New issue

Jump to bottom

XSS vulnerability #176

Open **enferas** opened this issue on Jul 18 · 1 comment

enferas commented on Jul 18

Hello,

I would like to report for XSS vulnerability.

In file
https://github.com/cobub/razor/blob/2c991aff4a9c83f99e77a03e26056715706f15c0/web/application/controllers/manage/product.php

```
//line 98
function uploadchannel()
    {
        $platform = $_POST['platform'];
        $channel = $this->channel->getchanbyplat($platform);
        echo json_encode($channel);
    }
```

In file

razor/web/application/models/channelmodel.php
Line 421 in 2c991af

```
421     function getchanbyplat($platform)
```

```
//line 421
function getchanbyplat($platform)
    {
        $userid=$this->common->getUserId();
        $sql="select * from  ".$this->db->dbprefix('channel')."  where active=1 and platform='$platfo
        select * from  ".$this->db->dbprefix('channel')."  where active=1 and platform='$platform' an
        $query = $this->db->query($sql);
        if ($query!=null&&$query->num_rows()>0) {
                return $query->result_array();
        }
```

```
            return null;
        }
```

We can see that the $platform variable is used inside the the sql query without sanitization.
So the attacker can use the UNION command inside the platform to join a harmful input to the results of the query.
For example: $platform = 'something' UNION select '<script>alert(document.cookie)<\script>' AS '.

Thus the XSS will happen at `echo json_encode($channel);`

I recommend to have a check and delete for the character (') in the platform variable.

enferas commented on Sep 13 — Author

CVE-2022-36747 is assigned for this vulnerability.

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant