将Oracle10g内置的安全特性用于PHP PDF转换可能丢失图片或格式，建议阅读原文
https://www.100test.com/kao_ti2020/142/2021_2022__E5_B0_86Oracle1_c102_142835.htm 当今大多数 Web 应用程序都需要至少采用某种基本的安全策略。例如，提供用口令保护的内容的网站、仅具有管理员后端的网站、网志和个人杂志、电子商务网站、企业内联网，等等。 构建这些类型的 Web 应用程序最常用的设计方法是将安全策略整合到 Web 应用程序的业务逻辑中，即由应用程序决定某个用户是否有权访问数据库中的某个数据。在这种情形下，数据库的角色仅为存储数据和依请求提供数据。换句话说，如果 Web 应用程序命令数据库提供特定信息，则数据库会直接执行该命令而不检查用户的权限。 在该文中，您将学习如何利用 Oracle 内置的安全特性在数据库级执行应用程序安全规则，以提高应用程序的整体安全性。作为附带的好处，直接在数据库中实现数据访问安全不但有助于提高应用程的安全性，而且有助于降低复杂性。 对数据库端安全性的需求 从 Web 应用程序控制数据访问会怎么样？大多数情况下没有问题；这是个不错的解决方案，尤其是在涉及的数据为非任务关键或绝密的时候。许多书和在线资源中都用到了该方法。实际上，有本很受欢迎的 PHP/MySQL 书明确反对每个应用程序创建一个以上的数据库用户帐户，这是因为“ 额外的用户或复杂的权限会因某个操作在继续前要检查更多的信息而降低 MySQL 的执行速度 ”。确实如此；但是，在放弃将安全性整合到数据库逻辑中的想法前可能要考虑几件事情。我们来看以下示例。 假设创建一个内容管理系统 (CMS)。其中使用数据库来存储网站上发布

的内容。大部分数据是公开的，允许匿名 Web 用户读取；但只允许编辑更改数据。使用单一数据库帐户访问和修改数据库中的记录，并通过用口令保护仅管理员可以访问的页面的访问权限用 PHP 代码控制安全性。 如果 Web 应用程序的公共端遭受了一个诸如公共搜索表单（即编码不够严密的表单）上的 SQL 注入的攻击，则该入侵者可能能够对该公共帐户可以访问的数据库对象执行任意 SQL 语句。当然，就这里的情形而言，执行 SELECT 语句不会造成什么大问题，这是因为数据本来就是公共的。但由于公共权限和管理权限使用同一数据库帐户，因此入侵者还能执行 UPDATE 和 DELETE 语句，甚至是从数据库中删除表。 怎么才能防止该情况的发生呢？最简单的方法就是彻底限制公共数据库帐户修改数据的权限。我们来看看 Oracle 是如何解决这个问题的。 Oracle 安全性基本概述 Oracle 数据库为 Web 开发人员提供了控制数据访问的许多方法，从管理对特定数据库对象（如表、视图和过程）的访问到控制个别行或列的数据的访问。很显然，对 Oracle 每个安全特性或可用选项的讨论超出了本文的范围。在这里，我们将不涉及过多细节，而仅介绍 Oracle 数据访问安全性的最基本方面： 验证和用户帐户 权限 角色 验证和用户帐户。 与其他数据库一样，请求访问 Oracle 的每个用户（数据库帐户）必须通过验证。验证工作可以由数据库、操作系统或网络服务来做。除基本的验证（口令验证）外，Oracle 还支持强验证机制，如Kerberos、CyberSafe、RADIUS，等等。 角色。 Oracle 角色是一个权限的有名集。尽管可以直接授予用户帐户权限，但使用角色可以极大简化用户管理，尤其是需要管理大量用户时。创建易管理的小角色，然后根据用

户的安全级别授予用户一个或多个角色，这样做的效率非常高。更不用说修改权限变得如何简单了 只需修改角色关联的角色即可，无需修改每个用户帐户。 为了简化新用户创建初期的工作，Oracle 自带了三个预定义的角色： CONNECT 角色 该角色使用户可以连接数据库以及执行基本的操作，如创建自己的表。默认情况下，该角色不能访问其他用户的表。 RESOURCE 角色 RESOURCE 角色与 CONNECT 角色相似，但它允许用户拥有较多的系统权限，如创建触发器或存储过程。 DBA 角色 允许用户拥有所有系统权限。 使用中的授权和权限 在本部分中，我们将讨论如何使用 Oracle 的授权和权限来提高本文开头部分讨论的那个简单 CMS 示例的安全性。假定，提供给应用程序用户的内容存储在 WEB_CONTENT 表中。 首先，创建该表。启动 Oracle 数据库特别版，以系统管理员身份登录。如果还没有释放示例 HR 用户，请将其释放。按照特别版安装附带的入门指南中的指示操作。请注意，默认情况下，HR 用户被赋予 RESOURCE 角色。在这里，赋予该用户 DBA 角色，这样就可以使用该帐户管理 CMS 应用程序的数据库方面了。当然，不会使用 HR 用户帐户进行在线访问，只用它管理数据库。 现在，可以使用对象浏览器或通过执行 SQL Commands 窗口创建新表。下面是创建该表的代码： CREATE TABLE WEB_CONTENT (page_id NUMBER PRIMARY KEY,page_content VARCHAR2(255)). 由于该表是使用 HR 用户帐户创建的，因此该表归 HR 帐户所有并位于 HR 模式中，并且在明确授予其他用户访问该表的权限前，其他用户无法访问该表。如果不信，可以创建一个新用户，用该用户访问 WEB_CONTENT 表试试。 现在，创

建两个新用户，CMS_USER 和 CMS_EDITOR。最终，将授予 CMS_USER 对 WEB_CONTENT 表的只读权限，并将该用户用作为匿名 Web 用户提供内容的数据库帐户。CMS_EDITOR 帐户将在该表上拥有更多权限，将被用作 CMS 编辑的帐户（该帐户需要更改和维护该表中的数据）。 可以使用 XE 的图形界面或通过执行以下命令创建新用户： CREATE USER cms_user IDENTIFIED BY cms_user.CREATE USER cms_editor IDENTIFIED BY cms_editor. （出于简化的目的，此处的口令与用户名对应。） 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com