Oracle OCI Vault (HSM) Setup

CBOM Secure · HSM Sensor (Discover_HSM)

Overview

This guide configures Oracle Cloud Infrastructure (OCI) Vault so the CBOM Secure Discover_HSM sensor can discover HSM-protected keys via the OCI REST API, read-only:

- Master Encryption Keys (MEKs) - HSM-protected AES, RSA, and ECDSA keys in Virtual Private Vaults
- Key versions - active and historical versions, including rotation history
- Key metadata - algorithm, length, protection mode (HSM vs SOFTWARE), lifecycle state, timestamps
- Vault metadata - vault type, compartment, and key counts

The sensor operates with read-only (inspect) IAM permissions. No key material is ever exported or decrypted.

Prerequisites

- OCI Console access with Administrator (or equivalent IAM) privileges to create users, groups, and policies.
- A tenancy subscribed to a region supporting OCI Vault with Virtual Private Vault (HSM) protection.
- The OCI CLI installed locally if using CLI steps.
- Your Tenancy OCID (Profile > Tenancy).
- The CBOM platform deployed with access to the sensor configuration interface.

Step-by-Step Guide

Step 1: Confirm a Virtual Private Vault (HSM)

- In the OCI Console, open Security > Vault and select the correct compartment.
- Click Create Vault (or reuse an existing one).
- Set a Name (e.g. cbom-discovery-vault) and Vault Type = Virtual Private Vault (the HSM-backed option; not Default).
- Create, wait for Active status, and note the Vault OCID.

Note Virtual Private Vault incurs extra cost. If one already exists, point the sensor at that vault's compartment instead of creating a new one.

Step 2: Create a Dedicated IAM Group and User

```
# Console: Identity & Security > Groups > Create Group
Name: cbom-readonly-group
Description: Read-only group for CBOM cryptographic asset discovery

# Console: Identity & Security > Users > Create User
Name: cbom-sensor-user
Description: Service account for CBOM HSM sensor - read-only vault access
# then: add cbom-sensor-user to cbom-readonly-group
```

Step 3: Write a Least-Privilege IAM Policy

Create policy cbom-vault-readonly-policy scoped to the vault's compartment with inspect-only statements:

```
Allow group cbom-readonly-group to inspect vaults in compartment <your-compartment>
Allow group cbom-readonly-group to inspect keys in compartment <your-compartment>
Allow group cbom-readonly-group to inspect key-versions in compartment <your-compartment>
```

Note inspect (not read) lists and views resource metadata without accessing key material - exactly what the sensor needs. Use in tenancy only if cross-compartment discovery is required.

Step 4: Generate an API Signing Key

```
mkdir -p ~/.oci/cbom
openssl genrsa -out ~/.oci/cbom/cbom_api_key.pem 2048
chmod 600 ~/.oci/cbom/cbom_api_key.pem
openssl rsa -pubout -in ~/.oci/cbom/cbom_api_key.pem -out
~/.oci/cbom/cbom_api_key_public.pem
```

In the Console, open cbom-sensor-user > API Keys > Add API Key, paste the public key, and record the user OCID, tenancy OCID, region, and fingerprint from the configuration preview. Also note the Compartment OCID.

Step 5: Configure the CBOM Secure Sensor

```
sensors:
  - name: Discover_HSM
    type: oci_vault
    enabled: true
    connection:
      tenancy_ocid: "ocid1.tenancy.oc1..aaaa..."
      user_ocid: "ocid1.user.oc1..aaaa..."
      region: "us-ashburn-1"
      fingerprint: "a1:b2:c3:d4:e5:f6:..."
      private_key_path: "/etc/cbom/keys/cbom_api_key.pem"
    scope:
      compartment_ocid: "ocid1.compartment.oc1..aaaa..."
      include_subcompartments: true
      vault_types: [VIRTUAL_PRIVATE]
      key_states: [ENABLED, DISABLED, PENDING_DELETION]
    discovery:
      include_key_versions: true
      include_key_metadata: true
      collect_rotation_history: true
    schedule:
      interval_hours: 24
      start_time: "02:00"
```

private_key_path must be readable only by the CBOM service account (chmod 600, chown cbom:cbom).

Step 6: Validate

```
# CBOM UI: Sensors > Discover_HSM > Run Now, then check logs for:
# Authenticated as user ... / Found N vault(s) type VIRTUAL_PRIVATE / Discovered N key(s)
# Then Inventory > Cryptographic Assets, filter Source: oci_vault
```

Confirm keys show Protection Mode HSM and the expected algorithms, cross-check the key count against the OCI Console, and verify no 401/403 errors appear.

Common Errors

Authorization failed or requested resource not found (404)

Cause: OCI conflates 403/404 - the user lacks permission on the resource, or the resource does not exist in the given compartment.

Resolution: Confirm the compartment_ocid, verify the policy has inspect vaults/keys/key-versions, and confirm cbom-sensor-user is in cbom-readonly-group.

NotAuthenticated (401) on Startup

Cause: The fingerprint in the YAML does not match the uploaded public key, or the private key file is inaccessible.

Resolution: Match the fingerprint against Users > API Keys, confirm private_key_path exists and is readable by the service account, and re-upload if the key was regenerated.

No Vaults Found Despite Vaults in the Console

Cause: The vault_types filter excludes the actual vault type, or the vault is in a child compartment with include_subcompartments false.

Resolution: Match vault_types to the vault's actual type (VIRTUAL_PRIVATE or DEFAULT) and set include_subcompartments: true.

Security Recommendations

- Restrict the RSA private key to the CBOM service account (chmod 600, chown cbom:cbom).
- Use a dedicated cbom-sensor-user - never reuse an admin or developer account.
- Scope the IAM policy to the minimum compartment rather than the whole tenancy.
- Rotate the API signing key at least annually (OCI keys do not auto-expire) and update both IAM and the host.
- Enable OCI Audit for the user and alarm on any call beyond inspect operations; avoid instance-principal auth in shared environments.

Conclusion

With a Virtual Private Vault confirmed, a dedicated inspect-only IAM user and group, an API signing key, and the Discover_HSM sensor configured, CBOM Secure maintains a continuous read-only inventory of every HSM-protected key and version in OCI Vault - enabling cryptographic visibility, compliance reporting, and algorithm risk assessment across Oracle Cloud.