**西安财经大学信息学院**

**姓名 王婧**

**学号 1831050171**

**班级 软件工程1802**

《**数据库系统** 》  **实验报告**

**实验名称** 数据库安全性的管理 **实验地点** 实验楼316 **实验日期** 2019-11-13

|  |
| --- |
| **一、实验目的及要求**  **目的：**  1． 掌握 SQL Server 身份验证模式。  2． 掌握创建登录账户、数据库用户的方法。  3． 掌握使用角色实现数据库安全性的方法。  4． 掌握权限的分配。  **要求：**  1． 写出与上述任务相对应的 SQL 语句。  2． 思考数据库服务器上的角色有哪些？权限类型有哪些？  3． 并记录在实验过程中遇到的问题、解决办法及心得体会  **二、实验环境**  **SQL server 2012**  **三、实验内容**  **1．设置身份验证模式**  （1）依次单击开始－>所有程序－>SQL Server 2008－>SQL Server Management Studio 启动 SQL Server 2008 数据库管理系统；  （2）连接成功后，右键你的实例，选择"属性" ；  （3）在"属性"窗口中，转到"Security"(安全性)项，在"服务器身份验证"中设置为"SQL Server 和 Windows 身份验证模式"，确定，根据提示，重新启动 sql 服务；  （4）然后执行下面的语句启用 sa 用户,同时清除 sa 的密码(能成功登陆后再根据你的需要设置)  EXEC sp\_password null,null,'sa'  ALTER LOGIN sa ENABLE  （5）语句执行完成后, 再用 sa 连接实例,应该就没有问题了。  **2．设置登录账户**  **创建使用 Windows 身份验证（SSMS）的 SQL Server 登录名**  （1）在 SSMS 中，打开对象资源管理器并展开要在其中创建新登录名的服务器实例的文件夹。  （2）右键单击“安全性”文件夹，指向“新建”，然后单击“登录名”。  （3）在“常规”页上的“登录名”框中输入一个 Windows 用户名。  （4）选择“Windows 身份验证”。  （5）单击“确定”。   **创建使用 SQL Server 身份验证（SSMS）的 SQL Server 登录名**  （1）在 SQL Server Management Studio 中，打开对象资源管理器并展开要在其中创建新登录名的服务器实例的文件夹。  （2）右键单击“安全性”文件夹，指向“新建”，然后单击“登录名”。  （3）在“常规”页上的“登录名”框中输入一个新登录名的名称。  （4）选择“SQL Server 身份验证”。  （5）输入登录名的密码。  （6）选择应当应用于新登录名的密码策略选项。通常，强制密码策略是更安全的选择。   **通过 Transact-SQL 创建使用 Windows 身份验证的 SQL Server 登录名**  （1）在查询编辑器中，输入以下 Transact-SQL 命令：  CREATE LOGIN <name of Windows User>  FROM WINDOWS;  GO  **通过 Transact-SQL 创建使用 SQL Server 身份验证的 SQL Server 登录名**  （1）在查询编辑器中，输入以下 Transact-SQL 命令：  CREATE LOGIN <login name>  WITH PASSWORD = '<password>';  GO   1. **设置数据库用户**   **创建数据库用户**  （1）在 SQL Server Management Studio 中，打开对象资源管理器并展开要在其中创建新登录名的服务器实例的文件夹。  （2）依次展开数据库，某个具体数据库，“用户”  （3）右键单击用户，选择“新建用户”  （4）输入用户名，选择登录名，确定  **通过 Transact-SQL 创建数据库用户（create user）：**  （1） 输入 SQL 语句：  create user dba  for login dba  with default\_schema=dbo  （2）并指定数据库用户“dba” 的默认 schema 是“dbo”。这意味着用户“dba” 在执行“select \* from t”，实际上执行的是 “select \* from dbo.t”。  **4．设置数据库角色 **  **新建角色**  （1）在 SQL Server Management Studio 中，打开对象资源管理器并展开要在其中创建新登录名的服务器实例的文件夹。  （2）依次展开数据库，某个具体数据库，“角色”  （3）右击选中新建角色   **通过 Transact-SQL 新建角色：**   1. CREATE ROLE role\_name    **在对象资源管理器中设置数据库角色**   1. 在 SQL Server Management Studio 中，打开对象资源管理器并展开要在其中创建新登录名的服务器实例的文件夹。 2. 依次展开数据库，某个具体数据库，“用户”选中某用户，右击“属性”，在展 开的数据库用户属性对话框中进行设置 ****   **通过 Transact-SQL 设置数据库角色**  （1）通过加入数据库角色，赋予数据库用户“dba”权限： exec sp\_addrolemember 'db\_owner', 'dba'  （2）此时，dba 就可以全权管理数据库 mydb 中的对象了。  （3）如果想让 SQL Server 登陆帐户“dba”访问多个数据库，比如 mydb2。可以让 sa 执行下面的语句：  use mydb2  go  create user dba  for login dba  with default\_schema=dbo  go exec sp\_addrolemember 'db\_owner', 'dba' go  （4）此时，dba 就可以有两个数据库 mydb, mydb2 的管理权限了！  **5．设置管理对象的权限 **  在对象资源管理器中授权：   1. 选中需要授权的登录名、用户名或角色名 2. 右键单击选择“属性”，在弹出的该对象的属性窗口中选择“安全对象”，进行 权限分配    通过 Transact-SQL 分配权限：   1. 通过 GRANT 和 REVOKE 语句   **实验任务**   1. 创建角色 ProgramerRole，拥有创建表，存储过程，视图权限，拥有对 Salary 表的查询、修改、插入权限   任务1  任务1.1   1. 创建一个登录账号 Testlogin   任务2   1. 创建对应于这个登录账号的数据库用户 TestUser   任务3   1. 将用户 TestUser 添加到 TestRole 角色中   任务4  **四、实验总结**  数据库服务器上还有 Windows用户组或用户账户、SQL server登录、其他角色。在实验的过程中遇到最大的问题就是新建登录账号和角色不熟练。还的加强练习。 |