

用 .NET 实现对 SQL SERVER 数据库安全的动态监控

胡开明, 陈建华

(广东松山职业技术学院计算机系, 广东 韶关 512126)

摘要: 为了保障数据库服务器的安全, 提出通过设计 .NET 程序来实现对 SQL SERVER 数据库服务器的动态监控, 及时监控 SQL SERVER 数据库服务器的变化, 从而了解 SQL SERVER 数据库服务器的安全状态, 了解服务器上的数据是否发生变化。分析变化原因, 及时采取应对措施, 达到保障数据库服务器安全的目的。

关键词: .NET; SQL SERVER; 数据库; 动态监控

中图分类号: TP311.1

文献标识码: A

引言

随着计算机技术的飞速发展, 数据库的应用十分广泛, 深入到各个领域。然而各种应用系统的数据库中大量数据的安全问题、敏感数据的防窃取和防篡改问题, 越来越引起人们的高度重视^[1]。数据库系统作为信息的聚集体, 是计算机信息系统的核心部件, 其安全性至关重要, 关系到企业兴衰、成败。如何及时发现问题并保障数据库系统的安全就显得尤为重要。为保障数据库的安全, 除采取必要的安全策略外, 及时监控数据库服务器的动态变化, 不失为有益尝试。微软的 SQL SERVER 是一种广泛使用的数据库, 很多电子商务网站、企业内部信息化平台等都是基于 SQL SERVER 上的, 文章探讨如何通过设计 .NET 程序来实现对 SQL SERVER 数据库服务器的动态监控, 从而保障数据库的安全。

1 设计基本思路

首先获取 SQL SERVER 数据库服务器中的所有登录账号, 定期检查登录账号及其对应密码有无异常; 其次获取 SQL SERVER 数据库服务器中的所有数据库, 监控其是否发生变化; 然后选择指定数据库, 获取能访问该数据库的所有用户并检查有无异常; 同时显示数据库

中的数据表, 单击数据表, 动态显示并监控数据表中的内容。或将指定数据库的 SQL 查询语句传递到程序中, 查询指定内容, 有无异常。从各方面了解 SQL SERVER 数据库服务器的安全状态, 了解 SQL SERVER 数据库服务器数据是否发生变化。分析变化原因, 及时采取措施, 从而实现对 SQL SERVER 数据库安全的动态监控。

2 实现过程

2.1 监控 SQL SERVER 登录帐号及数据库权限控制

2.1.1 监控登录帐号

SQL SERVER 的安全控制策略是一个层次结构系统的集合。只有满足上一层系统的安全性要求之后, 才可以进入下一层^[2]。图 1 给出了 SQL SERVER 安全控制策略示意图。

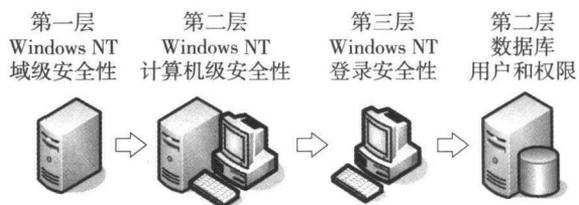


图 1 SQL SERVER 安全性控制策略

各层 SQL SERVER 安全控制策略是通过各层安全控制系统的身份验证实现的^[3]。身份验证是指当用户

访问系统时,系统对该用户的账号和口令的确认过程。身份验证的内容包括确认用户的账号是否有效、能否访问系统和能访问系统的哪些数据等。SQL SERVER的验证是把一组帐户、密码与 Master数据库 Syslogins表中的一个清单进行匹配, Syslogins是 Master数据库中的一个系统视图,用于提供该 SQL实例上的帐号相关信息,其 Base Table为 Syslogins,此表记录了系统的所有登录帐户的有关信息。使用如下方法可以监控 SQL SERVER中的登录帐号,如图 2所示。

```
OleDbConnection1.ConnectionString = " Provider =
SQLOLEDB; DataSource= server; Integrated Security = SS-
PI; Initial Catalog= master"
OleDbConnection1.Open()
Dim sqlstr As String= "select name from sysxlogins"
OleDbDataAdapter1.SelectCommand.CommandText =
sqlstr
OleDbDataAdapter1.SelectCommand.Connection =
OleDbConnection1
OleDbDataAdapter1.Fill(DataSet1, "sysxlogins")
DataGrid1.DataSource = DataSet1.Tables("sysxlog-
ins")
OleDbConnection1.Close()
```

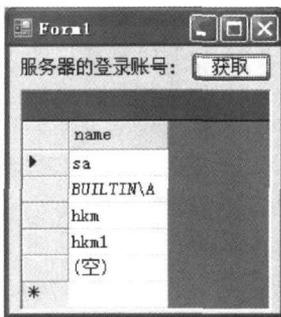


图 2 获取 SQL SERVER 的所有登录帐号

2.1.2 监控 SQL SERVER 数据库权限控制

SQL SERVER 数据库服务器的权限控制是基于 Master库的 Syslogins表^[4],拥有所有权限的帐号是 Sa 其他还有 Sysadmin, Db_owner 等不同权限帐号。但是,SQL SERVER 数据库最高权限帐号 Sa 的默认密码是空,这样如果安装的时候不注意,就会给数据带来毁灭性的灾难。恶意攻击者可以修改,删除所有数据,更加重要的是 SQL SERVER 帐号可以利用扩展执行系统命令。我们可以利用如下 SQL 查询语句来检查所有帐号是否密码为空,检查对存储过程和扩展存储过程的执行权,提防不必要的执行权限扩散,尽可能的删除存储扩展,防止本地用户利用存储扩展执行恶意命令。

(1)检查所有登陆帐号,是否有空口令帐号存在。

```
Select name Password from syslogins where password is
null
```

(2)检查对存储过程和扩展存储过程的执行权,提防不必要的执行权限扩散。

```
Select sysobjects name From sysobjects sysprotects
Where sysprotects uid= 0 AND xtype N ( 'X', 'P' ) AND
sysobjects id= sysprotects id
```

2.2 监控 SQL SERVER 数据库的变化

只要有权限,用户就可以操作指定内容,如建立、删除和修改数据库,建立数据表更新数据等。SQL SERVER 安装完毕后默认安装了 master model model northwind pubs 和 tempdb 这 6 个系统数据库,用户可以建立自己的数据库,用户数据库和系统数据库的相关信息均保存在系统数据库 Master 的 Sysdatabases 表中^[5]。因此,可以通过获取 SQL SERVER 数据库服务器中的所有数据库,监控其是否发生变化,从而了解数据库服务器的安全状态。使用下面的 SQL 语句可以查询 SQL SERVER 服务器中用户数据库。

```
Select name, createdate, filename from sysdatabases
where status= 16
```

其中 name, crdate, filename 分别表示数据库名、建立时间和数据库对应文件的存放路径; status= 16 表示是用户数据库, status < > 16 表示系统数据库。

使用如下方法将 SQL SERVER 服务器中所有数据库信息列表显示。如图 3 所示。

```
OleDbConnection1.ConnectionString = " Provider =
SQLOLEDB; DataSource= server; Integrated Security = SS-
PI; Initial Catalog= master"
```

```
OleDbConnection1.Open()
```

```
Dim sqlstr As String= "select name, crdate, filename
from sysdatabases"
```

```
OleDbDataAdapter1.SelectCommand.CommandText =
sqlstr
```

```
OleDbDataAdapter1.SelectCommand.Connection =
OleDbConnection1
```

```
OleDbDataAdapter1.Fill(DataSet1, "sysdatabases")
```

```
DataGrid1.DataSource = DataSet1.Tables("sysdatab-
ases")
```

```
OleDbConnection1.Close()
```

2.3 监控能访问该数据库的所有用户

在对应数据库中, Sysusers 表中存放着能访问该数据库中的所有用户信息,在上面获取 SQL SERVER 数据库服务器中的所有数据库后,选择指定数据库,通过 Sy-



图 3 获取 SQL SERVER 的所有数据库

users表我们可以获取到能访问该数据库的所有用户,通过动态监控所有用户的变化,了解 SQL SERVER 数据库服务器的安全状态。

可以通过如下 SQL 语句来访问对应数据库中的“sysusers”表。

```
Select name, createdate, updatedate from sysusers
where issqluser= 1
```

其中 name, createdate, updatedate 分别表示能访问该数据库的用户名、建立时间和更新时间; issqluser 如果该帐户是 SQL SERVER 用户,则为 1。

利用上面 SQL 语句,再使用如下方法,将访问该数据库的所有用户列举出来,如图 4 所示。

```
OleDbConnection1.ConnectionString = " Provider =
SQLOLEDB; Data Source= server; Integrated Security= SS-
PI; Initial Catalog= " & mdbname
```

mdbname 为上面所选择的数据库名

```
OleDbConnection1.Open()
```

```
Dim sqlstr As String = "Select name, createdate, up-
datedate from sysusers where issqluser= 1"
```

```
OleDbDataAdapter1.SelectCommand.CommandText =
sqlstr
```

```
OleDbDataAdapter1.SelectCommand.Connection =
OleDbConnection1
```

```
OleDbDataAdapter1.Fill(DataSet1, "sysusers")
```

```
DataGrid1.DataSource= DataSet1.Tables(" sysusers")
```

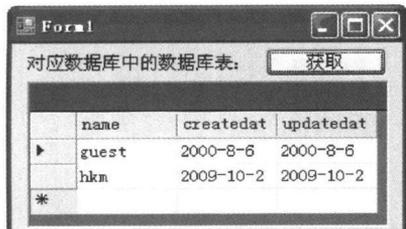


图 4 获取能访问对应数据库的所有用户

2.4 监控数据库的相关表的变化

在对应数据库中,“sysobjects”中存放着该数据库中的所有数据表的信息,在上面获取 SQL SERVER 数据库服务器中的所有数据库后,选择指定数据库,我们只要随时动态监控系统数据库的相关表的变化,就可以随时了解 SQL SERVER 数据库服务器的安全状态。

下面的查询语句可以得到该数据库中的用户表。

```
strSQL = " SELECT name FROM sysobjects WHERE
xtype= 'U' "
```

下面的查询语句可以得到该数据库中的系统表。

```
strSQL = " SELECT name FROM sysobjects WHERE
xtype= 'S' "
```

如果要得到所有表,去掉 WHERE 语句即可。

再使用如下方法显示对应数据库中的所有的数据表列表,如图 5 所示。

```
OleDbConnection1.ConnectionString = " Provider =
SQLOLEDB; Data Source= server; Integrated Security= SS-
PI; Initial Catalog= " & mdbname
```

mdbname 为上面所选择的数据库名

```
OleDbConnection1.Open()
```

```
Dim sqlstr As String = " select name FROM sysobjects
where xtype= 'U' "
```

```
OleDbDataAdapter1.SelectCommand.CommandText =
sqlstr
```

```
OleDbDataAdapter1.SelectCommand.Connection =
OleDbConnection1
```

```
OleDbDataAdapter1.Fill(DataSet1, "sysobjects")
```

```
DataGrid1.DataSource = DataSet1.Tables(" sysob-
jects")
```

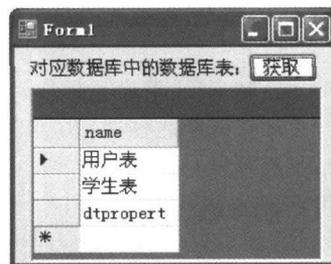


图 5 获取对应数据库中的所有数据表

2.5 监控数据表的相关内容的变化

2.5.1 动态显示数据表中的内容

选择相应数据表,动态显示数据表中的内容。以了解服务器数据是否发生变化,分析变化原因,及时采取措施。使用如下方法显示对应数据表中的内容,如图 6 所示。

```
OleDbConnection1.ConnectionString = " Provider =
```

```

SQLOLEDB; Data Source= server; Integrated Security= SS-
PI; Initial Catalog= " & mdbname

```

mdbname为上面所选择的数据库名

```
OLEDBConnection1.Open()
```

```
Dim sqlstr As String = " select * FROM " & ta-
blename

```

tablename为上面所选择对应数据表名

```
OLEDBDataAdapter1.SelectCommand.CommandText =
sqlstr

```

```
OLEDBDataAdapter1.SelectCommand.Connection =
OLEDBConnection1

```

```
OLEDBDataAdapter1.Fill(DataSet1, tablename)

```

```
DataGrid1.DataSource= DataSet1.Tables(tablename)

```

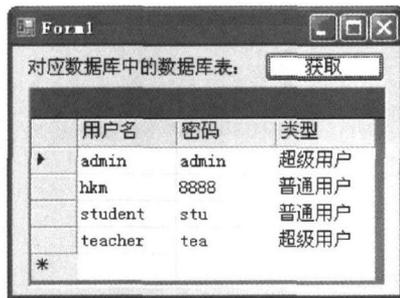


图 6 获取对应数据表中的内容

2.5.2 SQL 查询语句作为参数

为使程序更具有灵活性和可操作性, 可以设置一个

登录表单, 直接由用户确定要执行查询的内容, 然后由查询内容自动生成 SQL 语句或直接输入 SQL 语句, 然后通过提交后, 接受 SQL 语句并执行查询, 显示查询内容。以了解服务器数据是否发生变化。

3 结束语

保护数据不受内部和外部侵害是数据库管理的一项重要的工作, 如何进行 SQL SERVER 数据库服务器安全管理工作, 本文提出了一个新的思路, 用 .NET 设计程序实现对数据库的监控和管理, 是数据库服务的安全管理方面和安全防范策略的一种补充和延伸, 在实际工作中具有重要意义。

参考文献:

- [1] 萨师焯, 王珊. 数据库系统概论 [M]. 北京: 高等教育出版社, 2000
- [2] 郑阿奇. SQL Server 教程 [M]. 北京: 清华大学出版社, 2005
- [3] 郑阿奇, 彭作民. Visual Basic .NET 程序设计教程 [M]. 北京: 机械工业出版社, 2007.
- [4] 徐人凤, 曾建华. SQL Server 2000 数据库及应用 [M]. 北京: 高等教育出版社, 2004
- [5] 邹明, 赵刚. 基于 MATLAB 创建、删除远程 SQL 数据库的 ADO 技术 [J]. 四川理工学院学报: 自然科学版, 2007, 20(2): 43-46

Dynamic Monitoring of Security SQL Server Based on .NET

HUKaiming, CHEN Jian-hua

(Department of Computer Science, Guangdong Songshan Vocational College, Shaoguan 512126, China)

Abstract To protect the security of the database server the dynamic monitoring of SQL SERVER database server can be achieved by the design of .NET program, which monitors the changes in database server in time and reflects security status of database server and data changes on the server; and then analysis of the reasons for change and timely measure can be taken to safeguard the security of the database server

Key words .NET; SQL server; database; dynamic monitoring