Comprehensive Implementation Guide: Oracle Database Security with Data Safe and Database Vault (PRJ-OCI-DB-093)

1. Project Overview

The **PRJ-OCI-DB-093** project delivers a robust, multi-layered security architecture for Oracle Databases deployed on Oracle Cloud Infrastructure (OCI). This solution integrates native OCI security services—specifically **Oracle Data Safe** and **OCI Vault**—with core Oracle Database security features like **Transparent Data Encryption (TDE)** and **Oracle Database Vault**. The primary objective is to establish a continuous security and compliance posture for sensitive data assets, mitigating the risks associated with data breaches, unauthorized access, and compliance failures.

This guide provides a detailed, production-ready blueprint for implementing this framework, ensuring data is protected at rest and in transit, while privileged user access is strictly controlled.

Key Security Components

Component	Function	Security Benefit
Oracle Data Safe	Unified control center for security assessment, user assessment, data discovery, data masking, and activity auditing.	Continuous compliance monitoring, risk detection, and data de-sensitization for non-production environments.
OCI Vault	Managed service for storing and managing encryption keys and secrets.	Customer-managed key control for TDE, ensuring separation of duties and compliance with key management policies.
Transparent Data Encryption (TDE)	Encrypts data files at rest, making data unreadable without the encryption key.	Protects against unauthorized access to data files outside the database, a critical requirement for many regulatory frameworks.
Oracle Database Vault	Prevents privileged users (e.g., `SYS`, `SYSTEM`) from accessing application data, enforcing separation of duties.	Mitigates the risk of insider threats and privileged account abuse, a key control for SOC 2 and ISO 27001.

2. Business Context and Value Proposition

Database security is a foundational requirement for any enterprise, yet manual security management is often complex, error-prone, and non-scalable. This project addresses these challenges by automating and centralizing key security controls, delivering significant quantifiable business value.

Quantified Business Value and ROI

Metric	Description	Value Proposition
Risk Mitigation (Data Breach)	Reduces the probability and impact of a data breach by implementing mandatory encryption and access controls.	**ROI:** A single data breach can cost millions (average cost in 2024 is ~$4.45 million USD). This solution acts as a critical insurance policy, protecting brand reputation and avoiding massive regulatory fines (e.g., GDPR, HIPAA).
Compliance Automation	Automates the collection of audit data and provides continuous security assessments via Oracle Data Safe.	**Efficiency Gain:** Reduces manual effort for compliance reporting and auditing by up to 70%. Auditors can directly access Data Safe reports, significantly shortening audit cycles.
Insider Threat Prevention	Database Vault enforces separation of duties, preventing database administrators (DBAs) from viewing sensitive application data.	**Cost Savings:** Eliminates the risk of data theft or sabotage by privileged users, a major vector for corporate espionage and data loss.
Operational Efficiency	Centralized security management through Data Safe reduces the complexity of managing security across multiple databases.	**Time Savings:** Estimated 10-15 hours per week saved on manual security checks, patching, and audit log aggregation, allowing security teams to focus on threat hunting.

Risk Mitigation Scenarios

The solution is specifically designed to mitigate the following high-priority risks:

1. **Unauthorized Access and Privilege Abuse:** Database Vault prevents privileged users from accessing application data, even if they have OS-level access to the database files.
2. **Data at Rest Exposure:** TDE, with keys managed in OCI Vault, ensures that if the underlying storage is compromised, the data remains encrypted and unreadable.
3. **SQL Injection and Application-Level Attacks:** Continuous monitoring via Data Safe helps detect and alert on anomalous database activity indicative of an

attack.

4. **Compliance Violations:** Automated security assessments identify configuration drift and ensure the database remains compliant with internal and external standards.

3. GRC Mapping (Governance, Risk, and Compliance)

This solution directly addresses critical controls across major compliance frameworks, providing a clear path to audit readiness.

NIST SP 800-53 Rev. 5 Mapping

Control Family	Control ID	Description	Solution Component
Access Control (AC)	AC-3 (Access Enforcement)	Enforce approved authorizations for controlling access to information and system resources.	Oracle Database Vault Realms and Command Rules.
Audit and Accountability (AU)	AU-2 (Audit Events)	Determine, document, and disseminate a list of auditable events.	Oracle Data Safe Activity Auditing and unified audit trail.
System and Communications Protection (SC)	SC-13 (Cryptographic Protection)	Implement cryptographic mechanisms to protect the confidentiality of data at rest.	Transparent Data Encryption (TDE) with OCI Vault-managed keys.
Configuration Management (CM)	CM-6 (Configuration Settings)	Establish and enforce security configuration settings.	Oracle Data Safe Security Assessment for continuous monitoring of configuration drift.

ISO/IEC 27001:2022 Mapping

Annex A Control	Control ID	Description	Solution Component
Organizational Controls	5.15 (Access Control)	Define and implement rules for access control.	Database Vault and OCI IAM policies for OCI Vault access.
Technological Controls	8.12 (Data Leakage Prevention)	Implement measures to prevent data leakage.	Data Masking and Subsetting (Data Safe) for non-production environments.
Technological Controls	8.24 (Use of Cryptography)	Implement cryptography to protect the confidentiality, authenticity, and integrity of information.	TDE using OCI Vault.

SOC 2 Trust Services Criteria Mapping

Criteria	Description	Solution Component
Security (CC6.1)	Logical access security measures are in place to protect the entity's information and systems.	Database Vault Realms and Command Rules to restrict privileged access to sensitive data.
Security (CC6.7)	Data is classified and protected based on its sensitivity.	TDE for all sensitive data at rest; Data Safe for data discovery and classification.
Confidentiality (C1.2)	Data is protected during storage and transmission.	TDE for storage; OCI network security and Data Safe Private Endpoint for secure transmission.

4. Prerequisites

Successful deployment requires the following accounts, tools, and permissions to be configured prior to starting the implementation.

Required Accounts and Resources

1. **OCI Tenancy:** An active Oracle Cloud Infrastructure account.
2. **Target Database:** An existing Oracle Database (Autonomous Database, Exadata Cloud Service, or Database Cloud Service) to be secured.
3. **Compartment OCID:** The OCID of the compartment where the database and security resources will reside.
4. **Network Configuration:** A Virtual Cloud Network (VCN) and Subnet for deploying the Data Safe Private Endpoint.

Required Tools

1. **OCI Command Line Interface (CLI):** Must be installed and configured with appropriate credentials and region settings.

```
# Installation (Example for Linux/macOS)
bash -c "$(curl -L https://raw.githubusercontent.com/oracle/oci-cli/master/install.sh)"
# Configuration
oci setup config
```

2. **SQL Client:** A tool like SQL*Plus, SQL Developer, or SQLcl to connect to the target Oracle Database.

Required IAM Policies

The following IAM policies must be in place to allow the necessary services to interact:

Policy Statement	Purpose	Service
`Allow group <DB_ADMIN_GROUP> to manage vaults in compartment <COMPARTMENT_NAME>`	Allows administrators to create and manage OCI Vault and Keys.	Identity
`Allow dynamic-group <DB_SYSTEM_DG> to use keys in compartment <COMPARTMENT_NAME>`	Allows the database system to access the TDE master key in OCI Vault.	Database/KMS
`Allow group <DATA_SAFE_ADMIN_GROUP> to manage data-safe-private-endpoints in compartment <COMPARTMENT_NAME>`	Allows administrators to set up the Data Safe Private Endpoint.	Data Safe
`Allow service datasafesystem to manage target-databases in compartment <COMPARTMENT_NAME>`	Allows the Data Safe service to register and manage target databases.	Data Safe

5. Architecture Overview

The architecture is designed around a principle of centralized security management and distributed enforcement.

Solution Components and Data Flow

1. **Target Oracle Database:** The central asset. It enforces **TDE** using the key provided by OCI Vault and enforces access control via **Database Vault**.
2. **OCI Vault (Key Management):** Stores the TDE master key. The database system communicates with the Vault's management endpoint to retrieve the key, ensuring the key lifecycle is separate from the database lifecycle.
3. **Oracle Data Safe:** Acts as the security control plane. It connects to the target database to perform:
    - **Security Assessment:** Reads configuration metadata.
    - **User Assessment:** Reads user and privilege metadata.
    - **Activity Auditing:** Pulls audit logs from the database's unified audit trail.
    - **Data Masking:** Connects to mask data in non-production clones.

4. **Data Safe Private Endpoint:** For databases in a private subnet, this endpoint is deployed within the VCN to provide a secure, private connection path for Data Safe to communicate with the target database, avoiding exposure over the public internet.

Security Boundaries

- **Data at Rest:** Protected by TDE, with the key secured in OCI Vault.
- **Privileged Access:** Controlled by Database Vault, preventing DBAs from accessing application data.
- **Monitoring and Compliance:** Centralized in Data Safe, which acts as an independent security auditor.

6. Step-by-Step Implementation

The deployment is structured into three critical stages: Key Management (TDE), Database Vault Configuration, and Data Safe Integration.

Stage 1: Configure TDE with OCI Vault

This stage ensures the database uses a customer-managed master encryption key, a critical requirement for many compliance regimes.

6.1. Define Environment Variables

Replace the placeholder values with your actual OCIDs.

```
# OCI Compartment OCID where resources will be deployed
COMPARTMENT_OCID="ocid1.compartment.oc1..exampleuniqueID"
# OCID of the target Autonomous Database or Database Cloud Service
DB_OCID="ocid1.autonomousdatabase.oc1.phx.exampleuniqueID"
# Display name for the OCI Vault
VAULT_DISPLAY_NAME="DB_Security_Vault_PRJ093"
# Display name for the TDE Master Key
KEY_DISPLAY_NAME="TDE_Master_Key_PRJ093"
```

6.2. Create an OCI Vault

The vault will be a Virtual Private Vault to ensure high security.

```
echo "Creating OCI Vault: ${VAULT_DISPLAY_NAME}"
VAULT_OCID=$(oci kms management vault create \
    --compartment-id $COMPARTMENT_OCID \
    --display-name $VAULT_DISPLAY_NAME \
    --vault-type "VIRTUAL_PRIVATE" \
    --wait-for-state "ACTIVE" \
    --query 'data.id' --raw-output)

if [ -z "$VAULT_OCID" ]; then
    echo "Error: Vault creation failed."
    exit 1
fi
echo "Vault created with OCID: $VAULT_OCID"
```

6.3. Create a Master Encryption Key

The key is created within the vault and will be used as the TDE master key. We use a 256-bit AES key with HSM protection.

```
echo "Creating Master Encryption Key: ${KEY_DISPLAY_NAME}"
# Retrieve the management endpoint for the newly created vault
MANAGEMENT_ENDPOINT=$(oci kms management vault get --vault-id $VAULT_OCID --
query 'data."management-endpoint"' --raw-output)

KEY_INFO=$(oci kms management key create \
    --compartment-id $COMPARTMENT_OCID \
    --display-name $KEY_DISPLAY_NAME \
    --key-shape '{"algorithm":"AES","length":256}' \
    --protection-mode "HSM" \
    --endpoint $MANAGEMENT_ENDPOINT \
    --wait-for-state "ENABLED" \
    --query 'data."key-id" | [0] | {KEY_OCID: id, KEY_VERSION_OCID:
"current-key-version"}' --raw-output)

KEY_OCID=$(echo $KEY_INFO | jq -r '.KEY_OCID')
KEY_VERSION_OCID=$(echo $KEY_INFO | jq -r '.KEY_VERSION_OCID')

if [ -z "$KEY_OCID" ] || [ -z "$KEY_VERSION_OCID" ]; then
    echo "Error: Key creation failed."
    exit 1
fi
echo "Key created with OCID: $KEY_OCID and Version OCID: $KEY_VERSION_OCID"
```

6.4. Configure Database to use the OCI Vault Key

This command links the database to the OCI Vault key, enabling customer-managed TDE.

```
echo "Configuring Database ${DB_OCID} to use OCI Vault Key..."
oci db autonomous-database configure-key \
    --autonomous-database-id $DB_OCID \
    --kms-key-id $KEY_OCID \
    --kms-key-version-id $KEY_VERSION_OCID \
    --wait-for-state "AVAILABLE"

echo "Database successfully configured for TDE with OCI Vault."
```

Stage 2: Configure Oracle Database Vault

Database Vault is configured inside the database using SQL commands to enforce separation of duties.

6.5. Connect and Enable Database Vault

Connect to the database as a user with `SYSDBA` privileges.

```
-- 1. Enable Database Vault
EXEC DBMS_MACADM.ENABLE_DV;

-- 2. Create the Database Vault Owner and Account Manager
-- These users manage DV policies and user accounts, respectively, but
cannot access application data.
CREATE USER dv_owner IDENTIFIED BY "SecurePassword123#" CONTAINER=ALL;
GRANT CONNECT, RESOURCE, CREATE SESSION TO dv_owner;
EXEC DBMS_MACADM.AUTHORIZE_DV_OWNER('DV_OWNER');

CREATE USER dv_acctmgr IDENTIFIED BY "SecurePassword456#" CONTAINER=ALL;
GRANT CONNECT, RESOURCE, CREATE SESSION TO dv_acctmgr;
EXEC DBMS_MACADM.AUTHORIZE_DV_ACCTMGR('DV_ACCTMGR');

-- 3. Restart the database to activate Database Vault
SHUTDOWN IMMEDIATE;
STARTUP;
```

6.6. Create a Production-Ready Realm

A Realm protects a set of schemas, objects, and roles from unauthorized access, even by privileged users.

```
-- Connect as DV_OWNER
-- 4. Create a Realm to Protect Sensitive Application Data
BEGIN
  DBMS_MACADM.CREATE_REALM(
    realm_name    => 'Application Data Realm',
    description   => 'Protects core application schemas (APP_SCHEMA) and
sensitive tables',
    enabled       => DBMS_MACADM.Y_OPTION,
    audit_options => DBMS_MACADM.AUDIT_FAIL);
END;
/

-- 5. Add Objects to the Realm (e.g., the main application schema)
-- Replace APP_SCHEMA with the actual schema name
BEGIN
  DBMS_MACADM.ADD_OBJECT_TO_REALM(
    realm_name  => 'Application Data Realm',
    object_owner => 'APP_SCHEMA',
    object_name => '%',
    object_type => 'SCHEMA');
END;
/

-- 6. Authorize Users/Roles to Access the Realm
-- Typically, the application role (APP_ROLE) and the application user
(APP_USER) are authorized.
BEGIN
  DBMS_MACADM.ADD_AUTH_TO_REALM(
    realm_name  => 'Application Data Realm',
    grantee     => 'APP_ROLE',
    rule_set_name => 'ANY_ACCESS');

  DBMS_MACADM.ADD_AUTH_TO_REALM(
    realm_name  => 'Application Data Realm',
    grantee     => 'APP_USER',
    rule_set_name => 'ANY_ACCESS');
END;
/
```

Stage 3: Integrate with Oracle Data Safe

This stage connects the database to the Data Safe service for continuous monitoring and assessment.

6.7. Create a Data Safe Private Endpoint

If the database is in a private subnet, a Private Endpoint is required.

```
# Define network variables
SUBNET_OCID="ocid1.subnet.oc1..examplesubnetID"
VCN_OCID="ocid1.vcn.oc1..examplevcnID"

echo "Creating Data Safe Private Endpoint..."
DS_PE_OCID=$(oci data-safe private-endpoint create \
    --compartment-id $COMPARTMENT_OCID \
    --display-name "DS_Private_Endpoint_PRJ093" \
    --subnet-id $SUBNET_OCID \
    --vcn-id $VCN_OCID \
    --wait-for-state "ACTIVE" \
    --query 'data.id' --raw-output)

if [ -z "$DS_PE_OCID" ]; then
    echo "Error: Data Safe Private Endpoint creation failed."
    exit 1
fi
echo "Data Safe Private Endpoint created with OCID: $DS_PE_OCID"
```

6.8. Prepare the Target Database User for Data Safe

Data Safe requires a dedicated user with specific roles to perform its functions.

```
-- Connect as SYSDBA
CREATE USER DS_TARGET_USER IDENTIFIED BY "DataSafeSecurePwd789#" DEFAULT
TABLESPACE USERS QUOTA UNLIMITED ON USERS;
GRANT CREATE SESSION TO DS_TARGET_USER;

-- Grant required roles for Data Safe functionality
-- For Security Assessment, User Assessment, and Data Discovery
GRANT SELECT ANY DICTIONARY TO DS_TARGET_USER;
GRANT SELECT ON V_$DATABASE TO DS_TARGET_USER;
GRANT SELECT ON DBA_USERS TO DS_TARGET_USER;

-- For Activity Auditing (required roles depend on the audit policy)
GRANT AUDIT_ADMIN TO DS_TARGET_USER;
GRANT EXECUTE ON DBMS_AUDIT_MGMT TO DS_TARGET_USER;
```

6.9. Register the Target Database with Data Safe

Use the OCI CLI to register the database, referencing the Private Endpoint.

```
echo "Registering Target Database with Data Safe..."
oci data-safe target-database create \
    --compartment-id $COMPARTMENT_OCID \
    --display-name "TargetDB-PRJ093" \
    --database-details
'{"databaseType":"DATABASE_CLOUD_SERVICE","dbSystemId":"
<DB_SYSTEM_OCID>","infrastructureType":"DATABASE_CLOUD_SERVICE"}' \
    --connection-option
'{"connectionType":"PRIVATE_ENDPOINT","datasafePrivateEndpointId":"'$DS_PE_OCI
\
    --credentials
'{"userName":"DS_TARGET_USER","password":"DataSafeSecurePwd789#"}' \
    --wait-for-state "ACTIVE"

echo "Target Database registered successfully with Data Safe."
```

7. Validation & Testing

A robust validation process is essential to confirm that all security controls are correctly implemented and functioning as intended.

7.1. TDE Validation

Test Case: Verify the database is using the OCI Vault key and that new tablespaces are encrypted.

Step	Command/Action	Expected Result
1. Check Key Source	`SELECT KEY_ID, KEY_VERSION_ID FROM V$ENCRYPTION_KEYS;`	The `KEY_ID` and `KEY_VERSION_ID` should match the OCIDs captured in Stage 1.
2. Test New Tablespace	`CREATE TABLESPACE test_tde DATAFILE 'test_tde.dbf' SIZE 10M ENCRYPTION USING 'AES256';`	Command succeeds.
3. Verify Encryption	`SELECT TABLESPACE_NAME, ENCRYPTED FROM DBA_TABLESPACES WHERE TABLESPACE_NAME = 'TEST_TDE';`	The `ENCRYPTED` column for `TEST_TDE` should show `YES`.

7.2. Database Vault Validation

Test Case: Verify Database Vault is active and that privileged users are blocked from accessing protected data.

Step	Command/Action	Expected Result
1. Check DV Status	`SELECT * FROM V$DV_STATUS;`	The status should show `TRUE`.
2. Test Realm Protection (Negative Test)	Connect as `SYS` or `SYSTEM`. Attempt to `SELECT * FROM APP_SCHEMA.SENSITIVE_TABLE;`	The query should fail with an `ORA-01031: insufficient privileges` or a specific Database Vault error (`ORA-47400`).
3. Test Authorized Access (Positive Test)	Connect as `APP_USER` (authorized in the Realm). Attempt to `SELECT * FROM APP_SCHEMA.SENSITIVE_TABLE;`	The query should succeed.

7.3. Data Safe Validation

Test Case: Confirm Data Safe connectivity and the ability to perform security assessments.

Step	Command/Action	Expected Result
1. Check Target Status	Navigate to the Data Safe service in the OCI Console. Check the status of **TargetDB-PRJ093**.	Status should be **Active**.
2. Run Security Assessment	Initiate a Security Assessment for the target database.	The assessment should complete successfully, providing a detailed report on the database's security posture.
3. Verify Auditing	Check the Activity Auditing dashboard.	Audit records should be flowing from the target database to Data Safe.

8. Troubleshooting

This section addresses common issues encountered during the deployment and operation of the security framework.

Issue	Potential Cause	Resolution
TDE Key Not Found	IAM policy missing for the database to access OCI Vault.	Ensure the dynamic group for the database system has a policy to `use keys in compartment <COMPARTMENT_NAME>`. Verify the database is a member of the correct dynamic group.
Data Safe Registration Fails	Incorrect connection details, Private Endpoint issues, or network security group (NSG) rules blocking traffic.	1. Verify the `DS_TARGET_USER` credentials. 2. Check the NSG/Security List rules for the Data Safe Private Endpoint subnet and the database subnet. They must allow traffic on the database listener port (typically 1521).
Database Vault Blocks SYS	Realm configuration is too restrictive or the user is attempting an unauthorized action.	This is the intended behavior. If a privileged user needs access, connect as the `DV_OWNER` or `DV_ACCTMGR` to temporarily modify Realm authorizations or use a Command Rule exception.
OCI CLI Timeout	Long-running OCI CLI commands (e.g., `vault create`, `configure-key`) may time out.	Use the `--wait-for-state` flag as shown in the implementation steps. If the issue persists, check the resource status in the OCI Console and proceed manually.
Data Safe Auditing Lag	Audit data collection is delayed or incomplete.	Verify the audit policy is correctly provisioned in Data Safe and the `DS_TARGET_USER` has the necessary `AUDIT_ADMIN` privileges. Check the database's unified audit trail status.

9. Cost Optimization

While security is paramount, optimizing the cost of the underlying services is crucial for long-term operational efficiency.

1. **Oracle Data Safe Licensing:** Data Safe is often included with Oracle Database Cloud Service subscriptions. However, the storage of collected audit data is a potential cost driver.
    - **Optimization:** Implement a targeted audit policy. Use Data Safe's features to collect only the necessary audit events (e.g., failed logins, DDL changes,

access to sensitive tables) rather than collecting all activity. Archive or purge older audit data regularly.

2. **OCI Vault Costs:** OCI Vault charges are based on the number of key versions and the volume of cryptographic API calls.
    - **Optimization:** Minimize unnecessary key rotations. While rotation is a security best practice, excessive rotation (e.g., daily) can increase costs. A quarterly or semi-annual rotation schedule is typically sufficient.

3. **Database Sizing:** The underlying Oracle Database is the largest cost component.
    - **Optimization:** Use the Autonomous Database's auto-scaling feature to pay only for the compute resources consumed. Regularly review performance metrics to ensure the database is not over-provisioned in terms of CPU or storage.

4. **Network Costs:** Data transfer costs between the database and Data Safe Private Endpoint are minimal but should be monitored.
    - **Optimization:** Ensure all components (Database, Vault, Data Safe Private Endpoint) are deployed within the same OCI region to avoid cross-region data transfer charges.

10. Security Best Practices

Implementing the solution is the first step; maintaining a strong security posture requires adherence to ongoing best practices.

1. **Principle of Least Privilege (PoLP):**
    - Ensure the `DS_TARGET_USER` has only the minimum required privileges for auditing and assessment. Never grant `DBA` or `SYSDBA` to this user.
    - Regularly review the privileges of all users, especially those with `DV_OWNER` or `DV_ACCTMGR` roles.

2. **Key Rotation Policy:**
    - Establish a mandatory, automated policy for the regular rotation of the OCI Vault TDE master key (e.g., every 90 days). This is a critical control for many compliance standards.
    - Use OCI's built-in key rotation features to simplify this process.

3. **Database Vault Realm Maintenance:**
    - Define and enforce Realms for *all* sensitive application schemas and critical database objects.
    - Regularly review Realm authorizations to ensure only necessary users and roles have access.
    - Use **Command Rules** within Database Vault to restrict powerful SQL commands (e.g., `ALTER SYSTEM`, `CREATE USER`) based on factors like time, client IP, or program.
4. **Data Masking for Non-Production:**
    - **NEVER** use production data in development, testing, or staging environments.
    - Utilize Data Safe's **Data Masking and Subsetting** feature to create realistic, but de-sensitized, copies of the production database for non-production use.
5. **Continuous Monitoring and Alerting:**
    - Regularly review Data Safe's **Security Assessment** and **User Assessment** reports to detect configuration drift, new vulnerabilities, and risky user behavior.
    - Configure alerts in Data Safe and OCI Monitoring for critical events, such as unauthorized access attempts (blocked by DV) or changes to the TDE key configuration.
6. **Patch Management:**
    - Ensure the underlying Oracle Database and OCI infrastructure are kept up-to-date with the latest security patches to mitigate known vulnerabilities.

11. Cleanup (Optional)

To completely remove the deployed resources and revert the database configuration:

1. **Deregister Target Database from Data Safe**:

```
oci data-safe target-database delete --target-database-id
<TARGET_DB_OCID> --force
```

2. **Delete Data Safe Private Endpoint**:

```
oci data-safe private-endpoint delete --private-endpoint-id
<DS_PE_OCID> --force
```

3. **Delete OCI Vault Key and Vault**:

```
# Schedule key for deletion (required before vault deletion)
oci kms management key schedule-deletion --key-id $KEY_OCID --time-of-
deletion 2026-01-30T00:00:00Z
# Delete the vault
oci kms management vault delete --vault-id $VAULT_OCID --force
```

4. **Disable Database Vault** (if required):

```
-- Connect as DV_OWNER
EXEC DBMS_MACADM.DISABLE_DV;
-- Restart the database
SHUTDOWN IMMEDIATE;
STARTUP;
```

12. References

[1] Oracle Cloud Infrastructure Documentation. *Manage Database Security with Oracle Data Safe*. [2] Oracle Cloud Infrastructure Documentation. *Using Customer-Managed Keys for Transparent Data Encryption*. [3] Oracle Database Documentation. *Introduction to Oracle Database Vault*. [4] NIST Special Publication 800-53, *Security and Privacy Controls for Information Systems and Organizations*. [5] ISO/IEC

27001:2022, *Information security, cybersecurity and privacy protection — Information security management systems — Requirements*.