构造SQLServer安全门 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/460/2021_2022__E6_9E_84_E9_80_A0SQLS_c98_460948.htm 在改进SQL Server 7.0系列所实现的安全机制的过程中，Microsoft建立了一种既灵活又强大的安全管理机制，它能够对用户访问SQL Server服务器系统和数据库的安全进行全面地管理。按照本文介绍的步骤，你可以为SQL Server 7.0(或2000)构造出一个灵活的、可管理的安全之门，而且它的安全性经得起考验。 一、验证方法选择 本文对验证(authentication)和授权(authorization)这两个概念作不同的解释。验证是指检验用户的身份标识.授权是指允许用户做些什么。在本文的讨论中，验证过程在用户登录SQL Server的时候出现，授权过程在用户试图访问数据或执行命令的时候出现。 构造安全策略的第一个步骤是确定SQL Server用哪种方式验证用户。SQL Server的验证是把一组帐户、密码与Master数据库Sysxlogins表中的一个清单进行匹配。Windows NT/2000的验证是请求域控制器检查用户身份的合法性。一般地，如果服务器可以访问域控制器，我们应该使用Windows NT/2000验证。域控制器可以是Win2K服务器，也可以是NT服务器。无论在哪种情况下，SQL Server都接收到一个访问标记(Access Token)。访问标记是在验证过程中构造出来的一个特殊列表，其中包含了用户的SID(安全标识号)以及一系列用户所在组的SID。正如本文后面所介绍的，SQL Server以这些SID为基础授予访问权限。注意，操作系统如何构造访问标记并不重要，SQL Server只使用访问标记中的SID。也就是说，不论你使用SQL Server 2000、SQL Server 7.0、Win2K还是NT进行验证都

无关紧要，结果都一样。 如果使用SQL Server验证的登录，它最大的好处是很容易通过Enterprise Manager实现，最大的缺点在于SQL Server验证的登录只对特定的服务器有效，也就是说，在一个多服务器的环境中管理比较困难。使用SQL Server进行验证的第二个重要的缺点是，对于每一个数据库，我们必须分别地为它管理权限。如果某个用户对两个数据库有相同的权限要求，我们必须手工设置两个数据库的权限，或者编写脚本设置权限。如果用户数量较少，比如25个以下，而且这些用户的权限变化不是很频繁，SQL Server验证的登录或许适用。但是，在几乎所有的其他情况下(有一些例外情况，例如直接管理安全问题的应用)，这种登录方式的管理负担将超过它的优点。 二、Web环境中的验证 即使最好的安全策略也常常在一种情形前屈服，这种情形就是在Web应用中使用SQL Server的数据。在这种情形下，进行验证的典型方法是把一组SQL Server登录名称和密码嵌入到Web服务器上运行的程序，比如ASP页面或者CGI脚本.然后，由Web服务器负责验证用户，应用程序则使用它自己的登录帐户(或者是系统管理员sa帐户，或者为了方便起见，使用Sysadmin服务器角色中的登录帐户)为用户访问数据。 这种安排有几个缺点，其中最重要的包括：它不具备对用户在服务器上的活动进行审核的能力，完全依赖于Web应用程序实现用户验证，当SQL Server需要限定用户权限时不同的用户之间不易区别。如果你使用的是IIS 5.0或者IIS 4.0，你可以用四种方法验证用户。第一种方法是为每一个网站和每一个虚拟目录创建一个匿名用户的NT帐户。此后，所有应用程序登录SQL Server时都使用该安全环境。我们可以通过授予NT匿名帐户合适的权限，改进审核和

验证功能。 第二种方法是让所有网站使用Basic验证。此时，只有当用户在对话框中输入了合法的帐户和密码，IIS才会允许他们访问页面。IIS依靠一个NT安全数据库实现登录身份验证，NT安全数据库既可以在本地服务器上，也可以在域控制器上。当用户运行一个访问SQL Server数据库的程序或者脚本时，IIS把用户为了浏览页面而提供的身份信息发送给服务器。如果你使用这种方法，应该记住：在通常情况下，浏览器与服务器之间的密码传送一般是不加密的，对于那些使用Basic验证而安全又很重要的网站，你必须实现SSL(Secure Sockets Layer，安全套接字层)。 在客户端只使用IE 5.0、IE 4.0、IE 3.0浏览器的情况下，你可以使用第三种验证方法。你可以在Web网站上和虚拟目录上都启用NT验证。IE会把用户登录计算机的身份信息发送给IIS，当该用户试图登录SQL Server时IIS就使用这些登录信息。使用这种简化的方法时，我们可以在一个远程网站的域上对用户身份进行验证(该远程网站登录到一个与运行着Web服务器的域有着信任关系的域)。 最后，如果用户都有个人数字证书，你可以把那些证书映射到本地域的NT帐户上。个人数字证书与服务器数字证书以同样的技术为基础，它证明用户身份标识的合法性，所以可以取代NT的Challenge/Response(质询/回应)验证算法。Netscape和IE都自动在每一个页面请求中把证书信息发送给IIS。IIS提供了一个让管理员把证书映射到NT帐户的工具。因此，我们可以用数字证书取代通常的提供帐户名字和密码的登录过程。 由此可见，通过NT帐户验证用户时我们可以使用多种实现方法。即使当用户通过IIS跨越Internet连接SQL Server时，选择仍旧存在。因此，你应该把NT验证作为首选的用户身份验证

办法。 三、设置全局组 构造安全策略的下一个步骤是确定用户应该属于什么组。通常，每一个组织或应用程序的用户都可以按照他们对数据的特定访问要求分成许多类别。例如，会计应用软件的用户一般包括：数据输入操作员，数据输入管理员，报表编写员，会计师，审计员，财务经理等。每一组用户都有不同的数据库访问要求。 控制数据访问权限最简单的方法是，对于每一组用户，分别地为它创建一个满足该组用户权限要求的、域内全局有效的组。我们既可以为每一个应用分别创建组，也可以创建适用于整个企业的、涵盖广泛用户类别的组。然而，如果你想要能够精确地了解组成员可以做些什么，为每一个应用程序分别创建组是一种较好的选择。例如，在前面的会计系统中，我们应该创建Data Entry Operators、Accounting Data Entry Managers等组。请记住，为了简化管理，最好为组取一个能够明确表示出作用的名字。除了面向特定应用程序的组之外，我们还需要几个基本组。基本组的成员负责管理服务器。按照习惯，我们可以创建下面这些基本组：SQL Server Administrators，SQL Server Users，SQL Server Denied Users，SQL Server DB Creators，SQL Server Security Operators，SQL Server Database Security Operators，SQL Server Developers，以及 DB_Name Users(其中DB_Name是服务器上一个数据库的名字)。当然，如果必要的话，你还可以创建其他组。 创建了全局组之后，接下来我们可以授予它们访问SQL Server的权限。首先为SQL Server Users创建一个NT验证的登录并授予它登录权限，把Master数据库设置为它的默认数据库，但不要授予它访问任何其他数据库的权限，也不要把这个登录帐户设置为任何服务器角色的成员。接

着再为SQL Server Denied Users重复这个过程，但这次要拒绝登录访问。在SQL Server中，拒绝权限始终优先。创建了这两个组之后，我们就有了一种允许或拒绝用户访问服务器的便捷方法。 为那些没有直接在Sysxlogins系统表里面登记的组授权时，我们不能使用Enterpris Managr，因为Enterprise Manager只允许我们从现有登录名字的列表选择，而不是域内所有组的列表。要访问所有的组，请打开Query Analyzer，然后用系统存储过程sp_addsrvrolemember以及sp_addrolemember进行授权。 对于操作服务器的各个组，我们可以用sp_addsrvrolemember存储过程把各个登录加入到合适的服务器角色：SQL Server Administrators成为Sysadmins角色的成员，SQL Server DB Creators成为Dbcreator角色的成员，SQL Server Security Operators成为Securityadmin角色的成员。注意sp_addsrvrolemember存储过程的第一个参数要求是帐户的完整路径。例如，BigCo域的JoeS应该是bigco\joes(如果你想用本地帐户，则路径应该是server_name\joes)。 要创建在所有新数据库中都存在的用户，你可以修改Model数据库。为了简化工作，SQL Server自动把所有对Model数据库的改动复制到新的数据库。只要正确运用Model数据库，我们无需定制每一个新创建的数据库。另外，我们可以用sp_addrolemember存储过程把SQL Server Security Operators加入到db_securityadmin，把SQL Server Developers加入到db_owner角色。 注意我们仍然没有授权任何组或帐户访问数据库。事实上，我们不能通过Enterprise Manager授权数据库访问，因为Enterprise Manager的用户界面只允许我们把数据库访问权限授予合法的登录帐户。SQL Server不要求NT帐户在我们把它设置为数据库角色

的成员或分配对象权限之前能够访问数据库，但Enterprise Manager有这种限制。尽管如此，只要我们使用的是sp_addrolemember存储过程而不是Enterprise Manager，就可以在不授予域内NT帐户数据库访问权限的情况下为任意NT帐户分配权限。 到这里为止，对Model数据库的设置已经完成。但是，如果你的用户群体对企业范围内各个应用数据库有着类似的访问要求，你可以把下面这些操作移到Model数据库上进行，而不是在面向特定应用的数据库上进行。 