Oracle Database Security 23ai

# Oracle Database Security features (default)

**User accounts**. When you create user accounts, you can secure them in a variety of ways. You can also create password profiles to better secure password policies for your site.

**Authentication methods**. Oracle Database provides several ways to configure uthentication for users and database administrators. For example, you can authenticate users on the database level, from the operating system, and on the network

**Privileges and roles**. You can use privileges and roles to restrict user access to data.

**Application security**. The first step to creating a database application is to ensure that it is properly secure

**User session information using application context**. An application context is a name-value pair that holds the session information. You can retrieve session information about a user, such as the username or terminal, and restrict database and application access for that user based on this information

**Database access on the row and column level using Virtual Private Database**. A Virtual Private Database policy dynamically imbeds a WHERE predicate into SQL statements the user issues

**Classify and protect data in different categories**. You can find all table columns in a database that hold sensitive data (such as credit card or Social Security numbers), classify this data, and then create a policy that protects this data as a whole for a given class.

**Network data encryption**. Manually Encrypting Data, explains how to use the DBMS\_CRYPTO PL/SQL package to encrypt data as it travels on the network to prevent unauthorized access to that data.

**Thin JDBC client network configuration**. You can configure thin Java Database Connectivity (JDBC) clients to securely connect to Oracle databases

**Strong authentication**. You can configure your databases to use strong authentication with Oracle authentication adapters that support various third-party authentication services, including SSL with digital certificates

## Managing User Authentication and Authorization

### About User Security

* Username: unique (domain name)
* Minimum Database Privileges, No admin option
  + GRANT CREATE SESSION TO xxxxx;
* Minimum Object Privileges, No admin option
* Correct role
* Profile type:
  + Profile\_system
  + Profile\_application;
  + Profile\_enduser;
* Profile has:
  + IDLE\_TIME
  + CONNECT\_TIME
  + SESSIONS\_PER\_USER
  + FAILED\_LOGIN\_ATTEMPTS
  + PASSWORD\_LIFE\_TIME
  + PASSWORD\_REUSE\_TIME
  + PASSWORD\_REUSE\_MAX
  + PASSWORD\_VERIFY\_FUNCTION
  + PASSWORD\_LOCK\_TIME
  + PASSWORD\_GRACE\_TIME
* Changing password for user
  + Normal user, , alter user sys identified by …
  + Sys user: ORAPWD, alter user sys identified by …

### Creating User Accounts

### Altering User Accounts

### Configuring User Resource Limits

### Dropping User Accounts

### Predefined Schema User Accounts Provided by Oracle Database

### Predefined Schema User Accounts Provided by Oracle Database

## Application Development Security

## Controlling Access to Data

## Securing Data on the Network

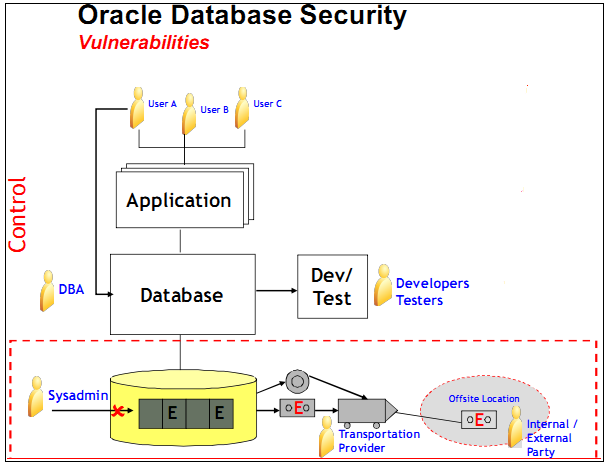
## Managing Strong Authentication

## Monitoring Database Activity with Auditing

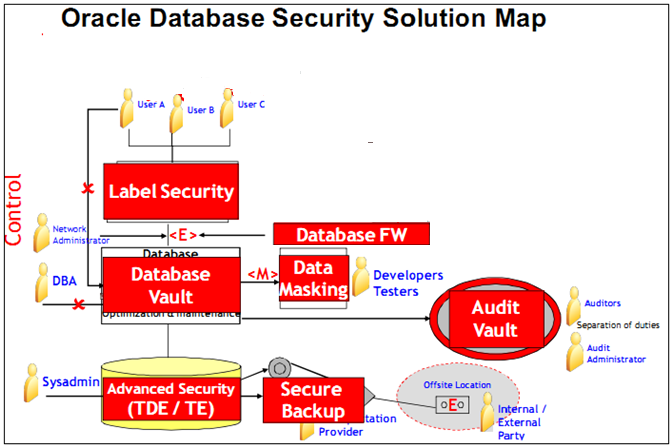
# Other database security products.

## Oracle Security product architechturen summary

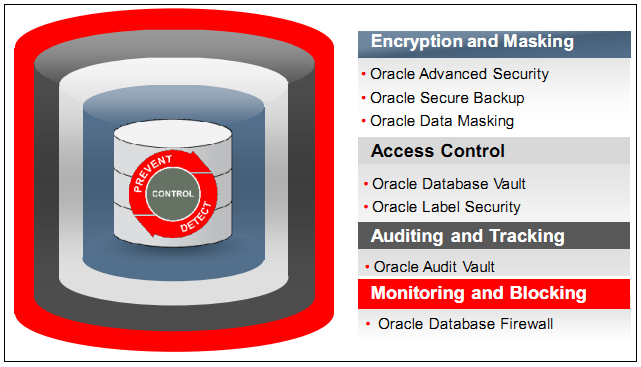
**Mô hình tổng quan của database**



**Mô hình Oracle Security tương ứng:**



* Mô hình Security tổng quan



**Oracle Label Security**. Oracle Label Security applies classification labels to data, allowing you to filter user access to data at the row level

**Oracle Database Vault**. Oracle Database Vault provides fine-grained access  
control to your sensitive data, including protecting data from privileged users Oracle Enterprise User Security. Oracle Enterprise User Security enables you to manage user security at the enterprise level.

**Oracle Enterprise Manager Data Masking and Subsetting Pack**. Data Masking and Subsetting Pack helps reduce this risk by irreversibly replacing the original sensitive data with fictitious data so that production data can be shared safely with IT developers or offshore business partners

**Oracle Audit Vault and Database Firewall**. Oracle Audit Vault and Database  
Firewall collects database audit data from sources such as Oracle Database audit trail tables, database operating system audit files, and database redo logs

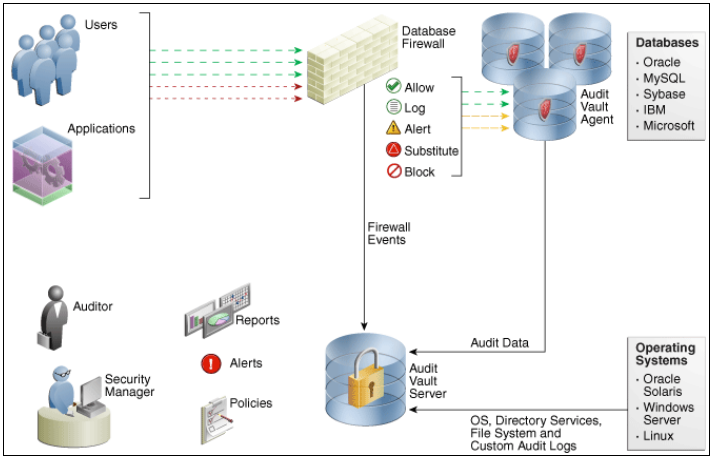
**Oracle Key Vault**. Oracle Key Vault enables you to accelerate security and  
encryption deployments by centrally managing encryption keys, Oracle wallets, Java keystores, and credential files. It is optimized for Oracle wallets, Java keystores, and Oracle Advanced Security Transparent Data Encryption (TDE) master keys.

## Oracle Audit Vault

Oracle Audit Vault and Database Firewall bảo vệ database và các thành phần liên quan (OS) bằng 3 cách:

* Cung cấp firewall có thể theo dõi, cho phép hoặc ngăn chặn các truy vấn SQL tới database ngay ở tầng mạng dựa trên cách chính sách của nó.
* Thu thập dữ liệu Audit từ các database và tạo báo cáo audit
* Cung cấp hàng chục build-in cho phép tùy chỉnh cấu hình các thông báo và cảnh báo phù hợp

Kiến trúc tổng quan của AVDF



* Đôi với mỗi DB cần bảo vệ, sẽ có 1 Audit Vault Agent tương ứng và có 1 Fire Wall nằm ở trên mạng (ngăn cách ứng dụng và DB)
* Dữ liệu từ AV Agent sẽ được gửi về tổng hợp ở Audit Vault Server
* Fire Wall có khả năng theo dõi lưu lượng dữ liệu tới DB, nó có khả năng cảnh báo hoặc chặn những truy vấn SQL dựa trên các polices có sẵn

**Audit Vault Server**

* Bao gồm các tool cần thiết cho quá trình cấu hình AVDF. Nó lưu trữ các file cấu hình của AVFW và thu thập các dữ liệu audit, có thể coi nó như 1 data warehouse nội bộ. Người sử dụng có thể cấu hình và tùy chỉnh báo cáo dựa trên nhưng thông tin có sẵn ở Audit Vault Server
* Sử dụng 1 Oracle Database nhúng để lưu trữ và quản lý dữ liệu. Database này được enable tính năng Database Vault dùng để quản lý truy cập người dùng. Người sử dụng được khuyến cáo không nên quản lý database này mà để tự nó quản lý

Các tính năng chính:

* Thu thập, quản lý dữ liệu audit
* Quản lý AV agent
* Quản lý FW
* Quản lý các chính sach bảo mật của AVFW
* Quản lý các thông báo và cảnh báo
* Lưu trữ dữ liệu, các phương pháp audit
* Tạo, lưu báo cáo
* Quản lý User sử dụng
* Liên kết với các chương trình thứ 3

Phân loại user:

* Super Administrator:
  + Được phép tạo admin hoặc super admin
  + Cài đăt, cấu hình tham số
* Administrator
  + Chỉ được phép xem dữ liệu

## Oracle Database FireWall

Concept:

* Là 1 hệ thống tường lửa có thể bảo vệ tính an toàn cho các database (Oracle, MySQL, Microsoft SQL Server …) trên nền tảng (Linux, UNIX, and Microsoft Windows)
* Nó ngăn chặn các cuộc tấn công, ghi lại log của các hoạt động, cung cấp các tool tăng cường các tính năng bảo mật của DB hiện có như mã hóa dữ liệu và xác thực người dùng

Các chế độ hoạt động:

* DPE Mode: đưa ra cảnh báo và cho phép/ ngăn chăn các truy vấn SQL đến
* DAM Mode: chỉ cảnh báo
* Proxy

Nguyên lý hoạt động:

* FW làm việc dựa trên việc phân tích ý nghĩa câu truy vấn mà client gửi tới. Điều này cung cấp 1 khả năng bảo vệ tốt hơn rất nhiều
* Không giới hạn độ dài độ dài của các câu truy vấn cần phải phân tích. Hiển thị 2000 ký tự đầu tiên và áp dụng cho mọi loại truy vấn (ví dụ như ANSI SQL...)
* FW ko làm ảnh hưởng tới hiệu năng làm việc của hệ thống

Các tính năng chính của Firewall:

* Tạo các chính sách bảo mật (policy file) đến và đi thông qua firewall
* Firewall sẽ dựa trên các chính sách bảo mật và thông tin của các truy vấn SQL để quyết định các hành động tiếp theo (cho phép, chặn, cảnh báo ...)
* Firewall có khả năng tích lũy những kết quả từ trước (lưu giữ nhưng cấu hình chính sách và kêt quả thực thi đối với những dữ liệu từ trước) để sinh ra những chứng sách thích hợp. Từ đó,chúng chia những câu lệnh SQL thành các nhóm, mỗi nhóm sẽ áp dụng các chính sách riêng
* FireWall ghi lại log để:
  + Giám sát và tạo báo cáo t
  + Tạo mớivà cải tiến các chính sách bảo mật
  + Không phụ thuộc vào log của database (ko ảnh hưởng tới DB)
* Các loại log bao gồm
  + Traffic log: (user, ip, Os, client program...)
  + Event log (các sự kiện xảy ra đối với FW)
  + Admin log (các thao tác của admin đối vs FW)

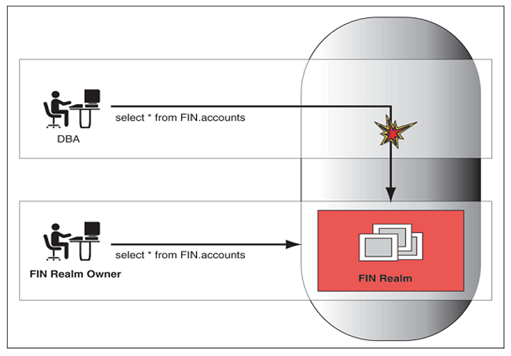
**Audit Vault Agent**

* Được cài đặt trên mỗi DB
* Làm nhiệm vụ thu thập audit file của mỗi DB và gửi về cho Audit Vault Server
* Mỗi khi Agent bị stop thì DB vẫn sinh audit trail file và khi nó được resume trở lại thì Agent sẽ chuyển những audit file ngay sau khi nó bị stop cho Audit Vault Server ( Cơ chế giống DataGuard chuyển archivelog file)

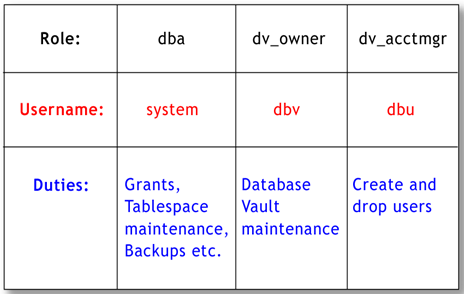
## Oracle Database Vault

Khái niệm:

* Oracle Database Vault điều khiển truy cập dữ liệu của DB từ mọi user (kể cả những user có quyền hạn lớn như admin...)
* Tăng cường khả năng bảo mật của database, bảo vệ chống lại các mối đe dọa từ bên trong và tuân thủ nghiêm ngặt các quy định.
* Phân chia quyền lực một cách rõ ràng: User bình thường chịu trách nhiệm với ứng dụng,dữ liệu. DBA chỉ có khả năng quản trị hệ thống



Các User - quyền hạn



Các thành phần của Oracle Database Vault:

* Oracle Database Vault Access Control Components:
  + Nhiệm vụ: Điều khiển truy câp dữ liệu trong database
  + Bao gồm các thành phần:
    - Realms: một tập hợp các thành phần như object, schema, roles cần được bảo mật
    - Command rules. Luật đặc biệt có thể được tạo ra kiểm soát cách người dùng thực hiện hầu hết các câu lệnh SQL như: SELECT, ALTER SYSTEM, DDL, DML
    - Factors: Tên của 1 biến hoặc 1 thuộc tính như: địa chỉ user, địa chỉ IP... Oracle Database Vault có thể nhận diện và đảm bảo an toàn.
    - Rule set: Một bộ quy tắc là một tập hợp của một hoặc nhiều quy tắc mà có thể kết hợp trong một lĩnh vực nào đ
    - Secure application roles: Một roles đặc biệt có thể được sử dụng dựa trên 1 tập luật (rule set)
* Oracle Database Vault Administrator (DVA)
  + Là 1 ứng dụng java được dựng cho người quản trị có thể cấu hình các chính sách thông qua các interface của người dùng
  + Tập hợp các báo cáo thông tin cấu hình, hoạt động của hệ thống
* Oracle Database Vault Configuration Assistant (DVCA)
  + Hiện thị, hỗ trợ quá trình cài đặt DV
* Oracle Database Vault DVSYS and DVF Schemas
  + Lưu trữ nhưng object được secure bởi DV (roles, views, accounts, functions...)
  + DVF Schema chứa những fuction public và khởi động/khôi phục nó trong quá trình hoạt động
* Oracle Database Vault PL/SQL Interfaces and Packages
  + Cho phép người quản trị DV hoặc người lập trình ứng dụng có thể cấu hình các chính sách cần thiết
* Oracle Database Vault and Oracle Label Security PL/SQL APIs
  + Cho phép người quản trị DV có thể nhậm biết chính sách của label và áp dụng nó vào database
* Oracle Database Vault Reporting and Monitoring Tools
  + Tạo báo cáo định kỳ vê hoạt động của DV

## Oracle Label Security

Giới thiệu:

* Trong DB có thể chứa nhiều dữ liệu nhảy cảm như: thông tin cá nhân, kế hoạch thương mại, kế hoạch tiếp thị ....
* Cho phép được truy cập bởi những người sử dụng không phù hợp có thể gây tổn hại cho người sở hữu thông tin hoặc cả hệ thống
* Tuy nhiên, dữ liệu nhạy cảm và ít nhạy cảm thường xuyên xen kẽ với nhau. Hạn chế quyền sử dụng, hoặc cách ly dữ liệu nhạy cảm tạo môi trường làm việc khó khăn và tốn kém tài nguyên

Khái niệm:

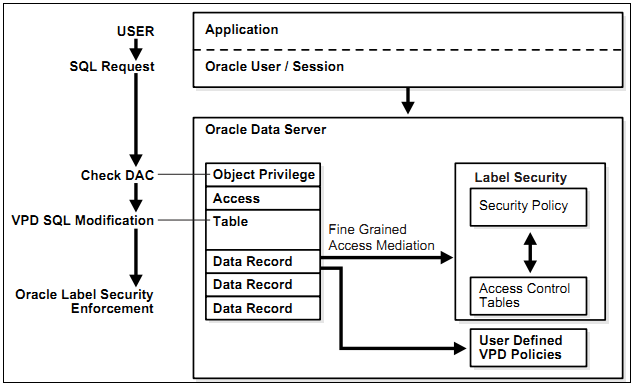
* Oracle Label Security giải quyết bài toán này bằng cách kiểm soát chế độ truy cập theo “row level” dựa trên nền tảng database riêng ảo (virtual private database technology)
* Nó kiểm soát truy cập nội dung các row dựa trên sự kết hợp nhãn dán của row với nhãn dán của người sử dụng và quyền hạn của họ. Admin có thể dễ dàng tùy chỉnh bằng giao diện đồ họa EM

Nguyên lý hoạt động

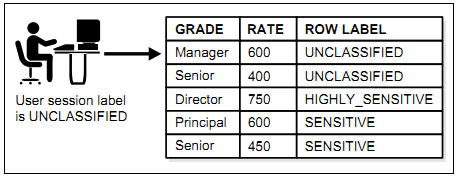
* Oracle Label Security sẽ tạo thêm 1 trường mới lưu trữ nhãn của row đó. Nhãn của row sẽ phản ánh độ nhạy cảm của từng row
* Cấp độ truy cập của người sử dụng được xác định bằng việc kết hợp giữa định danh người sử dụng và nhãn của row

Kiến trúc tổng quan

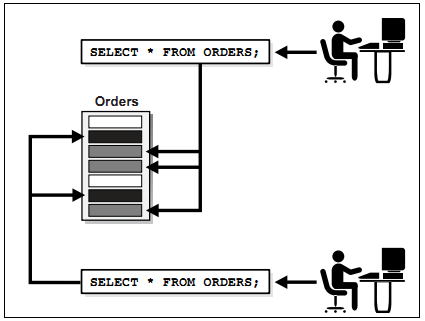
* DAC: Discretionary Access Control: cơ chế truy cập dữ liệu cơ bản mà Oracle DB cung cấp thông qua các quyền (SELECT, INSERT, UPDATE, DELETE)
* VPD: Virtual private Database: được sử dụng để tạo 1 DB ảo nhằm sửa đổi các câu truy vấn đến, điều khiển truy cập đến đúng địa điểm được phép. Thực chất là các chính sách được gán cho bảng
* Oracle Label Security: sử dụng polices để tạo nhãn. Kết hợp nhãn của người sử dụng và nhãn của các row để sửa đổi câu truy vấn
  + Có thể có nhiều polices cho 1 bảng, mỗi polices có 1 trường riêng
  + Một polices có thể áp dụng cho nhiều bảng



* Ví dụ: Oracle Label Security



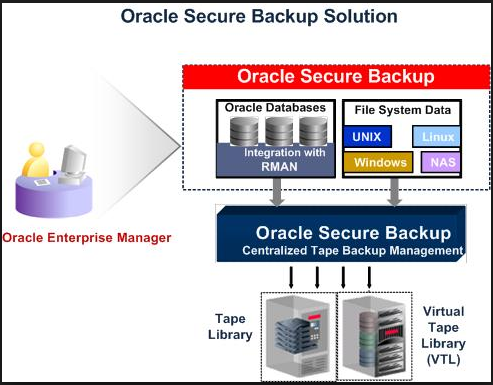
* Ví dụ: Virtual private Database



## Oracle Secure Backup

Khái niệm:

* Oracle cung cấp các phương pháp backup các file system và Oracle Database file đáng tin cậy và quản lý tập trung trên tape thông qua RMAN
* Oracle Secure Backup hỗ trợ gần như tất cả các tape driver trong các môi trường SAN và SCSI

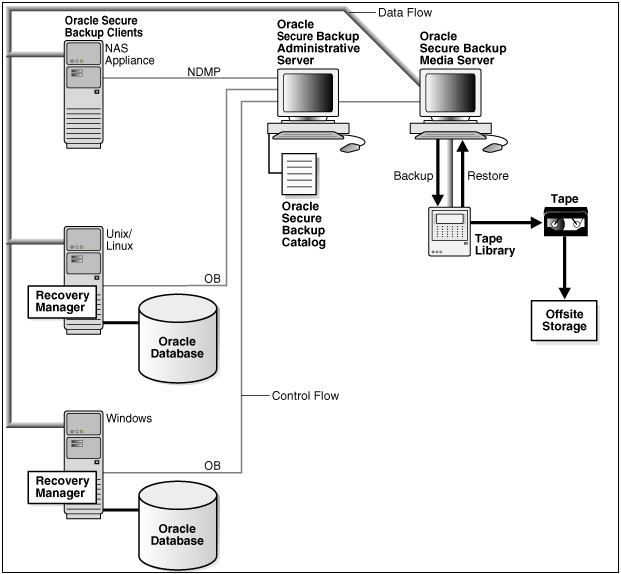


Các tính năng chính:

* Backup ra tape từ nhiều hệ điều hành, môi trường khác nhau
* Cho phép mã hóa dữ liệu, nén dữ liệu khi backup
* Xác định chính xác nhưng đối tượng muốn sao lưu
* Hỗ trợ nhiều mức độ incremental backup
* Hỗ trỡ Duplex database backups (dữ liệu được sao lưu thành nhiều bản và lưu vào các tape)
* Tối ưu hóa tài nguyên và tự động chia sẽ driver của tape
* Restore 1 cách nhanh chóng
* Giới hạn người sử dụng
* Làm việc theo chính sách được định nghĩa sẵn

Kiến trúc tổng thể:

* Oracle Secure Backup Administrative Server: quản lý tập trung cho cả hệ thống
  + Lưu trữ thông tin của các thiết bị, user của từng thiết bị trong hệ thống
  + Lưu trữ thông tin backup của các Database (Catalog)
  + Cấu hình chính sách backup
  + Cấu hình lịch backup
  + Cấu hình mã hóa hóa
* Oracle Secure Client là các Database cần được backup ra tape
  + Khi được gọi, Rman ở mỗi thiết bị sẽ thực thi backup database và chuyển dữ liệu tới MediaServer
* Oracle Secure Backup Media Server: Thiết bị kết nối trực tiếp với Tape
  + Nhận dữ liệu từ Client và move dữ liệu vào tape



## Oracle Data Masking

Khái niệm:

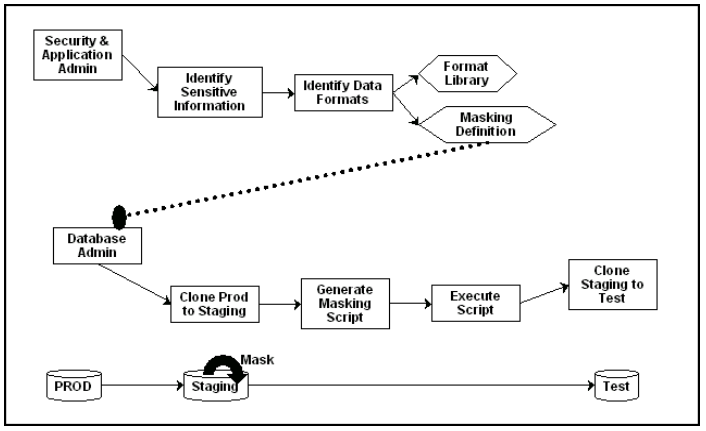
* Oracle Data Masking cung cấp 1 phương thức bảo vệ dữ liệu cho quá trình tester mà vẫn đảm bảo sự hiệu quả của công việc.
* Oracle Data Masking thay thế các dữ liệu nhạy cảm thành các dữ liệu giả (hư cấu) để các dữ liệu trong môi trường production có thể chia sẻ một cách an toàn với môi trường no-production.
* Những dữ liệu giả có đặc điểm tương tự dữ liệu thật để đảm bảo tính toàn vẹn của chúng. Tester sẽ chỉ được thao tác với những dữ liệu giả, ko bao giờ được thao tác với dữ liệu thật

Vai trò của người sử dụng Data Masking

* Application database administrator hoặc application developer: Những người sử dụng có có hiểu biết về các ứng dụng và các đối tượng trong DB. Những người này có thể tùy chỉnh (thêm, sửa, xóa...) vào các ứng dụng
* Information security administrator: Người sử dụng này quy định các chính sách bảo mật thông tin, thực thi và đưa ra các khuyến cáo các loại dữ liệu cẩn bảo vệ

Các tính năng chính của Data Masking Pack

* Masking format library:
  + Chứa 1 tập hợp các định dạng masking đã sẵn sàng được sử dụng. Định dang này có thể do người sử dụng tạo ra, cũng có thế do Oracle cung cấp
  + Mỗi tổ chức cần có những định dang masking để bảo vệ dữ liệu
* Masking definitions:
  + Định nghĩa kiểu masking data sẽ áp dụng cho một hoặc nhiều bảng trong DB. Để sử dụng, chỉ cần chỉ ra trường nào cần được masking và kiểu để masking, trường đó sẽ được đảm bảo các tính chất đã tồn tại như (unique, contraint...)
  + Có thể export và import vào các hệ thống khác nhau
* Quá trình làm việc với Data Masking



## Oracle Advanced Security

Khái niệm: Oracle Advanced Security cung cấp các tính năng cho phép bảo vệ dữ liệu bằng cách ngăn chặn các hành vi truy cập không hợp lệ. Bao gồm các công nghệ chính:

### Oracle Data Redaction:

* Oracle Data Redaction cho phép che giấu các kết quả truy vấn của các dữ liệu nhạy cảm trước khi được hiển thị bởi ứng dụng
* Sử dụng để bảo vệ dữ liệu khởi những mối nguy hiểm nội bộ
* Khi dữ liệu được load bên bộ nhớ, nó sẽ convert sang các ký tự đặc biệt (dựa vào các chính sách). Chính vì vậy ko ảnh hưởng tới độ chính xác của dữ liệu.
* Các kiểu Redaction
  + Full redaction: che giấu toàn bộ nội dung của các cột
  + Partial redaction: che giấu dữ liệu 1 phần: ví dụ: \*\*\*abcd
  + Regular expressions: kết hợp cả 2 loại
  + Random redaction: che giấu ngẫu nhiên, phụ thuộc vào loại dữ liệu
  + No redaction: ko che giấu, sử dụng để test

### Transparent Data Encryption:

* Mã hóa dữ liệu để chỉ có những user được ủy quyền mới có thể truy cập.
* Sử dụng để mã hóa để bảo vệ dữ liệu nhạy cảm trong một môi trường có khả năng không được bảo vệ, như đặt dữ liệu tại nơi backup được lưu trữ offsite
* Có thể mã hóa một table hoặc cả tablespace.
* Khi sử dụng Transparent Data Encryption, không cần phải sửa đổi các ứng dụng TDE tự động mã hóa dữ liệu khi ghi vào đĩa, và sau đó tự động giải mã dữ liệu khi ứng dụng truy cập nó. Key được tự động sinh và quản lý
* Quá trình mã hóa và giải mã là cực kỳ nhanh chóng bởi vì TDE có thể tối ưu hóa tốc độ bằng cách sử dụng bộ nhớ đệm (cache) của database

Oracle TDE (Transparent Data Encryption) là một tính năng bảo mật trong Oracle Database giúp mã hóa dữ liệu lưu trữ (data at rest) để bảo vệ thông tin nhạy cảm khỏi các truy cập trái phép. TDE hoạt động một cách "transparent" (trong suốt), nghĩa là các ứng dụng và người dùng không cần phải thay đổi cách thức truy cập dữ liệu khi mã hóa được áp dụng. TDE được thiết kế để giúp tuân thủ các quy định về bảo mật dữ liệu và bảo vệ dữ liệu khỏi các mối đe dọa tiềm ẩn.

Các Thành Phần Chính của Oracle TDE

1. **TDE Tablespace Encryption**:

* Mã hóa toàn bộ tablespace, bao gồm tất cả các bảng và chỉ mục bên trong tablespace đó.
  + Khi một tablespace được mã hóa, tất cả các dữ liệu được lưu trữ trong tablespace đó sẽ tự động được mã hóa khi ghi và giải mã khi đọc.

1. **TDE Column Encryption**:
   * Mã hóa các cột cụ thể trong một bảng.
   * Được sử dụng khi chỉ có một số cột chứa thông tin nhạy cảm cần mã hóa.
2. **Encryption Keys**:
   * **Master Encryption Key**: Là khóa mã hóa chính, được sử dụng để mã hóa các khóa mã hóa dữ liệu (data encryption keys). Khóa này được quản lý bởi Oracle Wallet hoặc Oracle Key Vault.
   * **Data Encryption Keys**: Được sử dụng để mã hóa dữ liệu thực tế trong các tablespace hoặc cột.
3. **Oracle Wallet**:
   * Là một nơi lưu trữ an toàn để giữ Master Encryption Key. Oracle Wallet cần được mở để cơ sở dữ liệu có thể truy cập các khóa mã hóa và thực hiện mã hóa hoặc giải mã dữ liệu.

Cách Hoạt Động của Oracle TDE

1. **Thiết lập Oracle Wallet**:
   * Trước khi sử dụng TDE, bạn cần cấu hình và mở Oracle Wallet để lưu trữ Master Encryption Key.

sql

Copy code

ADMINISTER KEY MANAGEMENT CREATE KEYSTORE 'path\_to\_wallet' IDENTIFIED BY 'password';

ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY 'password';

1. **Tạo Master Encryption Key**:
   * Tạo Master Encryption Key và lưu trữ nó trong Oracle Wallet.

sql

Copy code

ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY IDENTIFIED BY 'password';

1. **Mã hóa Tablespace**:
   * Khi tạo một tablespace mới và mã hóa nó.

sql

Copy code

CREATE TABLESPACE encrypted\_tbs DATAFILE 'path\_to\_datafile' SIZE 10M ENCRYPTION USING 'AES256' DEFAULT STORAGE(ENCRYPT);

1. **Mã hóa Cột**:
   * Mã hóa một cột cụ thể trong bảng.

sql

Copy code

CREATE TABLE employees (

emp\_id NUMBER,

emp\_name VARCHAR2(100),

emp\_ssn VARCHAR2(11) ENCRYPT USING 'AES256'

);

Lợi Ích của Oracle TDE

1. **Bảo mật dữ liệu**:
   * Bảo vệ dữ liệu nhạy cảm khỏi các truy cập trái phép, ngay cả khi ổ đĩa lưu trữ bị đánh cắp.
2. **Tuân thủ quy định**:
   * Giúp tuân thủ các quy định và tiêu chuẩn bảo mật dữ liệu như GDPR, HIPAA, và PCI DSS.
3. **Triển khai dễ dàng**:
   * Không yêu cầu thay đổi ứng dụng, mã hóa được thực hiện tự động và trong suốt đối với người dùng và ứng dụng.
4. **Hiệu suất**:
   * Oracle TDE được tối ưu hóa để giảm thiểu ảnh hưởng đến hiệu suất hệ thống, đảm bảo mã hóa và giải mã dữ liệu nhanh chóng và hiệu quả.
5. **Oracle Wallet**:
   * **Tệp Wallet (ewallet.p12)**: Tệp này lưu trữ các khóa mã hóa chính (Master Encryption Key) và các chứng chỉ bảo mật khác. Wallet là nơi chứa an toàn cho các khóa mã hóa cần thiết để mã hóa và giải mã dữ liệu trong cơ sở dữ liệu Oracle.
   * **Cách sử dụng**: Oracle Wallet phải được mở để cơ sở dữ liệu có thể truy cập các khóa mã hóa và thực hiện các hoạt động mã hóa và giải mã.
   * **Đường dẫn cấu hình**: Đường dẫn tới Wallet được xác định trong tệp cấu hình sqlnet.ora bằng tham số ENCRYPTION\_WALLET\_LOCATION.
6. **Oracle Wallet Configuration File**:
   * **Tệp cấu hình sqlnet.ora**: Tệp này chứa cấu hình của Oracle Net, bao gồm cả vị trí của Oracle Wallet. Trong tệp sqlnet.ora, bạn sẽ xác định đường dẫn đến Wallet, cho phép cơ sở dữ liệu biết vị trí tìm kiếm các khóa mã hóa.
   * **Cách sử dụng**: Cấu hình trong tệp sqlnet.ora là cần thiết để thiết lập môi trường cho Oracle Wallet và đảm bảo rằng cơ sở dữ liệu có thể truy cập Wallet để lấy các khóa mã hóa.

Ví dụ về Cấu hình Oracle Wallet và sqlnet.ora

1. **Tạo Oracle Wallet**:

sql

Copy code

ADMINISTER KEY MANAGEMENT CREATE KEYSTORE '/u01/app/oracle/admin/wallet' IDENTIFIED BY "MyPassword";

ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY "MyPassword";

ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY IDENTIFIED BY "MyPassword" WITH BACKUP;

1. **Cấu hình tệp sqlnet.ora**:
   * Tệp sqlnet.ora thường nằm trong thư mục $ORACLE\_HOME/network/admin/. Thêm hoặc chỉnh sửa mục sau để chỉ định đường dẫn đến Wallet:

plaintext

Copy code

ENCRYPTION\_WALLET\_LOCATION =

(SOURCE =

(METHOD = FILE)

(METHOD\_DATA =

(DIRECTORY = /u01/app/oracle/admin/wallet)

)

)

File

* **Oracle Wallet (ewallet.p12)**: Chứa các khóa mã hóa chính và các chứng chỉ bảo mật, đảm bảo an toàn cho việc mã hóa và giải mã dữ liệu.
* Auto-login wallet được thiết kế để tự động mở khóa các khóa mã hóa chính (Master Encryption Key) khi cơ sở dữ liệu khởi động, loại bỏ nhu cầu nhập mật khẩu mỗi khi mở Oracle Wallet. Được lưu trữ dưới dạng tệp cwallet.sso (định dạng auto-login).