PKCS#12 / PFX Keystore Setup

CBOM Secure · Keystore Sensor (Discover_Keystore)

Overview

This guide configures the CBOM Secure Discover_Keystore sensor to discover PKCS#12 (.p12, .pfx) keystores — password-protected containers bundling a private key with its certificate chain. The sensor scans configured paths read-only and extracts certificate subject/issuer DNs, serial number and validity, public key algorithm and size, signature algorithm, SANs, key usage, and fingerprints. It never reads, stores, or transmits private key material.

Prerequisites

- CBOM platform access and the Discover_Keystore agent installed on the host(s) with the .p12/.pfx files.
- At least one .p12 or .pfx file in the target directories.
- The store password for every keystore the sensor will open (supplied as secrets, never logged).
- Java 8+ or OpenSSL 1.1+ only if you need to create/convert keystores.
- A dedicated least-privilege OS account (cbom-sensor) with read-only access — never root.

Step-by-Step Guide

Step 1: Locate PKCS#12 Files

```
find /opt/certs /etc/ssl /var/lib/certs -type f \( -name "*.p12" -o -name "*.pfx" \) 2>/dev/null
# verify a file is a valid password-protected bundle (lists bags, no key material):
openssl pkcs12 -info -in /opt/certs/server.p12 -noout -passin pass:changeit
```

Step 2: (Optional) Export a PKCS#12 from JKS or PEM

If certificates are in a JKS keystore, export to PKCS#12; or bundle separate PEM cert/key files:

```
keytool -importkeystore \
  -srckeystore keystore.jks -srcstoretype JKS -srcstorepass <jks-pw> \
  -destkeystore out.p12 -deststoretype PKCS12 -deststorepass <p12-pw> -noprompt

openssl pkcs12 -export -in cert.pem -inkey key.pem -certfile chain.pem \
  -out bundle.p12 -name "myserver" -passout pass:<p12-pw>
```

Step 3: Grant Read-Only Filesystem Permissions

```
sudo useradd --system --no-create-home --shell /sbin/nologin cbom-sensor
sudo setfacl -R -m u:cbom-sensor:rX /opt/certs
sudo -u cbom-sensor openssl pkcs12 -info -in /opt/certs/server.p12 -noout -passin pass:<pw>
```

Step 4: Configure the CBOM Secure Sensor

```
sensors:
```

```
- name: Discover_Keystore
  enabled: true
  schedule: "0 2 * * *"
  options:
    scan_paths: [/opt/certs, /etc/ssl/private, /var/lib/tomcat/conf]
    file_extensions: [.p12, .pfx]
    recursive: true
    keystore_credentials:
      - path: /opt/certs/server.p12
        password_env: CBOM_KS_SERVER_PASSWORD
      - path: /opt/certs/wildcard.pfx
        password_env: CBOM_KS_WILDCARD_PASSWORD
    default_password_env: CBOM_KS_DEFAULT_PASSWORD
    exclude_paths: [/opt/certs/archive, /opt/certs/expired]
  output:
    format: cbom-json
    destination: platform
```

Set each password variable in a restricted environment file (chmod 640, root:cbom-sensor), then reload the agent.

Step 5: Validate

```
sudo journalctl -u cbom-agent -n 100 --no-pager
sudo cbom-ctl sensor run Discover_Keystore
```

Confirm the scan finds the expected PKCS#12 files and publishes assets. In the portal (Assets → Certificates), verify subject/expiry/algorithm are populated and no private-key field exists. Wrong-password files log a MAC verify warning.

Common Errors

MAC verify failure / incorrect password

Cause: The supplied password does not match, or the file uses legacy RC2/3DES encryption incompatible with the host OpenSSL.

Resolution: Verify with openssl pkcs12 -info; re-export legacy files with modern encryption.

Permission denied reading keystore

Cause: cbom-sensor lacks read permission on the file or its parent directory.

Resolution: Grant read via setfacl or group membership; re-run the sensor.

0 assets found despite files present

Cause: scan_paths or file_extensions don't match the actual files (e.g. missing leading dot).

Resolution: Compare a find command's output with the config; correct mismatches and re-run.

Security Recommendations

- Reference passwords via password_env — never embed them in the YAML.
- Run the sensor as a dedicated read-only account with no shell or sudo.

- Restrict the environment file to mode 640, root:cbom-sensor.
- Rotate PKCS#12 store passwords per policy and restart the agent.
- Add archived/expired keystore directories to exclude_paths.

Conclusion

The Discover_Keystore sensor provides continuous, read-only visibility into PKCS#12 keystores — extracting certificate and public-key metadata without ever touching private-key material — surfacing expiring certs, weak key sizes, and deprecated signature algorithms in CBOM Secure.