Oracle® Database

Security Assessment Tool User Guide

Release 2.2.1

F21286-04

October 2020

About the Oracle Database Security Assessment Tool

The Oracle Database Security Assessment Tool (Oracle DBSAT) analyzes database configurations, users, their entitlements, security policies and identifies where sensitive data resides to uncover security risks and improve the security posture of Oracle Databases within your organization.

Benefits of Using Oracle Database Security Assessment Tool

Using Oracle DBSAT, you can:

- Quickly and easily assess the current security status and identify sensitive data within the Oracle Database.

- Reduce risk exposure using proven Oracle Database Security best practices, CIS benchmark recommendations and STIG rules.

- Leverage security findings to accelerate compliance with EU GDPR and other regulations.

- Improve the security posture of your Oracle Databases and promote security best practices.

> **✏ Note:**
>
> DBSAT is a lightweight utility that will not impair system performance in a measurable way.

You can use Oracle DBSAT report findings to:

- Fix immediate short-term risks

- Implement a comprehensive security strategy

- Support your regulatory compliance program
- Promote security best practices

Oracle Database Security Assessment Tool Components

The Oracle DBSAT consists of the following components:

- **Collector:**

  The **Collector** executes SQL queries and runs operating system commands to collect data from the system to be assessed. It does this primarily by querying database dictionary views. The collected data is written to a JSON file that is used by the DBSAT Reporter in the analysis phase.
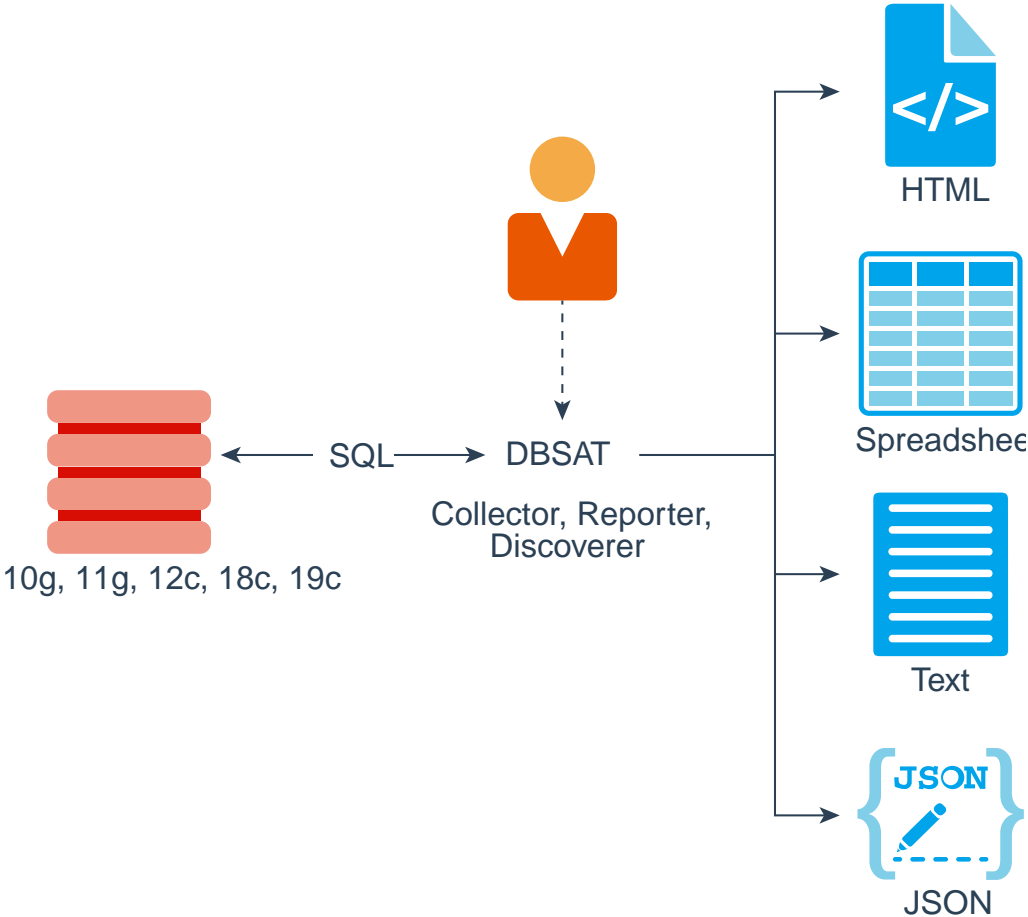
- **Reporter:**

  The **Reporter** analyzes the collected data and generates the Oracle Database Security Assessment Report in HTML, Excel, JSON, and Text formats. The Reporter can run on any machine: PC, laptop, or server. You are not limited to running the Reporter on the database server or the same machine as the Collector.

- **Discoverer:**

  The **Discoverer** executes SQL queries and collects data from the system to be assessed, based on the settings specified in the configuration files. It does this primarily by querying database dictionary views. The collected data is then used to generate the Oracle Database Sensitive Data Assessment Report in HTML and CSV formats. The Discoverer can run on any machine: PC, laptop, or server. You are not limited to running the Discoverer on the database server or the same machine as the Collector or Reporter.

The following figure shows the components, sources, and reports of the Oracle Database Security Assessment Tool.

Figure   Oracle DBSAT Components, Sources, and Reports

HTML

Spreadsheet

SQL → DBSAT

Collector, Reporter, Discoverer

Text

10g, 11g, 12c, 18c, 19c

JSON

For more information about the Collector, Reporter, and Discoverer, see Using the Collector and Reporter.

Prerequisites

The following sections outline the prerequisites for the Oracle Database Security Assessment Tool:

Supported Operating Systems

The database configuration collection queries run on most supported Oracle Database platforms. However, currently the OS data collection will be skipped on Windows platforms.

Oracle DBSAT runs on:

- Solaris x64 and Solaris SPARC64
- Linux x86-64
- Windows x64
- HP-UX IA (64-bit)
- IBM AIX (64-bit) & Linux on zSeries (64-bit)

Supported Database Versions

You can run the Oracle DBSAT on Oracle Database 10.2.0.5 and later releases on on-premises or in the Cloud, on Oracle Database Standard Edition 2 and Oracle Database Enterprise Edition. Oracle DBSAT can also be run against Oracle Cloud DBCS (PaaS) and Autonomous Databases.

> **Note:**
>
> Oracle Database Standard Edition 2 is available starting with Oracle Database 12c Release 1 (12.1.0.2). For 12.1.0.1, Oracle Database Standard Edition One and Oracle Database Standard Edition are available.
>
> Ref: https://docs.oracle.com/database/121/DBLIC/editions.htm#DBLIC109

Security Requirements

Oracle DBSAT output files are sensitive because they may reveal weaknesses in the security posture of your database. To prevent unauthorized access to these files, you must implement the following security guidelines:

- Ensure that the directories holding these files are secured with the appropriate permissions.

- Delete the files securely after you implement the recommendations they contain.

- Share them with others in their (by default) encrypted form.

- Grant user permissions to the Oracle DBSAT user on a short-term basis and revoke these when no longer necessary.

  For more information about Oracle DBSAT user privileges, see Collector Prerequisites.

> ⚠ **Caution:**
>
> This tool is intended to assist you in identifying potential sensitive data and vulnerabilities in your system. Further, the output generated by this tool may include potentially sensitive system configuration data and information that could be used by a skilled attacker to penetrate your system. You are solely responsible for ensuring that the output of this tool, including any generated reports, is handled in accordance with your company's policies.

Oracle Database Security Assessment Tool Prerequisites

DBSAT requires bash shell to be installed on Unix / Linux systems.

Bash Shell Prerequisites

Oracle DBSAT requires bash shell to be installed on Unix / Linux systems.

Zip and UnZip

Oracle DBSAT uses Zip and Unzip to compress or decompress the generated files. Oracle DBSAT searches for Zip and Unzip utilities in the default locations shown below. In order to use other Zip and Unzip utilities, update the following lines in the relevant script.

Windows (dbsat.bat script):

```
SET ZIP_CMD=%ORACLE_HOME%\bin\zip.exe
SET UNZIP_CMD=%ORACLE_HOME%\bin\unzip.exe
```

> ✎ **Note:**
>
> The Unzip utility is not included in Oracle Database 12.2 and higher. Ensure that you have installed a utility such as WinZip or WinRar, and add the path to the utility in the `SET UNZIP_CMD` parameter.

All other platforms (dbsat script):

```
ZIP=/usr/bin/zip
UNZIP=/usr/bin/unzip
DBZIP=${ORACLE_HOME}/bin/zip
```

The following are the prerequisites for the components of the Oracle Database Security Assessment Tool:

Collector Prerequisites

In order to collect complete data, the Oracle DBSAT Collector must be run on the server that contains the database, because it executes some operating system commands to collect process and file system information that cannot be obtained from the database. In addition, the Oracle DBSAT Collector must be run as an OS user with read permissions on files and directories under `ORACLE_HOME` in order to collect and process file system data using OS commands.

The Oracle DBSAT Collector collects most of its data by querying database views. It must connect to the database as a user with sufficient privileges to select from these views. Grant the DBSAT user the following privileges:

- `CREATE SESSION`

- `READ` or `SELECT` on `SYS.REGISTRY$HISTORY`

- Role `SELECT_CATALOG_ROLE`

- Role `DV_SECANALYST` (if Database Vault is enabled or if Database Vault Operations Control is enabled)

- Role `AUDIT_VIEWER` (12*c* and later)

- Role `CAPTURE_ADMIN` (12*c* and later)

- `READ` or `SELECT` on `SYS.DBA_USERS_WITH_DEFPWD` (11*g* and later)

> **Note:**
>
> If you plan to run only the Collector component, you can assign only the following privileges:
>
> - `CREATE SESSION`
>
> - Role `SELECT_CATALOG_ROLE`
>
> In order to successfully collect Database Vault information in a Database Vault protected environment, you must connect as a non-SYS user with the `DV_SECANALYST` role.

Reporter Prerequisites

The Reporter is a platform-independent Python program and requires Python 2.6 or later to run.

Discoverer Prerequisites

The Discoverer is a Java program and requires the Java Runtime Environment (JRE) 1.8 (jdk8-u172) or later to run.

The Discoverer collects metadata from database dictionary views and matches them against the patterns specified to discover sensitive data. The Discoverer must connect to the database as a user with sufficient privileges to select from these views. For more information about DBSAT user privileges, see Collector Prerequisites.

> **✎ Note:**
>
> The Discoverer relies on table statistics to get row counts. In order to get accurate row count results, `DBMS_STATS` should be executed by the Database Administrator before the DBSAT user runs the Discoverer.

Installing the Oracle Database Security Assessment Tool

To install the Oracle DBSAT:

1. Log in to the database server.
2. Create the `dbsat` directory:

   ```
   mkdir –p /home/oracle/dbsat
   ```

3. Download or copy the `dbsat.zip` file to the database server, and unzip the file.

   ```
   unzip dbsat.zip –d /home/oracle/dbsat
   ```
   Where `–d` refers to the directory path.

These commands are for Linux / Unix. If the installation takes place on Windows, you will use similar commands for Windows.

The Oracle DBSAT is installed on the database server.

You can run the Collector, Reporter, and Discoverer from the `/home/oracle/dbsat` directory.

You can also add this directory to your `PATH` and skip the step of going to the directory every time you want to run the tool.
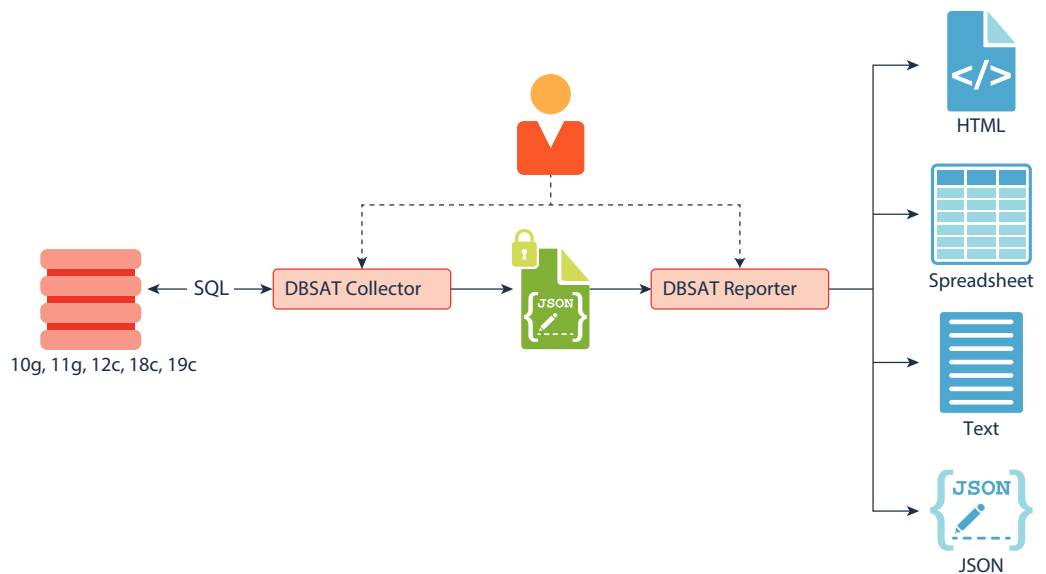
Using the Collector and Reporter

You can generate the Oracle Database Security Assessment Report and the Oracle Database Sensitive Data Assessment Report with the Collector, Reporter, and Discoverer components.

Oracle Database Security Assessment Report

The Collector and Reporter components are used to generate the Oracle Database Security Assessment Report.

The following figure shows the components and architecture of the Collector and Reporter.

Figure   Collector and Reporter Components and Architecture

Running the Collector

The Collector queries the database to collect data that will be analyzed by the Reporter.

> ✎ **Note:**
>
> The Collector connects to the database. Ensure that the target database and listener are running before running the Collector.

To run the Collector, do the following:

1. Specify the arguments to run the Collector:

```
$ dbsat collect <database_connect_string> <output_file>
```

The `dbsat collect` command has the following options and arguments:

- *database_connect_string*

  Specifies the connection string to connect to the database.

  Example: `system@ORCL`

- *output_file*

  Specifies the location and file name for the Database Security Assessment report. Do not add an extension.

  Example: `/home/oracle/dbsat/output_ORCL`

2. Run the Collector.

```
$ ./dbsat collect system@ORCL output_ORCL
```

The following output is displayed:

```
Connecting to the target Oracle database...

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Nov 14 10:02:09
2019
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.3.0.0.0

Setup complete.
SQL queries complete.
OS commands complete.
Disconnected from Oracle Database 19c Enterprise Edition Release
19.0.0.0.0 - Production
Version 19.3.0.0.0
DBSAT Collector completed successfully.

Calling /u01/app/oracle/product/version/db_1/bin/zip to encrypt
output_ORCL.json...

Enter password:
Verify password:
  adding: output_ORCL.json (deflated 88%)
```

```
zip completed successfully.
$
```

> **✎ Note:**
>
> If you do not want to encrypt the file invoke the `dbsat collect` script with
> the `-n` option. This is not recommended.
>
> Running the Collector in the root container in a multitenant container
> database collects data specific to the root container and not from
> its pluggable databases. If you need to access specific pluggable
> databases, you must run the Collector for these pluggable databases
> separately.

Running the Reporter

The Reporter analyzes the data collected by the Collector and makes
recommendations to improve the security of the database.

You can invoke the Reporter with `dbsat report`.

To run the Reporter, do the following:

1. Check that Python version is 2.6 or later is installed.

   ```
   $ python -V
   ```

   A similar output is displayed:

   ```
   Python 2.7.11rc1
   ```

2. Specify the arguments to run the Reporter.

   ```
   $ dbsat report [-a] [-n] [-g] [-x <section>] <input_file>
   ```

   Where the argument *input_file* stands for the full or relative path to the data file
   `db04` produced by the Oracle DBSAT Collector. If this file was encrypted during
   data collection, you will need to supply the encryption password when prompted
   by the Reporter.

   The Reporter supports the following command-line options:

   - *-a*

     Runs the reports for all the database accounts.

   - *-n*

     Specifies no encryption for output.

> **✎ Note:**
>
> For security reasons, this is not recommended.

- *-g*

  Shows all grants including common grants in a pluggable database.

- *-x*

  Excludes a section from the report.

  Valid sections are:

  – `USER` : **User Accounts**

  – `PRIV` : **Privileges and Roles**

  – `AUTH` : **Authorization Control**

  – `CRYPT` : **Encryption**

  – `ACCESS` :**Fine-Grained Access Control**

  – `AUDIT` : **Auditing**

  – `CONF` : **Database Configuration**

  – `NET` : **Network Configuration**

  – `OS` : **Operating System**

  To exclude multiple sections use a comma-separated list, for example:

  `-x USER,PRIV`

  Or:

  `-x USER -x PRIV`

  Omitting this option will include all sections of the report.

The same path name is used to generate the report files produced by the Reporter in HTML, Excel, JSON, and Text formats with the appropriate file extensions.

3. Run the Reporter.

```
$ ./dbsat report output_ORCL
```

The following output appears:

```
Archive:  output_ORCL.zip
[output_ORCL.zip] output_ORCL.json password:
  inflating: output_ORCL.json
DBSAT Reporter ran successfully.
Calling /usr/bin/zip to encrypt the generated reports...
Enter password:
Verify password:
    zip warning: output_ORCL_report.zip not found or empty
  adding: output_ORCL_report.txt (deflated 82%)
```

```
   adding: output_ORCL_report.html (deflated 86%)
   adding: output_ORCL_report.xlsx (deflated 3%)
   adding: output_ORCL_report.json (deflated 85%)
zip completed successfully.
```

4. Specify a password to encrypt the output report `.zip` file.

   The `.zip` file is created.

   > **Note:**
   >
   > The `.zip` file is used for Reporter and Discoverer output. To avoid
   > confusion, it is recommended that you use the same password while
   > creating both outputs.

5. Extract the contents of the `.zip` file to access the Oracle Database Security
   Assessment Report. When prompted, enter the password to decrypt the `.zip` file
   specified in Step *4*.

   The contents of the `.zip` file are extracted.

6. Use the appropriate tools to read the recommendations from the report files.

   Example: Use `vi` on Linux to read the `.txt` files.

   Example: Use a browser to display the `.html` files.

Oracle Database Security Assessment Report

The Collector and Reporter components are used to generate the Oracle Database
Security Assessment Report in HTML, Excel, JSON, and Text formats.

The HTML report provides detailed results of the assessment in a format that is
easy to navigate. The Excel format provides a high-level summary of each finding
without the detailed output included in the HTML report. It also allows you to add
columns for your tracking and prioritization purposes. A report in text format makes it
convenient to copy portions of the output for other usage. Finally, a JSON document
containing the report contents is provided for easier filtering, comparison, aggregation,
and integration with other tools.

Oracle Database Security Assessment Report — Summary

The Oracle Database Security Assessment Report — Summary section contains the
following information:

Section	Description
Assessment Time & Date	Displays the date on which the data was collected and the date on which the final Database Security Assessment report was generated. The DBSAT Reporter version is also displayed.

Section	Description
Database Identity	Displays the details of the database assessed by DBSAT.
Summary	Displays a high level summary of the resulting analysis.

The following figure displays an example of the Oracle Database Security Assessment Report — Summary section.

Figure   Oracle Database Security Assessment Report — Summary

Assessment Date & Time

Date of Data Collection	Date of Report	Reporter Version
Wed Apr 29 2020 03:06:00	Wed Apr 29 2020 03:07:30	2.2.1 (March 2020) – 80a4

Database Identity

Name	Container (Type:ID)	Platform	Database Role	Log Mode	Created
ORCL	PDB1 (PDB:3)	Linux x86 64–bit	PRIMARY	ARCHIVELOG	Tue Mar 31 2020 03:00:00

Summary

Section	Pass	Evaluate	Advisory	Low Risk	Medium Risk	High Risk	Total Findings
Basic Information	0	0	0	0	0	1	1
User Accounts	7	0	2	2	2	0	13
Privileges and Roles	17	4	1	0	0	0	22
Authorization Control	0	0	2	0	0	0	2
Fine–Grained Access Control	0	0	5	0	0	0	5
Auditing	0	6	6	0	0	0	12
Encryption	0	1	1	0	0	0	2
Database Configuration	8	3	0	2	0	0	13
Network Configuration	1	1	3	0	0	0	5
Operating System	2	2	0	0	1	0	5
Total	**35**	**17**	**20**	**4**	**3**	**1**	**80**

The Summary section is followed by the Basic Information section.

Oracle Database Security Assessment Report — Basic Information

The Oracle Database Security Assessment Report — Basic Information section contains the following information:

Section	Finding ID	Description
Databas e Version	-	Displays the version of the database assessed by the Collector and Reporter.
Security Features	-	Displays the security features and indicates if they are in use.
Patch Check	INFO.PAT CH	Displays information about the patches installed. It is vital to keep the database software up-to-date with security fixes as they are released. Oracle issues comprehensive patches in the form of Release Updates, Patch Set Updates, and Bundle Patches on a regular quarterly schedule. These updates should be applied as soon as they are available.

The following figure displays an example of the Oracle Database Security Assessment Report — Basic Information section.

Figure   Oracle Database Security Assessment Report — Basic Information

Basic Information

Database Version

Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 – Production

Security options used: Database Vault, Label Security

Security Features Utilized

Feature	Currently Used
USER AUTHENTICATION	
Password Authentication	Yes
Global Authentication	No
External Authentication	No
AUTHORIZATION CONTROL	
Database Vault	Yes
Privilege Analysis	No
ENCRYPTION	
Tablespace Encryption	No
Column Encryption	No
Network Encryption	No
AUDITING	
Unified Audit	Yes
Fine Grained Audit	No
Traditional Audit	N/A
FINE-GRAINED ACCESS CONTROL	
Virtual Private Database	No
Real Application Security	No
Label Security	Yes
Data Redaction	No
Transparent Sensitive Data Protection	No

ORACLE

The Basic Information section is followed by the User Accounts section.

Oracle Database Security Assessment Report — User Accounts

The Oracle Database Security Assessment Report — User Accounts section displays the following information:

Name	Finding ID	Description
User Accounts	-	Displays the user accounts and the following information about each account: • User Name — Displays the name of the user. • Status — Displays whether the account is Open, Locked, or Expired. • Profile — Displays the profile type. • Tablespace — Displays the tablespace used by the account. • Oracle Defined — Displays whether an Oracle-defined tablespace is used by the account. • Authorization Type — Displays the type of authorization used.
User Schemas in SYSTEM or SYSAUX Tablespace	USER.TBLSPACE	Displays information about the regular user accounts that use the reserved Oracle-supplied tablespaces. The `SYSTEM` and `SYSAUX` tablespaces are reserved for Oracle-supplied user accounts. To avoid a possible denial of service caused by exhausting these resources, regular user accounts should not use these tablespaces. Prior to Oracle Database 12.2, the SYSTEM tablespace cannot be encrypted, and this is another reason to avoid user schemas in this tablespace.
Sample Schemas	USER.SAMPLE	Displays information about the user accounts that use sample schemas such as `SCOTT`, `HR`, `OE`, `SH`, `PM`, `IX`, `ADAMS`, `BLAKE`, `CLARK`, and `BI`. Sample schemas are well-known accounts provided by Oracle to serve as simple examples for developers. They generally serve no purpose in a production database and should be removed because they unnecessarily increase the attack surface of the database.

Name	Finding ID	Description
Inactive Users	USER.INACTIVE	Displays information about the user accounts that are not in use and also accounts that are not configured to be locked when inactive.
		If a user account is no longer in use, it increases the attack surface of the system unnecessarily while providing no corresponding benefit. Furthermore, unauthorized use is less likely to be noticed when no one is regularly using the account. Accounts that have been unused for more than 30 days should be investigated to determine whether they should remain active. A solution is to set `INACTIVE_ACCOUNT_TIME` in the profiles assigned to users.
Case-Sensitive Passwords	USER.CASE	Displays whether case-sensitive passwords are enabled.
		Case-sensitive passwords are recommended because including both upper and lower-case letters greatly increases the set of possible passwords that must be searched by an attacker who is attempting to guess a password by exhaustive search. Setting `SEC_CASE_SENSITIVE_LOGON` to `TRUE` ensures that the database distinguishes between upper and lower-case letters in passwords.
Users with Expired Passwords	USER.EXPIRED	Displays information about the user accounts with expired passwords.
		Password expiration is used to ensure that users change their passwords on a regular basis. If a user's password has been expired for more than 30 days, it indicates that the user has not logged in for at least that long. Accounts that have been unused for an extended period of time should be investigated to determine whether they should remain active.
Users with Default Passwords	USER.DEFPWD	Displays information about the user accounts with default passwords.
		Default account passwords for predefined Oracle accounts are well known. Active accounts with default passwords provide a trivial means of entry for attackers, but well-known passwords should be changed for locked accounts as well.

Name	Finding ID	Description
Minimum Client Authentication Version	USER.AUTHVERS	Displays information about the user accounts that do not have minimum client version specified in the `ALLOWED_LOGON_VERSION_SERVER` parameter in the `sqlnet.ora` file.
		Over time, Oracle releases have added support for increasingly secure versions of the algorithm used for password authentication of user accounts. In order to remain compatible with older client software, the database continues to support previous password versions as well. The `sqlnet.ora` parameter `ALLOWED_LOGON_VERSION_SERVER` determines the minimum password version that the database will accept. For maximum security, this parameter should be set to the highest value supported by the database once all client systems have been upgraded.
Password Verifiers	USER.VERIFIER	Displays information about the user accounts with obsolete password verifiers.
		For each user account, the database may store multiple verifiers, which are hashes of the user password. Each verifier supports a different version of the password authentication algorithm. Every user account should include a verifier for the latest password version supported by the database so that the user can be authenticated using the latest algorithm supported by the client. When all clients have been updated, the security of user accounts can be improved by removing the obsolete verifiers. HTTP password verifiers are used for XML Database authentication. Use the `ALTER USER` command to remove these verifiers from user accounts that do not require this access.
User Parameters	USER.PARAM	Displays information about the user account initialization parameters.
		`SEC_MAX_FAILED_LOGIN_ATTEMPTS` configures the maximum number of failed login attempts in a single session before the connection is closed. This is independent of the user profile parameter `FAILED_LOGIN_ATTEMPTS`, which controls locking the user account after multiple failed login attempts. `RESOURCE_LIMIT` should be set to `TRUE` to enable enforcement of any resource constraints set in user profiles.
User Profiles	-	Displays information about the user profiles.

Name	Finding ID	Description
Users with Unlimited Password Lifetime	USER.NO EXPIRE	Displays information about user profile password expiration enforcement.
		Password expiration is used to ensure that users change their passwords on a regular basis. It also provides a mechanism to automatically disable temporary accounts. Passwords that never expire may remain unchanged for an extended period of time. When passwords do not have to be changed regularly, users are also more likely to use the same passwords for multiple accounts.
Account Locking after Failed Login Attempts	USER.NO LOCK	Displays information about user profile failed login attempt enforcement.
		Attackers sometimes attempt to guess a user's password by simply trying all possibilities from a set of common passwords. To defend against this attack, it is advisable to use the `FAILED_LOGIN_ATTEMPTS` and `PASSWORD_LOCK_TIME` profile resources to lock user accounts for a specified time when there are multiple failed login attempts without a successful login.
Password Verification Functions	USER.PA SSWD	Displays information about password verification functions enforcement.
		Password verification functions are used to ensure that user passwords meet minimum requirements for complexity, which may include factors such as length, use of numbers or punctuation characters, difference from previous passwords, etc. Oracle supplies several predefined functions, or a custom PL/SQL function can be used. Every user profile should include a password verification function.
Users with Unlimited Concurrent Sessions	USER.SE SSIONS	Displays all users that have a Profile Resource Limit for `SESSIONS_PER_USER` set to `UNLIMITED`. With `SESSIONS_PER_USER = UNLIMITED` users can have any number of concurrent sessions.

> **Note:**
>
> Predefined Oracle accounts which are locked are not included in this report. To include all user accounts, run the report with the `-a` option.

The following figure displays an example of the Oracle Database Security Assessment Report — User Accounts section.

Figure   Oracle Database Security Assessment Report — User Accounts

User Name	Status	Profile	Tablespace	Oracle Defined	Auth Type
ADDM	OPEN	DEFAULT	SYSAUX	No	PASSWORD
ANONYMOUS_USER	OPEN	DEFAULT	SYSTEM	No	PASSWORD
C##DVA	OPEN	DEFAULT	SYSTEM	No	PASSWORD
C##DVO	OPEN	DEFAULT	SYSTEM	No	PASSWORD
EAP	OPEN	DEFAULT	SYSTEM	No	PASSWORD
HR	OPEN	DEFAULT	SYSTEM	Yes	PASSWORD
JIM	OPEN	DEFAULT	SYSTEM	No	PASSWORD
OE	OPEN	DEFAULT	SYSTEM	Yes	PASSWORD
PDB_ADMIN	OPEN	DEFAULT	SYSTEM	No	PASSWORD
SCOTT	OPEN	DEFAULT	SYSTEM	Yes	PASSWORD
SYS	OPEN	DEFAULT	SYSTEM	Yes	PASSWORD
SYSTEM	OPEN	DEFAULT	SYSTEM	Yes	PASSWORD
TEST	OPEN	DEFAULT	SYSTEM	No	PASSWORD
TOM	OPEN	DEFAULT	SYSTEM	No	PASSWORD
VAULT_USER	OPEN	DEFAULT	SYSTEM	No	PASSWORD

The User Accounts section is followed by the Privileges and Roles section.

Oracle Database Security Assessment Report — Privileges and Roles

The Oracle Database Security Assessment Report — Privileges and Roles section displays the following information:

Name	Finding ID	Description
System Privilege Grants	PRIV.SYSTEM	Displays the system privileges granted to users.
		System privileges provide the ability to access data or perform administrative operations for the entire database. Consistent with the principle of least privilege, these privileges should be granted sparingly. System privileges should be granted with admin option only when the recipient needs the ability to grant the privilege to others.
		`-g` option reports all grants including common grants in a PDB. The report displays `(*)` for privileges being granted with admin option, `(D)` for privileges being granted directly, and `(C)` for privileges being granted commonly.
All Roles	PRIV.ROLES	Displays all roles granted to users.
		Roles are a convenient way to manage groups of related privileges, especially when the privileges are required for a particular task or job function. Beware of broadly defined roles, which may confer more privileges than an individual recipient requires. Roles should be granted with admin option only when the recipient needs the ability to modify the role or grant it to others.
Code Based Access Control	PRIV.CBAC	Displays all program units granted CBAC roles.
		Code Based Access Control(CBAC) can be used to grant additional privileges on program units. CBAC allows you to attach database roles to a PL/SQL function, procedure, or package. These database roles are enabled at run time, enabling the program unit to execute with the required privileges in the calling user's environment.
Account Management Privileges	PRIV.ACCT	Displays account management privileges granted to users.
		User management privileges (`ALTER USER`, `CREATE USER`, `DROP USER`) can be used to create and modify other user accounts, including changing passwords. This power can be abused to gain access to another user's account, which may have greater privileges.
Role and Privilege Management Privileges	PRIV.MGMT	Displays privilege management privileges granted to users.
		Users with privilege management privileges (`ALTER ANY ROLE`, `CREATE ROLE`, `DROP ANY ROLE`, `GRANT ANY OBJECT PRIVILEGE`, `GRANT ANY PRIVILEGE`, `GRANT ANY ROLE`) can change the set of privileges granted to themselves and other users. This ability should be granted sparingly, since it can be used to circumvent many security controls in the database.

Name	Finding ID	Description
Database Management Privileges	PRIV.DBMGMT	Displays database management privileges granted to users. Database management privileges (`ALTER DATABASE`, `ALTER SYSTEM`, `CREATE ANY LIBRARY`, `CREATE LIBRARY`) can be used to change the operation of the database and potentially bypass security protections. This ability should be granted only to trusted administrators.
Audit Management Package	PRIV.AUDMGMT	Displays audit management tool access granted to users. The `DBMS_AUDIT_MGMT` package allow for execution of Audit management tools. Access should be strictly limited and granted only to users with a legitimate need for this functionality.
Audit Management Privileges	PRIV.AUDIT	Displays audit management privileges granted to users. Audit management privileges (`AUDIT ANY`, `AUDIT SYSTEM`) can be used to change the audit policies for the database. This ability should be granted sparingly, since it may be used to hide malicious activity.
Broad Data Access Privileges	PRIV.DATA	Displays data access privileges granted to users. Users with data access privileges (`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`) have very broad access to data stored in any schema. Most administrative tasks do not require access to the data itself, so these privileges should be granted rarely even to administrators. In addition to minimizing grants of these privileges, consider the use of Database Vault realms to limit the use of these privileges to access sensitive data.
Access Control Exemption Privileges	PRIV.EXEMPT	Displays access control exemption privileges that are enforced. Users with exemption privileges (`EXEMPT ACCESS POLICY`, `EXEMPT REDACTION POLICY`) can bypass the row and column access control policies enforced by Virtual Private Database and Data Redaction. Most administrative tasks do not require access to the data itself, so these privileges should be granted rarely even to administrators.
Access to Password Verifier Tables	PRIV.PASSWD	Displays access to password verifier tables granted to users. Users with these privileges can access objects that contain user password verifiers. The verifiers can be used in offline attacks to discover user passwords.

Name	Finding ID	Description
Write Access to Restricted Objects	PRIV.OBJ	Displays access to restricted objects granted to users.
		Users with these privileges can directly modify objects in the `SYS`, `DVSYS`, `AUDSYS` or `LBACSYS` schemas. Manipulating these system objects may allow security protections to be circumvented or otherwise interfere with normal operation of the database. Object permissions granted to `PUBLIC` must be restricted for objects in the `SYS`, `DVSYS`, `AUDSYS` or `LBACSYS` schemas.
Access to Audit Objects	PRIV.AUDOBJ	Displays access to audit objects granted to users.
		Users with these privileges can directly access and modify objects containing audit information. Access to these objects may allow a malicious user deduce privilege settings for other users and to manipulate the audit information by replacing or deleting audit records.
User Impersonation Privilege	PRIV.USER	Displays the user accounts that have been granted rights to impersonate other users.
		The `BECOME USER` privilege and these PL/SQL packages (`DBMS_AQADM_SYS`, `DBMS_AQADM_SYSCALLS`, `DBMS_IJOB`, `DBMS_PRVTAQIM`, `DBMS_REPCAT_SQL_UTL`, `DBMS_SCHEDULER`, `DBMS_STREAMS_ADM_UTL`, `DBMS_STREAMS_RPC`, `DBMS_SYS_SQL`, `INITJVMAUX`, `LTADM`, `WWV_DBMS_SQL`, `WWV_EXECUTE_IMMEDIATE`) allow for execution of SQL code or external jobs using the identity of a different user. Access should be strictly limited and granted only to users with a legitimate need for this functionality.
Data Exfiltration	PRIV.EXFIL	Displays the user accounts that have been granted rights to access or copy any data from a client or server.
		These PL/SQL packages (`DBMS_BACKUP_RESTORE`, `UTL_DBWS`, `UTL_ORAMTS`) can send data from the database using the network or file system. Access should be granted only to users with a legitimate need for this functionality.
System Privileges Granted to PUBLIC	PRIV.SYSPUB	Displays the system privileges granted to all users.
		Privileges granted to `PUBLIC` are available to all users. This generally should include few, if any, system privileges since these will not be needed by ordinary users who are not administrators.
Roles Granted to PUBLIC	PRIV.ROLEPUB	Displays the roles granted to all users.
		Roles granted to `PUBLIC` are available to all users. Most roles contain privileges that are not appropriate for all users.
Column Privileges Granted to PUBLIC	PRIV.COLPUB	Displays the column access privileges granted to all users.
		Privileges granted to `PUBLIC` are available to all users. This should include column privileges only for data that is intended to be accessible to everyone.

Name	Finding ID	Description
Users with DBA Role	PRIV.DBA	Displays the user accounts that have been granted the DBA role.
		The DBA role is very powerful and can be used to bypass many security protections. It should be granted to only a small number of trusted administrators. Furthermore, each trusted user should have an individual account for accountability reasons. As with any powerful role, avoid granting the DBA role with admin option unless absolutely necessary.
Users with Powerful Roles	PRIV.BIG ROLES	Displays the user accounts that have been granted roles with maximum data access privileges.
		Like the DBA role, these roles (`AQ_ADMINISTRATOR_ROLE`, `EM_EXPRESS_ALL`, `EXP_FULL_DATABASE`, `IMP_FULL_DATABASE`, `SELECT_CATALOG_ROLE`, `EXECUTE_CATALOG_ROLE`, `DELETE_CATALOG_ROLE`, `OEM_MONITOR`) contain powerful privileges that can be used to bypass security protections. They should be granted only to a small number of trusted administrators.
Java Permissions	PRIV.JAVA	Displays the user accounts that have been granted privileges to execute Java classes within the database.
		Java permission grants control the ability of database users to execute Java classes within the database server. A database user executing Java code must have both Java security permissions and database privileges to access resources within the database. These resources include database resources, such as tables and PL/SQL packages, operating system resources, such as files and sockets, Oracle JVM classes, and user-loaded classes. Make sure that these permissions are limited to the minimum required by each user.
Users with Administrative Privileges SYS* Privileges	PRIV.ADMIN	Displays the administrative privileges granted to user accounts.
		Administrative privileges allow a user to perform maintenance operations, including some that may occur while the database is not open. The `SYSDBA` privilege allows the user to run as `SYS` and perform virtually all privileged operations. Starting with Oracle Database 12.1, less powerful administrative privileges were introduced to allow users to perform common administrative tasks with less than full `SYSDBA` privileges. To achieve the benefit of this separation of duty, each of these administrative privileges should be granted to at least one user account.

The following figure displays an example of the Oracle Database Security Assessment Report — Privileges and Roles section.

Figure   Oracle Database Security Assessment Report — Privileges and Roles
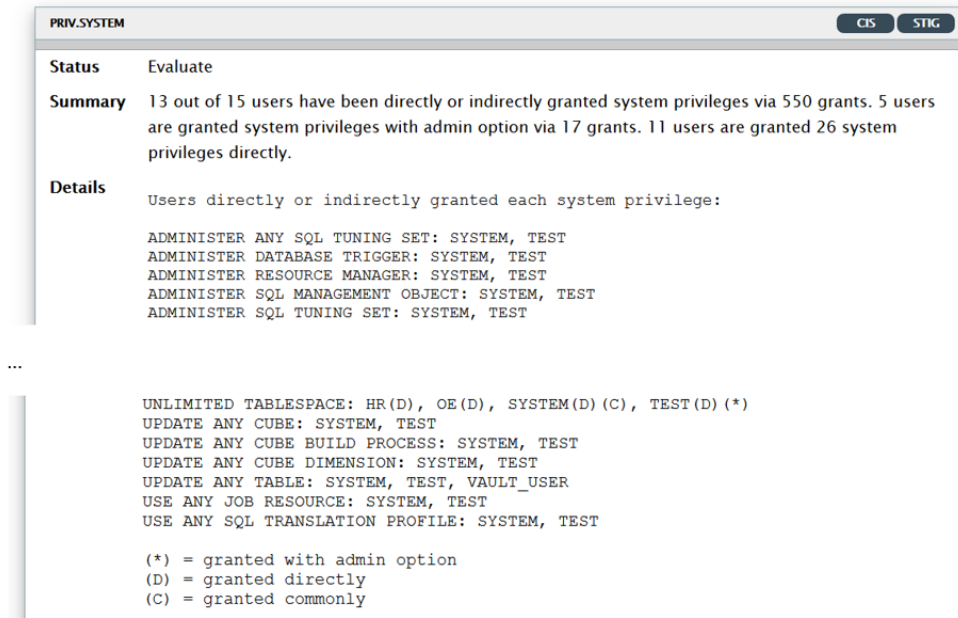
The Privileges and Roles section is followed by the Authorization Control section.

Oracle Database Security Assessment Report — Authorization Control

The Oracle Database Security Assessment Report — Authorization Control section displays the following information:

Name	Finding ID	Description
Database Vault	AUTH.DV	Displays whether Oracle Database Vault is enabled.
		Oracle Database Vault provides for configurable policies to control the actions of privileged administrative users.
Privilege Analysis	AUTH.PRIV	Displays whether Privilege Analysis is enabled.
		Privilege Analysis records the privileges used during a real or simulated workload.

The following figure displays an example of the Oracle Database Security Assessment Report — Authorization Control section.

Figure   Oracle Database Security Assessment Report — Authorization Control

The Authorization Control section is followed by the Fine-Grained Access Control section.

Oracle Database Security Assessment Report — Fine-Grained Access Control

The Oracle Database Security Assessment Report — Fine-Grained Access Control section displays the following information:

Name	Finding ID	Description
Data Redaction	ACCESS.REDACT	Displays whether Data Redaction policies are enabled.
		Data Redaction automatically masks sensitive data found in the results of a database query.
Virtual Private Database	ACCESS.VPD	Displays whether Virtual Private Database (VPD) policies are enabled.
		VPD allows for fine-grained control over the rows and columns of a table are visible to a SQL statement.

Name	Finding ID	Description
Real Application Security	ACCESS.RAS	Displays whether Real Application Security (RAS) policies are enabled.
		Similar to Virtual Private Database (VPD), RAS provides fine-grained control over the rows and columns of a table that are visible to a SQL statement.
Label Security	ACCESS.OLS	Displays whether Oracle Label Security is enabled.
		Oracle Label Security provides the ability to tag data with a data label or a data classification. Access to sensitive data is controlled by comparing the data label with the requesting user's label or security clearance.
Transparent Sensitive Data Protection	ACCESS.TSDP	Displays whether Transparent Sensitive Data Protection (TSDP) policies are enabled.
		TSDP was introduced in Oracle Database 12.1, and allows a data type to be associated with each column that contains sensitive data. TSDP can then apply various data security features to all instances of a particular type so that protection is uniform and consistent.

The following figure displays an example of the OracleDatabase Security Assessment Report — Fine-Grained Access Control section.

Figure   Oracle Database Security Assessment Report — Fine-Grained Access Control

Data Redaction

Virtual Private Database

The Fine-Grained Access Control section is followed by the Auditing section.

Oracle Database Security Assessment Report — Auditing

The Oracle Database Security Assessment Report — Auditing section displays the following information:

Name	Finding ID	Description
Audit Records	AUDIT.RECORDS	Displays information about audit trails.
		Auditing is an essential component for securing any system. The audit trail allows for monitoring the activities of highly privileged users.

Name	Finding ID	Description
Audit SQL Statements	AUDIT.STMT	Displays information about SQL statements audited by enabled audit policies.
Audit Object Actions	AUDIT.OBJ	Displays information about the object access audited by enabled audit policies.
Audit System Privileges	AUDIT.PRIV	Displays information about the privileges audited by enabled audit policies.
Audit Administrative (SYS*) Users	AUDIT.ADMIN	Displays whether the actions of the `SYS` user are audited by enabled audit policies. It is important to audit administrative actions performed by the `SYS` user. Traditional audit policies do not apply to `SYS`, so the `AUDIT_SYS_OPERATIONS` parameter must be set to record `SYS` actions to a separate audit trail.
Audit Privilege Management	AUDIT.PRIVMGMT	Displays whether the actions related to privilege management are audited by enabled audit policies. Granting additional privileges to users or roles potentially affects most security protections and should be audited. Each action or privilege listed should be included in at least one enabled audit policy.
Audit Account Management Activities	AUDIT.ACCTMGMT	Displays whether the actions related to account management are audited by enabled audit policies. Creation of new user accounts or modification of existing accounts can be used to gain access to the privileges of those accounts and should be audited. Each action or privilege listed should be included in at least one enabled audit policy.
Database Management Audit	AUDIT.DBMGMT	Displays whether the actions related to database management are audited by enabled audit policies. Actions that affect the management of database features should always be audited. Each action or privilege listed should be included in at least one enabled audit policy.
Audit Powerful Privileges	AUDIT.PRIVUSE	Displays whether the use of powerful system privileges are audited by enabled audit policies. The use of powerful system privileges should always be audited. Each privilege listed should be included in at least one enabled audit policy.
Audit User Logon / Logoff	AUDIT.CONN	Displays whether Database connections are audited by enabled audit policies. Successful user connections to the database should be audited to assist with future forensic analysis. Unsuccessful connection attempts can provide early warning of an attacker's attempt to gain access to the database.
Fine Grained Audit	AUDIT.FGA	Displays whether fine grained audit policies are enabled. Fine Grained Audit policies can record highly specific activity, such as access to particular table columns or access that occurs under specified conditions. This is a useful way to monitor unexpected data access while avoiding unnecessary audit records that correspond to normal activity.

Name	Finding ID	Description
Unified Audit Policies	AUDIT.UNIFIED	Displays whether unified audit policies are enabled. Unified Audit, available in Oracle Database 12.1 and later releases, combines multiple audit trails into a single unified view. It also introduces new syntax for specifying effective audit policies.

The following figure displays an example of the Oracle Database Security Assessment Report — Auditing section.

Figure   Oracle Database Security Assessment Report — Auditing

Auditing

Audit Records

AUDIT.RECORDS		CIS GDPR STIG
Status	Evaluate	
Summary	Examined 3 audit trails. Found records in 2 audit trails. No errors found in audit initialization parameters.	
Details	Traditional Audit Trail: In use, 19 records found (Oct 15 2018 - Oct 15 2018) FGA Audit Trail: No records found Unified Audit Trail: In use, 527 records found (Oct 15 2018 - Feb 18 2019) AUDIT_FILE_DEST=/u01/app/oracle/admin/orclcdb/adump AUDIT_SYSLOG_LEVEL is not set. AUDIT_TRAIL=DB	
Remarks	Auditing is an essential component for securing any system. The audit trail allows for monitoring the activities of highly privileged users. For any attack that exploits gaps in other security policies, auditing cannot prevent the attack but it forms the critical last line of defense by detecting the malicious activity. Oracle Database 12c introduced Unified Auditing and this is the recommended auditing mode moving forward (pure Unified Audit mode). It adds several benefits as centralized audit logs in a single audit trail, improved performance, simplified management and security. The AUDIT_FILE_DEST controls the OS directory to which the audit trail is written if using AUDIT_TRAIL=os, xml, or xml,extended. This directory should be prevented from any unauthorized access. Sending audit data to a remote system is recommended in order to prevent any possible tampering with the audit records. The AUDIT_SYSLOG_LEVEL parameter can be set to send an abbreviated version of some audit records to a remote syslog collector. A better solution is to use Oracle Audit Vault and Database Firewall to centrally collect full audit records from multiple databases.	
References	CIS Oracle Database 12c Benchmark v2.0.0: Recommendation 2.2.2 EU GDPR 2016/679: Article 30, 33, 34 Oracle Database 12c STIG v1 r10: Rule SV-75899r1, SV-76111r1, SV-76117r1, SV-76121r1, SV-76123r1, SV-76125r1, SV-76127r1, SV-76129r1, SV-76455r3	

Unified Audit

AUDIT.UNIFIED		STIG
Status	Evaluate	
Summary	Found 9 unified audit policies. 1 is enabled. 1 objects or statements being audited.	

The Auditing section is followed by the Encryption section.

Oracle Database Security Assessment Report — Encryption

The Oracle Database Security Assessment Report — Encryption section displays the following information:

Name	Finding ID	Description
Transparent Data Encryption	CRYPT.TDE	Displays whether encryption of tablespace and column data is enabled.
		Encryption of some sensitive data is a requirement in certain regulated environments. Transparent Data Encryption automatically encrypts data as it is stored and decrypts it upon retrieval. This protects sensitive data from attacks that bypass the database to read data files directly.
Encryption Key Wallet	CRYPT.WALLET	Displays wallet information.
		Wallets are encrypted files used to store encryption keys, passwords, and other sensitive data. Wallet files should not be stored in the same directory with database data files, to avoid accidentally creating backups that include both encrypted data files and the wallet containing the master key protecting those files. For maximum separation of keys and data, consider storing encryption keys in Oracle Key Vault instead of wallet files.

The following figure displays an example of the Oracle Database Security Assessment Report — Encryption section.

Figure   Oracle Database Security Assessment Report — Encryption

Encryption

Transparent Data Encryption

CRYPT.TDE		GDPR	STIG

Status Advisory

Summary Found 6 unencrypted tablespaces. No encrypted columns found. Examined 1 initialization parameter.

Details

```
Unencrypted tablespaces: APEX_1581590175460392, SYSAUX, SYSTEM, TEMP,
    UNDOTBS2, USERS
Encrypted tablespaces: (none)

ENCRYPT_NEW_TABLESPACES=CLOUD_ONLY. Recommended value is ALWAYS in the
    Oracle Cloud or in an on-premise database.
```

Remarks Encryption of sensitive data is a requirement in most regulated environments. Transparent Data Encryption automatically encrypts data as it is stored and decrypts it upon retrieval. This protects sensitive data from attacks that bypass the database and read data files directly. Encryption keys may be stored in wallets on the database server itself, or stored remotely in Oracle Key Vault for improved security. The ENCRYPT_NEW_TABLESPACES parameter ensures that TDE tablespace encryption is applied to all newly created tablespaces. Setting this parameter to ALWAYS is recommended in order to protect all data regardless of the options specified when the tablespace is created.

References EU GDPR 2016/679: Article 6, 32, 34; Recital 83
Oracle Database 12c STIG v1 r10: Rule SV-76157r2, SV-76245r2, SV-76251r1, SV-76261r2, SV-76263r3

Encryption Key Wallet

CRYPT.WALLET		GDPR	STIG

Status Evaluate

Summary Found 1 wallet. No wallets are stored in the data file directory.

Details

```
Encryption wallet location:
Wallet type: FILE
Status: NOT_AVAILABLE
Keystore type: UNKNOWN
Wallet order: SINGLE

Data file directory: /u01/app/oracle/product/version/db_1/dbs
```

Remarks Wallets are encrypted files used to store encryption keys, passwords, and other sensitive data. Wallet files should not be stored in the same directory with database data files, to avoid accidentally creating backups that include both encrypted data files and the wallet containing the master key protecting those files. For maximum separation of keys and data, consider storing encryption keys in Oracle Key Vault instead of wallet files.

References EU GDPR 2016/679: Article 6, 32, 34; Recital 83
Oracle Database 12c STIG v1 r10: Rule SV-76015r1, SV-76223r2, SV-76233r2

The Encryption section is followed by the Database Configuration section.

Oracle Database Security Assessment Report — Database Configuration

The Oracle Database Security Assessment Report — Database Configuration section displays the following information:

Name	Finding ID	Description
Initialization Parameters for Security	-	Displays the Database initialization parameters and their values.
Access to Dictionary Objects	CONF.SYSOBJ	Displays whether access to dictionary objects is properly limited.
		When `O7_DICTIONARY_ACCESSIBILITY` is set to `FALSE`, tables owned by `SYS` are not affected by the `ANY TABLE` system privileges. This parameter should always be set to `FALSE` because tables owned by `SYS` control the overall state of the database and should not be subject to manipulation by users with `ANY TABLE` privileges.
Inference of Table Data	CONF.INFER	Displays whether data inference attacks are properly blocked.
		When SQL92_SECURITY is set to TRUE, UPDATE and DELETE statements that refer to a column in their WHERE clauses will succeed only when the user has the privilege to SELECT from the same column. This parameter should be set to TRUE so that this requirement is enforced in order to prevent users from inferring the value of a column which they do not have the privilege to view.
Access to Password File	CONF.PWDFILE	Displays whether the password file is configured correctly.
		The `REMOTE_LOGIN_PASSWORDFILE` set to `EXCLUSIVE` allows the password file to contain distinct entries for each administrative user allowing them to be individually audited and tracked for their actions. It also allows passwords to be updated using the `ALTER USER` command.
Database link global names	CONF.GLBLNM	Displays whether database link names are different from or the same as database names.
		The `GLOBAL_NAMES` parameter is used to set the requirement for database link names to be the same name as the remote database whose connection they define. By using the same name for both, ambiguity is avoided and unauthorized or unintended connections to remote databases are less likely.

ORACLE®

Name	Finding ID	Description
Network Communications	CONF.NETCOM	Displays information about the initialization parameters for the local network listener.
		`REMOTE_LISTENER` allows a network listener running on another system to be used. This parameter should normally be unset to ensure that the local listener is used. The `SEC_PROTOCOL_ERROR` parameters control the database server's response when it receives malformed network packets from a client. Because these malformed packets may indicate an attempted attack by a malicious client, the parameters should be set to log the incident and terminate the connection.
		`SEC_RETURN_SERVER_RELEASE_BANNER` should be set to `FALSE` to limit the information that is returned to an unauthenticated client, which could be used to help determine the server's vulnerability to a remote attack.
External Authorization	CONF.EXTAUTH	Displays information about the initialization parameters for remote authentication.
		The `OS_ROLES` parameter determines whether roles granted to users are controlled by `GRANT` statements in the database or by the database server's operating system. `REMOTE_OS_AUTHENT` and `REMOTE_OS_ROLES` allow the client operating system to set the database user and roles. All of these parameters should be set to `FALSE` so that the authorizations of database users are managed by the database itself.
Trace Files	CONF.TRACE	Displays information about the initialization parameters for trace files.
		The hidden parameter `_TRACE_FILES_PUBLIC` determines whether trace files generated by the database should be accessible to all OS users. Since these files may contain sensitive information, access should be limited by setting this parameter to `FALSE`.
Instance Name Check	CONF.INSTNM	Displays whether the instance name contains the Database version number.
		Instance names should not contain Oracle version numbers. Service names may be discovered by unauthenticated users. If the service name includes version numbers or other database product information, a malicious user may use that information to develop a targeted attack.
Triggers	CONF.TRIG	Displays information about logon triggers.
		A trigger is code that executes whenever a specific event occurs, such as inserting data in a table or connecting to the database. Disabled triggers are a potential cause for concern because whatever protection or monitoring they may be expected to provide is not active.

Name	Finding ID	Description
Disabled Constraints	CONF.CONST	Displays information about disabled constraints. Constraints are used to enforce and guarantee specific relationships between data items stored in the database. Disabled constraints are a potential cause for concern because the conditions they ensure are not enforced.
External Procedures	CONF.EXTPROC	Displays information about external procedures and services. External procedures allow code written in other languages to be executed from PL/SQL. Note that modifications to external code cannot be controlled by the database. Be careful to ensure that only trusted code libraries are available to be executed. Although the database can spawn its own process to execute the external procedure, it is advisable to configure a listener service for this purpose so that the external code can run as a less-privileged OS user. The listener configuration should set `EXTPROC_DLLS` to identify the specific shared library code that can be executed rather than using the default value `ANY`.
Directory Objects	CONF.DIR	Displays information about directory objects. Directory objects allow access to the server's file system from PL/SQL code within the database. Access to files that are used by the database kernel itself should not be permitted, as this may alter the operation of the database and bypass its access controls.
Database Links	CONF.LINKS	Displays information about database links. Database links allow users to execute SQL statements that access tables in other databases. This allows for both querying and storing data on the remote database. It is advisable to set `GLOBAL_NAMES` to `TRUE` in order to ensure that link names match the databases they access.
Network Access Control	CONF.NETACL	Displays information about Network Access Control Lists (ACLs). Network ACLs control the external servers that database users can access using network packages such as UTL_TCP and UTL_HTTP. Specifically, a database user needs the connect privilege to an external network host computer if he or she is connecting using the `UTL_TCP`, `UTL_HTTP`, `UTL_SMTP`, and `UTL_MAIL` utility packages. To convert between a host name and its IP address using the `UTL_INADDR` package, the `Resolve` privilege is required. Make sure that these permissions are limited to the minimum required by each user.

Name	Finding ID	Description
XML Database Access Control	CONF.XMLACL	Displays information about XML Database Access Control Lists (ACLs).
		XML ACLs control access to database resources using the XML DB feature. Every resource in the Oracle XML DB Repository hierarchy has an associated ACL. The ACL mechanism specifies a privilege-based access control for resources to principals, which are database users or roles. Whenever a resource is accessed, a security check is performed, and the ACL determines if the requesting user has sufficient privileges to access the resource. Make sure that these privileges are limited to the minimum required by each user.
Database Backup	CONF.BKUP	Displays information about Database backup records.
		Database should be backed up regularly to prevent loss of data in the event of a system failure. Oracle Recovery Manager (RMAN) allows performing backup and recovery tasks on your databases. Unencrypted backup data should not be transported on tape or disk to offsite storage for safekeeping. Oracle Secure Backup (OSB) may also be used for tape data protection in a distributed environment.

The following figure displays an example of the Oracle Database Security Assessment Report — Database Configuration section.

Figure   Oracle Database Security Assessment Report — Database Configuration

Database Configuration

Initialization Parameters for Security

Name	Value
ADG_ACCOUNT_INFO_TRACKING	LOCAL
AUDIT_FILE_DEST	/u01/app/oracle/admin/ORCL/adump
AUDIT_SYSLOG_LEVEL	
AUDIT_SYS_OPERATIONS	TRUE
AUDIT_TRAIL	DB
COMPATIBLE	19.0.0
CURSOR_BIND_CAPTURE_DESTINATION	memory+disk
DBFIPS_140	FALSE
DISPATCHERS	(PROTOCOL=TCP) (SERVICE=ORCLXDB)
ENCRYPT_NEW_TABLESPACES	CLOUD_ONLY
GLOBAL_NAMES	FALSE
LDAP_DIRECTORY_ACCESS	NONE
LDAP_DIRECTORY_SYSAUTH	no
O7_DICTIONARY_ACCESSIBILITY	
OS_AUTHENT_PREFIX	ops$
OS_ROLES	FALSE
OUTBOUND_DBLINK_PROTOCOLS	ALL
PDB_LOCKDOWN	
PDB_OS_CREDENTIAL	
REMOTE_DEPENDENCIES_MODE	TIMESTAMP
REMOTE_LISTENER	
REMOTE_LOGIN_PASSWORDFILE	EXCLUSIVE
REMOTE_OS_AUTHENT	FALSE
REMOTE_OS_ROLES	FALSE
RESOURCE_LIMIT	TRUE
SEC_CASE_SENSITIVE_LOGON	TRUE
SEC_MAX_FAILED_LOGIN_ATTEMPTS	3
SEC_PROTOCOL_ERROR_FURTHER_ACTION	(DROP,3)
SEC_PROTOCOL_ERROR_TRACE_ACTION	TRACE

The Database Configuration section is followed by the Network Configuration section.

Oracle Database Security Assessment Report — Network Configuration

The Oracle Database Security Assessment Report — Network Configuration section displays the following information:

Name	Finding ID	Description
Network Encryption	NET.CRYPT	Displays information about network encryption.
		Network encryption protects the confidentiality and integrity of communication between the database server and its clients. Either Native Encryption or TLS should be enabled. For Native Encryption, both `ENCRYPTION_SERVER` and `CRYPTO_CHECKSUM_SERVER` should be set to `REQUIRED`. If TLS is used, TCPS should be specified for all network ports and `SSL_CERT_REVOCATION` should be set to `REQUIRED`.
Client Nodes	NET.CLIENTS	Displays information about which client nodes can connect to the database server.
		`TCP.VALIDNODE_CHECKING` should be enabled to control which client nodes can connect to the database server. Either an allowlist of client nodes allowed to connect (`TCP.INVITED_NODES`) or a blocklist of nodes that are not allowed (`TCP.EXCLUDED_NODES`) may be specified. Configuring both lists is an error; only the invited node list will be used in this case.
SQLNET Banners	NET.BANNER	Displays whether SQLNET connect banner messages are configured.
		These banner messages are used to warn connecting users that unauthorized access is not permitted and that their activities may be audited.
Network Listener Configuration	NET.COST	Displays information about network listener configuration.
		These parameters are used to limit changes to the network listener configuration.
		`ADMIN_RESTRICTIONS` should be enabled to prevent parameter changes to the running listener. One of the following restrictions on service registration should be implemented:
		• Prevent changes by disabling `DYNAMIC_REGISTRATION`
		• Limit the nodes that can make changes by enabling `VALID_NODE_CHECKING_REGISTRATION`
		• Limit the network sources for changes using the `COST` parameters `SECURE_PROTOCOL`, `SECURE_CONTROL`, and `SECURE_REGISTER`. `CONNECTION_RATE` determines rate enforced across all the endpoints that are rate limited

Name	Finding ID	Description
Listener Logging Control	NET.LIST ENLOG	Displays information about network listener logging configuration. The `LOGGING_LISTENER` parameter enables logging of listener activity. Log information can be useful for troubleshooting and to provide early warning of attempted attacks.

The following figure displays an example of the Oracle Database Security Assessment Report — Network Configuration section.

Figure   Oracle Database Security Assessment Report — Network Configuration

Network Configuration

Network Encryption

NET.CRYPT	STIG

Status	Medium Risk
Summary	Native encryption is accepted but not required. Integrity check using checksums is accepted but not required.

Details	`SQLNET.ENCRYPTION_SERVER is not set (default value = ACCEPTED).`
	`SQLNET.CRYPTO_CHECKSUM_SERVER is not set (default value = ACCEPTED).`
	
	`Examined 1 listener.`
	
	`LISTENER: IPC (1), TCP (1), TCPS (0)`
	
	`SSL_CERT_REVOCATION is not set (default value = NONE).`

Remarks	Network encryption protects the confidentiality and integrity of communication between the database server and its clients. Either Native Encryption or TLS should be configured to ensure that the connections from clients are encrypted. For Native Encryption, both ENCRYPTION_SERVER and CRYPTO_CHECKSUM_SERVER should be set to REQUIRED. For ease of deployment and compatibility, Oracle Database servers and clients are set to ACCEPT encrypted connections out of the box. This means that you can enable the desired encryption and integrity settings for a connection pair by configuring just one side of the connection, server-side or client-side. So, for example, if there are many Oracle clients connecting to an Oracle database instance, you can configure the required encryption and integrity settings for all these connections by making the appropriate sqlnet.ora changes at the server end. You do not need to implement configuration changes for each client separately. However, in this case, the risk of having plaintext data passed over the network still exists. Keep in mind that whether the security service is enabled or not it is based on a combination of client and server configuration parameters. If TLS is used, TCPS should be specified for all network ports and SSL_CERT_REVOCATION should be set to REQUIRED.
References	Oracle Database 12c STIG v1 r10: Rule SV-75937r2, SV-76035r5, SV-76165r1, SV-76193r2, SV-76195r2, SV-76203r4, SV-76205r4, SV-76231r3, SV-76233r2, SV-76239r1, SV-76241r1, SV-76305r4

Client Nodes

NET.CLIENTS	STIG

Status	Medium Risk
Summary	Valid node check is not enabled, can accept connections from any client. Neither TCP.INVITED_NODES nor TCP.EXCLUDED_NODES is set.

Details	`TCP.VALIDNODE_CHECKING is not set (default value = NO). Recommended value`
	`    is YES.`
	`TCP.INVITED_NODES is not set.`
	`TCP.EXCLUDED_NODES is not set.`

The Network Configuration section is followed by the Operating System section.

Oracle Database Security Assessment Report — Operating System

The Oracle Database Security Assessment Report — Operating System section displays the following information:

Name	Finding ID	Description
OS Authentication	OS.AUTH	Displays information about database connections that use OS authentication. OS authentication allows operating system users within the specified user group to connect to the database with administrative privileges. This shows the OS group names and users that can exercise each administrative privilege. OS users with administrative privileges should be reviewed to prevent any unauthorized, malicious or unintentional access to the database.
Process Monitor Process	OS.PMON	Displays whether Process Monitor (PMON) processes are running under the `ORACLE_HOME` owner account. The PMON process monitors user processes and frees resources when they terminate. This process should run with the user ID of the `ORACLE_HOME` owner.
Agent Processes	OS.AGENT	Displays whether Agent process owners overlap with Listener or Process Monitor (PMON) process owners. Agent processes are used by Oracle Enterprise Manager to monitor and manage the database. These processes should run under a user ID separate from the database and listener processes.
Listener Processes	OS.LISTEN	Displays whether Listener process owners overlap with Agent or Process Monitor (PMON) process owners. Listener processes accept incoming network connections and connect them to the appropriate database server process. These processes should run with a user ID separate from the database and agent processes. These processes should be administered only through local OS authentication.
File Permissions in ORACLE_HOME	OS.FILES	Displays information about errors detected in the file permissions in `ORACLE_HOME`. The `ORACLE_HOME` directory and its subdirectories contain files that are critical to the correct operation of the database, including executable programs, libraries, data files, and configuration files. Operating system file permissions must not allow these files to be modified by users other than the `ORACLE_HOME` owner and must not allow other users to directly read the contents of Oracle data files.

> **Note:**
>
> On Windows, the DBSAT Collector collects data only from SQL queries. Since the data from the operating system commands is missing, the DBSAT Reporter runs a subset of rules on this data. Operating System findings are not available for databases running on Windows platform.

The following figure displays an example of the Oracle Database Security Assessment Report — Operating System section.

Figure   Oracle Database Security Assessment Report — Operating System

The Operating System section is followed by the Diagnostics section.

Oracle Database Security Assessment Report — Diagnostics

The Diagnostics section displays the checks which could not be executed.

> **Note:**
>
> This report provides information and recommendations that may be helpful in securing your Oracle database system. These recommendations reflect best practices for database security and should be part of any strategy for Data Protection by Design and by Default. These practices may help in addressing Articles 25 and 32 of the EU General Data Protection Regulation as well as other data privacy regulations. Technical controls alone are not sufficient for compliance. Passing all findings does not guarantee compliance.
>
> Oracle Database Vault, Oracle Advanced Security, Oracle Label Security, Oracle Data Masking and Subsetting Pack are database licensed options. Oracle Key Vault and Oracle Audit Vault and Database Firewall require separate licensing as well.
>
> The report provides a view on the current status. The results shown are provided for informational purposes only and should not be used as a substitute for a thorough analysis or interpreted to contain any legal or regulatory advice or guidance.
>
> You are solely responsible for your system, and the data and information gathered during the production of this report. You are also solely responsible for the execution of software to produce this report, and for the effect and results of the execution of any mitigating actions identified herein.
>
> Oracle provides this analysis on an "as is" basis without warranty of any kind and Oracle hereby disclaims all warranties and conditions whether express, implied or statutory.

Using the Discoverer

You can generate the Oracle Database Sensitive Data Assessment Report with the Discoverer component.

Oracle Database Sensitive Data Assessment Report

The Discoverer component is used to generate the Oracle Database Sensitive Data Assessment Report.

The following figure shows the components and architecture of the Discoverer.

Figure   Discoverer Components and Architecture

10g, 11g, 12c, 18c, 19c
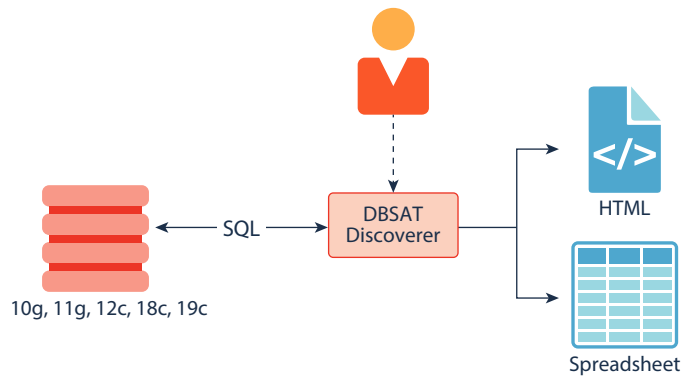
Using the Discoverer

The Discoverer executes SQL queries and collects data from the system to be assessed, based on the settings specified in the configuration and pattern files.

Configuring the Discoverer

Configuring dbsat.config

The settings in the configuration file determine the behavior of the Discoverer.

To configure the Discoverer, do the following:

1. Access the directory where Oracle DBSAT is installed.

2. Navigate to the `Discover/conf` directory. Make a copy of the `sample_dbsat.config` file and rename the file to match your site–specific requirements. For example, you can rename the file to `custom_dbsat.config`.

> **✎ Note:**
>
> Creating a duplicate file ensures that your custom settings are not overwritten during reinstallation.

3. Open `custom_dbsat.config`.

   The following are the contents of the configuration file:

```
[Database]
        TNS_ADMIN =
        NET_SERVICE_NAME =
        WALLET_LOCATION =

        DB_HOSTNAME = localhost
        DB_PORT = 1521
```

```
        DB_SERVICE_NAME =

        SSL_ENABLED = FALSE
        SSL_TRUSTSTORE =
        SSL_TRUSTSTORE_TYPE =
        SSL_KEYSTORE =
        SSL_KEYSTORE_TYPE =
        SSL_DN =
        SSL_VERSION =
        SSL_CIPHER_SUITES =

[Discovery Parameters]
        sensitive_pattern_files = sensitive_en.ini
        schema_scope = ALL
        minrows = 1
        exclusion_list_file =

[Sensitive Categories]
        Identification Info – National IDs = High Risk
        Identification Info – Personal IDs = High Risk
        Identification Info – Public IDs = High Risk
        Biographic Info – Address = High Risk
        Biographic Info – Family Data = High Risk
        Biographic Info – Extended PII = High Risk
        Biographic Info – Restricted Data = High Risk
        IT Info – User Data = High Risk
        IT Info – Device Data = Medium Risk
        Financial Info – Card Data = High Risk
        Financial Info – Bank Data = High Risk
        Health Info – Insurance Data = High Risk
        Health Info – Provider Data = Medium Risk
        Health Info – Medical Data = Medium Risk
        Job Info – Employee Data = High Risk
        Job Info – Org Data = Low Risk
        Job Info – Compensation Data = High Risk
        Academic Info – Student Data = High Risk
        Academic Info – Institution Data = Medium Risk
        Academic Info – Performance Data = Low Risk
```

> **✎ Note:**
>
> Keep the `[Database]`, `[Discovery Parameters]`, and `[Sensitive Categories]` entries for the sections. If you remove these lines, the execution of the tool will issue an error.

4. Configure the settings. For more information about the configuration settings, see Configuration Settings.

5. Save and close the configuration file.

Configuration Settings

Section	Key	Value	Description
[Database]	`TNS_ADMIN`	`<network service name location>`	Location from where network service names needs to be read
-	`NET_SERVICE_NAME`	`<net_service_name >`	Network Service name to be used to make connection
-	`WALLET_LOCATION`	`<SSL wallet location>` | `<SEPS wallet location>`	Location of wallets for secured connections via SSL or SEPS (Secure External Password Store)
-	`DB_HOSTNAME`	`<hostname>` | `<ip_address>`	Hostname or IP Address of the target database server
-	`DB_PORT`	`<portnumber>` The default is 1521.	Listener port number for the target database. If a port number is not specified, the default port 1521 is used.
-	`DB_SERVICE_NAME`	`<service_name>`	Service name for the target database
-	`SSL_ENABLED`	`TRUE` | `FALSE` The default is `FALSE`.	Specifies if SSL is enabled or disabled when connecting to the Database Server. This is an optional argument. It is recommended that the `SSL_ENABLED` value is set to `TRUE`. Retain the default `FALSE` value if you do not require an SSL connection to the Database Server. If `SSL_ENABLED` `= TRUE`, then `SSL_TRUSTSTORE` is mandatory.
-	`SSL_TRUSTSTORE`	`<Absolute path to the TrustStore/ TrustStore filename>` **Example:** `/opt/ oracle/wallets/ truststore.jks`	Specifies the absolute path to the TrustStore, and the TrustStore file name. Mandatory if `SSL_ENABLED = TRUE`.

Section	Key	Value	Description
-	SSL_TRUSTSTORE_TYPE	PKCS12 | JKS | SSO	Specifies the type of TrustStore. Use `PKCS12` if the Truststore is a Wallet. Use `JKS` if the Truststore is a Java KeyStore. Use `SSO` if the Truststore is an auto-login SSO Wallet.
-	SSL_KEYSTORE	`<Absolute path to the KeyStore/KeyStore filename>` **Example:** `/opt/ oracle/wallets/ keystore.jks`	Specifies the absolute path to the KeyStore, and the KeyStore file name. If `SSL_KEYSTORE` is not specified, the value specified in `SSL_TRUSTSTORE` is used. Mandatory if the Database server requires client authentication.
-	SSL_KEYSTORE_TYPE	PKCS12 | JKS | SSO	Specifies the type of KeyStore. Use `PKCS12` if the KeyStore is a Wallet. Use `JKS` if the KeyStore is a Java KeyStore. Use `SSO` if the KeyStore is an auto-login SSO Wallet.
-	SSL_DN	`<distinguished_name>`	Distinguished Name (DN) of the target Database server. Specify the DN if the server's DN needs to be checked. This is an optional argument.

ORACLE®

Section	Key	Value	Description
-	SSL_VERSION	1.0 | 1.1 | 1.2 The default is 1.2.	Specifies the version of the SSL protocol to use when connecting to the Database Server. This is an optional argument. Use 1.0 for SSL version TLSv1.0. Use 1.1 for SSL version TLSv1.1. Use 1.2 for SSL version TLSv1.2.
-	SSL_CIPHER_SUITES	`<cipher_suite1>,<cipher_suite2>` **Example:** TLS_RSA_WITH_AES_256_CBC_SHA256 , SSL_RSA_WITH_RC4_128_MD5	Specifies the Cryptographic Algorithms to be used. Multiple entries can be specified as a comma-separated list. This is an optional argument. For information about supported cryptographic suites, see https://docs.oracle.com/javase/8/docs/technotes/guides/security/SunProviders.html.
[Discovery Parameters]	SENSITIVE_PATTERN _FILES	`<file_name>` | `<file_name1>, <file_name2>` The default is sensitive_en.ini.	Specifies the pattern files to be used. Multiple files can be specified as a comma-separated list. The limit is 10 files. For more information about configuring the Sensitive Data Type pattern file, see Pattern File Configuration (Optional).
-	SCHEMA_SCOPE	ALL | `<schema1>,<schema2>` The default is ALL.	Specifies the schemas to be scanned. Multiple schemas can be specified as a comma-separated list.

Section	Key	Value	Description
-	`MINROWS`	`<numerical value>` The default is `1`.	Specifies the minimum number of rows in a table for that table to be scanned. Tables with a number of rows less than what is specified in the `minrows` parameter are excluded from the scan.
-	`EXCLUSION_LIST_FILE`	`<exclusion_list_filename>.ini`	Specifies the file to be used to exclude schemas, tables, or columns from the scan. For more information about configuring the Exclusion List file, see Configuring the Exclusion List File (Optional).
[Sensitive Categories]			The **[Sensitive Categories]** section defines which Sensitive Categories are used. Valid risk levels are: • `Low Risk` • `Medium Risk` • `High Risk` The types of sensitive data are defined in the Sensitive Data Type pattern file. For more information about configuring the Sensitive Data Type pattern file, see Pattern File Configuration (Optional).

Pattern File Configuration (Optional)

The Oracle Database Security Assessment Tool searches for the types of sensitive data defined in the Pattern file(s).

About Sensitive Types

Pattern files contain the patterns to search for. A Pattern file is grouped into sections, defined by the section heading format `[SENSITIVE_TYPE_NAME]`. Each section constitutes a Sensitive Type.

The following example shows a sample Sensitive Type section for `FULL NAME`.

```
[FULL NAME]
COL_NAME_PATTERN = ^(?!.*(ITEM|TAX|BALANCE)).*(FULL.*NAME)|(^|[_-])
(CUSTOMER|CUST|CLIENT|PATIENT|PERSON).?(NAME|NM)($|[_-])
COL_COMMENT_PATTERN = ^(?!.*(ITEM|TAX|BALANCE)).*(FULL.?NAME)|(CUSTOMER|
CUST|CLIENT|PATIENT|PERSON).?NAME
SENSITIVE_CATEGORY = Identification Info - Public IDs
```

The Sensitive Type name `[SENSITIVE_TYPE_NAME]` is displayed in the Sensitive Type column of the Database Sensitive Data Assessment Report — Sensitive Column Details section. For more information about the Database Sensitive Data Assessment Report, see Oracle Database Sensitive Data Assessment Report.

Each Sensitive Type is defined by the following three parameters: `COL_NAME_PATTERN`, `COL_COMMENT_PATTERN`, and `SENSITIVE_CATEGORY`.

COL_NAME_PATTERN

The `COL_NAME_PATTERN` parameter specifies the text to search for in the Regular Expression (`RegExp`) patterns of the database column names.

```
(^LNAME$)|((LAST|FAMILY|SUR|PATERNAL).*NAME$)
```

In the example above, the following text will be searched for in the `RegExp` patterns of the database column names:

- `(^LNAME$)` — Searches for a column titled `LNAME`.

- `((LAST|FAMILY|SUR|PATERNAL).*NAME$)` — Searches for column names that contain `LAST`, `FAMILY`, `SUR`, or `PATERNAL`, followed by any characters and ending with `NAME`. For example, `LAST_NAME` or `CUSTOMER_SURNAME`.

COL_COMMENT_PATTERN

The `COL_COMMENT_PATTERN` parameter specifies the text to search for in the Regular Expression (`RegExp`) patterns of the database column comments.

SENSITIVE_CATEGORY

The `SENSITIVE_CATEGORY` parameter specifies the type of sensitive data. The risk levels associated with exposing types of sensitive data are specified in the `sample_dbsat.config` file. The risk levels are:

- `Low Risk`

- `Medium Risk`

- High Risk

For more information about configuring the `sample_dbsat.config` file, see Configuration Settings.

Customizing the Pattern File

To customize the Pattern file, do the following:

1. Access the directory where Oracle DBSAT is installed.

2. Navigate to the `Discover/conf` directory. Make a copy of the `sensitive_en.ini` file and rename the file `my_sensitive_en.ini`.

> **✎ Note:**
>
> The `Discover/conf` directory also contains the following language-specific `.ini` files to help discover sensitive data in data dictionaries in major European languages (filename - country, language):
>
> - `sensitive_de.ini` - German, Germany.
> - `sensitive_el.ini` - Greek, Greece.
> - `sensitive_es.ini` - Spanish, Spain.
> - `sensitive_fr.ini` - French, France.
> - `sensitive_it.ini` - Italian, Italy.
> - `sensitive_nl.ini` - Dutch, Netherlands.
> - `sensitive_pt.ini` - Portuguese, Portugal.

3. Open `my_sensitive_en.ini`.

4. Customize the settings by adding new Sensitive Types and modifying existing Sensitive Types.

   For more information about adding new Sensitive Types and Sensitive Categories to the Pattern file, see About Sensitive Types and Configuration Settings.

5. Save and close `my_sensitive_en.ini`.

   The Pattern file is configured.

6. Include `my_sensitive_en.ini` in the Discoverer scan by adding a reference to the file in the `custom_dbsat.config` file.

   ```
   sensitive_pattern_files = my_sensitive_en.ini
   ```

For more information about referencing the Pattern file in the `custom_dbsat.config` file, see Configuring dbsat.config.

About Regular Expressions

The search parameters use regular expressions, sets of strings based on common characteristics shared by each string in the set. Regular expressions vary in complexity, but once you understand the basics of how they are constructed, you can decipher or create any regular expression. You can use character classes, capturing groups, quantifiers, boundary matchers, and logical operators to define regular expressions.

String Literals

The most basic form of pattern matching is the match of a string literal. For example, if the regular expression is `EMP` and the input string is `EMP`, the match succeeds because the strings are identical. This regular expression also matches any string containing `EMP`, such as `EMPLOYEE`, `TEMP`, and `TEMPERATURE`.

Metacharacters

You can also use some special characters that affect the way a pattern is matched. One of the most common ones is the dot (.) symbol, which matches any character. For example, `EMPLOYEE.ID` matches `EMPLOYEE_ID` and `EMPLOYEE-ID`, but not `EMPLOYEE_VERIFICATION_ID`. Here, the dot is a metacharacter — a character with special meaning interpreted by the matcher.

Some other metacharacters are: `^ $ ? + * \ - [ ] ( ) { }`.

If you want a metacharacter to be treated literally (as an ordinary character), you can use a backslash (\) to escape it. For example, the regular expression `9\+9` matches `9+9`.

Character Classes

A character class is a set of characters enclosed within square brackets. It specifies the characters that successfully match a single character from a given input string.

The following table describes some common regular expression constructs.

Construct	Description
`[abc]`	Matches one of the characters mentioned within square brackets. Example: `EMPLOYE[ER]` matches `EMPLOYEE` and `EMPLOYER`.
`[^abc]`	Matches any character except the ones mentioned within square brackets. Example: `[^BC]AT` matches `RAT` and `HAT`, but does not match `BAT` and `CAT`.
`[A-Z0-9]`	Matches any character in the range mentioned within square brackets. To specify a range, simply insert the dash metacharacter "-" between the first and last character to be matched; for example, `[1-5]` or `[A-M]`. You can also place different ranges beside each other within the class to further expand the match possibilities. Example: `[B-F]AT` matches `BAT`, `CAT`, `DAT`, `EAT`, and `FAT`, but does not match `AAT` and `GAT`.

> **✎ See Also:**
>
> - Character Classes
> - Predefined Character Classes

Capturing Groups

You can use capturing groups to treat multiple characters as a single unit. A capturing group is created by placing the characters to be grouped inside a set of parentheses. For example, the regular expression `(SSN)` creates a single group containing the letters `S`, `S`, and `N`.

> **✎ See Also:**
>
> Capturing Groups

Quantifiers

You can use quantifiers to specify the number of occurrences to match against.

The following table describes some common quantifiers.

Quantifier	Description
`X?`	Matches zero or one occurrence of the specified character or group of characters. Example: `SSN_NUMBERS?` matches strings `SSN_NUMBER` and `SSN_NUMBERS`.
`X*`	Matches zero or more occurrences of the specified character or group of characters. Example: `TERM.*DATE` matches strings like `TERMDATE`, `TERM_DATE` and `LAST_TERMINATION_DATE`.
`X+`	Matches one or more occurrences of the specified character or group of characters. Example: `TERM.+DATE` matches strings like `TERM_DATE` and `TERMINATION_DATE`, but not `TERMDATE`.
`X{n}`	Matches the specified character or group of characters exactly n times. Example: `9{3}` matches `999`, but not `99`.
`X{n,}`	Matches the specified character or group of characters at least n times. Example: `9{3,}` matches `999`, `9999`, and `99999`, but not `99`.
`X{n,m}`	Matches the specified character or group of characters at least n times but not more than m times. Example: `9{3,4}` matches `999` and `9999`, but not `99`.

An example of regular expression using character class is `SSN[0-9]+`, which matches strings like `SSN0`, `SSN1`, and `SSN12`. Here, `[0-9]` is a character class and is allowed one or more times. The regular expression does not match `SSN`.

> ✎ **See Also:**
>
>   Quantifiers

Boundary Matchers

You can use boundary matchers to make pattern matching more precise by specifying where in the string the match should take place. For example, you might be interested in finding a particular word, but only if it appears at the beginning or end of an input string.

The following table describes common boundary matchers.

Boundary Construct	Description
`^`	Matches the specified character or group of characters at the beginning of a string (starts with search).
	Example: `^VISA` matches strings beginning with `VISA`.
`$`	Matches the specified character or group of characters at the end of a string (ends with search).
	Example: `NUMBER$` matches strings ending with `NUMBER`.
`\b`	Marks a word boundary. Matches the character or group of characters specified between a pair of `\b` only if it is a separate word (as opposed to substring within a longer string).
	Example: `\bAGE\b` matches strings like `EMPLOYEE AGE` and `PATIENT AGE INFORMATION`, but does not match strings like `AGEING` and `EMPLOYEEAGE`.

If no boundary matcher is specified, a contains search is performed. For example, `ELECTORAL` matches strings containing `ELECTORAL`, such as `ELECTORAL_ID`, `ID_ELECTORAL`, and `ELECTORALID`.

An exact match search can be performed by using `^` and `$` together. For example, `^ADDRESS$` searches for the exact string `ADDRESS`. It matches the string `ADDRESS`, but does not match strings like `PRIMARY_ADDRESS` and `ADDRESS_HOME`.

> ✎ **See Also:**
>
>   Boundary Matchers

Logical Operators

You can use the pipe or vertical bar character (|) if you want to match any one of the characters (or group of characters) separated by pipe. For example, `EMPLOY(EE|ER)_ID` matches `EMPLOYEE_ID` and `EMPLOYER_ID`.

Examples

`^JOB.*(TITLE|PROFILE|POSITION)$` matches strings beginning with `JOB`, followed by zero or more occurrences of any character, and ending with `TITLE`, `PROFILE`, or `POSITION`.

`^[A-Z]{3}[0-9]{2}[A-Z0-9]$` matches strings beginning with three letters, followed by two digits, and ending with a letter or digit.

`BIRTH.?(COUNTRY|PLACE)|(COUNTRY|PLACE).*BIRTH` matches strings such as `BIRTH COUNTRY`, `PATIENT_BIRTH_PLACE`, `PLACE_OF_BIRTH`, and `EMPLOYEE'S COUNTRY OF BIRTH`.

> ✎ **See Also:**
>
> Regular Expressions

Configuring the Exclusion List File (Optional)

You can specify schemas, tables, or columns to exclude from the scan in the Exclusion List file.

This is an optional step but often required to fine tune the Discoverer to exclude false positives.

To create an Exclusion List file, do the following:

1. Create an Exclusion List file, and save it in the `Discover/conf` directory as `myexclusion_list`.

2. Specify the schemas, tables, or columns to exclude from the Discoverer scan.

   The following is a sample of the contents of the Exclusion List file.

   ```
   PAYROLL
   IT.ENTITLEMENTS
   HR.EMPLOYEE.MARITAL_STATUS
   HR.JOB.CANDIDATE
   ```

   Specify the schemas, tables, or columns to exclude using the format `SchemaName.TableName.ColumnName`. Type each exclusion entry on a new line.

   In the example above, PAYROLL excludes the `PAYROLL` schema from the discovery scan; IT.ENTITLEMENTS excludes the `ENTITLEMENTS` table in `IT` schema; HR.EMPLOYEE.MARITAL_STATUS excludes column `MARITAL_STATUS` from the `HR.EMPLOYEE` table. Similarly, HR.JOB.CANDIDATE excludes column `CANDIDATE` from `HR.JOB` table.

> **💡 Tip:**
>
> The Discoverer CSV report includes a column with the fully qualified column names (FULLY_QUALIFIED_COLUMN_NAME). This column can be used to create the exclusion list file contents and speed up the removal of unwanted columns or false positives from the report in a subsequent run.

3. Save and close the Exclusion List file.

4. Update the `exclusion_list_file` entry in your `custom_dbsat.config` file to
   `exclusion_list_file = myexclusion_list`

For more information about referencing the Exclusion List file, see Configuring dbsat.config.

Configuring Certificates and Wallets (Optional)

The Discoverer allows usage of Secure External Password Store to retrieve login credentials stored a wallet while connecting. Secure External Password Store can be used to connect to Database without entering the username and password. Secure External Password Store improves the security and allows automation of the execution of the Discoverer.

For increased security, Oracle Database provides Secure Sockets Layer (SSL) support to encrypt the connection between clients and the server. If SSL (TLS) encryption is configured on the Database Server, the Discoverer needs to be configured in order to connect and discover data. Configuration parameters for SSL can be found in the `dbsat.config` file.

To establish an SSL connection with the Discoverer, the Database Server sends its certificate, which is stored in its wallet. The client may or may not need a certificate or wallet, depending on the server configuration.

> **✎ Note:**
>
> Configuring certificates and wallets is an optional step and needs to be performed only when using SSL to connect to the Oracle Database server.

For more information about configuring certificates and wallets, see Support for SSL in the *Oracle Database JDBC Developer's Guide*.

Running the Discoverer

To run the Discoverer, do the following:

1. Specify the arguments to run the Discoverer:

    ```
    $ dbsat discover [-n] -c <config_file> <output_file>
    ```

    The `dbsat discover` command has the following options and arguments:

    -   `-n`

        Specifies that there is no encryption for output.

    -   `-c`

        Specifies the name of the configuration file used for discoverer. For more information about the `config_file` file, see Configuring dbsat.config.

    -   `output_file`

        Specifies the full or relative path name to create the `.zip` file. Do not add an extension.

        Example: `/home/oracle/dbsat/PDB1`

2. Run the Discoverer.

    ```
    $ ./dbsat discover -c Discover/conf/custom_dbsat.config PDB1
    ```

    The following output is displayed:

    ```
    DBSAT Discover ran successfully.
    Calling /usr/bin/zip to encrypt the generated reports...
    Enter password:
    Verify password:
      adding: PDB1_discover.html (deflated 86%)
      adding: PDB1_discover.csv (deflated 86%)
    Zip completed successfully.
    $
    ```

3. Specify a password to encrypt the `.zip` file.

    A zip file named `<destination>_report.zip` is created. If the file `<destination>_report.zip` exists, the discovery results are added to the existing zip file.

    > **Note:**
    >
    > The `.zip` file is used for Reporter and Discoverer output. To avoid confusion, it is recommended that you use the same password while creating both outputs.

4. Extract the contents of the `.zip` file to access the Database Sensitive Data Assessment Report. When prompted, enter the password to decrypt the `.zip` file specified in Step *3*.

    The contents of the `.zip` file are extracted.

5. Use the appropriate tools to read the Database Sensitive Data Assessment Report.

   Example: Use a browser to display the `.html` file.

   Example: Use a spreadsheet reader like `OpenOffice Calc` or `Excel` to open the `.csv` file.

Oracle Database Sensitive Data Assessment Report

The Discoverer component is used to generate the Oracle Database Sensitive Data Assessment Report in HTML and CSV formats.

The HTML report is the main report and contains the discovered sensitive data and its categories along with target database information and Discoverer parameters.

The CSV report can be loaded into Oracle Audit Vault and Database Firewall to add sensitive data context to the new Data Privacy reports. For more information about this functionality, see Importing Sensitive Data Into AVDF Repository in the *Oracle Audit Vault and Database Firewall Auditor's Guide*.

Oracle Database Sensitive Data Assessment Report — High-Level Summary

The Oracle Database Sensitive Data Assessment Report — High-Level Summary section contains the following information:

Table   Oracle Database Sensitive Data Assessment Report — High-Level Summary

Section	Description
Assessment Time & Date	Displays when the Sensitive Data Assessment report was generated. The DBSAT Discoverer version is also displayed.
Database Identity	Displays the details of the database assessed by the Discoverer.
Database Version	Displays the version of the database assessed by the Discoverer.
Discovery Parameters	Displays the Discovery Parameters specified in the configuration file. For more information about Discovery Parameters, see Configuration Settings.

The following figure displays the first four tables of the Oracle Database Sensitive Data Assessment Report — High-Level Summary section.

Figure   Oracle Database Sensitive Data Assessment Report — High-Level Summary

The High-Level Summary section is followed by the Summary section.

Oracle Database Sensitive Data Assessment Report — Summary

The Oracle Database Sensitive Data Assessment Report — Summary section displays information about the number of tables, columns, and rows identified as sensitive data, grouped by Sensitive Category.

The Database Sensitive Data Assessment Report — Summary section contains the following columns:

Table   Oracle Database Sensitive Data Assessment Report — Summary

Column Name	Description
Sensitive Category	Displays the name of the Sensitive Category
# Sensitive Tables	Displays the number of tables detected that contain sensitive data

Table   (Cont.) Oracle Database Sensitive Data Assessment Report — Summary

Column Name	Description
# Sensitive Columns	Displays the number of columns detected in the tables that contain sensitive data
# Sensitive Rows	Displays the number of rows detected in the tables that contain sensitive data

The following figure displays the information displayed in the Oracle Database Sensitive Data Assessment Report — Summary section:

Figure   Oracle Database Sensitive Data Assessment Report — Summary

Summary

Sensitive Category	# Sensitive Tables	# Sensitive Columns	# Sensitive Rows
BIOGRAPHIC INFO – ADDRESS	7	18	244
FINANCIAL INFO – CARD DATA	2	2	256
HEALTH INFO – PROVIDER DATA	1	1	149
IDENTIFICATION INFO – PERSONAL IDS	3	3	356
IDENTIFICATION INFO – PUBLIC IDS	3	12	321
IT INFO – USER DATA	1	1	149
JOB INFO – COMPENSATION DATA	7	10	527
JOB INFO – EMPLOYEE DATA	12	25	569
JOB INFO – ORG DATA	7	8	412
TOTAL	21*	80	989**

* Number of Unique Tables with Sensitive Data.

** Number of Unique Rows with Sensitive Data.

> **Note:**
>
> A single database table could contain columns or column comments that match more than one Sensitive Category, causing a higher number to be displayed in the `# Sensitive Tables` and `# Sensitive Rows` columns. However, the `Total` row displays the unique number of tables and rows identified as sensitive data.

For more information about configuring Sensitive Categories, see Pattern File Configuration (Optional).

The Summary section is followed by the Sensitive Data section.

Oracle Database Sensitive Data Assessment Report — Sensitive Data

The Oracle Database Sensitive Data Assessment Report — Sensitive Data section displays information about the schemas containing sensitive data.

The Oracle Database Sensitive Data Assessment Report — Sensitive Data section contains the following information:

Table   Oracle Database Sensitive Data Assessment Report — Sensitive Data
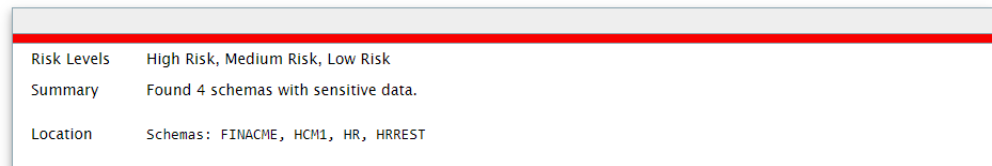
Section	Description
Risk Level(s)	Displays the Risk Level(s) of the sensitive data identified in the schema of the database assessed by the Discoverer.
Summary	Displays a summary of the occurrence of sensitive data in the schema.
Location	Displays the names of the schemas containing sensitive data.

The following figure shows the information displayed in the Oracle Database Sensitive Data Assessment Report — Sensitive Data section.

Figure   Oracle Database Sensitive Data Assessment Report — Sensitive Data

Findings belonging to each risk level are followed by a set of recommendations to secure the sensitive data. These recommendations lists various controls based on the Risk Levels, namely HIGH, MEDIUM, and LOW.

The following figure shows the information displayed in the Risk Level: High Risk section.

Figure   Sensitive categories grouped by Risk Level

Risk Level: High Risk

Security for Environments with High Value Data: Detective plus Strong Preventive Controls
Highly sensitive and regulated data should be protected from privileged users, and from users without a business need for the data. Activity of privileged accounts should be controlled to protect against insider threats, stolen credentials, and human error. Who can access the database and what can be executed should be controlled by establishing a trusted path and applying command rules. Sensitive data should be redacted on application read only screens. A Database Firewall ensures that only approved SQL statements or access by trusted users reaches the database – blocking unknown SQL injection attacks and the use of stolen login credentials.

Recommended controls include:

- **Audit all sensitive operations including privileged user activities**
- **Audit access to application data that bypasses the application**
- **Encrypt data to prevent out-of-band access**
- **Mask sensitive data for test and development environments**
- **Restrict database administrators from accessing highly sensitive data**
- **Block the use of application login credentials from outside of the application**
- **Monitor database activity for anomalies**
- **Detect and prevent SQL Injection attacks**
- *Evaluate: Oracle Audit Vault and Database Firewall, Oracle Advanced Security, Oracle Data Masking and Subsetting, Oracle Database Vault*

Each Risk Level section is followed by a list of the tables detected that contain sensitive data. The following information is displayed:

Table   Tables Detected within Sensitive Category: <Sensitive Category Name>

Name	Description
Risk Level	Displays the Risk Level
Summary	Displays a summary of the sensitive category data detected
Location	Displays the names of the tables that contain sensitive data

The following figure shows the information displayed in the Tables Detected within each Sensitive Category: <Sensitive Category Name> subsection.

Figure   Tables Detected within Sensitive Category: <Sensitive Category Name>

Tables Detected within Sensitive Category: JOB INFO – EMPLOYEE DATA

Risk Level	High Risk
Summary	Found JOB INFO – EMPLOYEE DATA within 9 Column(s) in 4 Table(s)
Location	Tables: HR.DEPARTMENTS, HR.EMPLOYEES, HR.JOBS, HR.JOB_HISTORY

The Sensitive Data section is followed by the Schema View section.

ORACLE

Oracle Database Sensitive Data Assessment Report — Schema View

The Oracle Database Sensitive Data Assessment Report — Schema View section displays information about the schemas, tables, columns, and rows containing sensitive data. The Sensitive Category is also displayed.

The Oracle Database Sensitive Data Assessment Report — Summary section contains the following columns:

Column Name	Description
Schema	Displays the name of the schema
Table Name	Displays the name of the table
Columns	Displays the number of columns in the table
Sensitive Columns	Displays the number of columns detected that contain sensitive data
Rows	Displays the number of rows in the table
Sensitive Category	Displays the category of sensitive data detected in each column

The following figure highlights the information displayed in the Oracle Database Sensitive Data Assessment Report — Schema View section:

Figure   Oracle Database Sensitive Data Assessment Report — Schema View

Schema View

Table Summary

Schema	Table Name	Columns	Sensitive Columns	Rows	Sensitive Category
FINACME	COMPANY_DATA	9	4	100	BIOGRAPHIC INFO – ADDRESS, IDENTIFICATION INFO – PERSONAL IDS
HCM1	COUNTRIES	3	1	25	BIOGRAPHIC INFO – ADDRESS
HCM1	DEPARTMENTS	4	1	27	JOB INFO – ORG DATA
HCM1	EMPLOYEES	11	8	107	IDENTIFICATION INFO – PUBLIC IDS, JOB INFO – COMPENSATION DATA, JOB INFO – EMPLOYEE DATA, JOB INFO – ORG DATA
HCM1	EMP_EXTENDED	3	3	107	FINANCIAL INFO – CARD DATA, IDENTIFICATION INFO – PERSONAL IDS, JOB INFO – EMPLOYEE DATA
HCM1	JOBS	4	4	19	JOB INFO – COMPENSATION DATA, JOB INFO – EMPLOYEE DATA
HCM1	JOB_HISTORY	5	4	10	JOB INFO – EMPLOYEE DATA, JOB INFO – ORG DATA
HCM1	LOCATIONS	6	4	23	BIOGRAPHIC INFO – ADDRESS
HCM1	SUPPLEMENTAL_DATA	6	5	149	FINANCIAL INFO – CARD DATA, HEALTH INFO – PROVIDER DATA, IDENTIFICATION INFO – PERSONAL IDS, IT INFO – USER DATA, JOB INFO – COMPENSATION DATA
HR	COUNTRIES	3	1	25	BIOGRAPHIC INFO – ADDRESS
HR	DEPARTMENTS	4	2	27	JOB INFO – EMPLOYEE DATA, JOB INFO – ORG DATA

The Schema View section is followed by the Sensitive Column Details section.

Oracle Database Sensitive Data Assessment Report — Sensitive Column Details

The Oracle Database Sensitive Data Assessment Report — Sensitive Column Details section displays information about the columns containing sensitive data. The Sensitive Category and Type are also displayed.

Column Name	Description
Schema Name	Displays the name of the schema

Column Name	Description
Table Name	Displays the name of the table
Column Name	Displays the name of the column
Column Comment	Displays the column comment
Sensitive Category	Displays the category of sensitive data detected in each column
Sensitive Type	Displays the type of sensitive data detected in each column
Risk Level	Displays the risk level

The following figure displays the information displayed in the Oracle Database Sensitive Data Assessment Report — Sensitive Column Details section.

Figure   Oracle Database Sensitive Data Assessment Report — Sensitive Column Details

Sensitive Column Details

Schema Name	Table Name	Column Name	Column Comment	Sensitive Category	Sensitive Type	Risk Level
FINACME	COMPANY_DAT A	CITY	--	BIOGRAPHIC INFO – ADDRESS	CITY	High Risk
FINACME	COMPANY_DAT A	STATE	--	BIOGRAPHIC INFO – ADDRESS	STATE	High Risk
FINACME	COMPANY_DAT A	TAX_PAYER_ID	--	IDENTIFICATIO N INFO – PERSONAL IDS	TAX ID NUMBER (TIN)	High Risk
FINACME	COMPANY_DAT A	ZIP	--	BIOGRAPHIC INFO – ADDRESS	POSTAL CODE	High Risk
HCM1	COUNTRIES	COUNTRY_NA ME	--	BIOGRAPHIC INFO – ADDRESS	COUNTRY	High Risk
HCM1	DEPARTMENTS	DEPARTMENT_ NAME	--	JOB INFO – ORG DATA	DEPARTMENT NAME	Low Risk
HCM1	EMPLOYEES	EMAIL	This is the e…	IDENTIFICATIO N INFO – PUBLIC IDS	EMAIL ADDRESS	High Risk
HCM1	EMPLOYEES	EMPLOYEE_ID	This is the u…	JOB INFO – EMPLOYEE DATA	EMPLOYEE ID NUMBER	High Risk
HCM1	EMPLOYEES	FIRST_NAME	--	IDENTIFICATIO N INFO – PUBLIC IDS	FIRST NAME	High Risk
HCM1	EMPLOYEES	HIRE_DATE	--	JOB INFO – ORG DATA	HIRE DATE	Low Risk
HCM1	EMPLOYEES	JOB_ID	--	JOB INFO – EMPLOYEE DATA	JOB CODE	High Risk
HCM1	EMPLOYEES	LAST_NAME	--	IDENTIFICATIO N INFO – PUBLIC IDS	LAST NAME	High Risk

Sample Script to Create a User with Minimum Privileges

You can create a user with required minimum privileges to run the Oracle Database Security Assessment Tool Collector with a script.

Purpose

Create a DBSAT user to run the Oracle DBSAT Collector script with required privileges.

Sample Script

```
create user dbsat_user identified by dbsat_user;
--If Database Vault is enabled, connect as DV_ACCTMGR to run this
command
grant create session to dbsat_user;
grant select_catalog_role to dbsat_user;
grant select on sys.registry$history to dbsat_user;
grant select on sys.dba_users_with_defpwd to dbsat_user; // 11g and 12c
grant audit_viewer to dbsat_user; // 12c
grant capture_admin to dbsat_user;// 12c covers sys.dba_priv_captures,
sys.priv_capture$, sys.capture_run_log$
--If Database Vault is enabled, connect as DV_OWNER to run this command
grant DV_SECANALYST to dbsat_user;
```

Best Practices

Collector - OS Commands

As a general best practice, you should not put username/password credentials in cleartext in an application or file. When you provide the password on the command line while executing `dbsat collect`, someone can retrieve credentials, either using history or executing the `ps` Unix command or any similar Windows command. Therefore, Oracle recommends that you enter the password when prompted.

Attribution for Third-Party Licenses

Third-party licenses used in the Database Security Assessment Tool Release 2.2.1.0.0

About Third-Party Licenses

For third party technology that you receive from Oracle in binary form which is licensed under an open source license that gives you the right to receive the source code for that binary, you can obtain a copy of the applicable source code from this page. If the source code for the technology was not provided to you with the binary, you can also receive a copy of the source code on physical media by submitting a written request to:

```
Oracle America, Inc.
Attn: Associate General Counsel
Development and Engineering Legal
500 Oracle Parkway, 10th Floor
Redwood Shores, CA 94065
```

Or, you may send an email to Oracle using this form. Your request should include:

```
The name of the component or binary file(s) for which you are requesting the
source code
The name and version number of the Oracle product
The date you received the Oracle product
Your name
Your company name (if applicable)
Your return mailing address and email
A telephone number in the event we need to reach you
```

We may charge you a fee to cover the cost of physical media and processing. Your request must be sent (i) within three (3) years of the date you received the Oracle product that included the component or binary file(s) that are the subject of your request, or (ii) in the case of code licensed under the GPL v3, for as long as Oracle offers spare parts or customer support for that product model

XlsxWriter, Version: 1.2.2

Copyright (c) 2013, John McNamara <`jmcnamara@cpan.org`>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.