SQL SERVER AUDIT

SQL SERVER AUDIT sem Overhead

RESUMO

Olá, sou DBA SQL Server Sênior, atuando com SQL há 10 anos. Criando documentações com ênfase em melhorias baseadas nos meu Cases.

Junior Moraes

SQL Server DBA Senior Consultant

Dias atras escutei nos corredores que a ferramenta de Auditoria do SQL era pesada e não funcionava corretamente pois causava um Overhead no ambiente. Causando concorrência ou em alguns casos podendo até derrubar o ambiente.

Diante disso, achei interessante analisar a questão, já que havia implantado o SQL Audit em outros ambientes e não sofri da mesma situação.

Passando a entender o cenário, verifiquei que o Audit criado não continha filtros, ou seja, auditava tudo que passava pelo SQL. Desse modo, realmente o Audit vai causar um overhead no ambiente, gerando um stress na corporação.

A auditoria é uma ferramenta que deve ser objetiva e ter seus filtros claros, para auditar somente aquilo que procura. Caso contrário, você criará uma arapuca para si mesmo.

Abaixo, irei mostrar como configurei o meu Audit com assertividade, leveza e o principal objetividade. Encontrando o que procurávamos sem precisar derrubar o ambiente :D.

No cenário em questão precisávamos capturar todos os objetos que sofreram alteração no ambiente. Para isso utilizamos o Audit Action Type "`SCHEMA_OBJECT_CHANGE_GROUP` e `SERVER_OBJECT_CHANGE_GROUP`".

Segue link de todos os tipos de auditoria: https://docs.microsoft.com/pt-br/sql/relational-databases/security/auditing/sql-server-audit-action-groups-and-actions?view=sql-server-ver16

Definido o que queríamos auditar, verificamos se teríamos espaço em disco pois o **Audit salva a sua coleta fisicamente. Estimar a volumetria também é importante**.

Em seguida iniciamos a criação do AUDIT com o script abaixo:

1-Create_ServerAudit

Percebam que no script, além do **ACTION TYPE**, inclui um **WHERE**. Isso mesmo, conseguimos incluir outros tipos de filtros no Audit Server, isso com certeza afunila a busca e nos leva com mais assertividade ao alvo. Sem contar que não causa overhead no ambiente.

No nosso caso, tirei as alterações dos Objetos provenientes da **UPDATE STATISTICS.**

```
WHERE (NOT [statement] like '%UPDATE STATISTICS%') -FILTRO
```

Bom, após criar o AUDIT SERVER, precisamos criar o **AUDIT SPECIFICATIONS**, que será especificamente onde vamos informar o AUDIT TYPE, o script abaixo já está pronto, com os TYPES que informei.

2_Create_AuditSpecifications.sql

```
ADD (SCHEMA_OBJECT_CHANGE_GROUP), --AUDIT TYPE 1
ADD (SERVER_OBJECT_CHANGE_GROUP) -- AUDIT TYPE 2
```

SERVER_OBJECT_CHANGE_GROUP - *Esse evento é gerado para operações CREATE, ALTER ou DROP em objetos de servidor.*

SCHEMA_OBJECT_CHANGE_GROUP - *Esse evento é gerado sempre que algum esquema de qualquer banco de dados é alterado.*

Após criar nossa Auditoria, será criado o arquivo no caminho que informamos no primeiro Script, já iniciando sua gravação.

Arquivo criado:

Certo, mas com o arquivo criado, o que fazer?

Pois bem, conseguimos fazer a leitura do arquivo e o que ele auditou por dentro do SQL, utilizando em resumo a query abaixo, na qual eu inseri filtros:

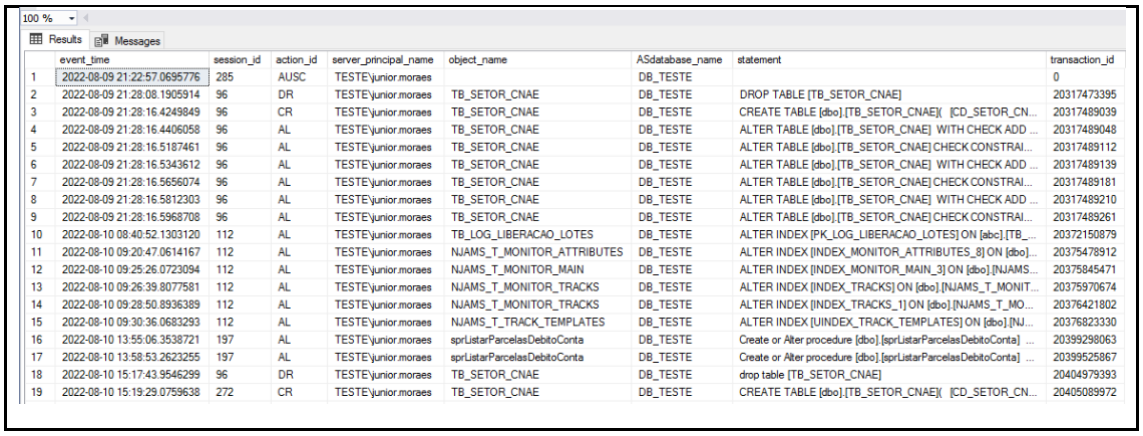
3_Select_Audit.sql

```
--SELECT INFORMANDO O ARQUIVO ESPECÍFICO
SELECT *
FROM Sys.fn_get_audit_file('F:\AUDIT_SQL\AUDIT_OBJECT_DEV_C3B993FF-AC7C-443F-
B0AB-CF9582893A57_0_133041166209690000.sqlaudit',default,default)


--SELECT DE TODOS OS ARQUIVOS
SELECT event_time, session_id, action_id, server_principal_name, object_name,
database_name, statement,
transaction_id
FROM Sys.fn_get_audit_file('F:\AUDIT_SQL\*.sqlaudit',default,default)
```

Segue result da query com as colunas selecionadas, e que eu particularmente prefiro executar.

Além disso, já está com as informações para análise, como: Database, Objeto alterado, usuário, comando executado, SPID e Transação ID.



100 %

Results | Messages

	event_time	session_id	action_id	server_principal_name	object_name	ASdatabase_name	statement	transaction_id
1	2022-08-09 21:22:57.0695776	285	AUSC	TESTE\junior.moraes		DB_TESTE		0
2	2022-08-09 21:28:08.1905914	96	DR	TESTE\junior.moraes	TB_SETOR_CNAE	DB_TESTE	DROP TABLE [TB_SETOR_CNAE]	20317473395
3	2022-08-09 21:28:16.4249849	96	CR	TESTE\junior.moraes	TB_SETOR_CNAE	DB_TESTE	CREATE TABLE [dbo].[TB_SETOR_CNAE]([CD_SETOR_CN...	20317489039
4	2022-08-09 21:28:16.4406058	96	AL	TESTE\junior.moraes	TB_SETOR_CNAE	DB_TESTE	ALTER TABLE [dbo].[TB_SETOR_CNAE] WITH CHECK ADD ...	20317489048
5	2022-08-09 21:28:16.5187461	96	AL	TESTE\junior.moraes	TB_SETOR_CNAE	DB_TESTE	ALTER TABLE [dbo].[TB_SETOR_CNAE] CHECK CONSTRAI...	20317489112
6	2022-08-09 21:28:16.5343612	96	AL	TESTE\junior.moraes	TB_SETOR_CNAE	DB_TESTE	ALTER TABLE [dbo].[TB_SETOR_CNAE] WITH CHECK ADD ...	20317489139
7	2022-08-09 21:28:16.5656074	96	AL	TESTE\junior.moraes	TB_SETOR_CNAE	DB_TESTE	ALTER TABLE [dbo].[TB_SETOR_CNAE] CHECK CONSTRAI...	20317489181
8	2022-08-09 21:28:16.5812303	96	AL	TESTE\junior.moraes	TB_SETOR_CNAE	DB_TESTE	ALTER TABLE [dbo].[TB_SETOR_CNAE] WITH CHECK ADD ...	20317489210
9	2022-08-09 21:28:16.5968708	96	AL	TESTE\junior.moraes	TB_SETOR_CNAE	DB_TESTE	ALTER TABLE [dbo].[TB_SETOR_CNAE] CHECK CONSTRAI...	20317489261
10	2022-08-10 08:40:52.1303120	112	AL	TESTE\junior.moraes	TB_LOG_LIBERACAO_LOTES	DB_TESTE	ALTER INDEX [PK_LOG_LIBERACAO_LOTES] ON [abc].[TB_...	20372150879
11	2022-08-10 09:20:47.0614167	112	AL	TESTE\junior.moraes	NJAMS_T_MONITOR_ATTRIBUTES	DB_TESTE	ALTER INDEX [INDEX_MONITOR_ATTRIBUTES_8] ON [dbo]...	20375478912
12	2022-08-10 09:25:26.0723094	112	AL	TESTE\junior.moraes	NJAMS_T_MONITOR_MAIN	DB_TESTE	ALTER INDEX [INDEX_MONITOR_MAIN_3] ON [dbo].[NJAMS...	20375845471
13	2022-08-10 09:26:39.8077581	112	AL	TESTE\junior.moraes	NJAMS_T_MONITOR_TRACKS	DB_TESTE	ALTER INDEX [INDEX_TRACKS] ON [dbo].[NJAMS_T_MONIT...	20375970674
14	2022-08-10 09:28:50.8936389	112	AL	TESTE\junior.moraes	NJAMS_T_MONITOR_TRACKS	DB_TESTE	ALTER INDEX [INDEX_TRACKS_1] ON [dbo].[NJAMS_T_MO...	20376421802
15	2022-08-10 09:30:36.0683293	112	AL	TESTE\junior.moraes	NJAMS_T_TRACK_TEMPLATES	DB_TESTE	ALTER INDEX [UINDEX_TRACK_TEMPLATES] ON [dbo].[NJ...	20376823330
16	2022-08-10 13:55:06.3538721	197	AL	TESTE\junior.moraes	sprListarParcelasDebitoConta	DB_TESTE	Create or Alter procedure [dbo].[sprListarParcelasDebitoConta] ...	20399298063
17	2022-08-10 13:58:53.2623255	197	AL	TESTE\junior.moraes	sprListarParcelasDebitoConta	DB_TESTE	Create or Alter procedure [dbo].[sprListarParcelasDebitoConta] ...	20399525867
18	2022-08-10 15:17:43.9546299	96	DR	TESTE\junior.moraes	TB_SETOR_CNAE	DB_TESTE	drop table [TB_SETOR_CNAE]	20404979393
19	2022-08-10 15:19:29.0759638	272	CR	TESTE\junior.moraes	TB_SETOR_CNAE	DB_TESTE	CREATE TABLE [dbo].[TB_SETOR_CNAE]([CD_SETOR_CN...	20405089972

Veja que nesse exemplo já temos registros consideráveis e na coluna **ACTION_ID**, informa o tipo da Alteração.

AL = ALTER
CR – CREATE
DR – DROP

Por fim, essa auditoria é um case implantado em ambiente DEV, onde sofre constantes alterações, e mesmo após um mês o arquivo principal ainda está com 116 MB e sem causar qualquer overhead no ambiente.

Isso ocorre por conta de as definições estarem bem atribuídas, ou seja, Audit Type e filtros estarem totalmente voltado par ao objetivo da auditoria.

Assim, sendo a coleta mínima e assertiva.

Por fim, acredito que seja valido um laboratório com Audit Server para entender as funcionalidades, antes de implementar em produção.

Por fim, espero ajudar com esse conteúdo.

Junior Moraes

DBA SQL Server Senior Consultant
Cel: +55 (41) 9 88816464
e-mail: jcjunior.dba@outlook.com