Tipps & Tricks: Spalten verschlüsseln

Bereich:	DBA, Security	Erstellung:	01/2006 MP
Versionsinfo:	10.2, 11.1, 11.2	Letzte Überarbeitung:	06/2009 MP

Spalten verschlüsseln

Übersicht der Themen:

- Allgemeines
- Wallet konfigurieren
- Wallet administrieren
- Spalten verschlüsseln
- Beispiel

>> Allgemeines

Wenn Sie einen Export (Dump) der Datenbank durchführen und ihn nicht sicher lagern, kann es passieren, dass jemand ihn kopiert und auf einer fremden Maschine wieder einspielt. Viele Administratoren sind bis heute der Meinung, dass man dazu das SYS oder SYSTEM Passwort der Ursprungsdatenbank benötigt. Dies ist jedoch falsch, es reicht das Passwort der Zieldatenbank, auf der das Dump-File eingespielt wird. Dort kann man also durch Anlegen einer eigenen Instanz in den Besitz der Dump-Daten kommen.

Bei sensiblen Daten wie Kreditkarteninformationen erzeugt dies ein großes Sicherheitsrisiko. Abhilfe schafft die Möglichkeit die Spalten ab Version 10.2 verschlüsselt in den Tablespaces abzulegen. Die Datenbank legt das Master-Passwort für alle verschlüsselten Spalten in ein Wallet.

>> Wallet konfigurieren

- Tragen Sie in die Datei SQLNET.ORA einen gültigen Wallet-Pfad ein
- <ORACLE_HOME>/Network/Admin:

```
WALLET_LOCATION =
(SOURCE =
(METHOD = FILE) (
METHOD_DATA =
(DIRECTORY = /opt/oracle/product/10.2.0/db_1/network/admin )
)
)
```

>> Wallet administrieren

- Wallet erzeugen:

```
ALTER SYSTEM SET ENCRYPTION KEY ["certificate_id" ]
IDENTIFIED BY "pwd" ;
```

- Wallet öffnen(Alle verschlüsselten Spalten sind normal lesbar):

```
ALTER SYSTEM SET ENCRYPTION WALLET OPEN
IDENTIFIED BY "password";
```

- Wallet schließen (Damit kann keiner Daten aus verschlüsselten Spalten lesen):

```
ALTER SYSTEM SET ENCRYPTION WALLET CLOSE;
```

>> Spalten verschlüsseln:

Tabelle mit verschlüsselten Spalten anlegen:

```
CREATE TABLE kunden (
kd_id NUMBER NOT NULL,
last_name varchar2(30) not null,
SSN varchar2(9) ENCRYPT USING 'AES128',
card_id number ENCRYPT USING 'AES128' );
```

Nachträglich einschalten:

```
ALTER TABLE kunden MODIFY (card_id ENCRYPT);
```

Nachträglich ausschalten:

```
ALTER TABLE kunden MODIFY (card_id DECRYPT);
```

>> Beispiel:

Nun werden die Spalteninhalte nur angezeigt, wenn das Wallet mit dem richtigen Passwort geöffnet wurde:

```
SQL>    ALTER SYSTEM SET ENCRYPTION WALLET OPEN
       IDENTIFIED BY muniqsoft;
SQL>    INSERT INTO kunden VALUES (1,'Test','ABCDEFG',1000);
SQL>    SELECT * FROM kunden;

ACC_NO       LAST_NAME           SSN        CARD_ID
----------    ------------------    ---------    --------
1           Test                 ABCDEFG      1000

SQL>    ALTER SYSTEM SET ENCRYPTION WALLET CLOSE;
SQL>    select * from kunden;
```

```
SELECT * FROM kunden
*
FEHLER in Zeile 1: ORA-28365: Wallet ist nicht geöffnet
```

Hinweis: Beim Datapump Export muss unbedingt, der ab Version 10.2 neue Parameter ENCRYPTION_PASSWORD <pwd> gesetzt werden, damit die Spalten auch verschlüsselt im Export-File abgelegt werden!!!

Ab Version 11.1 kann ein kompletter Tablespace per Default auf ENCRYPTION gesetzt werden. Dies wird im Kurs: Neuerungen der Version 11g besprochen.
Weitere Informationen zum Thema Verschlüsselung erhalten Sie in unserem Oracle 10g Neuerungen oder Security Kurs, der auch die Inhalte von Release 10.2 abdeckt!