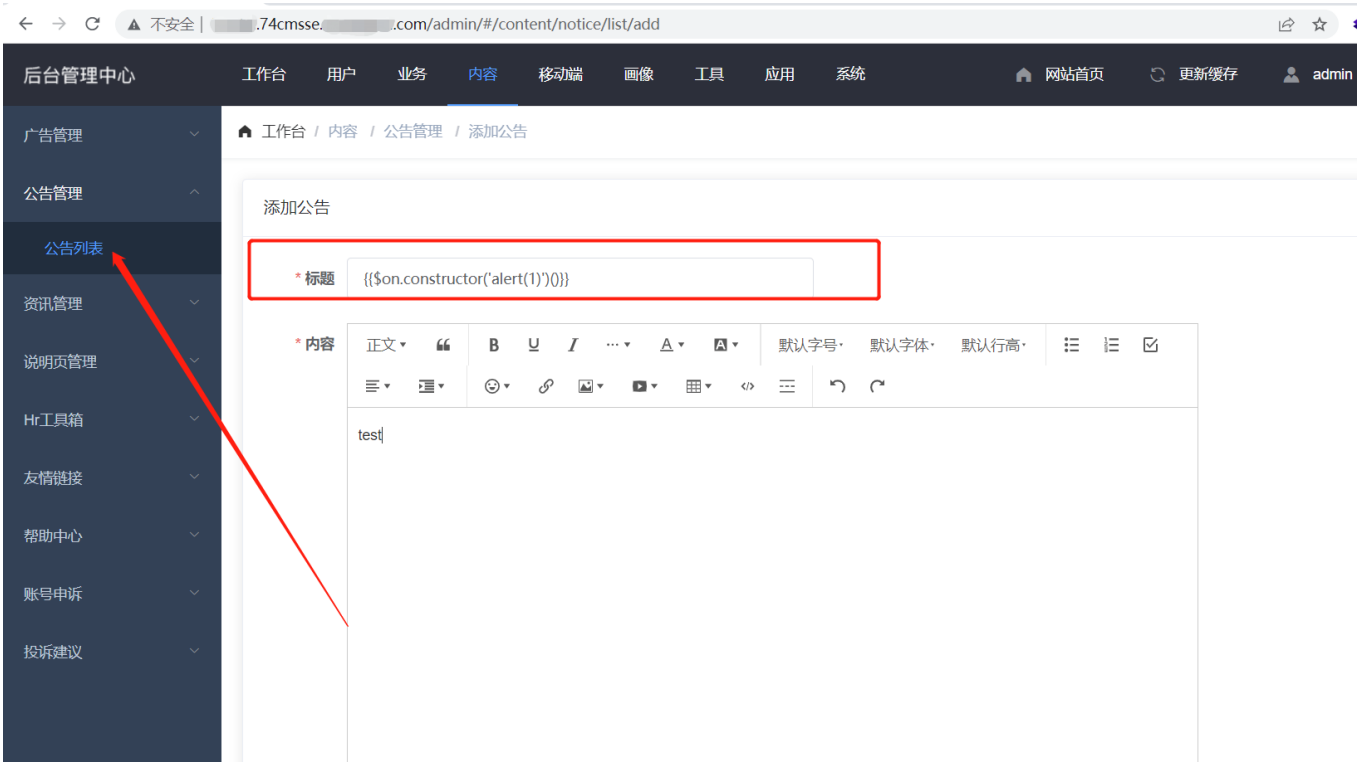
Vulnerability Description:
Add a bulletin to the background of 74cmSSE V3.12.0. Insert the XSS Payload into the header to store and trigger the XSS

Prove:

1.In the background of the website, add a bulletin and insert payload in the header

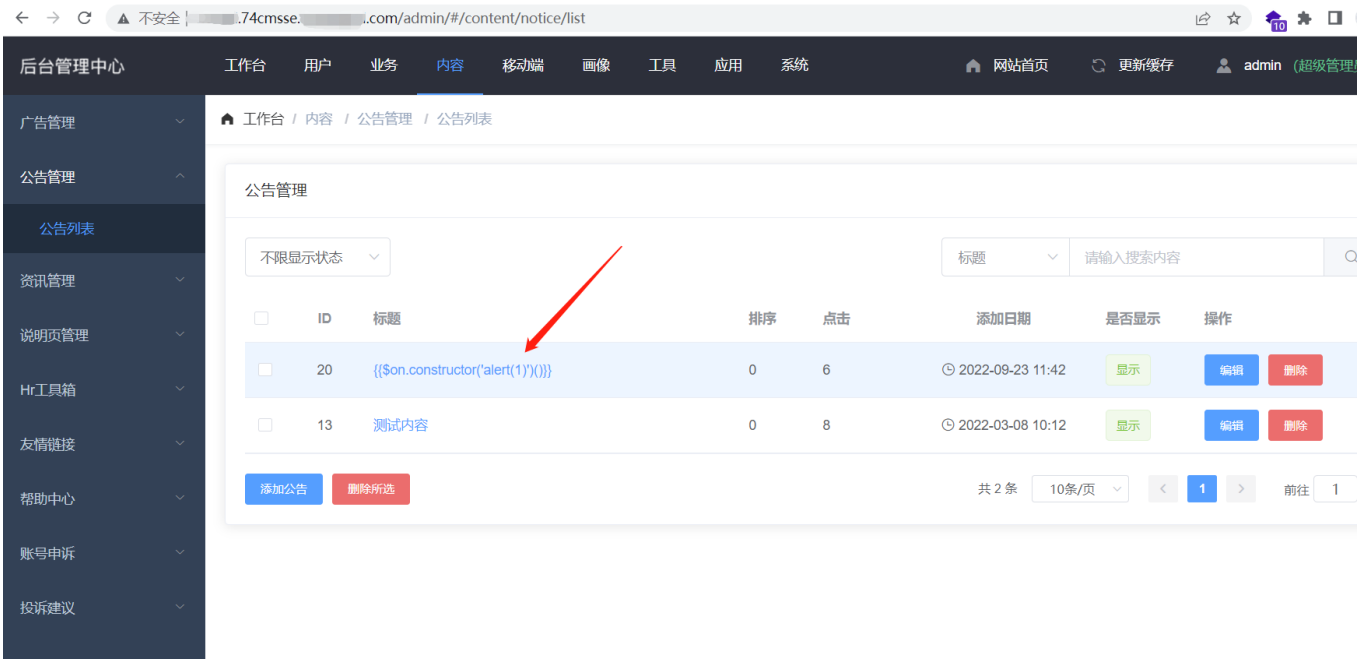
payload: {{$on.constructor('alert(1)')()}}





2.Check the parameter and find title

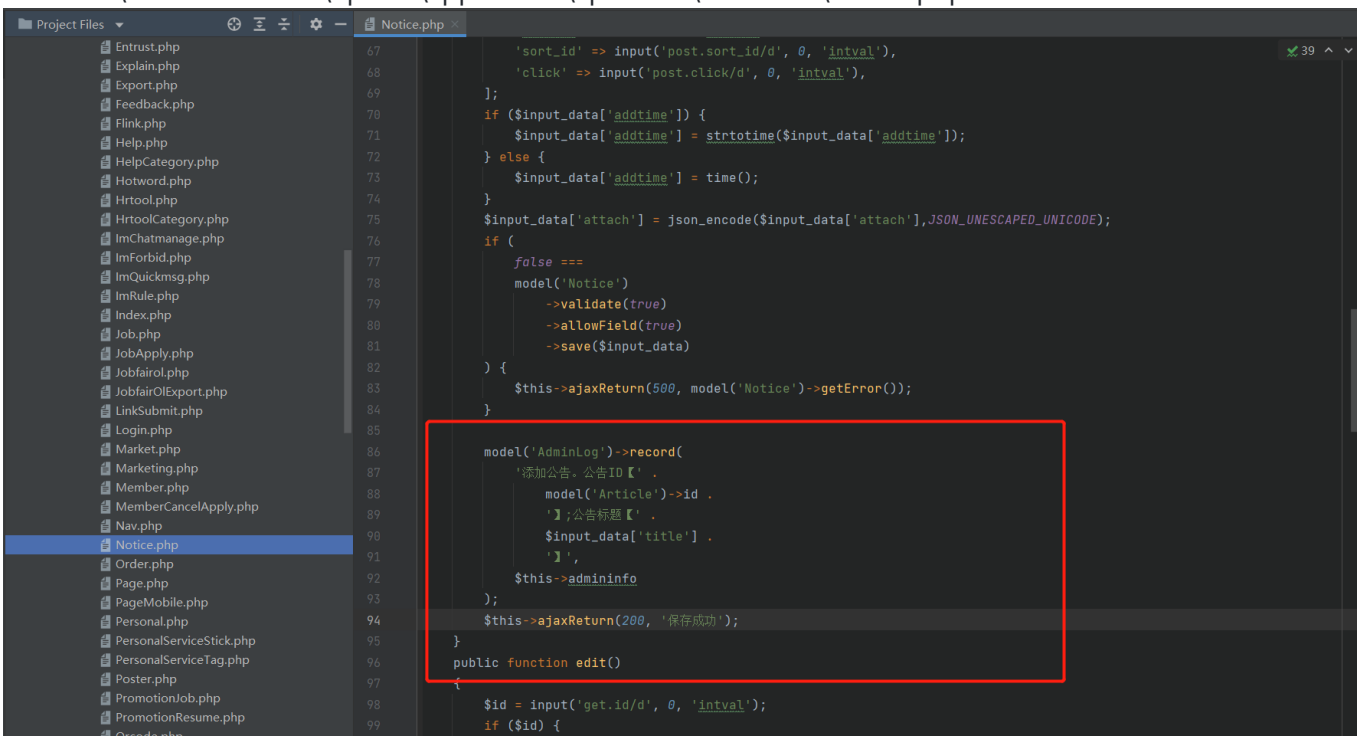
3.Save success



4.Click the title to trigger the XSS successfully

code:

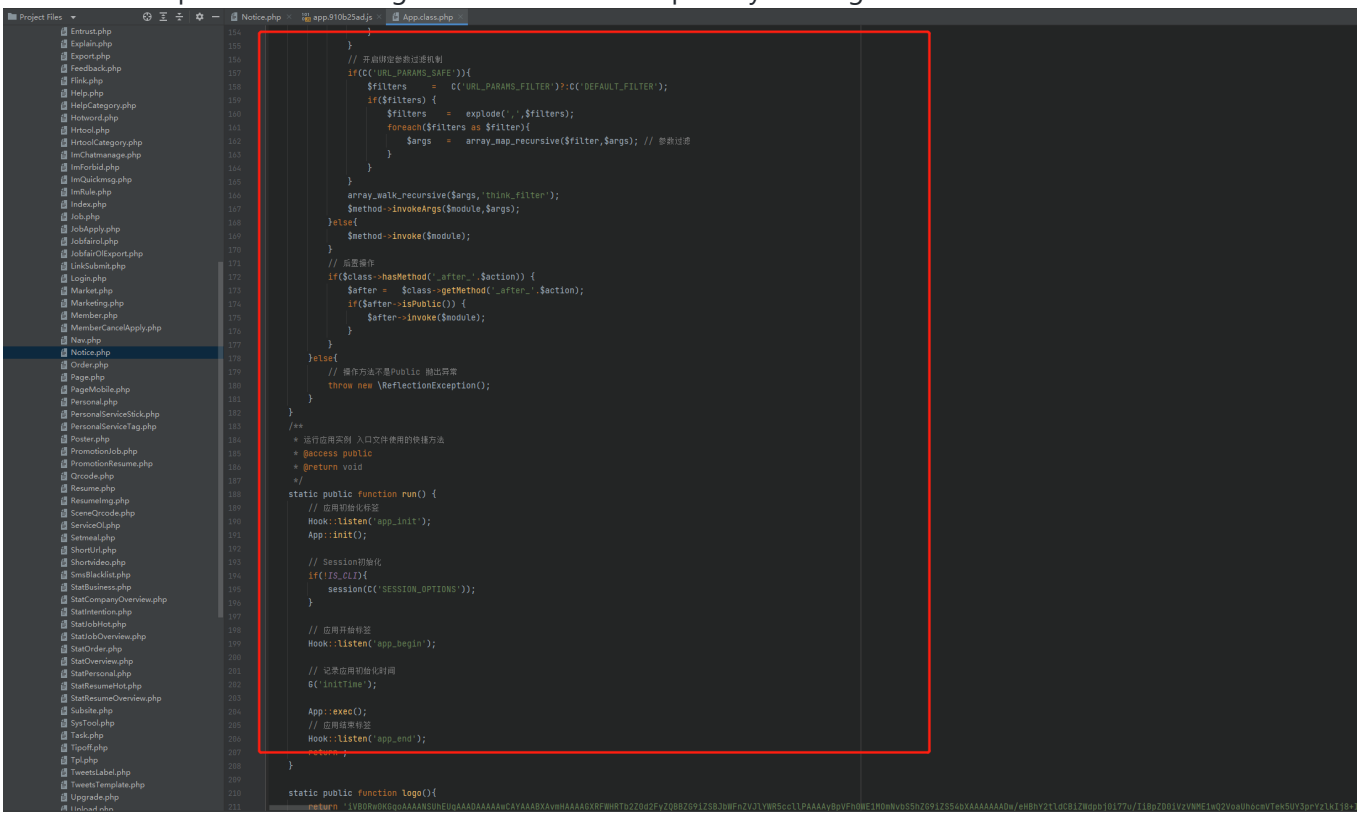
Position: \74cmsSEv3.12.0\upload\application\apiadmin\controller\Notice.php



```
            'sort_id' => input('post.sort_id/d', 0, 'intval'),
            'click' => input('post.click/d', 0, 'intval'),
        ];
        if ($input_data['addtime']) {
            $input_data['addtime'] = strtotime($input_data['addtime']);
        } else {
            $input_data['addtime'] = time();
        }
        $input_data['attach'] = json_encode($input_data['attach'],JSON_UNESCAPED_UNICODE);
        if (
            false ===
            model('Notice')
                ->validate(true)
                ->allowField(true)
                ->save($input_data)
        ) {
            $this->ajaxReturn(500, model('Notice')->getError());
        }

        model('AdminLog')->record(
            '添加公告。公告ID【' .
                model('Article')->id .
                '】;公告标题【' .
                $input_data['title'] .
                '】',
            $this->admininfo
        );
        $this->ajaxReturn(200, '保存成功');
    }
    public function edit()
    {
        $id = input('get.id/d', 0, 'intval');
        if ($id) {
```

Check the xss parameter filtering mechanism and escape only the angle brackets



I am using AngularJS sandbox escapes reflected.Therefore, the storage xss vulnerability was successfully triggered.

Assignees

No one assigned

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant