
ORACLE®

Oracle **Database** Technology Night
～集え！オラクルの力（チカラ）～

えっ、知らなかったでは済まされない。
データベース・セキュリティの勘所
～ 最低限これくらいはやっておきましょう ～

日本オラクル株式会社
クラウド・テクノロジー事業統括
Database & Exadata プロダクトマネジメント本部
データベースエンジニアリング部
福田 知彦

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

こんなデータベースはイヤだ！

セキュリティ的によろしくないデータベースの例

DBユーザーがアプリケーションのアカウントしかない

```
SQL> select username
  2    from dba_users
  3   where oracle_maintained='N';

USERNAME
----------------
SCOTT
APP1
APP2
APPADMIN
PDBADMIN
```

どんなことが起こる？

- ✓ 誰が何にアクセスできるかの権限管理ができない
- ✓ アカウントを共有すると監査やログから実際に誰が操作をおこなったかを特定できない
- ✓ 共有アカウントはパスワードを多くの人で共有していることが多い
- ✓ アプリケーションのアカウントはパスワードを変更できないことが多い

パスワードがデフォルト、もしくはユーザー名と同じ

```
SQL> connect system/manager
接続されました。
SQL> connect hr/hr
接続されました。
SQL>
```

どんなことが起こる？

- ✓ 本来利用する必要のないDBユーザーに簡単になり済まされる
- ✓ 誰が何にアクセスできるかの権限管理ができない
- ✓ 誤操作でデータやデータベースが破壊される
- ✓ 監査やログから実際に誰が操作をおこなったかを特定できない
- ✓ 攻撃されたときにアカウントを奪取されやすい

多くのDBユーザーがDB管理権限を持っている

```
SQL> select grantee, granted_role
  2    from dba_role_privs
  3   where granted_role='DBA';

GRANTEE              GRA
-------------------- ---
SYSTEM               DBA
SYS                  DBA
SCOTT                DBA
PDBADMIN             DBA
MY_DBA_ROLE          DBA
```

どんなことが起こる？

- ✓ 誰が何にアクセスできるかの権限管理ができない
- ✓ 本来アクセスする必要のないデータにアクセスされる
- ✓ 誤操作でデータやデータベースが破壊される
- ✓ 攻撃を受けたときに被害が拡大する

多くのDBユーザーが強力な権限を持っている

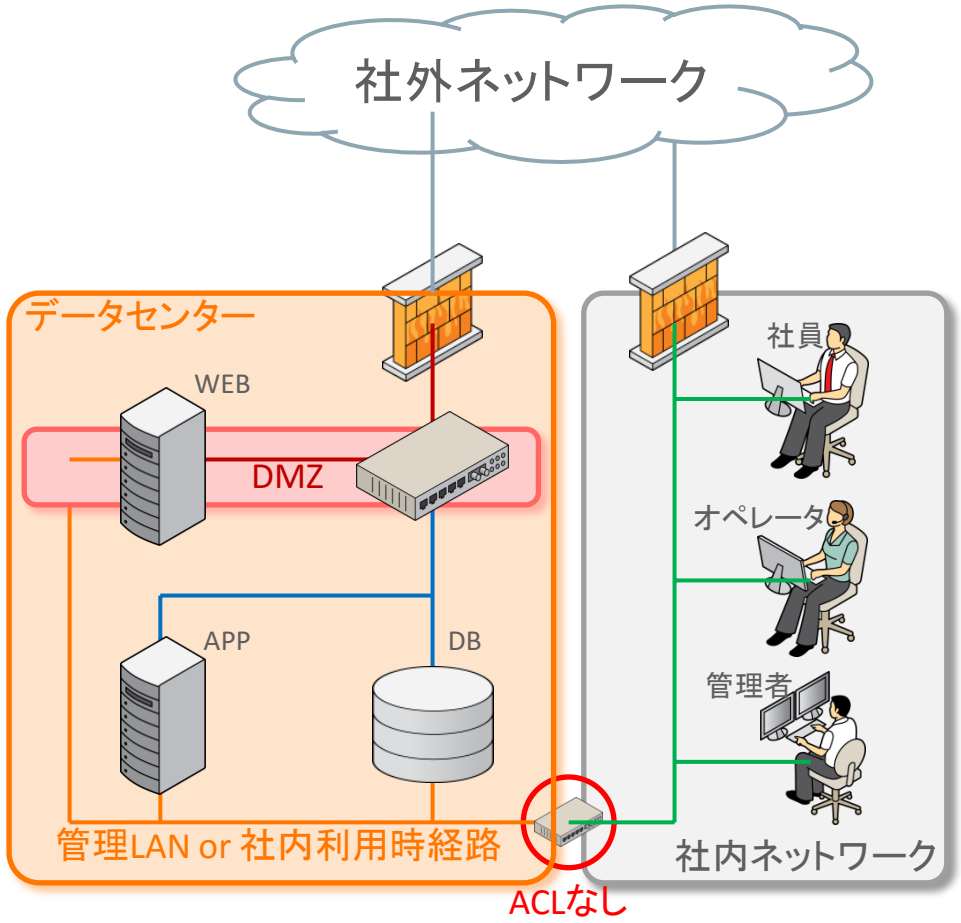
```
SQL> select grantee, privilege
  2    from dba_sys_privs
  3   where privilege='SELECT ANY
TABLE';

GRANTEE          PRIVILEGE
---------------- ----------------
SYSTEM           SELECT ANY TABLE
SYS              SELECT ANY TABLE
APP1             SELECT ANY TABLE
APP2             SELECT ANY TABLE
DBA              SELECT ANY TABLE
MY_APP_ROLE      SELECT ANY TABLE
```

どんなことが起こる？

- ✓ 誰が何にアクセスできるかの権限管理ができない
- ✓ 本来アクセスする必要のないデータにアクセスされる
- ✓ 誤操作でデータやデータベースが破壊される
- ✓ 攻撃を受けたときに被害が拡大する

社内のどこからでも本番データベースに接続できる



どんなことが起こる？

- ✓ 接続状況を知っていればデータベースに直接接続できる
- ✓ マルウェア感染した端末からデータベースサーバーに接続される

Agenda

1. データベース・セキュリティの基本的な考え方
2. データベースのセキュリティセルフチェックツール(DBSAT)紹介
3. DBSATレポートの読み方と考慮すべき項目
4. その他の要考慮事項
5. 参考資料ご紹介

データベース・セキュリティの基本的な考え方

- 共有ユーザー利用などの匿名性を排除する
- 重要なデータは集中管理し、アクセス経路を限定する
- 最小権限の原則に則ったアクセス制御を強制する
- 暗号化したほうがよいか迷うものは、すべて暗号化する
- アクセスは監査するだけでなく、監視する

各種ガイドラインでのデータベース・セキュリティの記載

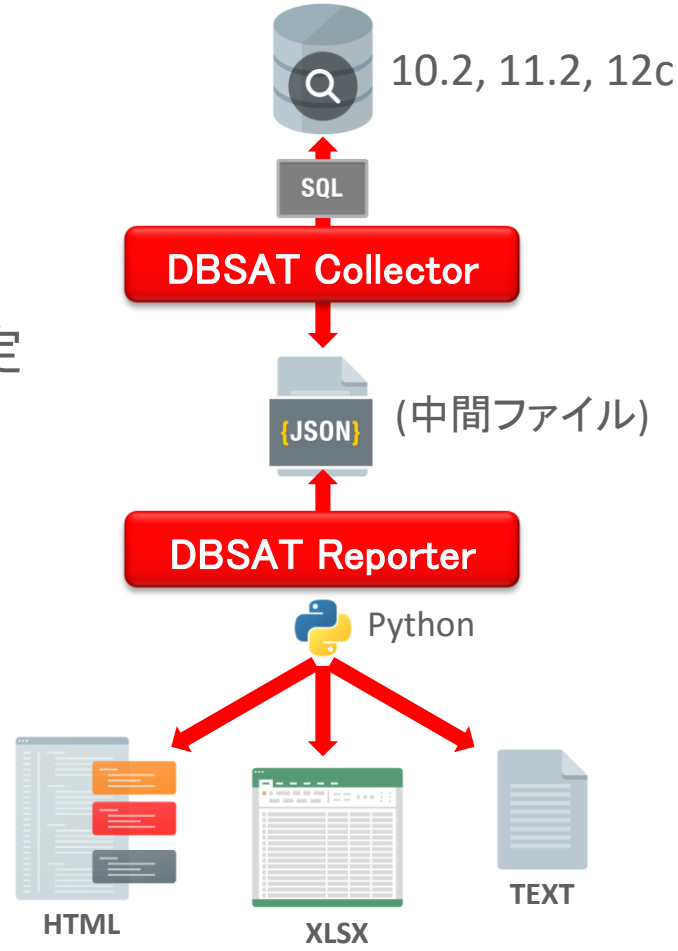
名称	セキュリティ強化への取り組み		対象
個人情報保護 （経済産業省）	2004年10月	2004年10月:経済産業分野のガイドライン策定	経済産業分野の 事業者 及び、業界団体
	2008年2月	ガイドライン改定。**暗号化**の記載が追加	
	2014年12月	ガイドライン改定。**データベースへのアクセス制御、 管理者権限分割、パッチ管理が追加**	
特定個人情報の 適正な取扱いに関する ガイドライン （個人情報委員会）	2013年5月	「行政手続における特定の個人を識別するための番号の利用等に関する法律（番号法）」が成立	行政機関・ 地方公共団体及び、 全事業者
	2014年12月	ガイドライン公開。技術的安全管理措置に特定個人情報への**アクセス制御、暗号化、監査**が記載	
サイバーセキュリティ 経営ガイドライン （経済産業省）	2015年12月	ガイドライン 公開。**多層防御と重要データ（データベース、 ファイル）への高度な暗号化、アクセス制御、監査**が記載	企業の経営者 （大企業・中小企業の ITシステムを供給する 企業と経営戦略上ITの 利活用が不可欠な企業）
政府機関等の 情報セキュリティ対策の ための統一基準	2005年9月	政府機関の情報セキュリティ対策のための統一基準 公開	政府機関等
	2016年8月	統一基準改定。**データベースの項目が新たに追加。 管理者権限分割、アクセス制御、監査、暗号化**が記載	

Agenda

1. データベース・セキュリティの基本的な考え方
2. データベースのセキュリティセルフチェックツール(DBSAT)紹介
3. DBSATレポートの読み方と考慮すべき項目
4. その他の要考慮事項
5. 参考資料ご紹介

Database Security Assessment Tool (DBSAT)

- データベースをスキャンし、DBのセキュリティを網羅的にチェックするツール
  - 基本構成、ユーザアカウント、権限管理、暗号化、監査・・・
- Pythonで作成された自動レポート生成スクリプト
- HTML, XLSX, TEXTの形式でそれぞれレポート出力
- 出力レポートは英語　※日本オラクルから日本語化サービス提供予定
- ダウンロード先
  - Doc ID 2138254.1
    Oracle Database Security Assessment Tool (DBSAT)
- ドキュメント
  - Database Database Security Assessment Tool User Guide
    http://docs.oracle.com/cd/E76178_01/index.htm



動作要件

- 対応OS
  - Solaris x64, Soloaris SPARC
  - Linux x86-64
  - Windows x64
  - HP UX IA(64bit)
  - IBM AIX, zSeries Based Linux

- データベース
  - Oracle Database 10.2.0.5以上

※データは参照SQLとOSコマンドをデータベースサーバ上で実行して収集される

Windowsの場合は、SQLコマンドのみ

- 実行ユーザに必要な権限
  - CREATE SESSION
  - SELECT on SYS.REGISTRY$HISTORY
  - Role SELECT_CATALOG_ROLE
  - Role DV_SECANALYST (Database Vaultを使用時)
  - Role AUDIT_VIEWER (12cのみ)
  - Role CAPTURE_ADMIN (12cのみ)
  - SELECT on SYS.DBA_USERS_WITH_DEFPWD (11g and 12c)
  - SELECT on AUDSYS.AUD$UNIFIED (12c のみ)

※Linux, Unixの場合は、DBSATを実行するOSユーザにORACLE_HOME以下のファイルへのRead権限が必要

OSユーザがoracle、DBユーザがSYSTEMで実行するのが最も簡単な実行方法

DBSAT Collectorの実行

1. My Oracle SupportからDbsat.zipをダウンロード
   - Doc ID 2138254.1 Oracle Database Security Assessment Tool (DBSAT)
2. dbsat.zip をデータベースサーバー上に解凍
   1. unzip dbsat.zip –d /home/oracle/dbsat
   2. cd /home/oracle/dbsat
3. DBSAT Collectorの実行
   1. 環境変数にORACLE_HOMEが設定されていることを確認
   2. LANGおよびNLS_LANG環境変数がUTF8であることを確認
      例) LANG=ja_JP.utf8; NLS_LANG=Japanese_Japan.AL32UTF8
   3. dbsat collect [DBUser/Password] [ZIP name] 例) dbsat collect system/oracle12c ora003

      -> ora003.zipファイルが作成される

      ※ 出力されたJSONファイルがZIPで暗号化される

DBSAT Collectorの実行例

```
$ ./dbsat collect system/oracle12c orac003
Connecting to the target Oracle database...SQL*Plus: Release 12.1.0.2.0 Production on 月 7月 11 00:43:01 2016
Copyright (c) 1982, 2014, Oracle.  All rights reserved.
最終正常ログイン時間: 月 7月  11 2016 00:16:06 +09:00
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
Real Application Testing and Unified Auditing options
に接続されました。
Setup complete.
SQL queries complete.
OS commands complete.
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
Real Application Testing and Unified Auditing optionsとの接続が切断されました。
DBSAT Collector completed successfully.
Calling /opt/app/oracle/product/12.1.0.2/dbhome_1/bin/zip to encrypt secvm3.json...
Enter password:  ZIPファイルのパスワード入力
Verify password: ZIPファイルのパスワード入力
adding: ora003.json (deflated 87%)
zip completed successfully.
```

DBSAT Reporterの実行

1. Python 2.6以上(Python 3.xは不可)がインストールされていることの確認
   1. phthon –V

```
$ python -V
Python 2.6.6
```

2. DBSAT Reporterの実行
   1. LANGおよびNLS_LANG環境変数がUTF8であることを確認
      例) LANG=ja_JP.utf8; NLS_LANG=Japanese_Japan.AL32UTF8
   2. dbsat report [ZIP name] 例) dbsat report ora003

      -> ora003.zipファイルが作成される

   ※実行後のレポートがZIPで暗号化される

DBSAT Reporterの実行例

```
$ ./dbsat report ora003
Archive:  ora003.zip
[ora003.zip] ora003.json password: ZIPファイルのパスワード入力
  inflating: ./ora003.json

DBSAT Reporter ran successfully.
Calling /usr/bin/zip to encrypt the generated reports...
Enter password: ZIPファイルのパスワード入力
Verify password: ZIPファイルのパスワード入力
  adding: ora003.txt (deflated 77%)
  adding: ora003.html (deflated 83%)
  adding: ora003.xlsx (deflated 3%)
zip completed successfully.
```

DBSATレポートサンプル (HTML)

Oracle Database Security Risk Assessment

Assessment Date & Time

Date of Data Collection	Date of Report	Reporter Version
Sat Sep 10 2016 02:36:00	Sat Sep 10 2016 21:19:54	1.0.1 (June 2016) - b855

Database Identity

Name	Platform	Database Role	Log Mode	Created
ORA007	Microsoft Windows x86 64-bit	PRIMARY	NOARCHIVELOG	Sat Sep 10 2016 02:17:00

Summary

Section	Pass	Evaluate	Unused	Some Risk	Significant Risk	Severe Risk	Total Findings
基本情報	0	0	0	0	0	0	**0**
ユーザーアカウント	5	0	0	0	4	0	**9**
権限とロール	6	12	0	0	0	0	**18**
権限付与のコントロール	0	0	1	0	0	0	**1**
データの暗号化	0	0	1	0	0	0	**1**
ファイングレイン・アクセス制御	0	0	2	0	0	0	**2**
監査	3	3	2	0	3	0	**11**
データベース構成	4	3	0	2	2	0	**11**
ネットワーク構成	0	0	0	0	0	0	**0**
オペレーティング・システム	0	0	0	0	0	0	**0**
Total	**18**	**18**	**6**	**2**	**9**	**0**	**53**

SYSTEM・SYSAUX 表領域のユーザーアカウント

USER.TBLSPACE

Status Significant Risk

Summary Found 5 users using SYSTEM or SYSAUX tablespace.

Details

```
Tablespace SYSTEM: MGMT_VIEW
Tablespace SYSAUX: APEX_030200, OWBSYS, OWBSYS_AUDIT, SYSMAN
```

Remarks SYSTEMとSYSAUX表領域は、Oracle Databaseの管理ユーザ用に保持された表領域です。これらの既存リソースを利用したDOS攻撃などを避けるために、通常のユーザ・アカウントがこの領域を使用することは避けて下さい。12.2より以前のリリースはSYSTEM領域は暗号化することができません。これはユーザのスキーマ情報をこの表領域に格納してはいけないもうひとつの理由です。

ユーザのなりすまし

PRIV.USER

Status Pass

Summary No grants of EXECUTE on restricted packages

Remarks PL/SQLパッケージ(DBMS_SCHEDULER, DBMS_SYS_SQL)は、SQLコードの実行や別のユーザの識別情報を使って外部ジョブを実行することを許可します。このアクセスは厳しく制限し、正当に必要なユーザのみ付与すべきです。

大小文字のパスワード

USER.CASE

Status Severe Risk

Summary 大小文字パスワードが使用されていません。

Details

```
Initialization parameter SEC_CASE_SENSITIVE_LOGON is set to FALSE.
```

Remarks 大小文字を区別するパスワードが推奨されます。なぜなら、大小文字を含むことで、攻撃者がパスワードを推測して攻撃しなければならないパスワードのリストが劇的に増加します。SEC_CASE_SENSITIVE_LOGONパラメータをTRUEにすることでデータベースはパスワードの大小文字を区別するようになります。

Agenda

1. データベース・セキュリティの基本的な考え方
2. データベースのセキュリティセルフチェックツール(DBSAT)紹介
3. DBSATレポートの読み方と考慮すべき項目
4. その他の要考慮事項
5. 参考資料ご紹介

ユーザーアカウント

User Accounts

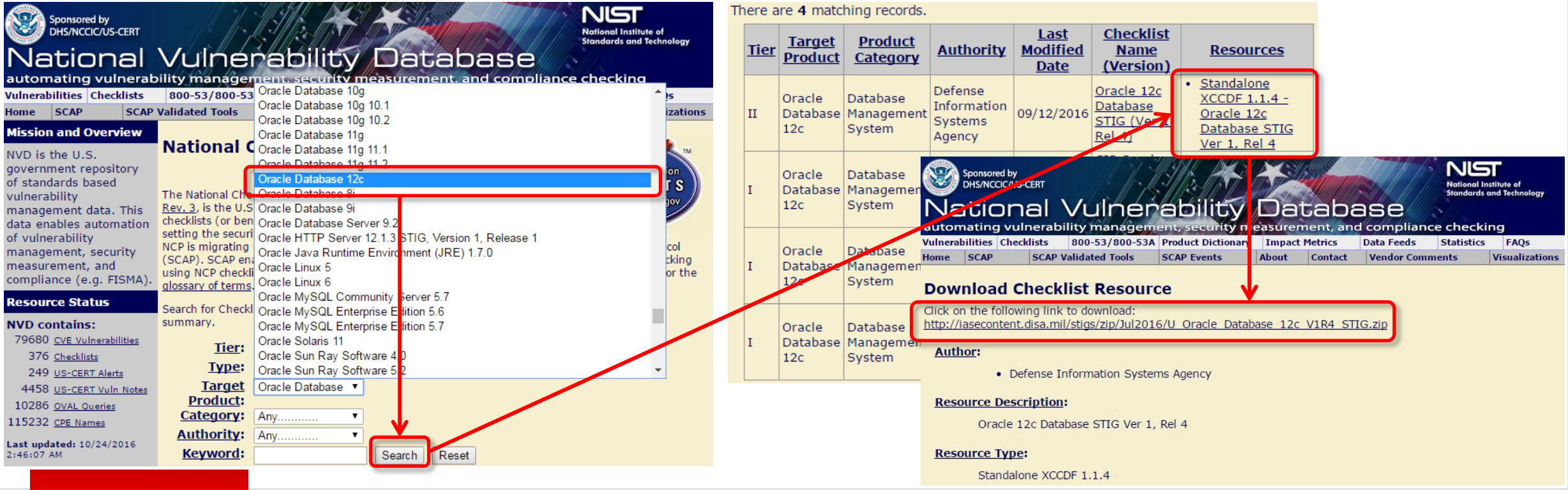
User Name	Status	Profile	Tablespace	Predefined	Type
APP	OPEN	DEFAULT	APP	No	PASSWORD
APP1	OPEN	DEFAULT	SYSTEM	No	PASSWORD
APP2	OPEN	DEFAULT	SYSTEM	No	PASSWORD
APPUSER	OPEN	DEFAULT	APP	No	PASSWORD
DBSNMP	OPEN	DEFAULT	SYSAUX	Yes	PASSWORD
EUSSHARED	OPEN	DEFAULT	SYSTEM	No	GLOBAL
HR	OPEN	DEFAULT	APP	Yes	PASSWORD
OPS$APP1DBA	OPEN	DEFAULT	SYSTEM	No	EXTERNAL
PDBADMIN	OPEN	DEFAULT	SYSTEM	No	PASSWORD
PUKU	OPEN	DEFAULT	SYSTEM	No	PASSWORD
SYS	OPEN	DEFAULT	SYSTEM	Yes	PASSWORD
SYSTEM	OPEN	DEFAULT	SYSTEM	Yes	PASSWORD

① ②

1. アプリケーション用ユーザーしかいない
   - 共有ユーザーを利用していないか
2. LDAPで認証されるグローバルユーザー、OSで認証される外部ユーザーがいる
   - 認証は十分に強固かどうか
   - 不要な強力な権限を割り当てていないか

参考) STIG：Security Technical Implementation Guide セキュリティ技術実装ガイド

- 米国国防総省（DoD：United States Department of Defense)のセキュリティ基準に準拠した製品に出される要件を満たすための実装ガイド
  - https://web.nvd.nist.gov/view/ncp/repository



権限とロール
～ 考え方

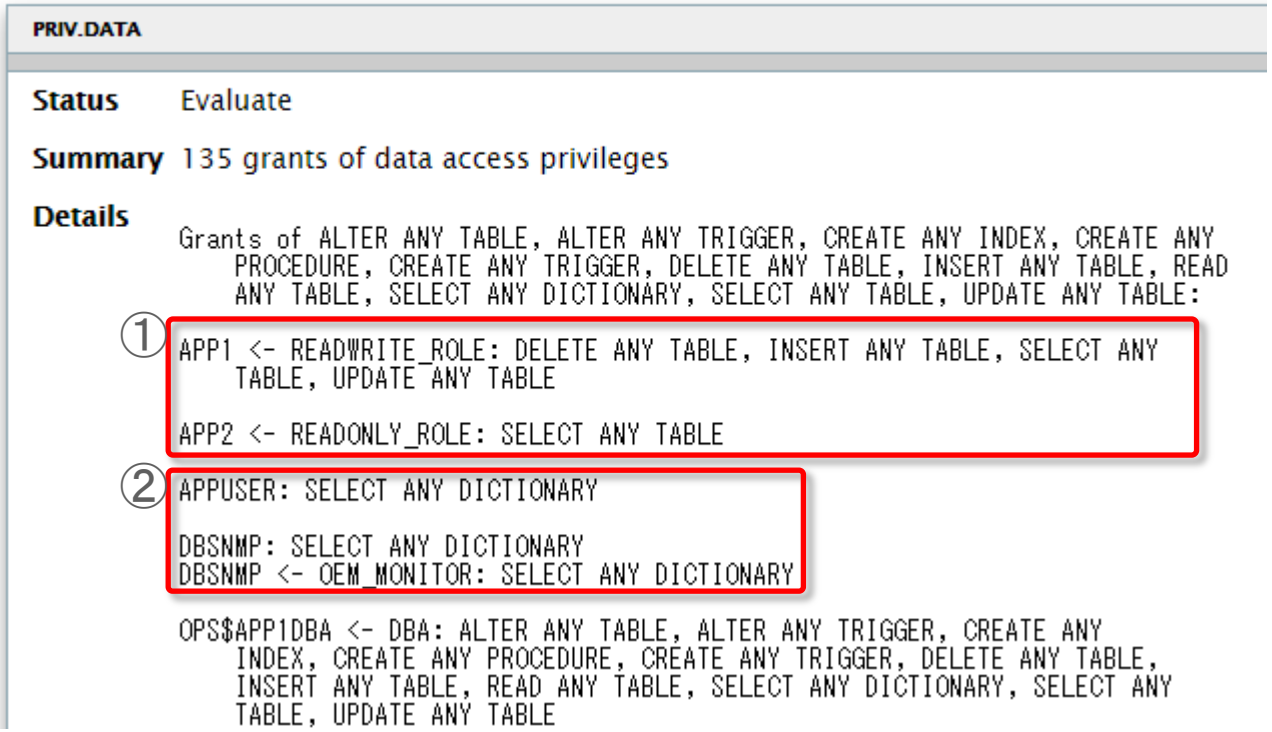
- 最小権限の原則
  - 各利用者が業務上必要なデータにしかアクセスできない (業務に必要な操作しかできない) ようにする設計の原則

    例：インフラ運用担当者は業務データにアクセスできない
- 職務分掌
  - ミス、不正を防ぐために一般利用者と承認者(権限設定する人)を明確に分離

    例： データにアクセスする人とデータへのアクセスを許可する人を分離する

権限とロール
～ ANYシステム権限

データアクセス権限



PRIV.DATA

Status Evaluate

Summary 135 grants of data access privileges

Details

Grants of ALTER ANY TABLE, ALTER ANY TRIGGER, CREATE ANY INDEX, CREATE ANY PROCEDURE, CREATE ANY TRIGGER, DELETE ANY TABLE, INSERT ANY TABLE, READ ANY TABLE, SELECT ANY DICTIONARY, SELECT ANY TABLE, UPDATE ANY TABLE:

①
APP1 <- READWRITE_ROLE: DELETE ANY TABLE, INSERT ANY TABLE, SELECT ANY TABLE, UPDATE ANY TABLE

APP2 <- READONLY_ROLE: SELECT ANY TABLE

②
APPUSER: SELECT ANY DICTIONARY

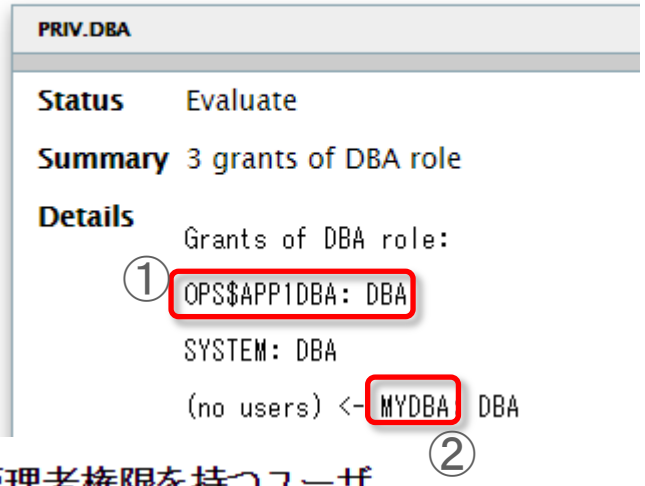
DBSNMP: SELECT ANY DICTIONARY
DBSNMP <- OEM_MONITOR: SELECT ANY DICTIONARY

OPS$APP1DBA <- DBA: ALTER ANY TABLE, ALTER ANY TRIGGER, CREATE ANY INDEX, CREATE ANY PROCEDURE, CREATE ANY TRIGGER, DELETE ANY TABLE, INSERT ANY TABLE, READ ANY TABLE, SELECT ANY DICTIONARY, SELECT ANY TABLE, UPDATE ANY TABLE

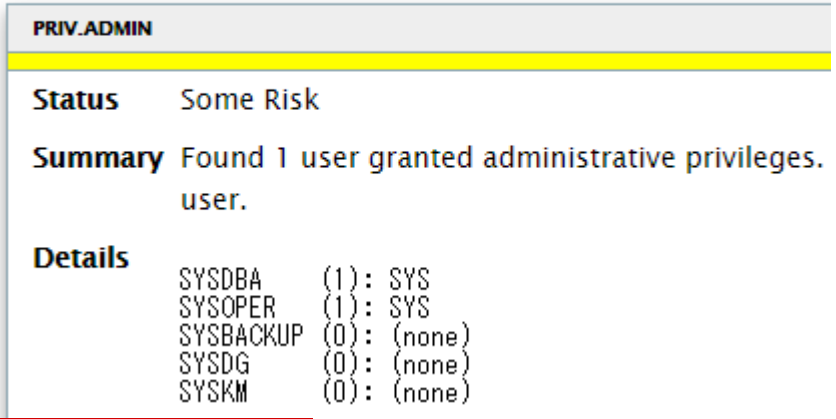
1. SELECT(UPDATE、DELETE) ANY TABLE権限がアプリケーションユーザーに割り当てられている
   - 同一データベース内の他のシステムのデータにアクセスできる
2. Enterprise Managerで接続するときに必要なSELECT ANY DICTIONARY権限割り当ては適切か

権限とロール
～ DBAロール、SYSDBA管理権限

DBAロール



管理者権限を持つユーザ



1. 不要なユーザーにDBAロールやSYSDBA管理権限は割り当てられていないか
   - インフラ管理をおこなわないアプリユーザーやアプリケーションデータの管理者に割り当てられていないか
2. ロール経由で暗黙的に管理権限が割り当てられてしまう事はないか
   - DBAロールはなるべく使わない
   - 利用する際には直接割り当てる

参考) DBAロール、SYSDBA管理権限、OS DBAグループ

DBAロール

- ユーザー名とパスワードで接続
- 強力なANYシステム権限を多数持つ
- SYSスキーマのオブジェクトにアクセスできないものがある

```
SQL> connect system/oracle@localhost/appdb
接続されました。
SQL> show user
ユーザーは"SYSTEM"です。
SQL> select count(*) from sys.user$;
select count(*) from sys.user$
                         *
行1でエラーが発生しました。:
ORA-01031: 権限が不足しています。
```

SYSDBA管理権限

- ユーザー名とパスワードにAS SYSDBA句をつけて接続
- SYSとして接続される
- 基本的に全ての操作が可能

```
SQL> connect puku/oracle@localhost/appdb
接続されました。
SQL> show user
ユーザーは"PUKU"です。
SQL> select count(*) from sys.user$;
select count(*) from sys.user$
                         *
行1でエラーが発生しました。:
ORA-00942: 表またはビューが存在しません。


SQL> connect puku/oracle@localhost/appdb as
  sysdba
接続されました。
SQL> show user
ユーザーは"SYS"です。
SQL> select count(*) from sys.user$;

  COUNT(*)
----------
       137
```

OS DBAグループ

- ユーザー名とパスワードを指定せずにAS SYSDBA句をつけて接続
- ユーザー名とパスワードが間違っていても接続できる
- SYSとして接続される
- 基本的に全ての操作が可能
- PDBにはパスワードなしでSYSDBA接続できない

```
SQL> connect / as sysdba
接続されました。
SQL> show user
ユーザーは"SYS"です。
SQL> select count(*) from sys.user$;

  COUNT(*)
----------
       121

SQL> connect nodoby/wrongPwd as sysdba
接続されました。
SQL> show user
ユーザーは"SYS"です。
SQL> connect /@localhost/appdb as sysdba
ERROR:
ORA-01017: ユーザー名/パスワードが無効です。ロ
  グオンは拒否されました。


警告: Oracleにはもう接続されていません。
```

アクセスマトリックス

- 最小権限の原則を整理するためにはアクセスマトリックスの整理が必要

誰が(主体)
人、ロール

何に(対象)
データ、各種管理操作

	経理データ１	経理データ２	経理データ３	顧客データ１	顧客データ２	顧客データ３
経理部社員	○	×	×	×	×	×
経理課長	○	△	△	△	×	×
経理部長	○	○	△	△	△	×
経理本部長	○	○	○	△	△	△
経理担当取締役	○	○	○	△	△	△
営業部社員	△	×	×	○	×	×
営業課長	△	△	×	○	△	×
営業部長	△	△	△	○	○	△
営業本部長	△	△	△	○	○	○
営業担当取締役	△	△	△	○	○	○

データの機密性レベル
１：機密性　低
２：機密性　中
３：機密性　高

アクセス権限種別
○：読みとり・書き込み可
△：読みとりのみ
×：アクセス不可

最小権限の原則と職務分掌実現時のユーザーの作成例

ユーザー種別	ユーザー名	付与する権限	アカウント管理方法	監査
デフォルトユーザー（非個人）	SYS	変更しない(SYDBA、DBA)	基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須
	SYSTEM	変更しない(DBA)	基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須
	OUTLN	変更しない(各ユーザーごと)	利用しないのであれば、EXPIRED & LOCKEDに。	すべて必須
	DBSNMP	変更しない(OEM_MONITORなど)	直接利用不可。パスワードは周知しない状態で厳密に管理	接続失敗は必須
アプリ用オブジェクト所有者（非個人）	APP1	必要なオブジェクト作成権限、領域割り当てのための権限	必要なオブジェクト作成後、EXPIRED & LOCKEDに。	すべて必須
アプリ接続用ユーザー（非個人）	APP1_BI	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_WEB1	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_WEB2	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_Batch1	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_Batch2	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
インフラ管理/アプリ管理・利用用個人アカウント	USER1	管理・利用に必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)と構成変更、重要なデータへのアクセスは必須
	USER2	管理・利用に必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)と構成変更、重要なデータへのアクセスは必須
	USER3	管理・利用に必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)と構成変更、重要なデータへのアクセスは必須
緊急時利用アカウント（非個人）	URGENT	DBA、SYSOPER	基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須

少なくともアプリ用と個人用のプロファイルを別に作成すべき

※ あくまで1例であり、OSユーザー等と組み合わせて個人を識別する方法もあります

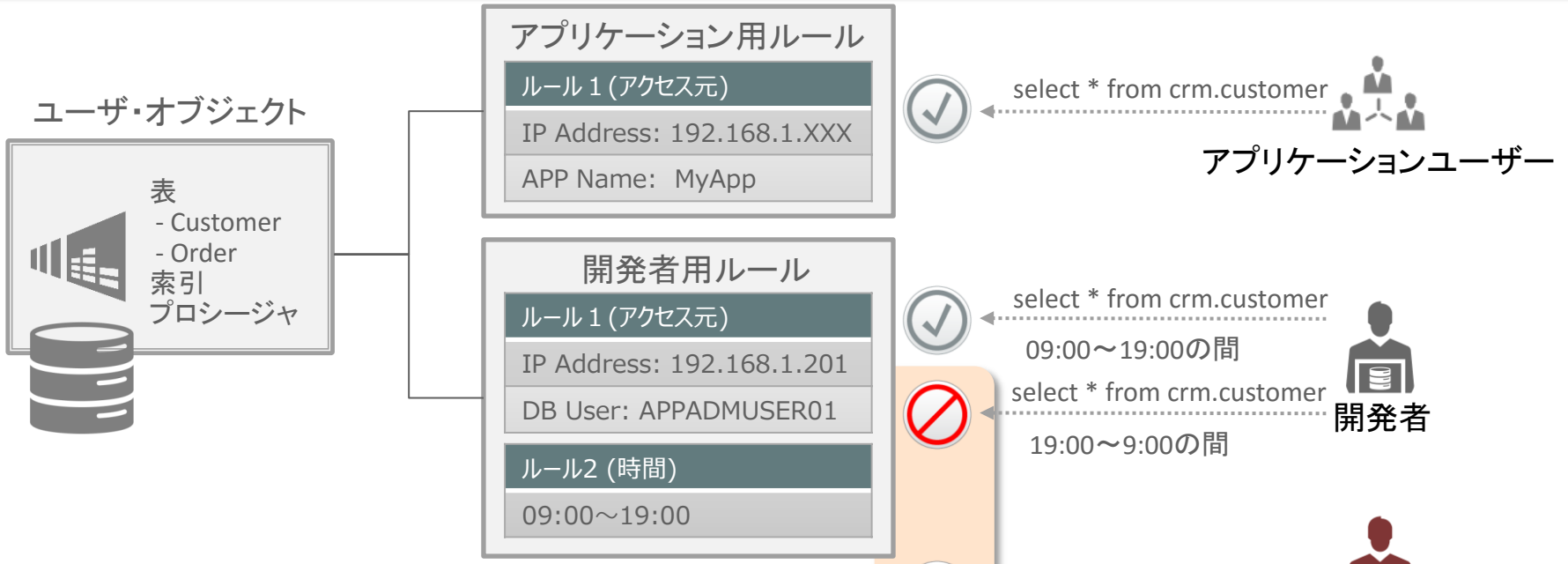
最小権限の原則と職務分掌の実現時の注意点

- CONNECT、RESOURCE、DBAなど事前定義のロールは利用しない
- アプリ接続用ユーザーには、アプリやバッチごとに利用に必要な最小限のオブジェクト権限をまとめたロールを作成し、それを付与する。システム権限が必要な場合には必ずロールを分ける
- インフラ管理用の個人ユーザーには、その人の役割ごとにインフラ管理に必要な最小限のシステム権限をまとめたロールを作成し、それを付与する。アプリデータへのアクセス権限は基本的に付与しない
- アプリ接続用、インフラ管理用のロールは、運用コストとのバランスを考えつつなるべく細分化する

DV

参考：Database Vault
～ やっぱりDBAロールを使いたい！

厳密なルール設定で不正アクセス、想定外アクセスを遮断。管理権限でも回避できないアクセス制御を提供



常に非認可

職務分掌 特権ユーザ (SYS, DBA権限) であっても情報にはアクセスさせない

透過的 既存アプリケーションの変更不要、どの経路からのアクセスも一律に制御

厳密 ユーザー、クライアント情報 (IPアドレス、アプリ名)、時間を組み合わせたポリシー設定

Simulation Mode (12.2新機能)
ポリシー違犯があった場合、ブロックするかわりにログを残す。
テストフェーズでポリシーの妥当性評価に便利

DV

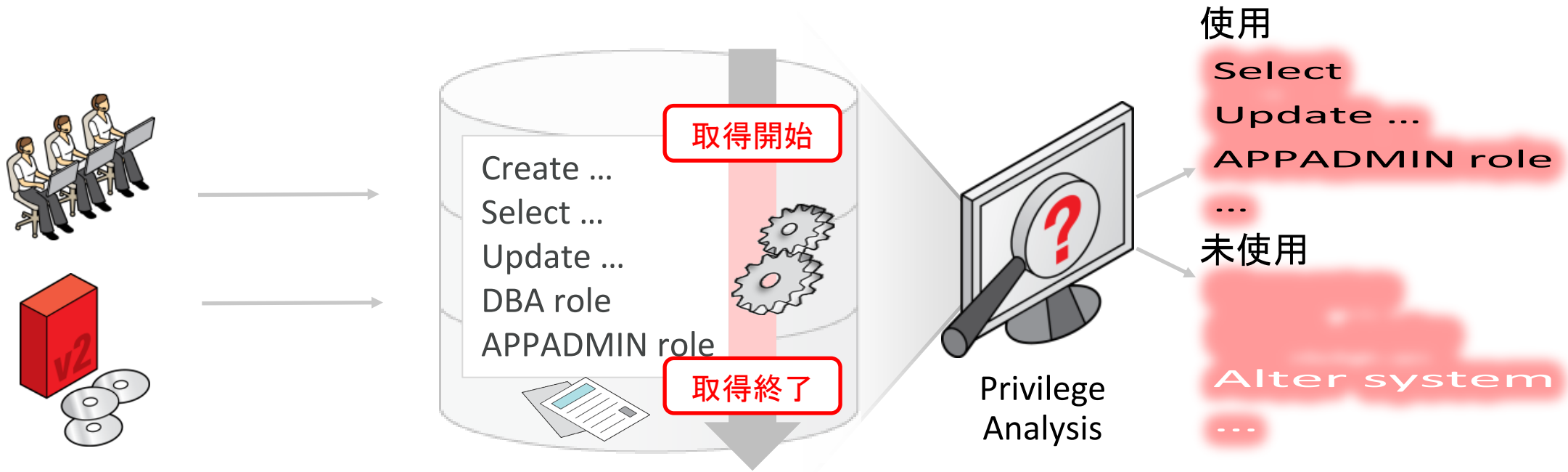
(参考) Database Vaultの利用による シンプルな最小権限の原則と職務分掌実現

ユーザー種別	ユーザー名	付与する権限	アカウント管理方法	監査
デフォルトユーザー（非個人）	SYS	変更しない(SYDBA、DBA)	基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須
	SYSTEM	変更しない(DBA)	基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須
	DVO	変更しない(DV_OWNER)	設定後は基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須
	OUTLN	変更しない(各ユーザーごと)	利用しないのであれば、EXPIRED & LOCKEDに。	すべて必須
	DBSNMP	変更しない(OEM_MONITORなど)	直接利用不可。パスワードは周知しない状態で厳密に管理	接続失敗は必須
アプリ用オブジェクト所有者（非個人）	APP1	必要なオブジェクト作成権限、領域割り当てのための権限	必要なオブジェクト作成後、EXPIRED & LOCKEDに。	すべて必須
アプリ接続用ユーザー（非個人）	APP1_BI	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_WEB1	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_WEB2	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_Batch1	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_Batch2	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
インフラ管理/アプリ管理・利用用個人アカウント	USER1	DBA、もしくは管理・利用に必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)と構成変更、重要なデータへのアクセスは必須
	USER2	DBA、もしくは管理・利用に必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)と構成変更、重要なデータへのアクセスは必須
	USER3	DBA、もしくは管理・利用に必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)と構成変更、重要なデータへのアクセスは必須
緊急時利用アカウント（非個人）	URGENT	DBA（アプリデータへのアクセス権限を含む）、SYSOPER	基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須

少なくともアプリ用と個人用のプロファイルを別に作成すべき

DV

参考：Privilege Analysis
～ 今現在、誰がどんな権限を利用しているのかの見える化



- ユーザーやロールに付与されたシステム権限、オブジェクト権限の使用・未使用を洗い出してレポーティング
- アプリケーションや開発者・管理者に本当に必要な権限のみを付与
- 最小権限の原則を実現し、不正アクセスの未然防止に

暗号化
～ どのようなデータを暗号化すべきか

- 法律/ガイドライン/ポリシー/コンプライアンス要件などで定められている項目
  - (特定)個人情報 ― マイナンバー法、個人情報保護法および関連ガイドライン
  - クレジットカード情報 ― PCI-DSS
  - 会社などの情報保護規定の指定する項目
- それ以外の機微情報
  - 営業機密 ― 情報が漏洩しても適切に保護されていなかった場合、機密と認められないことも
  - 会計(業績)情報
  - 顧客情報
  - 認証情報

など

暗号化したほうがよいか迷うものは、すべて暗号化する

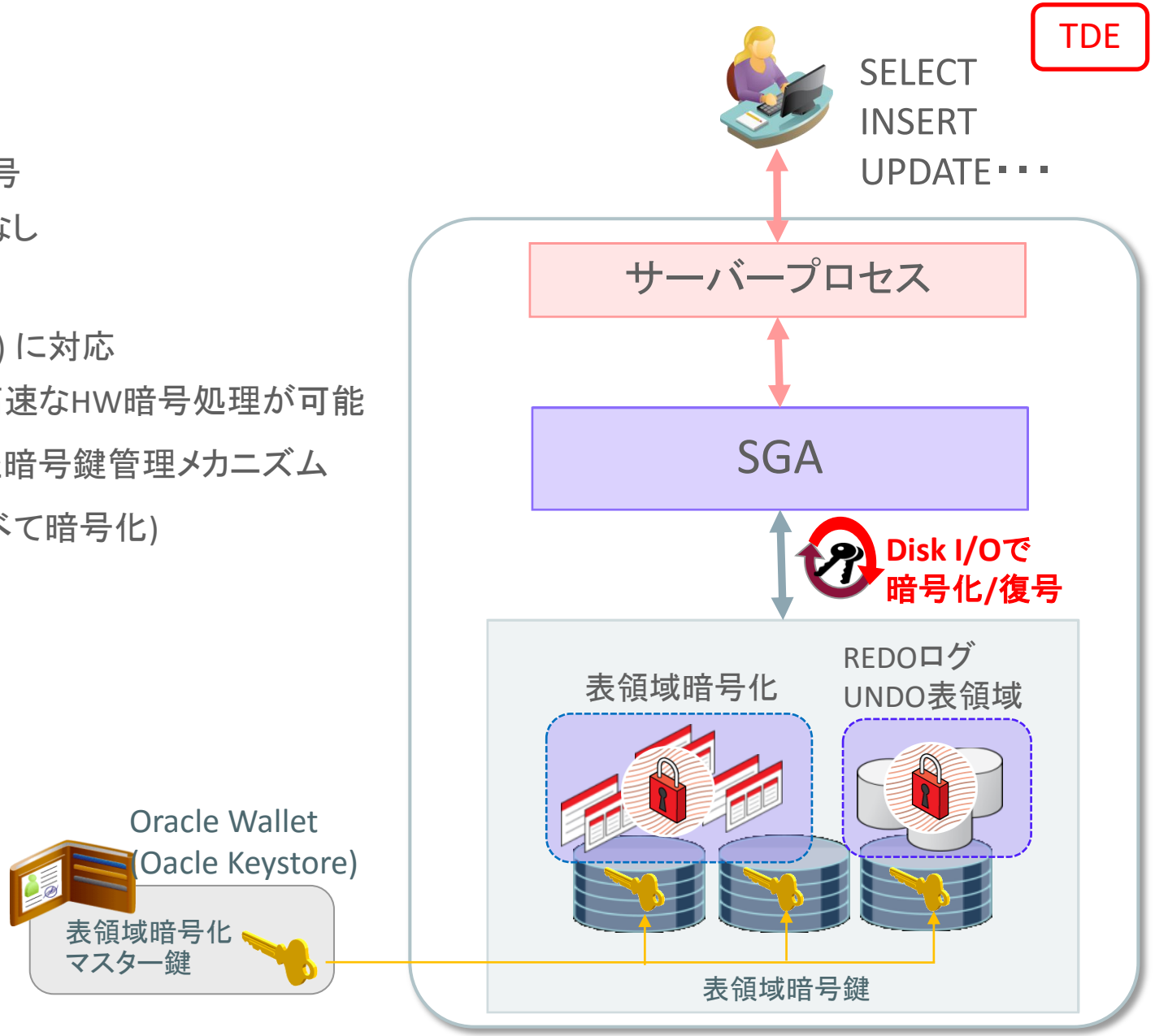
格納データの暗号化の3つの方式

方式	暗号化してから データベースに格納	データベース格納時に 暗号化	ストレージでの暗号化
実装例	•カスタムアプリ •暗号化製品	•データベース機能 (TDE表領域暗号化)	•ストレージ機能 •アプライアンス
対応できる脅威	•ストレージ機器の盗難 •データファイルの盗難 •データベースアクセスによる情報漏洩	•ストレージ機器の盗難 •データファイルの盗難 ← DBへのアクセス制御で対応	•ストレージ機器の盗難 ← OSファイルへのアクセス制御で対応
利用時の注意点	•暗号化によりデータ長やデータ型が変わる **•アプリケーションの書き換えが必要** **•索引の利用に制限** (チューニングが困難) •問い合わせごとに暗号化/復号処理が毎回おこなわれるため、暗号化列数が多い場合や戻り行数が多い場合には負荷大 **•データベースアクセスによる情報漏洩は防げても、不正書換、削除には対応できない** **•鍵管理・配布が複雑**	•ファイルI/O時に暗号化/復号がおこなわれるため、OSメモリ上のデータは暗号化されていない	**•OSからは元のデータが見える** •OS経由でバックアップすると暗号化されない •暗号化単位により余計なI/Oや暗号化/復号処理が発生する可能性 •問題発生時の切り分けが困難かつ、回避策として暗号化製品を利用しない運用をお願いする可能性あり
	重要なデータ(パスワードなど)では利用を考慮	有償オプションだがお薦め	OSファイルへのアクセス制御に要注意

※ OracleのCloudでは、どのライセンスでも格納データは暗号化されます

TDE表領域暗号化

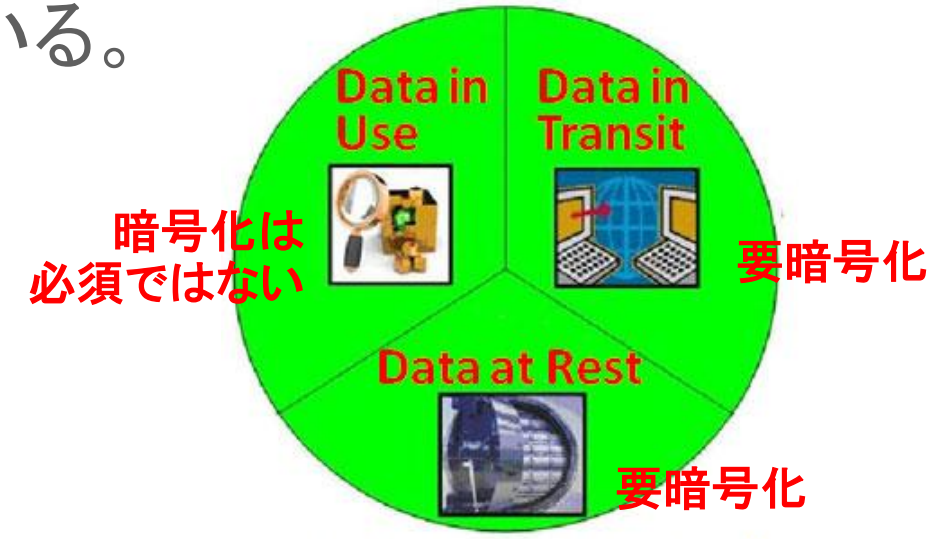
- アプリケーションからは透過的にデータの暗号化/復号
  - 既存のアプリケーション (SQL) を改修する必要はなし
- 強力な暗号アルゴリズムを利用した暗号化を実施
  - NISTの標準共通鍵暗号方式 AES (128/192/256bit) に対応
  - Intel AES-NI、Sparc 暗号化命令アクセラレータで高速なHW暗号処理が可能
- Oracle Wallet やHardware Security Moduleを利用した暗号鍵管理メカニズム
- 表領域単位での暗号化 (表領域内の表や索引がすべて暗号化)
- REDOログ、UNDO、アーカイブログも暗号化
- データブロックに対するI/Oで暗号化/復号
- SGAのバッファキャッシュ上は暗号化されていない
- 暗号化してもデータサイズは増加しない
- 表領域暗号鍵はデータファイルのヘッダーに格納
- 通常のチューニングが可能 (索引の利用に制限なし)
- ほとんどすべてのオブジェクトが暗号化可能
(BFILEのみ不可)



TDE

バッファキャッシュの暗号化は必要か？ ～ 3つのデータの状態と暗号化

- 米国国防のセキュリティ基準を制定しているNSA(National Security Agency)では、2003年にNet-Centric Data Strategyという文書の中でデータの状態をData at Rest(保存状態)、Data in Transit(転送状態)、Data in Use(利用状態)の3つに分類し、Data at RestとData in Transitの状態での暗号化を指示。その後のDoD(Department of Defense)の文書でもこの原則に従って暗号化のガイドがされている。



NSAによるデータの3モード

TDE

既存の表領域の暗号化
Offline Encryption Conversion

- SQLの発行だけで、高速に既存の表領域を暗号化
  - ALTER DATABASE DATAFILE '<データファイル名>' ENCRYPT; (11.2、12.1、12.2)
  - ALTER TABLESPACE <表領域名> ENCRYPTION OFFLINE ENCRYPT; (12.2のみ)
- 移行のための特別なディスク領域は必要なし
- 表領域、データファイルがオフライン状態の時に実施可能
  (12.2ではOnline Encryption Conversion機能を利用することでオンライン状態でも暗号化可能)
- データファイル単位でのパラレル実行が可能
- 暗号アルゴリズム はAES128固定
- SYSTEM, SYSAUX, UNDO表領域も暗号化可能 (12.2のみ)
- 11.2.0.4, 12.1.0.2は、以下のパッチ適用が必要
  - Enable Transparent Data Encryption (TDE) Using Fast Offline Conversion in 11.2.0.4 and 12.1.0.2
    （ドキュメントID 2148746.1）
  - 12.1は、12.1.0.2.160719 Database Proactive Bundle Patch (Jul 2016)に含まれる

TDE

データの暗号化
～ TDE性能劣化の勘所 (要は通常のSQLチューニング)

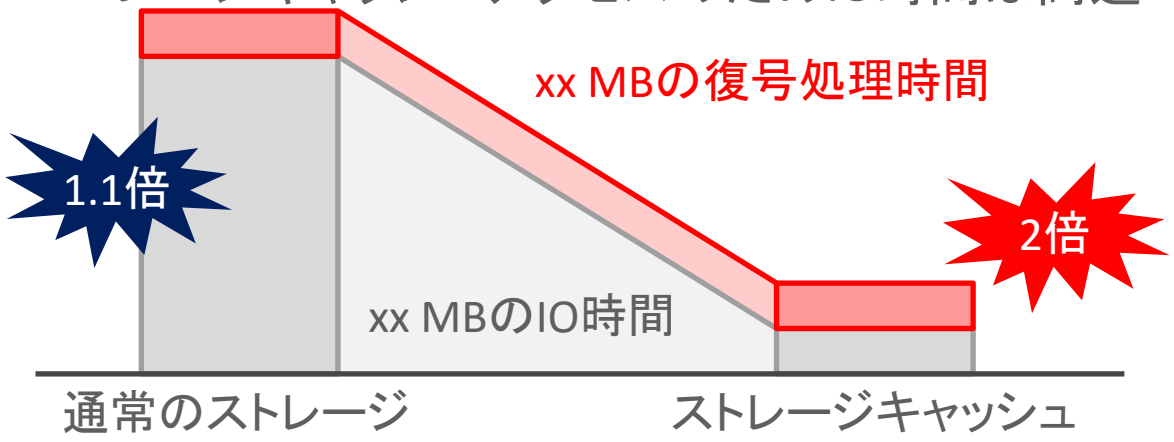
性能劣化はIO数に依存
(メモリ上のデータへのアクセスではオーバーヘッドは発生しない)

- 実行計画を確認したら毎回ディスクアクセス (Direct Path Read) していた例
  - 表のサイズがキャッシュサイズと比較して小さい場合でも実行計画を確認しましょう

```
実行計画
----------------------------------------
Plan hash value: 1445457117

----------------------------------------
| Id  | Operation         | Name      |
----------------------------------------
|   0 | SELECT STATEMENT  |           |
|   1 |  TABLE ACCESS FULL| EMPLOYEES |
----------------------------------------
```

- ストレージが高速(キャッシュ)で相対的にオーバーヘッドが大きく見えた例
  - オラクルから見るとディスクアクセスだがストレージキャッシュアクセスのためIO時間は高速



性能影響の検証はIOが多い処理 (もっとも性能影響を受ける可能性のある処理) を中心に
ただし、すべての処理のでそれだけのオーバーヘッドが発生するわけではありません

ファイングレイン・アクセス制御

1. Data Redaction
2. Virtual Private Database
3. Real Application Security
4. Client Identifier (エンドユーザーの情報をデータベースに渡す方法)

TDE

1. Data Redaction

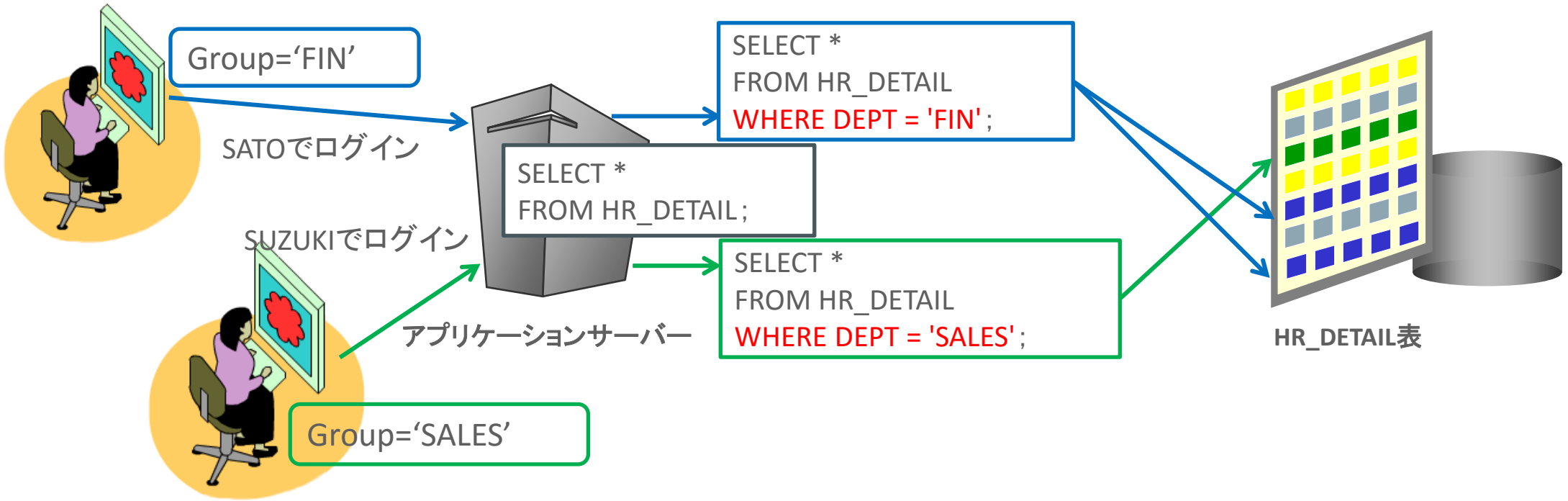
- ユーザーの権限やクライアント情報に応じてリアルタイムにデータを伏字化
- アプリケーションのコード修正は必要のないデータベース内で完結する列アクセス制御
- コールセンターやサポート業務などの職責に応じた顧客情報へのアクセス制御の実現や PCIDSSに対応したクレジットカード番号の表示、アプリ開発者の直接アクセスも制御

(注意点) アプリケーションが発行する固定化されたSQLで情報を隠す機能。自由検索時は推測攻撃を受ける可能性がある

EE

2. Virtual Private Database

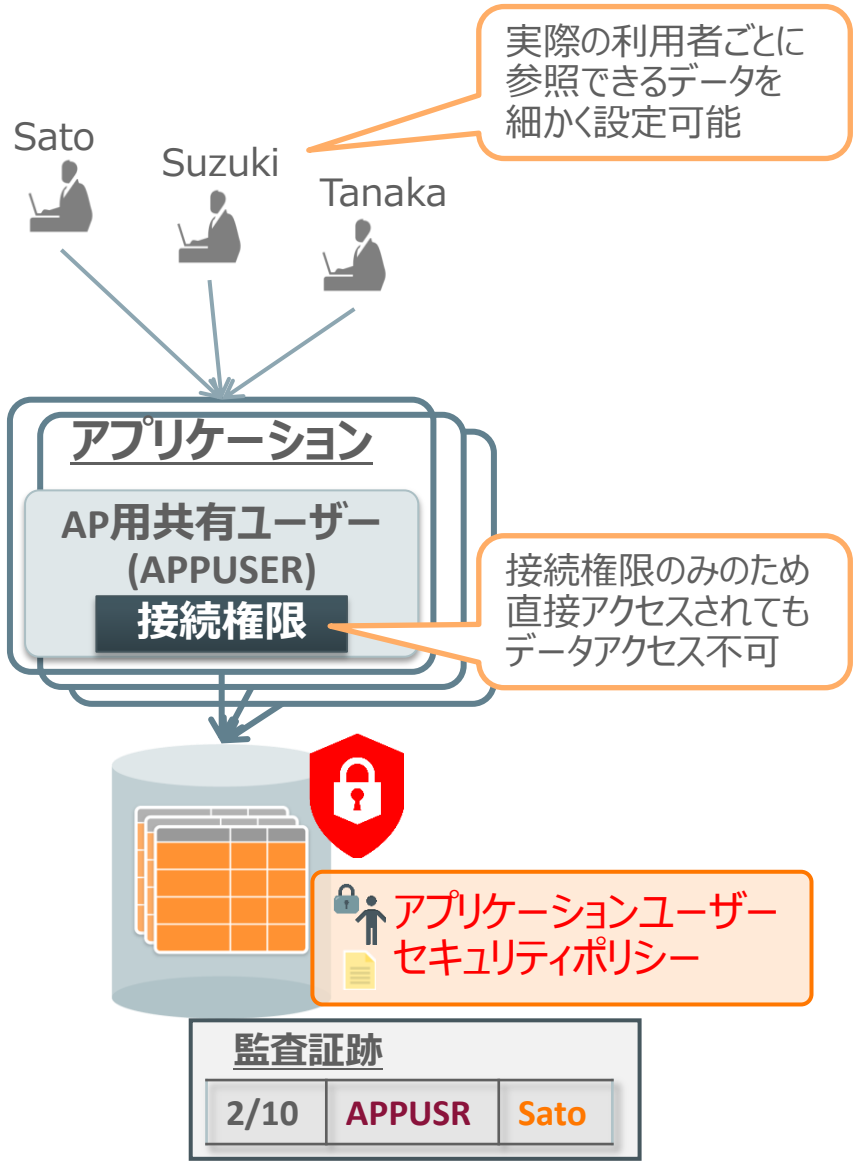
- 発行されたSQL文に内部的・強制的に条件句をつけて、アクセス制御を実施
- 条件に合致しないデータを戻さないアクセス制御を行および列レベルで適用



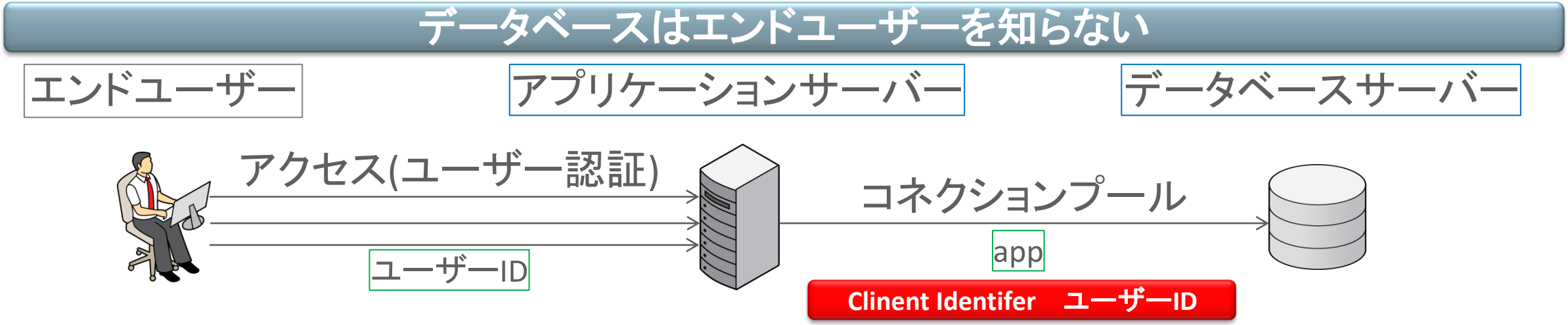
EE

3. Real Application Security

- データベース上でデータベース ユーザーではなく、実際の利用者 (アプリケーションユーザー)単位の アクセス制御を提供
  - アプリケーション屋アクセス経路に 同じセキュリティポリシーを適用可能
  - アプリケーションからの接続用共通 ユーザーへのアクセス権限付与は不要
  - アプリケーションユーザーに対して、 表の行や列単位の詳細なアクセス制御を提供
  - 監査証跡にアプリケーションユーザー名を 記録可能



4. Client Identifier
～ エンドユーザーの情報をデータベースに渡す方法



- エンドユーザーのIDをデータベース接続のセッションに紐づくクライアント情報の一つとして設定可能
  - 他にはDBユーザー名、クライアントホスト名、クライアントIPアドレス、クライアントモジュール名、OSユーザー名、利用言語設定など
- 設定した値はSQLで参照可能
  - データベース監査機能の証跡に自動的に格納

Client Identifierの設定方法

SQL*PLUSの場合

SQL*PLUSの場合、接続後に以下を実行

execute dbms_session.set_identifier(‘任意の値’)

Ex)
execute dbms_session.set_identifier(‘user=tanaka id=001234‘)

JDBCの場合

DBへ接続オープン後、以下を追加

JDBC 11g

```
String metrics[] =
new String[OracleConnection.END_TO_END_STATE_INDEX_MAX];
metrics[OracleConnection.END_TO_END_CLIENTID_INDEX] = "任意の値";
conn.setEndToEndMetrics(metrics, (short) 0);
```

JDBC 12c

```
conn.setClientInfo("E2E_CONTEXT.CLIENT_IDENTIFIER", "任意の値");
```

ユーザ識別子を取得

```
EXECUTE DBMS_SESSION.SET_IDENTIFIER('APP001');

PL/SQLプロシージャが正常に完了しました。

SELECT SYS_CONTEXT(‘USERENV‘, 'CLIENT_IDENTIFIER') FROM DUAL;

SYS_CONTEXT('USERENV','CLIENT_IDENTIFIER')
----------------------------------------------------------------
APP001
```

.NETの場合

DBへ接続オープン後、以下を追加

conn.ClientId = “任意の値” ‘

参考) Client Identifierをコネクションプーリングを利用しているアプリケーションで利用する場合の処理の流れ

1. アプリケーション開始
2. コネクションプール取得
3. Client Identifier設定

（2.〜3.：この部分はフレームワークでアプリケーションから隠ぺい可能）

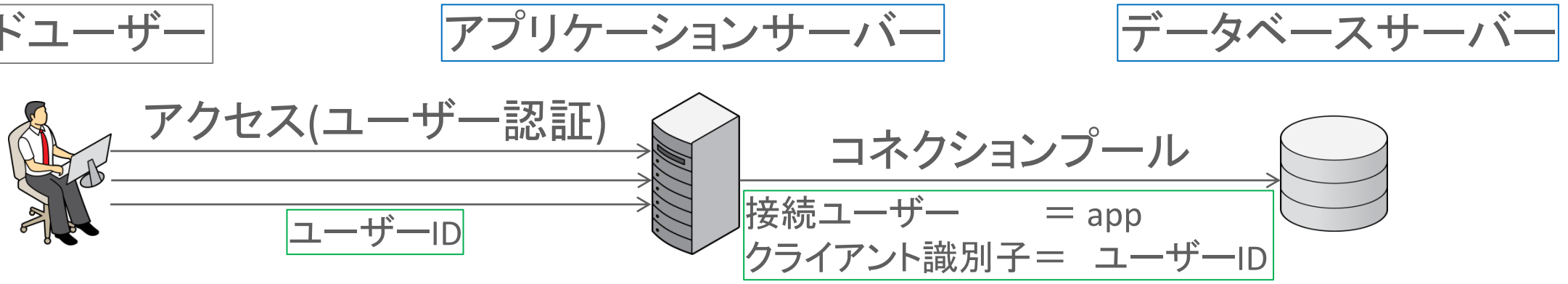
4. データベース処理(SQL実行)
5. Client Identifierの値を削除 (オプション)
6. コネクションプール開放

（5.〜6.：この部分はフレームワークでアプリケーションから隠ぺい可能）

7. アプリケーション終了

参考：WebLogicでのClient Identifierの自動設定

「接続時にクライアントIDを設定」機能



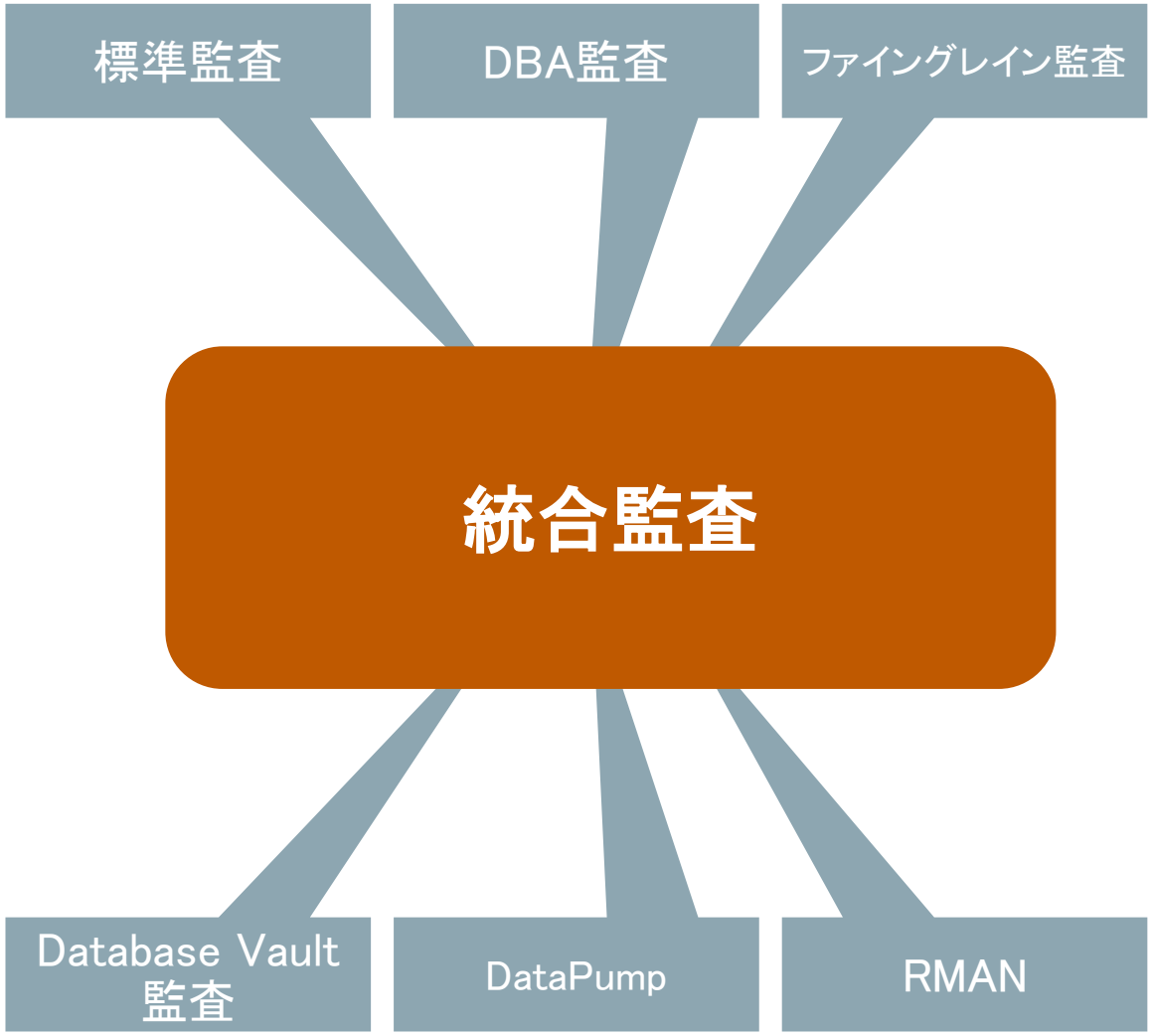
- DataSourceをgetConnectionする時にアプリケーションサーバーにログインしているユーザー名をデータベースに自動的に伝播する仕組み
  - WebLogic Serverで認証されたユーザー名が自動的にClient Identifierに設定される

監査
～ 11gまでの監査

	①必須監査（オペレーティング・システム監査）	②DBA監査	③標準監査（任意監査）	④ファイングレイン監査（任意監査）
必要Edition	全エディション	全エディション	全エディション	Enterprise Edition
監査対象	・インスタンス起動 ・インスタンス停止 ・管理者権限によるデータベース接続	・データベース管理者としてログインしたユーザーのデータベース操作	・データベースへの操作（ログイン、CREATE/ALTER/DROPなどのアクション、UPDATE、DELETEなどのオブジェクトへの操作）	・特定のデータ（列名、条件指定可能）へのアクセス（SELECT） ・Oracle10gからはUPDATE、DELETE、INSERTへも可能
監査証跡出力先	・OSファイル	・OSファイル / システムビューア（Win） ・Syslog（10gR2～）	・DBA_AUDIT_TRAILビュー ・OSファイル / システムビューア（Win） ・Syslog（10gR2～） ・XMLファイル（10gR2～）	・DBA_FGA_AUDIT_TRAILビュー ・ユーザー定義表 ・メール送信も可能 ・XMLファイル（10gR2～）
取得可能な監査証跡	・OSによって生成された監査レコード ・データベース監査証跡レコード ・常に監査されるデータベース関連のアクション	・時刻 ・操作（SQL文全体） ・データベースユーザー名/権限 ・OSユーザー名/端末 ・終了コード	・時刻 ・操作（SQL文の種類） ・データベースユーザー名/権限 ・OSユーザー名/端末 ・終了コード	・時刻 ・データベースユーザー ・OSユーザー名/端末 ・アクセスしたオブジェクト名 ・ファイングレイン監査ポリシー名 ・操作（SQL文全体） ・ユーザー定義アクション

監査
～ 12cからの新しい監査

- Unified Auditing (統合監査)
  - さまざまな監査やログをシンプルに統合
    - ポリシー定義、ログ出力先の一元化
    - 詳細な監査条件、監査除外条件を設定可能
    - 今まで取れなかった内容も取得可能
    - 非同期モードのサポートによる性能改善
    - 監査ログ保全の強化



監査

1. デフォルトで監査される項目
2. 監査の設定方針
3. アプリからのアクセスの監査
4. 監査証跡の分析例

1. デフォルトで監査される項目

- Oracle Database 11g以降では、データベースへの変更操作など重要な操作の監査がデフォルトで設定されています
  - Oracle Database 12cR1のデフォルトの監査設定 (標準監査)

    http://docs.oracle.com/cd/E57425_01/121/DBSEG/audit_config.htm#CHDGEIDA
  - Oracle Database 11gR2のデフォルトの監査設定 (Unified Auditing)

    http://docs.oracle.com/cd/E16338_01/network.112/b56285/auditing.htm#CHDEAEHA
  - 11gでは接続の監査を成功・失敗に関わらず取得。12cでは接続の監査は失敗時にのみ取得
  - 監査証跡のライフサイクル管理(過去の監査証跡の削除運用)に注意

2. 監査の設定方針

- データベース変更、アカウント管理、権限付与など重要な操作
- 重要なデータへのアクセス

- 管理者など強力な権限でDBに直接アクセスし、アドホックなクエリを実行できる人のすべての操作を不正抑止の観点から取得を検討
- 接続の監査は誰のものをどのような条件で取得するのか要検討

3. アプリからのアクセスの監査

- 実際にアクセスをしている利用者(エンドユーザー)をどのように把握するか？
  1. アプリケーションでログを取得する
     - アプリケーション以外からのアプリケーションユーザーでのアクセスログの取得は必要
     - データベース監査ログでアプリケーションからのアクセスログの取得を除外することを考慮
  2. データベースの監査ログにエンドユーザー名を記録する
     - Client Identifierの利用を考慮

(再掲) 最小権限の原則と職務分掌実現イメージでの監査の設定方針

ユーザー種別	ユーザー名	付与する権限	アカウント管理方法	監査
デフォルトユーザー（非個人）	SYS	変更しない(SYDBA、DBA)	基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須
	SYSTEM	変更しない(DBA)	基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須
	DVO	変更しない(DV_OWNER)	設定後は基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須
	OUTLN	変更しない(各ユーザーごと)	利用しないのであれば、EXPIRED & LOCKEDに。	すべて必須
	DBSNMP	変更しない(OEM_MONITORなど)	直接利用不可。パスワードは周知しない状態で厳密に管理	接続失敗は必須
アプリ用オブジェクト所有者（非個人）	APP1	必要なオブジェクト作成権限、領域割り当てのための権限	必要なオブジェクト作成後、EXPIRED & LOCKEDに。	すべて必須
アプリ接続用ユーザー（非個人）	APP1_BI	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_WEB1	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_WEB2	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_Batch1	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
	APP1_Batch2	アプリを利用するために必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)は必須
インフラ管理/アプリ管理・利用用個人アカウント	USER1	DBA、もしくは管理・利用に必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)と構成変更、重要なデータへのアクセスは必須
	USER2	DBA、もしくは管理・利用に必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)と構成変更、重要なデータへのアクセスは必須
	USER3	DBA、もしくは管理・利用に必要な最小の権限をまとめたロール	PROFILEで適切に管理	少なくとも接続(成功・失敗)と構成変更、重要なデータへのアクセスは必須
緊急時利用アカウント（非個人）	URGENT	DBA（アプリデータへのアクセス権限を含む）	基本的に利用不可。パスワードは周知しない状態で厳密に管理	すべて必須

通常利用しないユーザーのアクティビティは要監査

データへのアクセスも要考慮

4. 監査証跡の分析例

例1) 昨日、誰がどこからアクセスしてきた？

統合監査

```
select dbusername, os_username, userhost, return_code, count(*)
from unified_audit_trail
where action_name='LOGON'
and event_timestamp between trunc(systimestamp)-1 and trunc(systimestamp)
group by dbusername, os_username, userhost, return_code;
```

DBUSERNAME	OS_USERNAM	USERHOST	RETURN_CODE	COUNT(*)
SYS	oracle	dbsec.jp.oracle.com	28009	1
APP1	oracle	dbsec.jp.oracle.com	1017	1
PUKU	oracle	dbsec.jp.oracle.com	1017	2
USER01	oracle	dbsec.jp.oracle.com	1017	3
APP	oracle	dbsec.jp.oracle.com	1045	1
ORACLE	oracle	dbsec.jp.oracle.com	1017	2
HR	oracle	dbsec.jp.oracle.com	1017	1
SYSEM	oracle	dbsec.jp.oracle.com	1017	2
SYSTEM	oracle	dbsec.jp.oracle.com	1017	9

セッション監査(特にログイン失敗)は不正アクセス検知の第1歩。

1. 大量にログイン失敗しているアカウントはないか
2. 想定外のOSユーザー、ホストからアクセスがないか
   - 今回は12cデフォルトのログイン失敗のログしか取得していないが、ログイン成功でも想定外の場所からのログインは要確認

標準監査

```
select username, os_username, userhost, returncode, count(*)
from dba_audit_trail
where action_name='LOGON'
and timestamp between trunc(sysdate)-1 and trunc(sysdate)
group by username, os_username, userhost, returncode;
```

例2) 利用しているアプリケーションは？

統合監査

```
select dbusername, userhost, client_program_name, count(*)
from unified_audit_trail
where action_name='LOGON'
and event_timestamp between trunc(systimestamp)-1 and trunc(systimestamp)
group by dbusername, userhost, client_program_name;
```

DBUSERNAME	USERHOST	CLIENT_PROGRAM_NAME	COUNT(*)
SYS	dbsec.jp.oracle.com	sqlplus@dbsec.jp.oracle.com (TNS V1-V3)	6
APP1	apsvr.jp.oracle.com	JDBC Thin Client	32
APP1	apsvr.jp.oracle.com	sqlplus@dbsec.jp.oracle.com (TNS V1-V3)	1
APP2	dbsec.jp.oracle.com	sqlplus@dbsec.jp.oracle.com (TNS V1-V3)	2

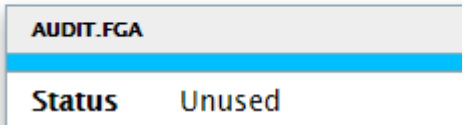
通常、特定のアプリケーションからしか接続してこないアカウントがSQL*Plusなどの別のアプリケーションから接続してきたら要注意

特定データへのアクセスログも同様にどのアプリケーションからアクセスしているかを確認することで、不正アクセスの検知につながるケースもある

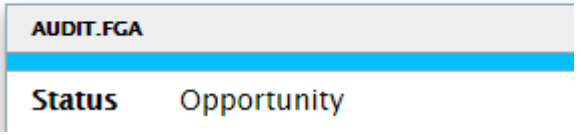
参考) DASATでの利用していない機能のstatus表記

- 日本語版では「Unused」、英語版では「Opportunity」
  - 「便利な機能なのでこの機会にぜひ利用をご検討ください」という意味です

ファイングレイン監査



Fine Grained Audit



Agenda

1. データベース・セキュリティの基本的な考え方
2. データベースのセキュリティセルフチェックツール(DBSAT)紹介
3. DBSATレポートの読み方と考慮すべき項目
4. その他の要考慮事項
5. 参考資料ご紹介

その他の要考慮事項

- アクセス経路の把握と制限
  - 想定外経路からのアクセスは攻撃の可能性がある → 要監視項目
  - アクセスマトリックスに「誰が」、「何に」に加え「どこから」を追加することを検討
- スクリプトにパスワードを書かない
  - ユーザーの認証情報の奪取は情報漏洩事件の原因のひとつ
  - 認証情報管理機能を持つ運用監視ツールの利用を検討
- 本番環境以外のセキュリティ (バックアップ、開発・テスト環境)
  - バックアップが暗号化されていない
  - 開発・テスト環境で本番データが利用されている
  - 開発・テスト環境と本番環境でパスワードが同じ
- ID管理

Agenda

1. データベース・セキュリティの基本的な考え方
2. データベースのセキュリティセルフチェックツール(DBSAT)紹介
3. DBSATレポートの読み方と考慮すべき項目
4. その他の要考慮事項
5. 参考資料ご紹介

各種ガイドライン

- Oracle Databaseセキュリティ・ガイド
  Oracle Databaseの安全性の維持

  http://docs.oracle.com/cd/E57425_01/121/DBSEG/guidelines.htm#CHDCEBFA (12c)

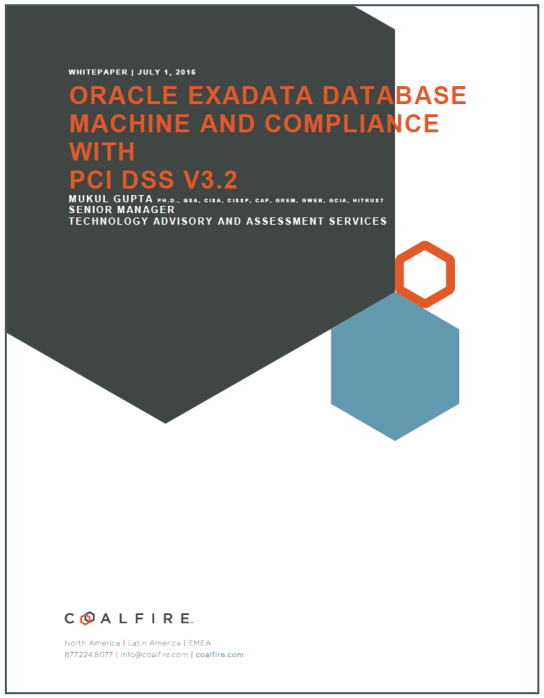
  http://docs.oracle.com/cd/E16338_01/network.112/b56285/guidelines.htm#CHDCEBFA (11gR2)

- データベース・セキュリティ・コンソーシアム
  データベースセキュリティガイドライン

  http://www.db-security.org/report/dbsc_guideline_ver2.0.pdf

参考情報

- **Oracle Exadata：**
  PCIDSS 3.2対応 White Paper



URL: https://www.oracle.com/engineered-systems/exadata/resources.html

- **これだけは押さえたい! データベースセキュリティ (マイナビ)**

第1回：データベース管理者、信じていいの?
性善説の運用を考える(1)

第2回：データベース管理者、信じていいの?
性善説の運用を考える(2)

第3回：物理セキュリティとデータアクセスで
安全性と可用性を両立

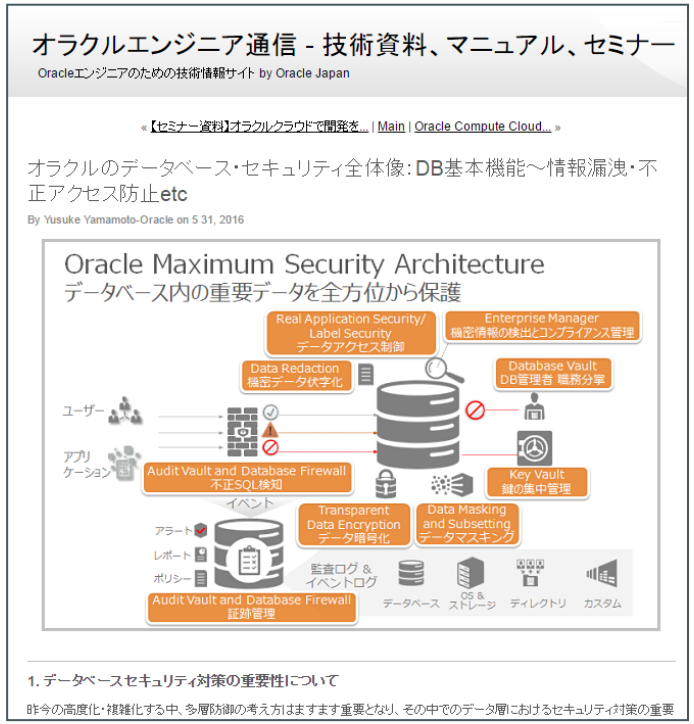
第4回：考慮不足だと危険? データベース
監査の設計を考える(前編)

第5回：考慮不足だと危険? データベース
監査の設計を考える(後編)

第6回：守りも万全! データベースの
暗号化のポイントを考える

URL: http://news.mynavi.jp/series/db_security/001/

- **オラクルエンジニア通信：**
  オラクルのデータベース・セキュリティ全体像



URL: https://blogs.oracle.com/oracle4engineer/

- **データベースインサイダー：**
  セキュリティ対策

URL: http://www.oracle.co.jp/dbi/

Safe Harbor Statement

The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Integrated Cloud

Applications & Platform Services


ORACLE®