WEB 平台设计缺陷分析

汪永生

（海军士官学校，安徽 蚌埠 233012）

摘 要：WEB 平台底层设计的缺陷可能会导致很多漏洞，有些漏洞会对平台产生严重的安全隐患，其中包括不安全的设计模式、用户密码没有加密、加密的随机性不强和很多重要信息的泄露等。开发者在平台设计中要增强安全意识，考虑周全，避免存在这些设计缺陷和问题，从而提升平台的安全性，也提高用户信数据息的安全性。

关键词：WEB 平台；设计；缺陷

中图分类号：TP311.52 **文献标识码：**A **文章编号：**1003-9767（2020）04-199-02

Analysis of Design Defects of Web Platform

Wang Yongsheng

(Naval Petty Officer School, Bengbu Anhui 233012, China)

Abstract: The defects of the underlying design of web platform may lead to many vulnerabilities, some of which will cause serious security risks to the platform, including insecure design mode, no encryption of user password, weak randomness of encryption, and leakage of many important information. In order to improve the security of the platform and the user's information and data information, developers should enhance the security awareness, consider carefully and avoid these design defects and problems.

Key words: WEB platform; design; defect

0 引言

WEB 平台的安全性问题，从有互联网就一直伴随着其发展，安全是一个相对的概念，就像矛和盾的关系，绝对安全的系统是不存在的。随着 WEB 攻击渗透技术的发展，原本很安全的 WEB 平台，由于平台设计时分析不够充分完善，从而导致出现很多设计缺陷[1]。本文主要探讨了 WEB 平台设计中容易被忽视的一些设计缺陷，这些缺陷比单纯的由代码编写错误产生的漏洞危害性更大，也更隐蔽。比如，平台的开发人员没有对用户数据进行有效的验证过滤，或者想当然认为用户不会对发送的数据进行篡改，就有可能会导致跨站脚本攻击和出现 sql 注入漏洞。如果攻击者发现了设计缺陷，可能会绕过用户身份验证或授权机制登录系统，从而会导致用户敏感信息泄露。在此背景下，人们要从平台的设计中分析可能存在缺陷，完善平台设计，加强其安全性。

1 不安全的设计模式

当出现平台设计的用户数据过滤程序，实际实现与该过滤程序设计实现的功能不一致时，就经常会发生用户直接绕过数据验证机制。所以在程序功能完成后要充分地进行数据测试，来验证程序功能是否与设计一致。

1.1 歧义、未定义引起的异常行为

在平台设计时，开发人员一般是按照标准的技术需求进行项目开发设计的，但是由于需求制定者的疏忽，总会出现一些模棱两可或者与实际场景不一致的需求。如果部分场景不符合实际情况或者没有考虑更为极端的数据输入情况，就会出现漏洞。例如，非法攻击者经常使用的查询字符串为 http://xxx.xxx.xx/search?a=1&a=3&a=6。在这个查询中，假如平台设计时要求变量 a 的合法值是 1，按照上面这种写法，数据的过滤程序可能判断 a 为合法变量，但是后台代码实际接收并处理的是 a=6 这个部分，那么 a 的值变成了 6，从而就造成了歧义，非法攻击者很有可能就绕过了数据的过滤程序或检测机制。面对这种情况要在设计上把数据的验证和实际使用的数据进行统一判断，从而保证验证后的数据和实际使用数据是一致的。

1.2 授权验证不充分

用户在 WEB 平台上进行任何一个操作都必须进行权限验证，只有这样才能保证用户在平台授权的范围内进行操作，从而防止发生越权事件。例如，用户 A 登录系统需要验证，但是如果在修改用户 A 的数据时不需要进行权限验证，那么其他用户在登录平台后便可以对用户 A 的数据进行修改，这就

作者简介：汪永生（1975—），男，安徽东至人，硕士研究生，高级工程师。研究方向：信息通信。

是一个典型的验证不充分导致的用户越权问题，导致用户信息数据安全风险很大。因此，用户权限验证要尽可能完整、完善。

1.3 数据清理不充分

很多平台在设计用户数据过滤清理时，直接使用了简单的字符串匹配方法，从而可能导致数据清理不充分，达不到实际数据清理的功能，例如要过滤字符串中的 hello 单词，如果数据是 hellhelloo 这个字符串 , 只简单匹配 hello 这个字符串，过滤完的数据变成 hello 这个字符串，完全没有达到数据过滤的实际目的。如果 hello 这个字符串能被非法攻击者利用并攻击，就会导致出现漏洞，面对这种情况，设计平台时需要考虑对数据使用其他字符串进行替换。

2 加密中的实现错误

很多 WEB 开发人员喜欢自己编写加密的算法，由于这些算法没有进行很深层次的研究和安全性分析，可靠性很低，被破解的可能性就非常大，容易产生密码泄露的安全隐患。开发人员应该使用成熟的加密算法，如果使用了不当的密码加密方法，依然会存在安全隐患。

2.1 用户密码未加密

密码加密保存是平台设计最基本的要求之一。明文密码泄露造成的影响很大，因为很多用户在不同的平台和应用中都使用了相同的密码，从而直接导致用户在其他平台也会泄露自己的密码，从而导致更多的敏感用户数据会泄露。因此，在设计平台时必须对用户的密码进行加密保存，这样在保护用户敏感数据安全的同时也保证了平台的正常运行。

2.2 加密方式使用不当

例如，很多用户在使用 MD5 直接对需要加密的信息进行加密，由于各种原因用户 MD5 对应的原始密码比较简单，在出现平台数据库泄露的情况下，非法攻击者可以拿着用户的密码密文进行撞库，很容易就可以知道用户的原始明文密码。如果能在平台设计时对用户的密码加上一个随机的值，并把这个随机值保存在数据库中，然后再使用 MD5 进行加密，产生的密文密码就很难被撞库了，即使撞库成功也很难得到原始明文密码，因为明文密码中包含了随机数。MD5 的加密方式是很安全的，但是由于很多用户的密码设置不可控，因此必须对 MD5 加密方法进行加固。

3 信息泄露的形式

信息泄露涉及很多方面，比如用户数据、服务器信息以及代码执行中的各种错误信息。这些信息中有很多错误信息，在非法攻击者进行攻击时，为其提供了信息参考和攻击方向，更加方便非法攻击者进行攻击。

3.1 不应该给用户显示的服务器信息

很多平台都没有隐藏服务器的信息，而是直接把服务器使用的操作系统、服务器的版本、数据库的版本信息直接显示出来，如果系统补丁更新不及时或者服务器、数据库使用的版本有漏洞，非法攻击者可以轻而易举地攻入服务器。因此在服务器配置时，要尽可能多地隐藏信息，尽量不显示不必要的信息，因为显示的越少越安全。

3.2 错误提示信息没有关闭

系统在开发环境中由于方便开发人员对代码进行调试，几乎所有的开发环境都会开启错误报告。由于开发者在平台设计时没有考虑线上环境的错误提示设置，导致线上运行环境中的代码错误提示没有关闭，非法攻击者会通过错误提示信息了解到代码结构、数据库名称甚至是数据库连接中使用的用户、表名以及字段，从而给平台和用户带来危害。所以在线上运行环境中必须关闭错误提示。

3.3 泄露后台管理地址

开发者很容易忽视 WEB 平台设计时后台管理地址的安全问题，很少会对后台管理地址进行隐藏或者伪装，容易造成后台管理员账号、密码泄露。非法攻击者可以利用账号、密码直接登录。甚至有些平台为了方便管理员登录，直接把后台登录的登录地址放置在首页中，是非常危险的一件事情。面对这个问题应该把后台登录地址修改为不常见的地址，现在基本通用的后台登录地址就是 http://xx.xx.xx/admin，可以修改为类似 http://xx.xx.xx/afas222 这样的地址，这种没有规律可言的后台登录地址，可以大大降低非法攻击者发现后台管理地址并进行攻击的概率。

3.4 泄露有价值的网站数据

网站中有价值的信息中包含各种文档，文档内容有可能包含了单位员工以及网站管理人员的信息，比如姓名、邮箱、电话、身份等重要信息。这些文件本来是用户需要授权后才可以访问的，但是系统对访问这类文件没有进行权限控制，从而导致搜索引擎随意抓取。泄露的文件信息可能会被非法攻击者利用，通过社工技术对这些人员进行渗透，从而获得 WEB 平台的登录账号以及密码。因此，WEB 平台设计时一定要把访问重要文件加入控制权限中，使用重要文件时必须对用户进行权限验证。

4 结 语

WEB 平台的设计缺陷是最容易被平台开发者忽视的，因为大部分平台开发者在设计时主要为了满足用户所定制的业务需求，并没有把数据安全放在平台设计的第一位。因此，平台开发者和用户要增强安全意识，在设计前期要考虑周全，避免在设计中存在本文所提到的各项设计缺陷和问题，才能保证平台和用户数据的安全性。

参考文献

[1][美]Patrick Engebrestson. 渗透测试实践指南 [M]. 缪纶 , 只莹莹 , 蔡金栋 , 译 . 北京 : 机械工业出版社 ,2012:346.