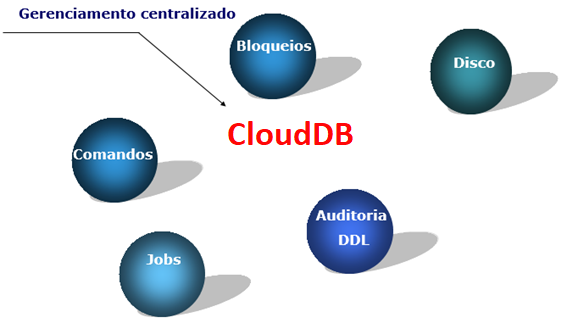
****

****



**Procedimento de criação de auditoria SQL Server**

No primeiro script criamos um auditor no nosso servidor para auditar logins no SQL Server, criamos a pasta C:\SQLAudits\ para armazenar os arquivos de auditoria, com o tamanho máximo de 500 MB para cada arquivo e definimos as regras de auditoria.

SERVER\_PERMISSION\_CHANGE\_GROUP

Esse evento é gerado quando um GRANT, REVOKE ou DENY é emitido para permissões no escopo do servidor. Equivalente à classe de evento GDR de escopo do servidor de auditoria.

SERVER\_PRINCIPAL\_CHANGE\_GROUP

Este evento é gerado quando entidades de servidor são criadas, alteradas ou eliminadas. Equivalente à Classe de Evento de Gerenciamento Principal do Servidor Audit.

SERVER\_ROLE\_MEMBER\_CHANGE\_GROUP

Esse evento é gerado sempre que um logon é adicionado ou removido de uma função de servidor fixa. Esse evento é gerado para os procedimentos armazenados sp\_addsrvrolemember e sp\_dropsrvrolemember. Equivalente à classe de evento Audit Add Login to Server Role.

SERVER\_OBJECT\_PERMISSION\_CHANGE\_GROUP

Esse evento é gerado sempre que um GRANT, REVOKE ou DENY é emitido para uma permissão de objeto de servidor por qualquer principal no SQL Server. Equivalente à classe de eventos GDR do objeto de servidor de auditoria.

SERVER\_OBJECT\_CHANGE\_GROUP

Inclui CREATE, ALTER e DROP para qualquer objeto de servidor (Database ou endpoint).

**Script 1**

-- Criar auditoria de logins

USE MASTER

GO

-- Create the server audit

CREATE SERVER AUDIT login\_perm\_audit

TO FILE ( FILEPATH ='C:\SQLAudits\' );

GO

--Define o tamanho de cada arquivo de log da auditoria

ALTER SERVER AUDIT [login\_perm\_audit]

TO FILE

(MAXSIZE = 500 MB)

-- Enable the server audit

ALTER SERVER AUDIT login\_perm\_audit

WITH (STATE = ON) ;

GO

CREATE SERVER AUDIT SPECIFICATION login\_audit\_spec

FOR SERVER AUDIT login\_perm\_audit

ADD (SERVER\_PERMISSION\_CHANGE\_GROUP),

ADD (SERVER\_PRINCIPAL\_CHANGE\_GROUP),

ADD (SERVER\_ROLE\_MEMBER\_CHANGE\_GROUP),

ADD (SERVER\_OBJECT\_PERMISSION\_CHANGE\_GROUP),

ADD (SERVER\_OBJECT\_CHANGE\_GROUP)

WITH ( STATE = ON )

No segundo script criamos um auditor no nosso servidor para auditar alteração de dados no SQL Server, criamos a pasta C:\SQLAudits\ para armazenar os arquivos de auditoria, com o tamanho máximo de 500 MB para cada arquivo e definimos as regras de auditoria.

DATABASE\_PERMISSION\_CHANGE\_GROUP

Esse evento é gerado sempre que um GRANT, REVOKE ou DENY é emitido para uma permissão de instrução por qualquer usuário no SQL Server para eventos somente de banco de dados, como conceder permissões em um banco de dados. Equivalente à classe de evento GDR do escopo do banco de dados de auditoria.

DATABASE\_OBJECT\_PERMISSION\_CHANGE\_GROUP

Este evento é gerado quando um GRANT, REVOKE ou DENY foi emitido para objetos de banco de dados, como assemblies e esquemas. Esse evento é gerado para qualquer alteração de permissão de objeto para qualquer banco de dados no servidor. Equivalente à Classe de Evento GDR do Objeto de Banco de Dados de Auditoria.

DATABASE\_CHANGE\_GROUP

Esse evento é gerado quando um banco de dados é criado, alterado ou descartado. Este evento é gerado sempre que qualquer banco de dados é criado, alterado ou descartado. Equivalente à Classe de Evento Audit Database Management.

DATABASE\_OBJECT\_CHANGE\_GROUP

Esse evento é gerado quando uma instrução CREATE, ALTER ou DROP é executada em objetos de banco de dados, como esquemas. Este evento é gerado sempre que qualquer objeto de banco de dados é criado, alterado ou descartado. Nota: Isso pode levar a quantidades muito grandes de registros de auditoria.

SCHEMA\_OBJECT\_CHANGE\_GROUP

Esse evento é gerado quando uma operação CREATE, ALTER ou DROP é executada em um esquema. Equivalente à Classe de Evento de Gerenciamento de Objeto de Esquema de Auditoria.

**Script 2**

--Criar a auditoria de alterações de dados

USE master ;

GO

-- Create the server audit.

-- Apenas nome para a task [Spec-DBA-DDL]

CREATE SERVER AUDIT [Spec-DBA-DDL]

TO FILE ( FILEPATH = 'C:\SQLAudits\' ) ;

GO

--Define o tamanho de cada arquivo de log da auditoria

ALTER SERVER AUDIT [Spec-DBA-DDL]

TO FILE

(MAXSIZE = 500 MB)

-- Enable the server audit.

ALTER SERVER AUDIT [Spec-DBA-DDL]

WITH (STATE = ON) ;

GO

-- Criar a auditoria de alterações de dados

USE [master]

GO

CREATE SERVER AUDIT SPECIFICATION [Spec-DBA-DDL]

FOR SERVER AUDIT [Spec-DBA-DDL]

ADD (DATABASE\_PERMISSION\_CHANGE\_GROUP),

ADD (DATABASE\_OBJECT\_PERMISSION\_CHANGE\_GROUP),

ADD (DATABASE\_CHANGE\_GROUP),

ADD (DATABASE\_OBJECT\_CHANGE\_GROUP),

ADD (SCHEMA\_OBJECT\_CHANGE\_GROUP)

WITH (STATE = ON)

GO

-- Criar dentro do banco que voce quer salvar os dados de auditoria

USE [AdventureWorks2019]

GO

-- nome do banco para executar a task [AdventureWorks2019\_AuditDatabase\_DML]

ALTER DATABASE AUDIT SPECIFICATION [AdventureWorks2019\_AuditDatabase\_DML]

WITH (STATE = OFF)

ALTER DATABASE AUDIT SPECIFICATION [AdventureWorks2019\_AuditDatabase\_DML]

FOR SERVER AUDIT [Spec-DBA-DDL]

ADD (DATABASE\_OBJECT\_CHANGE\_GROUP),

ADD (SCHEMA\_OBJECT\_CHANGE\_GROUP),

ADD (DELETE, INSERT, UPDATE ON DATABASE::[AdventureWorks2019] BY [dbo]),

ADD (DELETE, INSERT, UPDATE ON DATABASE::[AdventureWorks2019] BY [guest]),

ADD (DELETE, INSERT, UPDATE ON DATABASE::[AdventureWorks2019] BY [INFORMATION\_SCHEMA]),

ADD (DELETE, INSERT, UPDATE ON DATABASE::[AdventureWorks2019] BY [sys])

WITH (STATE = ON);

GO

No terceiro script criamos uma tabela temporária com as informações que queremos capturar na auditoria.

**Script 3**

USE AdventureWorks2019

GO

SET NOCOUNT ON;

IF OBJECT\_ID(N'tempdb..##INFORMACOES') IS NOT NULL

DROP TABLE ##INFORMACOES

CREATE TABLE ##INFORMACOES (

event\_time VARCHAR(MAX),

object\_id VARCHAR(MAX),

class\_type VARCHAR(MAX),

server\_principal\_name VARCHAR(MAX),

server\_instance\_name VARCHAR(MAX),

database\_name VARCHAR(MAX),

schema\_name VARCHAR(MAX),

object\_name VARCHAR(MAX),

statement VARCHAR(MAX),

application\_name VARCHAR(MAX)

)

DECLARE

@event\_time VARCHAR(MAX),

@object\_id VARCHAR(MAX),

@class\_type VARCHAR(MAX),

@server\_principal\_name VARCHAR(MAX),

@server\_instance\_name VARCHAR(MAX),

@database\_name VARCHAR(MAX),

@schema\_name VARCHAR(MAX),

@object\_name VARCHAR(MAX),

@statement VARCHAR(MAX),

@application\_name VARCHAR(MAX)

insert into ##INFORMACOES

SELECT

CAST(event\_time AS VARCHAR(MAX)),

CAST(object\_id AS VARCHAR(MAX)),

LTRIM(RTRIM(REPLACE(REPLACE(REPLACE(CAST(class\_type AS VARCHAR(MAX)), CHAR(9), ''), CHAR(10), ''), CHAR(13), ''))),

LTRIM(RTRIM(REPLACE(REPLACE(REPLACE(CAST(server\_principal\_name AS VARCHAR(MAX)), CHAR(9), ''), CHAR(10), ''), CHAR(13), ''))),

LTRIM(RTRIM(REPLACE(REPLACE(REPLACE(CAST(server\_instance\_name AS VARCHAR(MAX)), CHAR(9), ''), CHAR(10), ''), CHAR(13), ''))),

LTRIM(RTRIM(REPLACE(REPLACE(REPLACE(CAST(database\_name AS VARCHAR(MAX)), CHAR(9), ''), CHAR(10), ''), CHAR(13), ''))),

LTRIM(RTRIM(REPLACE(REPLACE(REPLACE(CAST(schema\_name AS VARCHAR(MAX)), CHAR(9), ''), CHAR(10), ''), CHAR(13), ''))),

LTRIM(RTRIM(REPLACE(REPLACE(REPLACE(CAST(object\_name AS VARCHAR(MAX)), CHAR(9), ''), CHAR(10), ''), CHAR(13), ''))),

LTRIM(RTRIM(REPLACE(REPLACE(REPLACE(CAST(statement AS VARCHAR(MAX)), CHAR(9), ''), CHAR(10), ''), CHAR(13), ''))),

LTRIM(RTRIM(REPLACE(REPLACE(REPLACE(CAST(application\_name AS VARCHAR(MAX)), CHAR(9), ''), CHAR(10), ''), CHAR(13), '')))

FROM sys.fn\_get\_audit\_file('C:\SQLAudits\\*.sqlaudit', DEFAULT, DEFAULT) as ta

WHERE

ta.event\_time > getdate () -30 -- Listar todos os registros dos últimos 30 dias

and (database\_name = 'AdventureWorks2019' or database\_name is null)

and schema\_name not in ('DBA')

and application\_name not in

(

'Database Engine Tuning Advisor'

)

and server\_principal\_name not in

(

'cloud.prime',

'usr\_solarwinds\_app\_01',

'NT SERVICE\SQLTELEMETRY$SQLSERVERPROD'

)

and OBJECT\_NAME not in

(

'nre\_fulltext',

'AdventureWorks2019',

'sysarticlecolumns',

'sysarticles',

'syspublications',

'sysreplservers',

'syssubscriptions',

'TOTVSAUDIT',

'tr\_MStran\_alterschemaonly',

'tr\_MStran\_altertable',

'tr\_MStran\_altertrigger',

'tr\_MStran\_alterview',

'tr\_MStran\_droptable',

'ZAUDITCHANGES',

'ZAUDITSCHEMAEVENTS')

order by 1 desc

SELECT

[event\_time],

[object\_id],

[class\_type],

[server\_principal\_name],

[server\_instance\_name],

[database\_name],

[schema\_name],

[object\_name],

[statement],

[application\_name]

FROM ##INFORMACOES

No terceiro script criamos uma tabela temporária com as informações que queremos capturar na auditoria

**Script 4**

CREATE TABLE EXEMPLO( A INT)

INSERT INTO EXEMPLO VALUES (1)

UPDATE EXEMPLO SET A = 2

DELETE EXEMPLO

DROP TABLE EXEMPLO

**Script 5**

--Consultas de auditorias existentes no ambiente

SELECT DATEADD(MINUTE, DATEDIFF(MINUTE, GETUTCDATE(), CURRENT\_TIMESTAMP), event\_time) AS event\_time\_afterconvert

,getdate() 'Current\_system\_time'

,\*

FROM fn\_get\_audit\_file('D:\DATA\\*', DEFAULT, DEFAULT)

--Listar as auditorias existentes

SELECT \* FROM sys.server\_file\_audits

GO

-- Listar os arquivos criados pela auditoria

SELECT \* FROM sys.fn\_get\_audit\_file('C:\SQLAudits\\*', NULL, NULL);

GO

--Listar as auditorias, informar o status e tamanho dos arquivos em bytes

SELECT \* FROM sys.dm\_server\_audit\_status

**Script 5**

--Consulta alterações em logins pela auditoria

SELECT DATEADD(MINUTE, DATEDIFF(MINUTE, GETUTCDATE(), CURRENT\_TIMESTAMP), event\_time) AS event\_time\_afterconvert

,getdate() 'Current\_system\_time',

session\_server\_principal\_name AS UserName ,

server\_instance\_name ,

database\_name ,

object\_name ,

statement

--FROM sys.fn\_get\_audit\_file('C:\SQLAudits\\*.sqlaudit', DEFAULT, DEFAULT) -- Listar alterações todas as auditorias

--FROM sys.fn\_get\_audit\_file('C:\SQLAudits\login\_perm\_audit\_800051E9-B96C-4E6D-A4BA-1A4A9D0FA822\_0\_133147363999240000.sqlaudit', DEFAULT, DEFAULT) -- Caso queira, informar um arquivo específico para consulta

FROM sys.fn\_get\_audit\_file('C:\SQLAudits\\*.sqlaudit', DEFAULT, DEFAULT) -- Listar alterações nas auditorias de alterações de dados

--where statement like '%login%'

order by 1 desc