链滴

# Solo 安全相关

作者：88250

原文链接：https://ld246.com/article/1538896576775

来源网站：链滴

本文是《Solo 从设计到实现》的一个章节，该系列文章将介绍 Solo 这款 Java 博客系统是如何从无有的，希望大家能通过它对 Solo 从设计到实现有个直观地了解、能为想参与贡献的人介绍清楚项目也希望能为给~~重复发明~~重新定义博客系统的人做个参考 
eart

## 权限校验

之前在登录会话的章节我们介绍过如何判断是否登录，解决了身份认证（Authentication），另外还一部分是功能授权（Authorization）。

在 ConsoleAuthAdvice.java 中判断是否登录后还进行了角色判断：

```
if (!Solos.isLoggedIn(context)) {
    final JSONObject exception401 = new JSONObject();
    exception401.put(Keys.MSG, "Unauthorized to request [" + context.requestURI() + "]");
    exception401.put(Keys.STATUS_CODE, HttpServletResponse.SC_UNAUTHORIZED);

    throw new RequestProcessAdviceException(exception401);
}

final JSONObject currentUser = Solos.getCurrentUser(context.getRequest(), context.getRespo
se());
final String userRole = currentUser.optString(User.USER_ROLE);
if (Role.VISITOR_ROLE.equals(userRole)) {
    final JSONObject exception403 = new JSONObject();
    exception403.put(Keys.MSG, "Forbidden to request [" + context.requestURI() + "]");
    exception403.put(Keys.STATUS_CODE, HttpServletResponse.SC_FORBIDDEN);

    throw new RequestProcessAdviceException(exception403);
}
```

通过 HTTP 状态码进行了异常情况的细分：

- 401：未登录
- 403：登录了，但是权限不足

有意思的是 401 常规返回的消息是“Unauthorized”，实际上它指的是 Unauthenticated。这两个态码的差异细节可参考 SO 讨论 403 Forbidden vs 401 Unauthorized HTTP responses。

## XSS

访客可输入的地方在前台只有一个，就是发布评论。所以防 XSS 主要就是防止评论内容带 XSS。后不存在这个问题，因为后台很多地方都是允许设置脚本的，比如文章内容、公告栏等。

对 XSS 处理我们是放在获取评论的时候，在访客发布评论的时候并没有进行过滤。也就是说，数据里保存的评论正文是可能带有攻击脚本的原文。这样设计的好处是如果发生了之前未知的安全过滤问，在新版本升级后不用更新已有数据。

## CSRF

目前暂时没有进行处理，后续可能会改进。