特集3 ここがポイント! Oracle DB 開発

Oracle meets .NET

第2回

見せたくないデータは隠せ

暗号化を極める

大田 浩
OTA, Hiroshi
日本オラクル株式会社
Oracle Direct テクニカルサービス部

Level

1 2 3 **4** 5

Technology Tools

- [x] Visual Basic
- [x] Visual C#
- [ ] Visual C++
- [ ] SQL Server
- [ ] Oracle
- [ ] Access
- [ ] ASP.NET
- [x] Other:
  Visual Studio .NET 2003
  Oracle 9i、10g

Samples

はじめに

今月は“暗号化”について説明します。

そもそも、アプリケーション側で認証やアクセス制御を実装し、セキュアなアプリケーションを開発したとしても、あの手この手でセキュリティが破られたり、あるいはそもそもデータベース管理者の不正アクセスにより、データが取得されたり改竄されたりする可能性は否定できません。もしくは、バックアップデータの入ったメディアを紛失してしまうなどの人的なトラブルに見舞われたりすることもあります。

そのようなトラブルに対処するためには、格納データそのものを保護する必要があります。格納データを保護するためのもっとも有力な手法は暗号化です。

そこで今回は、.NETアプリケーションからOracleの暗号化機能を利用するための方法を紹介します。

▶格納データを暗号化

暗号化はデータにアクセスされても情報を盗まれないようにするための機能です。アクセスそのものを防止する機能ではありません。暗号化することにより、以下のようなデータの保護が可能になります。

- データを列レベルで暗号化して格納
- データベース管理者からも秘匿可能
- データファイルが窃取された場合にも有効
  - サーバーからデータファイルをコピーされる
  - バックアップメディアを盗まれる

暗号化を行なうことにより、データの保護が実現できますが、格納されているすべてのデータを暗号化するのは現実的ではありません。データの暗号化／復号はCPUリソースの消費が多く、パフォーマンスに対する影響が高いため、利用するか否かは格納されている情報資産の価値や機密性の高さに応じて検討することになります。

たとえば、図1のような顧客データに

図1：暗号化ポリシーの策定

対しては、特に重要なクレジットカード番号のみ暗号化するといった、ポリシー策定があげられます。

暗号化の手段

暗号化のために用意されているOracleのPL/SQLパッケージには、

- DBMS_OBFUSCATION_TOOLKITパッケージ（8.1.6EE～、9iからはStandard Editionでも可）
- DBMS_CRYPTOパッケージ（10gの新パッケージ）

の2種類が用意されています。

Oracle Database 10gの新パッケージである、DBMS_CRYPTOパッケージは、DBMS_OBFUSCATION_TOOLKITパッケージに比べ多機能になっており、また暗号化アルゴリズムの種類が増えています（表1）。

今回は10gの新パッケージである、DBMS_CRYPTOパッケージを使用して、データを暗号化することにしましょう。

暗号化／復号のサンプル作成

サンプル作成の準備

データ暗号化のテストを行なう準備として、テスト用のテーブルを作成します。次に、そのテスト用のテーブルに対して、

①データの挿入時にデータの暗号化
②データの取得時に暗号化されたデータの復元

という操作を行なってみます。

テスト用テーブル作成

今回のテストを実行するためのテーブルを作成します。今回はSCOTTユーザーでテストします。SQL*PLUSからリスト1のコマンドを実行してください。暗号化対象列である「credit_id」は、RAW型で宣言します。

リスト1：テスト用スキーマの作成

```
SQLPLUS scott/tiger
SQL> CREATE TABLE customer(
  2   customer_id char(5) not null,
  3   customer_name varchar2(30),
  4   address varchar2(100),
  5   credit_id raw(20),
  6   PRIMARY KEY(customer_id)
  7 );
```

DBMS_CRYPTOパッケージに実行権限を付与

DBMS_CRYPTOパッケージは、デフォルトで「SYSスキーマ」にインストールされます。このパッケージを実行したいユーザーないしロールに、パッケージの実行権限を付与します。今回は以下のように、SCOTTユーザーにDBMS_CRYPTOパッケージの実行権限を付与しました。

```
SQLPLUS / as sysdba
SQL> -- 権限の付与
SQL> GRANT EXECUTE ON dbms_crypto TO scott;
```

暗号化／復号のためのファンクション作成

本稿でのテストはデータ暗号化規格（DES）を使用して、データを暗号化します。

暗号化の際には、DBMS_CRYPTOパッケージの、「ENCRYPT」ファンクションを使用し、逆に復号の際には「DECRYPT」ファンクションを使用します。両ファンクションとも、暗号化もしくは復号対象データと暗号化キーの値をRAW型で取得します（リスト2）。

また、RAW型は直接文字列で値を指定できないので、文字列からRAW型への変換には、UTL_I18Nパッケージの

表1：暗号化パッケージの比較

	DBMS_CRYPTO	DBMS_OBFUSCATION_TOOLKIT
暗号化アルゴリズム	DES、Triple-DES（2KEY／3KEY）、AES、RC4	DES、Triple-DES（2KEY／3KEY）
パディング方式	PKCS5、0（ゼロ）	なし
ブロック暗号連鎖モード	CBC、CFB、ECB、OFB	CBC
ハッシュアルゴリズム	MD5、SHA-1、MD4	MD5
ハッシュアルゴリズム（キーを必要とする）	HMAC_MD5、HMAC_SH1	なし
暗号化対象データ型	BLOB／CLOB／RAW	VARCHAR2／RAW