

Oracle Advanced Security テクニカル・ホワイト・ ペーパー

Oracle ホワイト・ペーパー
2007年6月

ご注意

本書は、オラクルの一般的な製品の方向性を示すことが目的です。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。下記の事項は、マテリアルやコード、機能の提供を確約するものではなく、また、購買を決定する際の判断材料とはなりません。オラクルの製品に関して記載されている機能の開発、リリース、および時期については、弊社の裁量により決定いたします。

Oracle Advanced Security 11g テクニカル・ホワイト・ペーパー

ご注意	2
はじめに	5
Oracle Databaseの暗号化の概要	5
ORACLE TRANSPARENT DATA ENCRYPTION	6
Transparent Data Encryption Benefits.....	7
暗号鍵管理の概要	7
実装手順	8
マスター鍵の初期化	8
Oracleウォレットを開く	8
マスター鍵の変更	9
ウォレットのパスワードの変更.....	9
機密データの特定	9
Oracle TDEのデータ型サポートの確認	9
外部鍵使用のチェック	9
Oracle TDEによるデータの暗号化	10
表/列の暗号鍵の変更	10
初めてデータを暗号化する場合のベスト・プラクティス	10
Oracle RMANによるバックアップ暗号化	11
Oracle RMANの暗号化モード	11
バックアップの透過的暗号化	11
パスワード暗号化バックアップ.....	12
デュアル・モード暗号化バックアップ.....	12
ORACLE TRANSPARENT DATA ENCRYPTIONの機能拡張	12
データ型サポートの追加	12
表領域暗号化	12

ハードウェア・セキュリティ・モジュール (HSM) マスター鍵保護	13
Oracle Data Pumpの暗号化.....	13
Oracle Streamsおよびロジカル・スタンバイのサポート	13
ネットワークの暗号化	14
業界標準暗号化およびデータ整合性.....	14
SSL	14
JDBCセキュリティ	15
容易な設定、アプリケーションの変更なし.....	15
Oracle Database 10gの厳密認証サービス	15
Kerberos認証.....	16
Kerberosの機能拡張.....	16
PKIのサポート.....	16
PKCS#12 のサポート	16
PKCS#11 のサポート、スマート・カード/ハードウェア・セキュリティ・モジュール	17
Oracle Database 10gエンタープライズ・ユーザー用のPKI認証	17
Oracle Internet Directoryに格納されたウォレット	17
複数の証明書のサポート	17
強力なウォレット暗号化	18
RADIUS (Remote Dial-in User Service)	18
まとめ	18

Oracle Advanced Security 11g テクニカル・ホワイト・ペーパー

はじめに

今日のグローバルなビジネス環境における経営は、セキュリティおよびコンプライアンスに関する非常に多くの課題を生み出しています。アウトソーシングを活用する際は、知的財産やプライバシー関連情報の十分な保護を伴う必要があります。近年、個人情報の盗難やクレジットカード詐欺の事件が多数発生しており、その被害総額は数千万ドルに達しています。このような脅威を防ぐには、意図的に透過性を持たせたセキュリティ・ソリューションが必要です。大学や健康管理機構は社会保障番号などの個人情報（PII）関連のセキュリティを厳重にし、小売業者はクレジットカード業界データ・セキュリティ法（PCI-DSS）への対応を進めています。Oracle Advanced Security は、ネットワーク、ディスク、およびバックアップ・メディア上のデータを保護する透過的な標準ベースのセキュリティを提供します。

Oracle Database の暗号化の概要

暗号化は多層防御の主要コンポーネントであり、転送データや保存データの保護にとって重要です。オラクルでは、Oracle8i Database において初めてデータベース暗号化 API を導入しました。Oracle データベースの暗号化 API は、機密アプリケーション・データを暗号化するために現在多数のユーザーによって使用されています。暗号化 API 使用時の透過性を実現するには、アプリケーション自体にファンクション・コールを組み込むか、事前に挿入されたデータベース・トリガーを使用する必要があります。アプリケーションがデータを受け取るためにアプリケーション・ビューによってはデータの復号化が必要な場合もあります。さらに、暗号鍵の管理はプログラムによって実行される必要があります。

Oracle Database 10g Release 2 で初めて導入された Oracle Advanced Security の Transparent Data Encryption (TDE) は、業界でもっとも高度な暗号化ソリューションです。Oracle TDE は、組込み暗号鍵管理と、機密アプリケーション・データの暗号化の完全な透過性を提供します。データベース暗号化プロセスは DDL コマンドを使用して開始され、アプリケーションの変更、プログラムによる鍵管理、データベース・トリガー、ビューをまったく必要としません。

パッケージ機能	DBMS OBFUSCATION TOOLKIT (Oracle8i Database 以降) SE および EE	DBMS CRYPTO (Oracle Database 10g R1 以降) SE および EE	Oracle Advanced Security Transparent Data Encryption EE のみのオプション
暗号化アルゴリズム	DES、3DES	DES、DES、AES、RC4、3DES_2KEY(1)	3DES、AES (128、192、および 256 ビット)
パディング方式	サポートなし	PKCS5、0 (ゼロ)	PKCS5(2)
ブロック暗号連鎖モード	CBC	CBC、CFB、ECB、OFB	CBC(2)
暗号化ハッシュ・アルゴリズム	MD5	SHA-1、MD4(1)、MD5(1)	SHA-1(2)
鍵付きハッシュ (MAC) アルゴリズム	サポートなし	HMAC_MD5、HMAC_SH1	該当なし
暗号化擬似乱数ジェネレータ	RAW、VARCHAR2	RAW、NUMBER、BINARY_INTEGER	該当なし
データベース型	RAW、VARCHAR2	RAW、CLOB、BLOB	OBJ、ADT、LOB 以外のすべて

1) 下位互換性のために提供されています。

2) 内部で使用されます。開発者が利用することはできません。

表 1. Oracle Database の暗号化の概要

ORACLE TRANSPARENT DATA ENCRYPTION

Oracle TDE は、データがディスクに書き込まれる前にデータを暗号化し、データがアプリケーションに返される前にデータを復号化します。暗号化および復号化プロセスは、アプリケーションおよびユーザーに対して完全に透過的に、SQL レイヤーで実行されます。ディスクまたはテーブルデータベース・ファイルがバックアップされる際、機密アプリケーション・データは暗号化されます。オプションとして、Oracle TDE を Oracle RMAN と併用して、ディスクへのバックアップ時に Oracle データベース全体を暗号化することもできます。

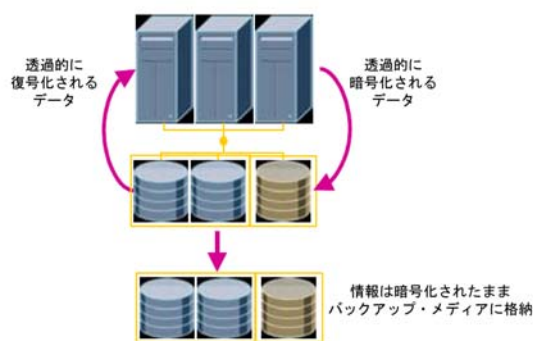


図 1. Oracle Transparent Data Encryption の概要

Transparent Data Encryption の利点

1. 組込み暗号鍵管理
2. 機密アプリケーション列の透過的暗号化
3. 表領域全体の透過的暗号化 (11g の新機能)
4. SecureFiles/LOBS の透過的暗号化 (11g の新機能)
5. ハードウェア・セキュリティ・モジュール (HSM) の統合 (11g の新機能)

暗号鍵管理の概要

Oracle TDE は、アプリケーション表列が暗号化される際に自動的に暗号鍵を作成します。暗号鍵は、表に対して固有です。1 つの表の複数の列が暗号化される場合、すべての列に同じ暗号鍵が使用されます。各表の暗号鍵は Oracle データ・ディクショナリに格納され、Oracle TDE のマスター暗号鍵を使用して暗号化されます。マスター暗号鍵は、データベースの外にある Oracle ウォレットの PKCS#12 フォーマット済みファイルに格納されます。このファイルは、セットアップ時に設定されたセキュリティ管理者または DBA によって指定されるパスワードを使用して暗号化されます。Oracle Database 11g Advanced Security は、PKCS#11 インタフェースを使用してマスター鍵を HSM デバイスに格納できる新機能を備えています。

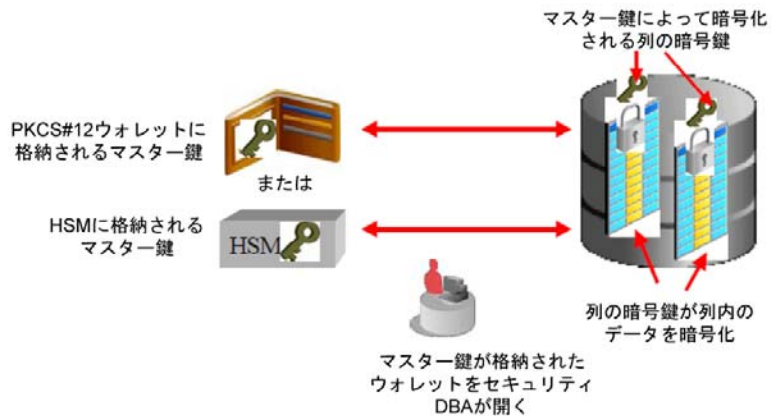


図 2. Oracle TDE の暗号鍵管理アーキテクチャ

実装手順

Oracle TDE は既存のアプリケーション・コードに対して透過的（データベース・トリガーおよびビューが不要）であるため、暗号化プロセスは従来の API ベースの暗号化ソリューションよりも単純です。次の手順で、Oracle TDE を適用できます。

1. マスター鍵を初期化します。
2. 暗号化する機密データ（PII データ、クレジットカード）を特定します。
3. Oracle TDE がデータ型をサポートすることを確認し、外部鍵の使用をチェックします。
4. Oracle TDE を使用して機密データを暗号化します。

マスター鍵の初期化

マスター鍵は、各データベースに対して固有です。ただし、マスター鍵がセカンダリ・データベース用にまだ確立されていないかぎり、どのマスター鍵もセカンダリ・データベースにコピーできます。マスター鍵を作成しないと、どのアプリケーション表も暗号化できません。マスター鍵を初期化する構文は、次のとおりです。

```
SQL> alter system set key identified by "password";
```

このコマンドによりウォレットが作成され、PKCS#5 標準の推奨事項に基づいてパスワードを使用して、ウォレットが暗号化されます。Oracle ウォレットには、使用されなくなったマスター暗号鍵の履歴が格納され、古い暗号鍵で暗号化されたデータをバックアップ・テープから読み出す際にはそれらの鍵を利用できます。

Oracle ウォレットを開く

マスター暗号鍵が格納されたウォレットを開かないと、データベースは表の暗号鍵を復号化してアプリケーション・データを暗号化または復号化することができません。データベースは、ウォレットが開いていなくても稼働しつづけることが

できます。ただし、暗号化されたデータにアクセスしようとすると、エラーが返されます。メンテナンス操作時にサポート担当者にデータベースへのアクセスを許可しなければならないときは、ウォレットを閉じると便利な場合があります。

マスター鍵の変更

マスター鍵は、alter system コマンドを再発行することによって変更できます。

```
SQL> alter system set key identified by "password";
```

マスター鍵を変更すると、Oracle データ・ディクショナリに格納された表の暗号鍵がすべて再暗号化されます。PCI データ・セキュリティ標準 (DSS) 1.1 は、「暗号鍵を頻繁に (少なくとも年 1 回) 更新すること」を要求します。マスター鍵を変更すると、新しいマスター暗号鍵を使用して列の暗号鍵が再暗号化されます。暗号化されているデータは、変更されません。

ウォレットのパスワードの変更

ウォレットのパスワードは、マスター暗号鍵とは無関係に変更できます。このパスワードは、ディスク上のウォレット・ファイルを暗号化するためにのみ使用されます。Oracle Wallet Manager (コマンド・ライン'owm'で起動) またはコマンド・ライン'orapki'を使用します。

機密データの特定

社会保障番号やクレジットカードなどの個人情報関連データの特定は、特に複雑なアプリケーションにおいては困難である場合があります。役に立つ可能性がある方法のひとつは、そのような情報を格納するためによく使用される列名およびデータ型を Oracle データ・ディクショナリで検索することです。

```
SQL> select column_name, table_name, data_type from
       dba_tab_cols where column_name like '%SOCIAL%' or
       column_name like '%SSN%' or column_name like '%SECNUM%' or
       column_name like "%SOC%" and owner='<owner>';
```

Oracle TDEのデータ型サポートの確認

Oracle TDE は、データベースで使用されるもっとも一般的なデータ型をサポートしています。これには、次のデータ型が含まれます。

VARCHAR2	CHAR	DATE
NUMBER	NVARCHAR2	NCHAR
RAW	RAW	SECUREFILES (LOBS)
BINARY_DOUBLE	BINARY_FLOAT	

外部鍵使用のチェック

Oracle TDE では、外部鍵で使用される列を暗号化することはできません。列が外部鍵の一部として使用されているかどうかを確認するには、Oracle データ・ディクショナリを調べます。

```
select A.owner, A.table_name, A.column_name, A.constraint_name from
       dba_cons_columns A, dba_constraints B
```

```
where A.table_name = B.table_name and
A.column_name = 'YOURCOLNAME' and
B.constraint_type = 'R';
```

Oracle TDE によるデータの暗号化

既存の列を暗号化するには、次のコマンドを発行します。

```
SQL> alter table customers modify (credit_card encrypt);
```

暗号化トランザクションの実行中は、読取り一貫性が維持されるので、選択（読み取り）操作を継続できます。暗号化トランザクションの実行中に実行された DML トランザクション（挿入、更新、削除）は、'オンライン再定義'を要求します。

暗号化された列で新しい表を作成することは簡単です。デフォルトの暗号化アルゴリズムは、AES192 です。

```
SQL> create table billing_information (
first_name varchar2(40)
,last_name varchar2(40)
,card_number varchar2(19) encrypt using 'AES256');
```

Oracle Transparent Data Encryption は等価検索に索引を使用するので、暗号化された列の検索のオーバーヘッドが最小限に抑えられます。

```
SQL> create index cust_idx on customers (credit_card);
```

索引付けされた列を暗号化する場合は、まず既存の索引を外してから列を暗号化し、最後に索引を再構築することを推奨します。

表/列の暗号鍵の変更

表や列の暗号鍵、鍵のサイズ、およびアルゴリズムは、ALTER TABLE コマンドを発行することによって、別々に変更できます。

```
SQL> ALTER TABLE employee REKEY;
SQL> ALTER TABLE employee REKEY USING 'AES256';
SQL> ALTER TABLE employee ENCRYPT USING 'AES128';
```

表や列の暗号鍵を変更すると、表に格納されている暗号化されたデータがすべて再暗号化されます。

初めてデータを暗号化する場合のベスト・プラクティス

表のライフタイム中に、データは、断片化、再配置、ソート、コピー、表領域内で移動などの処理が行われます。この結果、データのアクセス不能な古いコピーがデータベース・ファイル内の未使用データ・ブロックに存在することになる場合があります。既存の列を暗号化する場合、最新の'有効な'コピーだけが暗号化され、古いクリアテキストのバージョンが取り残される可能性があります。暗号化の前に作成された古いクリアテキストのコピーに関するリスクを最小限に抑えるために、新しい表領域を作成し、アプリケーション表を新しい表領域に移動させ、古い表領域を削除することを推奨します。

1. データベースを完全にバックアップします。
2. 新しいデータ・ファイルを指定して新しい表領域を作成します。
3. 元の表の機密列を暗号化します。

4. 機密列を含むすべての表について、手順 3 を繰り返します。
5. 元の表領域から新しい表領域に表を移動させます。
6. データ・ファイル・オプションを使用して元の表領域を削除します。オプションとして、'DROP TABLESPACE'コマンドで'... WITH DATAFILE'オプションを使用せずに、'SHRED'またはその他のプラットフォーム固有のコマンドを使用して、オペレーティング・システム上の古いデータ・ファイルを安全に削除することもできます。

'SHRED'などのオペレーティング・システム・コマンドを使用すると、オペレーティング・システムまたはストレージ・ファームウェアによって生成されるデータベース・ファイルのゴースト・コピーを検出できる可能性が低下します。

Oracle RMAN によるバックアップ暗号化

セキュリティを強化するために、バックアップ・セットとして作成される RMAN バックアップを Oracle Advanced Security によって暗号化できます。暗号化されたバックアップは、許可されていないユーザーが入手しても読み取ることができません。バックアップ・セットであるどの RMAN バックアップも暗号化できます。ただし、イメージ・コピー・バックアップは暗号化できません。

暗号化されたバックアップは、ユーザーが入力するパスワードまたは Oracle Encryption Wallet によって必要な復号鍵が提供される場合に、リストアおよびリカバリ操作時に自動的に復号化されます。

Oracle RMAN の暗号化を使用するには、ターゲット・データベースの COMPATIBLE 初期化パラメータを少なくとも 10.2.0 に設定する必要があります。

`backup backupset` コマンドを暗号化されたバックアップ・セットとともに使用すると、バックアップ・セットが、暗号化された形式でバックアップされます。`backup backupset` は、すでに暗号化されたバックアップ・セットをディスクまたはテープにコピーするだけなので、`backup backupset` 操作時に復元鍵は必要なく、操作中にデータが復元されることはありません。`backup backupset` コマンドでは、バックアップ・セットを暗号化することも復号化することもできません。

Oracle Transparent Data Encryption を使用してデータベースの一部の列が暗号化されている場合、バックアップ暗号化を使用してこれらの列をバックアップすると、これらの列はバックアップ時に 2 度暗号化されます。リストア時にバックアップ・セットが復号化されると、暗号化された列は、元の暗号化された形式に戻ります。暗号化アルゴリズムが指定されない場合のデフォルトの暗号化アルゴリズムは、128 ビット AES です。

Oracle RMAN の暗号化モード

Oracle RMAN は、透過モード、パスワード・モード、およびデュアル・モードの 3 つの暗号化モードを提供します。透過モードとデュアル・モードは、Oracle Encryption Wallet に基づきます。

バックアップの透過的暗号化

透過的暗号化では、必要な Oracle 暗号鍵管理インフラストラクチャを利用できる場合、DBA による操作なしに、暗号化されたバックアップを作成およびリストアできます。透過的暗号化は、バックアップ元のデータベースにバックアップがリ

ストアされる毎日のバックアップ操作に最適です。透過的暗号化は、RMAN 暗号化のデフォルト・モードです。

透過的暗号化を使用する場合は、Oracle の Oracle Transparent Data Encryption 機能に関するドキュメントで説明されているように、まず Oracle Encryption Wallet を設定する必要があります。Oracle Encryption Wallet の設定後は、DBA による操作なしに、暗号化バックアップを作成およびリストアできます。

パスワード暗号化バックアップ

パスワード暗号化では、暗号化バックアップの作成およびリストア時に DBA がパスワードを入力する必要があります。パスワード暗号化バックアップをリストアするには、バックアップの作成時に使用したパスワードが必要です。パスワード暗号化は、遠隔地でバックアップをリストアする場合に、転送時の安全を確保するのに便利です。パスワード暗号化を永続的に設定することはできません。パスワード暗号化を排他的に使用する場合、Oracle Encryption Wallet の設定は不要です。パスワード暗号化バックアップの暗号化に使用したパスワードを忘れてたりなくしたりすると、バックアップをリストアできません。

パスワード暗号化を使用するには、RMAN スクリプトで、SET ENCRYPTION ON IDENTIFIED BY passwordONLY コマンドを使用します。

デュアル・モード暗号化バックアップ

デュアル・モード暗号化バックアップは、透過的にリストアするかパスワードを指定してリストアすることができます。デュアル・モード暗号化バックアップは、通常は Oracle Encryption Wallet を使用してオンサイトでリストアするものの場合によっては Oracle Encryption Wallet を利用できないオフサイトでリストアしなければならないバックアップを作成するようなどに便利です。

デュアル・モード暗号化バックアップをリストアする場合は、Oracle Encryption Wallet が復号化用のパスワードを使用できます。

デュアル・モード暗号化バックアップの暗号化に使用したパスワードを忘れてたりなくしたりした場合、Oracle Encryption Wallet も喪失すると、バックアップをリストアできません。

デュアル・モード暗号化バックアップ・セットを作成するには、RMAN スクリプトで、SET ENCRYPTION ON IDENTIFIED BY password コマンドを使用します。

ORACLE TRANSPARENT DATA ENCRYPTION の機能拡張

Oracle Database 11g Advanced Security の Transparent Data Encryption では、いくつかの重要な機能拡張が実施されています。

データ型サポートの追加

Oracle Database 11g Advanced Security の Transparent Data Encryption では、新しい Oracle Database 11g SecureFiles を暗号化できます。これにより、データベースに格納されているスキャンされた医療画像、契約書、その他の機密性の高いドキュメントを透過的に暗号化することができます。

表領域暗号化

Oracle Database 11g Advanced Security の Transparent Data Encryption では、データ

ベース表領域全体を暗号化できるようになりました。表領域の暗号化により、アプリケーション表全体を透過的に暗号化できます。データ・ブロックは、データベースによってアクセスされる際に透過的に復号化されます。

新しい表領域のみを暗号化できます。

```
SQL> CREATE TABLESPACE securespace
DATAFILE '/home/user/oradata/secure01.dbf' SIZE 150M
ENCRYPTION USING 'AES192' DEFAULT STORAGE(ENCRYPT);
```

新しい表領域で作成されるすべてのデータベース・オブジェクトが暗号化されます。表領域暗号化では、列暗号化の外部鍵の制限がなく、暗号化データの範囲スキャンが可能です。

ハードウェア・セキュリティ・モジュール (HSM) マスター鍵保護

セキュリティをより強化するために、マスター暗号鍵を PKCS#11 準拠 HSM デバイスに格納できます。また、これにより、同じ暗号化データを共有する RAC 環境内の複数のデータベースまたはデータベース・インスタンスで同じマスター暗号鍵を共有できます。オラクルは PKCS#11 標準に準拠しているため、ユーザーはさまざまな HSM プロバイダを選択できます。

Oracle 10g R2 の TDE による列レベルの暗号化を Oracle 11g R1 の表領域暗号化にアップグレードする際は、暗号化列用の新しいマスター暗号鍵と表領域暗号化用の新しいマスター暗号鍵を生成する再作成操作を実行する必要があります。

Oracle ウォレットに格納されるマスター暗号鍵から HSM デバイスに格納されるマスター暗号鍵にアップグレードする場合は、次の手順を実行してください。

- 1.) マスター暗号鍵を再作成して、後で必要になった場合のために表領域暗号鍵を生成しておきます。
- 2.) sqlnet.ora ファイルを次のように変更します。

```
ENCRYPTION_WALLET_LOCATION=
SOURCE=(METHOD=HSM)
```
- 3.)

```
SQL> alter set key identified by "<user_id:password>" migrate from
<wallet password>
<user_id:password>
```

 は、HSM デバイスの認証情報です。

Oracle Data Pump の暗号化

Oracle Transparent Data Encryption を使用して、Oracle Data Pump の出力コンテンツ全体を暗号化できます。これは、Oracle Database 11g の新機能です。詳しくは、Oracle Data Pump のドキュメントを参照してください。

Oracle Streams およびロジカル・スタンバイのサポート

Oracle Database 11g Advanced Security の TDE は、Oracle Streams およびロジカル・スタンバイ・データベースをサポートしています。Oracle Database 10g Release 2 Advanced Security の TDE では、フィジカル・スタンバイのみがサポートされていました。

ネットワークの暗号化

Oracle Advanced Security は、データ・スニффイング、データ消失、リプレイ攻撃、PIM (Person-In-the-Middle) 攻撃などに対処することにより、ネットワーク上でのデータのプライバシーと機密を保護します。Oracle Database とのすべての通信は、Oracle Advanced Security で暗号化できます。データベースには非常に機密性の高い情報が含まれており、厳密認証によるアクセスの制限は防衛の最前線のひとつです。Oracle Advanced Security は、Kerberos、公開鍵暗号化、RADIUS、Oracle Database 10g 用 DCE など、ビジネスの既存のセキュリティ・インフラストラクチャを活用する厳密認証ソリューションを提供します。

業界標準暗号化およびデータ整合性

Oracle Advanced Security は、Oracle Database とのすべての通信を保護します。ビジネスに応じて、ネットワークを介するデータ保護の方法としては、Oracle Advanced Security のネイティブ暗号化/データ整合性アルゴリズムか、SSL の使用を選択できます。次に、ネットワーク・レベルの暗号化が必要な代表的な使用例を示します。

- データベース・サーバーはファイアウォールの背後にあるため、ユーザーはクライアント・サーバー・アプリケーションを介してサーバーにアクセスします。
- DMZ 内のアプリケーション・サーバーと 2 番目のファイアウォールの背後にあるデータベースの間の通信は、暗号化が必要です。

Oracle Advanced Security のネイティブ暗号化/データ整合性アルゴリズムは、PKI 配置が不要です。データベースの今後の各リリースでは、さらに新しい業界承認済の暗号化アルゴリズムが実装される予定です。最新の追加は Advanced Encryption Standard (AES) です。これは、DES 上でのセキュリティとパフォーマンスを向上させるアルゴリズムです。暗号化/データ整合性アルゴリズムの完全なリストを次に示します。

- AES (128、192、256 ビット)
- RC4 (40、56、128、256 ビット)
- 3DES (2 鍵および 3 鍵、168 ビット)
- MD5
- SHA1

SSL ベースの暗号化は、公開鍵インフラストラクチャの配置を選択した企業で使用できます。TLS 1.0 プロトコルに対するサポートは、Oracle Advanced Security 10g のリリースで導入されました。Oracle Advanced Security は、Oracle Database 10g の TLS 1.0 プロトコルで AES Cipher Suite の提供を開始しました。

SSL

オラクルは、データベース・クライアントとデータベースの間で交換されるデータの暗号化のために SSL プロトコルを実装しています。これには、Oracle Net Services (以前の Net8)、LDAP、シック JDBC、および IIOP 形式のデータが含まれます。SSL 暗号化は、Oracle7 以降の Oracle Advanced Security (以前の Advanced Networking Option) でサポートされてきたネイティブ Oracle Net Services 暗号化プロトコルに代わる方法として選択できます。事実上のインターネット標準であり、

Oracle Net Services 以外のプロトコルを使用するクライアントで使用できることが SSL の利点です。

3 階層システムでは、データベースでの SSL のサポートにより、中間層とデータベースの間で交換されるデータを SSL で暗号化できます。SSL プロトコルは、ユーザーからの信頼を得ており、おそらく現在使用されている暗号化プロトコルでは、もっとも広く配備されもっともよく理解されています。Oracle の SSL サポートは、匿名 (Diffie-Hellman)、X.509 証明書によるサーバーのみの認証、X.509 による相互 (クライアント-サーバー) 認証の 3 つの標準認証モードをサポートしています。

また、Oracle Application Server は、Oracle Application Server と Oracle Data Server の間だけではなく、シン・クライアントと Oracle Application Server の間の SSL 暗号化もサポートしています。Oracle における場合と同様に、匿名、サーバーのみ、および X.509 によるクライアント-サーバー認証がサポートされます。

JDBC セキュリティ

JDBC は、Java プログラムからリレーショナル・データベースに接続するための Java 標準を提供する業界標準 Java インタフェースです。オラクルは、個別プロバイダとして、Sun Microsystems が定義した JDBC 標準を実装し、独自の JDBC ドライバでこの標準を拡張しています。Oracle は、2 種類の JDBC ドライバを実装しています。ひとつは C ベースの Oracle Net Services クライアントに組み込まれるシック JDBC ドライバ、もうひとつはダウンロード可能なアプレットをサポートするシン (Pure Java) JDBC ドライバです。

シック JDBC はクライアントとサーバーの両方で Oracle Net Services 通信スタック全体を使用するので、既存の Oracle Advanced Security 暗号化および認証メカニズムを活用できます。シン JDBC ドライバは、インターネット経由で使用されるダウンロード可能なアプレット用に設計されているので、シン・クライアント用に Oracle Advanced Security 暗号化および整合性アルゴリズムの 100% Java 実装が含まれています。

容易な設定、アプリケーションの変更なし

サーバーやクライアントのネットワーク・パラメータを設定することにより、ネットワーク暗号化/整合性機能を有効化できます。このため、アプリケーションに変更を加える必要がなく、ほとんどの企業でこのテクノロジーを簡単に取り入れることができます。

Oracle Database 10g の厳密認証サービス

情報に対する無認可のアクセスは、非常に古い問題です。今日のビジネスは、テラバイトのデータの山から収集した情報により動いています。機密情報の保護は、ビジネスの競争力を維持する鍵です。貴重な情報が入った Oracle Database 10g のような重要なデータ・リポジトリへのアクセスは、ユーザーが正確に識別され認証されれば許可できます。ユーザー識別の検証には、通常ユーザー名とパスワード以外の情報も集めることが必要です。Oracle Advanced Security は、Kerberos、公開鍵インフラストラクチャ (PKI)、RADIUS などの既存のセキュリティ・インフラストラクチャを活用して、Oracle Database 11g への厳密な認証サービスの機能をビジネスに提供します。証明書失効リストは、ファイル・システムや Oracle Internet Directory に格納でき、CRL Distribution Points も使用できます。

Oracle Database Servers または Oracle Database Clients/Users は、業界の PKCS#11 標

準を使用するスマート・カードなどのハードウェア・ストレージ・モジュールに格納された PKI 証明書を使用できます。これにより、クライアント・サーバー・アプリケーションまたは Web アプリケーションを介したデータベースへのローミング・アクセスが可能になるため、ユーザーにとって特に便利です。ハードウェア・モジュールにサーバー証明書を格納することにより、一部の配置で必要なさらに高レベルのセキュリティが提供されます。

Kerberos 認証

Oracle Advanced Security には、任意の MIT v5 準拠 Kerberos サーバーまたは Microsoft KDC から発行される Kerberos v5 チケットと互換性を持つ Kerberos クライアントが含まれています。Oracle Advanced Security の Kerberos ソリューションを使用することにより、企業は異機種環境でも業務を継続できます。Oracle データベースを Kerberos サーバーに登録し、Kerberos サービスをサポートするように設定するだけで、その他の複雑な作業なしに、企業ユーザーはデータベースへの認証を実行できます。すでに Kerberos サーバーと Oracle Advanced Security の Kerberos アダプタを使用している企業は、外部データベース・ユーザーをディレクトリに移行して、ユーザーを集中管理できます。

Kerberos の機能拡張

Oracle Database 11g Advanced Security の Kerberos の機能拡張には、最長 2000 文字のプリンシパル名のサポートが含まれています。さらに、Oracle Database 11g Advanced Security は、ひとつのレルムの Kerberos プリンシパルによる別のレルムの Kerberos プリンシパルの認証を可能にする Kerberos プリンシパル・クロス・レルム・サポートを提供します。

ここで、外部認証される Oracle ユーザーを作成する例を示します。ユーザー名は、Kerberos ユーザーに対応している必要があります。

```
SQL> CONNECT / AS SYSDBA;
```

```
SQL> CREATE USER "KRBUSER@SOMECO.COM" IDENTIFIED  
EXTERNALLY;
```

```
SQL> GRANT CREATE SESSION TO "KRBUSER@SOMECO.COM";
```

Kerberos 認証のための Oracle データベースおよびクライアントの設定について詳しくは、Oracle Advanced Security の管理者ガイドを参照してください。

PKI のサポート

Oracle Advanced Security の SSL クライアントは、業界標準に準拠し、標準 PKCS7 証明書要求を受け入れて、X509v3 証明書を発行する任意の PKI で使用できます。Oracle Advanced Security は、Entrust アダプタを提供します。これにより、ビジネス・アプリケーションは、Oracle Database 11g で Entrust の PKI を活用できます。

Oracle Wallet Manager は、エンド・ユーザーのための証明書要求およびその他の証明書管理タスクにも使用できます。このリリースでは、証明書失効リスト (CRL) の管理およびその他の Oracle ウォレット操作で役立つ追加のコマンド・ライン・ユーティリティも用意されています。

LDAP サーバー、ファイル・システム、または URL に公開された証明書失効リストは、Oracle の SSL インフラストラクチャによってサポートされます。

PKCS#12 のサポート

Oracle Advanced Security は、PKCS #12 コンテナに格納された X.509 証明書をサ

ポートします。このため、Oracle ウォレットは Netscape Communicator 4.x や Microsoft Internet Explorer 5.x などのサード・パーティ・アプリケーションとの相互運用性を持ち、オペレーティング・システム間でのウォレットの移植性が実現されます。既存の PKI 証明書を持つユーザーは、それらの証明書を PKCS#12 形式でエクスポートして Oracle Wallet Manager で再利用したり、その逆の操作を実行したりできます。このように、PKCS#12 は、相互運用性を向上させ、企業の PKI 配備のコストを削減します。

PKCS#11 のサポート、スマート・カード/ハードウェア・セキュリティ・モジュール

Oracle ウォレットは、秘密鍵や証明書のその他のトラスト・ポイントを格納できるソフトウェア・コンテナです。Oracle Advanced Security 10g は、サポート PKCS#11 業界標準をサポートします。このため、以前はファイル・システムに格納されていた秘密鍵を、市場で入手できるハードウェア・セキュリティ・モジュールやスマート・カードなどの安全なデバイスで作成し、格納できます。

Oracle Database 10g エンタープライズ・ユーザー用の PKI 認証

Oracle8i 以降、Oracle Advanced Security は、ディレクトリに格納されたデジタル証明書による Oracle データベースへのディレクトリ・ユーザーの認証をサポートしてきました。

オラクルは、PKI の統合と相互運用性を次の方法で拡張します。

- PKCS#11 のサポート
- Oracle Internet Directory へのウォレットの格納
- ウォレットごとに複数の証明書
- 強力なウォレット暗号化
- OracleAS 認証局

Oracle Internet Directory に格納されたウォレット

Oracle Enterprise Security Manager は、ユーザー登録プロセスの一環としてユーザー・ウォレットを作成します。このウォレットは、Oracle Internet Directory またはその他の LDAP 準拠ディレクトリに格納できます。Oracle Wallet Manager は、LDAP ディレクトリにウォレットをアップロードし、LDAP ディレクトリからウォレットを取り出すことができます。

集中化された LDAP 準拠ディレクトリにウォレットを格納しておくことにより、ユーザー・ローミングがサポートされ、ユーザーは複数の場所またはデバイスから自身の資格証明にアクセスできます。これにより、ユーザー認証の整合性と信頼性が保証され、同時にウォレットのライフ・サイクル全体を通してウォレット管理が集中化されます。

複数の証明書のサポート

Oracle ウォレットは、1 つのウォレットに対して、次を含む複数の証明書をサポートします。

- S/MIME 署名証明書
- S/MIME 暗号化証明書

- Code 署名証明書

Oracle Wallet Manager バージョン 3.0 は、persona 内の 1 つのデジタル・エンティティに対して複数の証明書をサポートし、persona 内に複数の秘密鍵ペアを持ちます (1 つの秘密鍵に対応する証明書は 1 つのみです)。これにより、ユーザーの PKI 資格証明の統合とより安全な管理が実現されます。

強力なウォレット暗号化

X.509 証明書に関連付けられる秘密鍵には、安全なチャネルを介した強力な暗号化が必要です。オラクルでは、DES 暗号化を 3 つの鍵を使用する Triple-DES (3DES) に置き換えています。3DES は DES よりも極めて強力な暗号化アルゴリズムで、Oracle ウォレットに対してさらに優れたセキュリティを提供します。

RADIUS (Remote Dial-in User Service)

Oracle Advanced Security は、RADIUS クライアントを提供します。これにより、Oracle Database 11g は、RADIUS サーバーによりアサートされた認証と認可を利用できます。この機能は、知っていること (パスワードまたは PIN 情報) と持っているもの (いくつかのトークン・カード・メーカーにより提供されるトークン・カード) に基づいて識別を確立する 2 つの要素による認証に関心のある企業にとって、特に便利です。RADIUS (RFC #2138) は、ネットワーク・サービスへのリモート・アクセスを保護する分散システムで、長期にわたり、ネットワークへのリモート・アクセスおよびアクセス制御の業界標準として確立されています。RADIUS のユーザー資格証明およびアクセス情報は RADIUS サーバーで定義され、これにより、要求されたときにこの外部サーバーが、認証、認可、および会計サービスを実行します。

Oracle RADIUS は、データベースによる RADIUS ユーザーの認証、認可、および会計サービスを可能にする RADIUS Client プロトコルの実装をサポートします。Oracle RADIUS は、認証要求を RADIUS サーバーに送信し、サーバーの応答に基づいて動作します。認証は、同期認証モードでも非同期認証モードでも実行され、RADIUS をサポートするための Oracle の設定に含まれます。

Oracle Advanced Security は、Oracle データベースへのアクセス時に、認証を提供し、RADIUS に格納される認可と RADIUS ユーザーへの基本会計サービスの利用を可能にします。

まとめ

暗号化は多層防御の主要コンポーネントであり、オラクルは、個人情報保護に関する厳しいセキュリティ要件の拡大に対処するための革新的なソリューションを開発しつづけます。小売業者は Oracle Advanced Security の TDE を使用してクレジットカード業界データ・セキュリティ法 (PCI-DSS) に対応し、大学や健康管理機構は Oracle TDE を使用して社会保障番号やその他の機密情報を保護できます。暗号化は、転送データの保護において特に重要な役割を果たします。Oracle Advanced Security のネットワーク暗号化は、イントラネット上の転送データを、ネットワーク・スニффイングや変更から保護します。Oracle Advanced Security の TDE は、ディスク・ドライブやバックアップ・メディア上の機密データを不正アクセスから保護し、メディアの喪失や盗難による影響を削減します。



日本語タイトル
2007年6月
著者: Paul Needham
共著者: Peter Wahl

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

海外からのお問合せ窓口:
電話: +1.650.506.7000
ファックス: +1.650.506.7200
www.oracle.com

Copyright © 2007, Oracle. All rights reserved.
本文書は情報提供のみを目的として提供されており、ここに記載される内容は予告なく変更されることがあります。
本文書は一切間違いがないことを保証するものではなく、さらに、口述による明示または法律による黙示を問わず、特定の目的に対する商品性もしくは適合性についての黙示的な保証を含み、いかなる他の保証や条件も提供するものではありません。オラクル社は本文書に関するいかなる法的責任も明確に否認し、本文書によって直接的または間接的に確立される契約義務はないものとします。本文書はオラクル社の書面による許可を前もって得ることなく、いかなる目的のためにも、電子または印刷を含むいかなる形式や手段によっても再作成または送信することはできません。
Oracle は米国 Oracle Corporation およびその子会社、関連会社の登録商標です。その他の名称はそれぞれの会社の商標です。