Atitit 信息系统安全法

目录

[1. 常见的安全保护目标 1](#_Toc27065)

[2. 三大危害 2](#_Toc23972)

[2.1. 金额篡改 2](#_Toc11978)

[2.2. 信息泄露 2](#_Toc14936)

[2.3. 程序崩溃风险 2](#_Toc18847)

[3. WEB安全风险行为 2](#_Toc14627)

[3.1. Injection 2](#_Toc14594)

[3.2. Broker Authentication损坏的身份验证 2](#_Toc6865)

[3.3. Sensitive Data Exposure 3](#_Toc11020)

[3.4. XML External Entities (XXE) 3](#_Toc2968)

[3.5. Broken Access Control损坏的访问控制 3](#_Toc24497)

[3.6. Security Misconfiguration安全配置错误 3](#_Toc17886)

[3.7. Cross-Site Script (XSS) 3](#_Toc24796)

[3.8. Insecure Deserialization不安全的反序列化 3](#_Toc2176)

[3.9. Using Components with Known Vulnerabilities使用已知漏洞的组件 3](#_Toc17454)

[3.10. Insufficient Logging and Monitoring日志记录和监控不足 4](#_Toc25602)

[3.11. TOP4-不安全的对象直接引用 4](#_Toc3032)

[3.12. TOP5-跨站请求伪造（CSRF） 5](#_Toc16016)

[3.13. TOP7-限制URL访问失败（缺少功能级访问控制） 5](#_Toc10028)

[3.14. TOP8-未验证的重定向和转发 5](#_Toc11450)

[4. 应用安全 6](#_Toc22747)

[4.1. 账务系统安全法 6](#_Toc27978)

[5. 数据库安全风险行为25个 6](#_Toc14327)

[6. Rf 7](#_Toc2224)

# 常见的安全保护目标

网络安全，数据库安全 WEB安全 应用安全

# 三大危害

## 金额篡改

## 信息泄露

## 程序崩溃风险

# WEB安全风险行为

## Injection

## Broker Authentication损坏的身份验证

与认证和会话管理相关的应用程序功能往往得不到正确实施，导致了攻击者可以破坏密码，密钥，会话令牌或实施漏洞冒充其他用户身份。

危害如下：

这些漏洞可能导致部分甚至全部账户遭受攻击，一旦攻击成功，攻击者就能执行合法的任何操作

如何防范：

1.使用内置的会话管理功能

2.通过认证的问候

3.使用单一的入口点

4.确保在一开始登录SSL保护的网页

如何防范：

1.使用内置的会话管理功能

2.通过认证的问候

3.使用单一的入口点

4.确保在一开始登录SSL保护的网页

## Sensitive Data Exposure

由于代码编写不严谨或应用固有的功能，造成网站服务器信息被非法获取，属于一种低危漏洞。

## XML External Entities (XXE)

## Broken Access Control损坏的访问控制

## Security Misconfiguration安全配置错误

安全配置错误可以发生在一个应用程序堆栈的任何层面，包括平台，web服务器，应用服务器，数据库，架构和自定义的代码。攻击者通过访问默认账户，未使用的网页，未安装的补丁的漏洞，未被保护的文件和目录等，以获得对系统为授权的访问。

危害如下；

系统可能在未知的情况下被完全攻破，用户数据可能随着时间被全部盗走或篡改。甚至导致整个系统被完全破坏

如何防范：

1.自动化安装部署

2.及时了解并部署每个环节的软件更新和补丁信息

3.实施漏洞扫描和安全审计

## Cross-Site Script (XSS)

种类：

存储型，反射型，DOM型

如何防范：

1.验证输入

2.编码输出（用来确保输入的字符被视为数据，而不是作为html被浏览器所解析）

## Insecure Deserialization不安全的反序列化

## Using Components with Known Vulnerabilities使用已知漏洞的组件

 组件（例如库，框架和其他软件模块）以与应用程序相同的权限运行。如果利用易受攻击的组件，这种攻击可能会导致严重的数据丢失或服务器接管。使用具有已知漏洞的组件的应用程序和API可能会破坏应用程序防御并实现各种攻击和破坏。

如何防范：

1.识别正在使用的组件和版本，包括所有的依赖

2.更新组件或引用的库文件到最新

3.建立安全策略来管理组件的使用

## Insufficient Logging and Monitoring日志记录和监控不足

2.2经纪人身份验证

2.3。敏感数据公开

2.4。XML外部实体（XXE）

2.5。损坏的访问控制

2.6。安全配置错误

2.7。跨站脚本（XSS）

2.8。不安全的反序列化

2.9。使用已知漏洞的组件

2.10。日志记录和监控不足

## TOP4-不安全的对象直接引用

一个已经授权的用户通过更改访问时的一个参数，从而访问到原本其并没有得到授权的对象。

危害如下：

这种漏洞可以损坏参数所引用的所有数据

如何防范：

1.使用基于用户或会话的间接对象访问，这样可防止攻击者直接攻击为授权资源

2.访问检查：对任何来自不受信源所使用的所有对象进行访问控制检查

3.避免在url或网页中直接引用内部文件名或数据库关键字

4.验证用户输入和url请求，拒绝包含./ ../的请求

## TOP5-跨站请求伪造（CSRF）

跨站请求伪造，攻击者通过用户的浏览器注入额外的网络请求，破坏网络会话的完整性。浏览器的安全策略是允许当前界面发送到任何地址，因此也意味着如果用户浏览了自己无法控制的资源时，攻击者可以通过控制页面的内容来控制浏览器，发送其精心构造的请求，这些精心构造的请求很难和合法的请求区分开。

危害如下：

攻击者可以让受害者用户修改任何允许修改的数据，执行任何用户允许的操作，例如修改密码，登录注销等。

如何防范：

1.给每个HTTP请求添加一个不可预测的令牌，并保证该令牌对每个用户会话来说是唯一的。

最好的办法就是将独有的令牌包含在隐藏字段中，通过HTTP请求发送，避免在URL中暴露出来。

2.要求用户重新认证或判断他们是一个真实的用户

## TOP7-限制URL访问失败（缺少功能级访问控制）

这个漏洞也是与认证相关的，这种漏洞具体是指在系统已经对url的访问做了限制的情况下，但这种限制并没有生效。常见的例子是系统没有对用户进行角色的检查，以及用户通过修改URL的action并指向未被授权页面就能访问该页面同样是个漏洞

危害如下：

攻击者很容易就能把网址改成享有特权的网页，这样就可以使用匿名或普通用户访问未授保护的私人页面，从而提升未授权功能和相关数据信息

如何防范：

1.检查管理权限的过程并确保能够容易进行升级和审计

2.默认缺省情况下，应该拒绝所有访问的执行权限。对于每个功能得访问，需要明确的角色授权

3.检查每个功能分配的权限合理有效

## TOP8-未验证的重定向和转发

在Web应用中重定向是极为普通的，并且通常重定向所引发的目的是带有用户输入参数的目的url，而如果这些重定向未被验证，那么攻击者就可以引导用户访问他们想要用户访问的站点。

同样，转发也是极为普遍的，本质上转发是在同一个应用中对一个新页面发送请求，并且有时是用参数来定义目标页面的。同样，如果参数未被验证，那么攻击者就可以利用其来绕过认证或是授权检查

危害如下：

攻击者通过重定向可以试图安装恶意软件或诱使受害人泄露密码等铭感信息，通过转发可以绕过访问限制

如何防范：

1.避免使用重定向和转发

2.如果使用了，不要在确定目标时涉及到用户参数

3.如果无法避免使用用户参数，则应确保目标参数值对于当前用户是有效的并已授权

如果是需要登录的，可以从session当中获取登录信息，然后判断

# 应用安全

加密法

签名法

动态秘钥法 时间+salt

安全类库法提供cookie api

安全审查法

权限安全法

防止注入法

事务安全法

审核法

不法行为预警法

## 账务系统安全法

必须每笔有记录

，余额签名放篡改法

每次业务操作对比最后余额法

# 数据库安全风险行为25个

数据库安全风险行为

目录

1. 数据库安全风险行为 2

1.1. Sql注入SQLi）攻击行为 2

1.2. . 过多的、不适当的和未使用的特权 2

1.3. 2. 权限滥用（数据库系统管理员和开发人员）行为 2

1.4. 3.Web 应用程序风险行为 3

1.5. 4. 审计线索不足行为 3

1.6. 5. 不安全的存储介质 3

1.7. 侵犯隐私行为 3

1）身份认证安全问题 3

· 无密码的身份认证安全问题 3

· 有密码的身份认证安全问题 3

2）权限控制 3

3）通讯加密 4

4）审计 4

5）数据加密 4

6）NoSQL注入 4

7）自身安全漏洞 4

1.8. 糟糕的加密与数据泄露密不可分 2 5

1.9. 数据库被盗 2 5

1.10. 功能中的缺陷成为了数据库安全问题 2 5

1.11. 弱而复杂的数据库基础架构 2 5

1.12. 无限的管理访问 = 糟糕的数据保护 3 5

1.13. 密钥管理不足 3 5

1.14. 数据库中的不规范 3 4 5

1.15. 按照发生的可能性排序 1. 人为误操作，删除数据，更改数据 ==>可通过定期备份解决 5

1.16. 2. 数据库本身故障。 数据库当机。===> 可通过高可用性解决，RAC, HA, Dataguard。 5

1.17. 3. 服务器硬件故障 ==> 同上 5

1.18. 4 . 黑客攻击 ==> 可通过定期安装安全补丁，使用符合安全策略的密码等手段解决 5

1.19. 5. 灾难: 火山，地震，海啸。机房起火。失窃。 ===》 可以通过异地备份，或者祈祷来解 5

2. Ref 6

# Rf

Atitit 数据安全法

OWASP TOP 10（OWASP十大应用安全风险）.html

OWASP TOP 10（OWASP十大应用安全风险）.html

详解5大数据库安全威胁-InfoQ.html