上海久誉软件系统有限公司
Shanghai Jiuyu Software Systems CO.,Ltd.

上海久誉软件系统有限公司

管控台安全修补方案V2

- 现有的问题
- 解决的方案
- 该方案不解决的问题

现有问题

1. 业务级的事务问题（A，B同时查询一条记录后修改）
2. 恶意修改不该修改字段（现在只在页面js层限制，action层后没有控制）firebug插件，chrome 调试工具轻松修改ajax请求
3. 恶意修改不属于自己的记录（查询出来属于自己的 key是 a的记录，恶意将key字段修改成无权访问的b后提交） firebug插件，chrome 调试工具轻松修改ajax请求
4. Sql注入攻击 where参数由于ORM介入基本不存在问题，动态order by注入屡禁不止
5. XSS 注入攻击
6. 重播攻击

问题的产生

1. 意识：培训不足
2. 成本：自动代码工具的引入，如activerecord的概念引入
3. 技术：过度的技术演进

解决方案（有多种）

1.表上增加REC_TOKEN字段，查询时带到页面，提交时带回，update时匹配并重新生成随机数 （解决 问题 1，3，6）,顺便防御了csrf 一部分
2.action对一些关键字段进行范围约束（如 状态字段，这个action只能set成一个固定值）
3.service层 把提交的bean根据业务规则进行过滤（如果可修改的属性少则new一个对象，set一下，如果可修改的属性多，则set不可改的属性（除了key）为null，让ORM拼接时滤除 （解决 问题2）,insert 时候也要如此

解决方案（有多种）

5.Orderby 后字段做白名单或 属性，字段映射map 应对问题5

6.在update后，再根据查询条件查询一次，如果没有查到说明更新有问题3，整体回滚（可选项）

不解决的问题

1。Sql注入
2。Xss

扩展

千万记住：不能相信从页面过来的数据，除非经过本地（action，service）的严格检查或限制

One More Thing

上述解决方案还是有漏洞

Rec_token方案只能针对修改操作的重播，但对应插入操作的重播无法解决

解决方案

管控台对所有请求都进行临时令牌交互工作
类似spring security csrf的工作原理
1.每个post的头部都带上上一次请求返回的临时令牌
2.后台验证本次请求的临时令牌是否在session中存在，存在后减去1，直到0直接remove。如果不存在则直接拒绝。
为啥还要有次数冗余，直接remvoe有什么问题？

谢谢大家！