18 Transparent Data Encryption

A Criptografia Transparente de Dados (frequentemente abreviada como TDE) é uma tecnologia utilizada pela Microsoft, IBM e Oracle para criptografar arquivos de banco de dados. O TDE oferece criptografia no nível de arquivo. Ele resolve o problema de proteger dados em repouso, criptografando bancos de dados tanto no disco rígido quanto, consequentemente, em mídias de backup. No entanto, ele não protege dados em trânsito nem dados em uso. As empresas geralmente utilizam o TDE para atender a requisitos de conformidade, como o PCI DSS, que exige a proteção de dados em repouso.

Este capítulo fornece diretrizes sobre o uso do TDE.

O DBMaker oferece dois tipos de criptografia de dados:

- **Criptografia de banco de dados**: criptografa os arquivos de banco de dados .DB e .BB.
- **Criptografia de colunas**: suporta a palavra-chave ENCRYPT nos comandos CREATE TABLE e ALTER TABLE, permitindo que os usuários criem colunas criptografadas.

18.1 Database encryption

Quando os usuários desejam criar um banco de dados criptografado, o DBMaker fornece a palavra-chave **DB_TDEMd** no arquivo *dmconfig.ini*. O mecanismo de criptografia utilizado é o **AES-256** com a biblioteca **OpenSSL**.

Atenção: apenas os dados do usuário e os arquivos blob (.DB, .BB) serão criptografados. Arquivos SDB, SBB, TMPTABLESPACE, JNL e fileobj não possuem suporte para criptografia.

Para ativar a criptografia do banco de dados, os usuários devem definir a palavra-chave **DB_TDEMd=1** no arquivo *dmconfig.ini* antes de criar o banco de dados.

```
DB_TDEMd=1
```

Após iniciar o banco de dados criptografado, os usuários podem usar o seguinte comando para verificar se o banco de dados está criptografado:

```
dmsql> select * from SYSINFO where ID='0724';
```

```
     ID          INFO            VALUE

======== ========== ==============

   0724         TDE_DB           ON
```

18.2 Column encryption

A criptografia de colunas, às vezes chamada de criptografia em nível de coluna ou criptografia em nível de célula, é usada para prevenir o acesso não autorizado e garantir a integridade dos dados. Em um banco de dados criptografado, os usuários SYSADM e SYSDBA podem definir a senha de TDE e abrir/fechar colunas. Abrir uma coluna significa que a coluna criptografada agora está acessível, e todo usuário com privilégio na tabela pode inserir, atualizar, excluir ou selecionar a coluna criptografada. Fechar uma coluna significa que a coluna criptografada está protegida, e ninguém (incluindo o proprietário da tabela) pode acessá-la.

Set/Alter TDE password of column encryption

Antes de ativar a criptografia de colunas, o SYSDBA ou SYSADM deve definir a senha TDE. Essa senha é utilizada para abrir ou fechar a criptografia de colunas. Se os usuários tentarem abrir ou fechar colunas sem que a senha TDE tenha sido configurada, será retornado o seguinte erro:

"ERROR (6891): [DBMaker] please set new TDE password".

SYSADM e SYSDBA podem usar a seguinte consulta para definir ou alterar a senha TDE:

```
ALTER TDE PASSWORD old_password TO new_password;
```

O SYSADM pode usar a seguinte consulta para definir ou alterar a senha TDE. Diferente da consulta anterior, esta não requer a **old_password**, o que significa que o SYSADM pode redefinir a senha TDE caso a **old_password** seja perdida.

NOTA: *Ao configurar a senha TDE pela primeira vez, use* ***NULL*** *para representar a* ***OLD_PASSWORD****.*

Exemplo 1:

Alterar a senha TDE de **null** para **password**:

```
dmsql> ALTER TDE PASSWORD null TO password;
```

Exemplo 2:

Alterar a senha TDE de **password** para **new_password**:

```
dmsql> ALTER TDE PASSWORD password TO new_password;
```

Open/Close column encryption

Após a senha TDE ser configurada, o SYSADM e o SYSDBA podem abrir a coluna e permitir que os usuários criem tabelas com colunas criptografadas. As palavras-chave **ENCRYPT** e **DECRYPT** estarão disponíveis somente quando a coluna estiver aberta.

O SYSADM/SYSDBA pode definir parâmetros usando o comando:

```
CALL SETSYSTEMOPTION('tde_open','tde_password'); //Opens column encryption
CALL SETSYSTEMOPTION('tde_close','tde_password'); //Closes column encryption
```

A configuração padrão da criptografia de colunas é **fechada**, o que significa que o SYSADM/SYSDBA deve abrir a criptografia de colunas primeiro, caso os usuários precisem executar operações DML e DDL com criptografia de colunas.

A seguinte consulta pode abrir/fechar a criptografia de colunas automaticamente quando o banco de dados for iniciado:

```
CALL SETSYSTEMOPTION('tde_open_auto','tde_password'); //Opens column encryption automatically when database is started CALL SETSYSTEMOPTION('tde_close_auto','tde_password'); //Closes column encryption automatically when database is started (default)
```

Encrypt/Decrypt columns

Em um banco de dados criptografado, após a senha TDE ser configurada e a criptografia de colunas ser aberta, os usuários podem começar a usar as palavras-chave **ENCRYPT** e **DECRYPT** nas instruções **CREATE TABLE** e **ALTER TABLE**.

Para criar uma coluna criptografada, use a palavra-chave **ENCRYPT** nas instruções **CREATE TABLE** ou **ALTER TABLE**.

CREATE TABLE table-name (column-name data-type ENCRYPT) ALTER TABLE table-name ADD column-name data-type ENCRYPT ALTER TABLE table-name MODIFY column-name [ENCRYPT|DECRYPT]

Exemplo:

Criando uma tabela com uma coluna criptografad**a**

```
dmSQL> CREATE TABLE enc1(c1 int encrypt);
```

Aqui estão os exemplos para cada operação mencionada:

Exemplos:

Alterando a tabela com coluna criptografada.

Adicionar uma coluna criptografada:

```
dmSQL> ALTER TABLE enc1 ADD c2 int ENCRYPT;
```

Descriptografar uma coluna criptografada:

```
dmSQL> ALTER TABLE enc1 MODIFY c2 DECRYPT;
```

Criptografar uma coluna normal:

```
dmSQL> ALTER TABLE enc1 MODIFY c2 ENCRYPT;
```

Ao utilizar colunas criptografadas, os usuários devem considerar as seguintes restrições:

- A criptografia de colunas só pode ser usada em bancos de dados criptografados.
- A criptografia da coluna deve ser desbloqueada se houver operações DML/DDL envolvendo colunas criptografadas.
- Chaves estrangeiras e chaves primárias referenciadas não podem ser criptografadas.

Working with Encrypted Columns

Este capítulo ensinará os usuários a trabalhar com colunas criptografadas, incluindo operações DML e outras operações (domínio/gatilho/visualização/sinônimo).

Quando a criptografia da coluna estiver desbloqueada, os usuários poderão acessar normalmente essas colunas.

```
dmSQL> CALL SETSYSTEMOPTION('tde_open','password');
dmSQL> CREATE TABLE t1(c1 int ENCRYPT);
dmSQL> INSERT INTO t1 VALUES(1);
1 rows inserted


dmSQL> UPDATE t1 SET c1=2;
1 rows updated
```

```
dmSQL> DELETE FROM t1 WHERE c1=2;
1 rows deleted
```

Pelo contrário, todas as operações DML que envolvem colunas criptografadas serão bloqueadas quando a criptografia da coluna estiver fechada. A mensagem de erro “ERROR (6747): [DBMaker] O modo TDE não está aberto” será retornada se os usuários tentarem acessar colunas criptografadas quando a criptografia da coluna estiver fechada.

```
dmSQL> CALL SETSYSTEMOPTION('tde_close','password');
dmSQL> ALTER TABLE t1 MODIFY c1 DECRYPT;
ERROR (6747): [DBMaker] TDE mode is not open

dmSQL> INSERT INTO t1 VALUES(1);

ERROR (6747): [DBMaker] TDE mode is not open



dmSQL> UPDATE t1 SET c1=2;

ERROR (6747): [DBMaker] TDE mode is not open



dmSQL> DELETE FROM t1 WHERE c1=2;

ERROR (6747): [DBMaker] TDE mode is not open
```

Domínio, gatilho, visualização e sinônimo com coluna criptografada ainda podem ser criados corretamente quando a criptografia da coluna estiver fechada. No entanto, a criptografia da coluna deve ser aberta quando os usuários desejarem usar esses elementos para criar tabelas ou selecionar dados.

Exemplo:

Suponha que o TDE esteja fechado.

Domínio:

```
dmSQL> CREATE DOMAIN d1 AS int ENCRYPT;

dmSQL> CREATE TABLE t1 (c1 d1);

ERROR (6747): [DBMaker] TDE mode is not open
```

Trigger:

```
dmSQL> DEF TABLE t1;
create table SYSADM.T1 (
C1 INTEGER default null encrypt)
in DEFTABLESPACE lock mode row fillfactor 80;

dmSQL> DEF TABLE t2;
create table SYSADM.T2 (
C1 INTEGER default null)
in DEFTABLESPACE lock mode row fillfactor 80;

dmSQL> CREATE TRIGGER tr1 AFTER INSERT ON t2 FOR EACH ROW (INSERT INTO t1
VALUES(new.c1));

dmSQL> INSERT INTO t2 VALUES(1);
ERROR (6747): [DBMaker] TDE mode is not open
```

View:

```
dmSQL> CREATE VIEW v1 AS SELECT * FROM t1;
dmSQL> SELECT * FROM v1;
ERROR (6747): [DBMaker] TDE mode is not open
```

Synonym:

```
dmSQL> CREATE SYNONYM s1 FOR t1;
dmSQL> SELECT * FROM s1;
ERROR (6747): [DBMaker] TDE mode is not open
```