

BouncyCastle KeyStore (BKS) Setup

CBOM Secure · Keystore Sensor (Discover_Keystore)

Overview

This guide configures the CBOM Secure Discover_Keystore sensor to inventory cryptographic material in BouncyCastle KeyStore (BKS) files, widely used in Android applications and Java projects using the BouncyCastle provider. Read-only, the sensor extracts:

- Private keys and metadata (algorithm, key size, alias)
- Public key certificates (X.509)
- Certificate chains linked to key entries
- Trusted certificate entries
- Secret key entries (symmetric keys under protected aliases)

Prerequisites

- JDK 11 or later with keytool on PATH.
- The bcprov-jdk18on-*.jar provider JAR available at a stable path (use bcprov-jdk18on for JDK 11+).
- One or more .bks files with known paths and store passwords.
- Keystore store passwords (entry passwords may differ).
- A CBOM platform account able to manage sensor configurations.
- Read access for the sensor host to the BKS file directories, and the CBOM agent installed.

Step-by-Step Guide

Step 1: Locate BKS Files

```
find . -name "*.bks"
# scope to a parent directory:
find /opt/android-builds -name "*.bks" -type f
# include alternate extensions:
find /opt/android-builds -type f \( -name "*.bks" -o -name "*.keystore" \)
```

Step 2: Obtain the BouncyCastle Provider JAR

```
# example: version 1.78.1 for JDK 18+
curl -L -o /opt/cbom/bc/bcprov-jdk18on-178.jar \
  https://repo1.maven.org/maven2/org/bouncycastle/bcprov-jdk18on/1.78.1/bcprov-
jdk18on-1.78.1.jar
ls -lh /opt/cbom/bc/bcprov-jdk18on-178.jar
```

Step 3: Inspect a BKS Keystore with keytool

```
keytool -list \
  -keystore /opt/android-builds/app/src/main/res/raw/client_keystore.bks \
```

```
-storetype BKS \  
-storepass changeit \  
-provider org.bouncycastle.jce.provider.BouncyCastleProvider \  
-providerpath /opt/cbom/bc/bcprov-jdk18on-178.jar \  
-v
```

Record the alias names and entry types — these help validate sensor output later.

Step 4: Prepare the Sensor Host

```
mkdir -p /opt/cbom/bc  
cp bcprov-jdk18on-178.jar /opt/cbom/bc/  
chmod 644 /opt/cbom/bc/bcprov-jdk18on-178.jar  
java -version  
echo $JAVA_HOME  
export BKS_STORE_PASS="your-keystore-password" # prefer secret injection
```

Step 5: Configure the CBOM Secure Sensor

```
sensors:  
- name: Discover_Keystore  
  enabled: true  
  type: keystore  
  schedule: "0 2 * * *"  
  keystore_sources:  
    - path: /opt/android-builds/app/src/main/res/raw/client_keystore.bks  
      store_type: BKS  
      store_password: "${BKS_STORE_PASS}"  
      provider: org.bouncycastle.jce.provider.BouncyCastleProvider  
      provider_path: /opt/cbom/bc/bcprov-jdk18on-178.jar  
      label: client-keystore-production  
    - path: /opt/android-builds/build/outputs/keystore/release.bks  
      store_type: BKS  
      store_password: "${BKS_RELEASE_PASS}"  
      provider: org.bouncycastle.jce.provider.BouncyCastleProvider  
      provider_path: /opt/cbom/bc/bcprov-jdk18on-178.jar  
      label: release-keystore  
  discovery:  
    recursive_scan: true  
    scan_directories: [/opt/android-builds]  
    file_patterns: ["*.bks"]  
  cbom_platform:  
    endpoint: https://cbom.example.com/api/v1/ingest  
    api_key: "${CBOM_API_KEY}"  
    tls_verify: true  
sudo systemctl restart cbom-agent
```

Step 6: Validate

```
sudo journalctl -u cbom-agent -n 100 --no-pager | grep Discover_Keystore  
cbom-agent run-sensor --name Discover_Keystore
```

Confirm the sensor initialized with the configured sources, then check Inventory → Cryptographic Assets filtered by Source: BKS. Cross-check the entry count against the manual `keytool -list` output.

Common Errors

BKS not found (KeyStoreException)

Cause: The BouncyCastle provider JAR is not on the classpath, or `provider_path` is wrong/missing.

Resolution: Verify provider_path and JDK compatibility; reproduce with keytool using the same -providerpath.

Keystore was tampered with, or password was incorrect

Cause: store_password does not match, or the BKS file is corrupted.

Resolution: Verify the password with keytool -list; confirm the injected variable is exported in the agent's session; check the file was not modified.

NoClassDefFoundError: BouncyCastleProvider

Cause: The JAR variant is incompatible with the JDK (e.g. bcprov-jdk15on with JDK 18+).

Resolution: Use bcprov-jdk18on-*.jar for JDK 11+, update provider_path, and restart the agent.

Security Recommendations

- Never store keystore passwords in plaintext — use secret injection or a secrets manager.
- Restrict read access to BKS files to the sensor account only (no write/execute).
- Protect the provider JAR with strict ownership to prevent malicious replacement.
- Periodically re-run the find scan and reconcile against the sensor config.
- Use CBOM expiry/algorithm data to prioritize renewing weak certs (e.g. RSA-1024, MD5withRSA).

Conclusion

By locating BKS files, validating with keytool, and configuring the correct BouncyCastle provider JAR and credentials, the Discover_Keystore sensor delivers continuous, read-only visibility into keys and certificates across Android build artifacts and Java deployments — enabling expiry, algorithm, and lifecycle tracking in CBOM Secure.