Tipps & Tricks: Daten maskieren (Redaction)

Bereich:	SQL	Erstellung:	07/2013 MP
Versionsinfo:	12.1	Letzte Überarbeitung:	07/2013 MP

Daten maskieren (Redaction)

Im folgenden Tipp zu Neuerungen in Oracle 12c beschäftigen wir uns mit der Verschleierung bzw. Maskierung von Daten über das Package DBMS_REDACT.

Beachten Sie bitte, dass diese Funktion nur in der Enterprise Edition mit Advanced Security Option verwendet werden darf.
Ein Beispiel, das sich dazu nahezu aufdrängt, sind Kreditkartennummern, die nicht in ihrer vollen Schönheit in einem Report ausgegeben werden sollen. Wir erstellen also eine kleine Tabelle für die Kreditkarteninfos:

```
CREATE TABLE scott.credit_card(cust_name VARCHAR2(64), card_id VARCHAR2(64));
INSERT INTO scott.credit_card VALUES ('Marco','1234-1234-1234-1234');
INSERT INTO scott.credit_card VALUES ('Hans','5678-5678-5678-5678');
commit;
```

Zuständig für die Verschleierung der Daten ist ein neues Package mit Namen DBMS_REDACT.
Der Benutzer, der die sogenannten Redaction Rules verwaltet, benötigt das Execute-Recht an diesem Package.

```
GRANT EXECUTE ON DBMS_REDACT TO scott;
```

Alle Benutzer, die die Daten unmaskiert sehen dürfen (also z. B. der Chef :-)), können die Redaction komplett umgehen mit:

```
GRANT EXEMPT REDACTION POLICY TO chef;
```

Beispiel: Daten der Spalte durch einen regulären Ausdruck verschleiern:

Das REGEXP-Muster greift hier auf alle Strings, die 4 vierstellige Zahlengruppen, jeweils getrennt durch einen Bindestrich enthalten. Diese Muster wird ersetzt durch die Folge XXX-XX- gefolgt von der 3. Zahlengruppe der Kreditkartennummer.

```
BEGIN
  DBMS_REDACT.ADD_POLICY(
         OBJECT_SCHEMA => 'SCOTT',
           OBJECT_NAME => 'CREDIT_CARD',
           COLUMN_NAME => 'CARD_ID',
           POLICY_NAME => 'MASK_CREDIT_CARD_CARD_ID',
         FUNCTION_TYPE => DBMS_REDACT.REGEXP,
            EXPRESSION => '1=1',
        REGEXP_PATTERN => '(\d{4})-(\d{4})-(\d{4})-(\d{4})',
 REGEXP_REPLACE_STRING => 'XXX-XX-\3',
       REGEXP_POSITION => 1,
     REGEXP_OCCURRENCE => 0,
```

```
 REGEXP_MATCH_PARAMETER => 'ic');
END;
/

SELECT * FROM scott.kredit_card;
CUST_NAME  CARD_ID
---------- --------------------
Marco      XXX-XX-1234
Marco      XXX-XX-5678
```

Als `FUNCTION_TYPE` können Sie setzen:

`0 NONE` (Keine Verschleierung)
`1 FULL` (Feste Werte)
`2 PARTIAL` (Teile der Spalte verschleiern)
`4 RANDOM` (Jede Abfrage wird mit zufälligen Werten beantwortet)
`5 REGEXP` (Verschleierung basierend auf Regulären Ausdrücken)

Also testen wir das Ganze einmal mit einer anderen Option (Zufallsausgabe):

```
BEGIN
 DBMS_REDACT.ADD_POLICY(
    OBJECT_SCHEMA   => 'SCOTT',
    OBJECT_NAME     => 'CREDIT_CARD',
    COLUMN_NAME     => 'CARD_ID',
    POLICY_NAME     => 'MASK_CREDIT_CARD_CARD_ID',
    FUNCTION_TYPE   => DBMS_REDACT.RANDOM,
    EXPRESSION      => '1=1');
END;
/

SELECT * FROM scott.kredit_card;

CUST_NAME  CARD_ID
---------- --------------------
Marco      o(X8}ZJzw0`@&bR# 8h
Marco      :#&[K<QIB+.cSDYQJ36
```

Wenn Sie die Expression Klausel ändern, kann die Regel nur für bestimmte Benutzer ausgeführt werden. So wird nur der Benutzer SCOTT die veränderten Spalteninhalte sehen, alle anderen bekommen die Spalten im Original.

```
EXPRESSION => q'!sys_context('USERENV','SESSION_USER')='SCOTT' !'
```

Einige Prüfungen sind schon vordefiniert, jedoch machen in unseren Breitengraden nicht alle Sinn. So wird bei uns ein anderes Sozialversicherungsnummer-Format verwendet als in den USA.

Sinnvolle Prüfungsvarianten (Auswahl)
`DBMS_REDACT.REDACT_ZIP_CODE` (5stellige Postleitzahl in VARCHAR2 Spalte wird durch XXXXX ersetzt)
`DBMS_REDACT.REDACT_NUM_ZIP_CODE` (5stellige Postleitzahl in NUMBER Spalte wird durch XXXXX ersetzt)
`DBMS_REDACT.REDACT_DATE_MILLENNIUM` (Datum wird ersetzt durch 01-JAN-00)
`DBMS_REDACT.REDACT_DATE_EPOCH` (Datum wird ersetzt durch 01-JAN-70)

`DBMS_REDACT.REDACT_CCN16_F12` (ersetzt eine 16stellige Kreditkartenummer in der Form 1234123412341234 durch XXXXXXXXXXXX1234)

Bei den Verwaltungsviews hat sich Oracle mal etwas Neues einfallen lassen.
Die Views heißen:

```
REDACTION_COLUMNS
REDACTION_POLICIES
REDACTION_VALUES_FOR_TYPE_FULL
```

Ja, Sie haben richtig gelesen, da steht kein USER_, ALL_ bzw. DBA_ davor :-)

Die drei Views sind an die SELECT_CATALOG_ROLE gebunden.

Anmerkungen:
Die Benutzer SYS/SYSTEM haben automatisch das Recht `EXEMPT REDACTION POLICY` und können damit jede existierende Redaction Regel umgehen und die Daten im Original sehen.
Data Redaction Policies dürfen nicht auf SYS Objekte definiert werden.

Sie sehen, das ist ein weiteres hochspannendes Thema von Oracle 12c, das wir in unserem Oracle Kurs Neuerungen 12c vertiefen können.

Für das Selbststudium ist auch die Oracle Doku empfehlenswert:
http://docs.oracle.com/cd/E16655_01/network.121/e17729/redaction_config.htm#ASOAG654