Linux System Trust Store Setup

CBOM Secure · Files Sensor (Discover_Files)

Overview

This guide configures the CBOM Secure Discover_Files sensor to scan Linux system trust store directories for certificate assets, read-only:

- CA root and intermediate certificates (PEM and DER)
- Custom CA certificates added by administrators
- Trust anchor certificates used by the OS and applications
- Expired, revoked, or soon-to-expire certificates in the trust store

Directories covered: /etc/ssl/certs/ and /usr/local/share/ca-certificates/ (Debian/Ubuntu), /etc/pki/tls/certs/ and /etc/pki/ca-trust/source/ (RHEL family).

Prerequisites

- root or sudo access on the host to be scanned.
- A supported distribution (Debian 10+/Ubuntu 20.04+, or RHEL/CentOS/Alma/Rocky/Fedora).
- The CBOM sensor agent installed (or agentless network access).
- The CBOM platform endpoint and a valid sensor token; outbound HTTPS (443); openssl on the host.

Step-by-Step Guide

Step 1: Create a Low-Privilege Scanner Account

```
sudo useradd -r -s /sbin/nologin -d /nonexistent -c "CBOM Scanner Service Account" cbom-scanner
id cbom-scanner
```

Step 2: Verify Read Access to Trust Store Directories

Trust store directories are typically world-readable. Verify, and grant read+execute if missing:

```
ls -ld /etc/ssl/certs/ /usr/local/share/ca-certificates/
sudo chmod o+rx /etc/ssl/certs/                # only if not already readable
sudo -u cbom-scanner openssl x509 -in /etc/ssl/certs/DigiCert_Global_Root_CA.pem -noout -subject -dates
```

Step 3: Configure the CBOM Secure Sensor

```
sensor:
  name: Discover_Files
  enabled: true
connection:
  platform_url: "https://cbom.your-organization.com"
```

```
  auth_token: "${CBOM_SENSOR_TOKEN}"
  verify_tls: true
identity:
  run_as_user: cbom-scanner
scan_targets:
  - id: debian_ubuntu_system_certs
    path: /etc/ssl/certs/
    recursive: true
    file_patterns: ["*.pem", "*.crt", "*.cer", "*.der", "ca-certificates.crt"]
  - id: debian_ubuntu_custom_cas
    path: /usr/local/share/ca-certificates/
    file_patterns: ["*.crt", "*.pem"]
  - id: rhel_system_certs
    path: /etc/pki/tls/certs/
    file_patterns: ["*.pem", "*.crt", "ca-bundle.crt"]
  - id: rhel_trust_anchors
    path: /etc/pki/ca-trust/source/
    file_patterns: ["*.pem", "*.crt", "*.p7b"]
scan_options:
  parse_bundles: true
  include_expired: true
  schedule_cron: "0 2 * * *"
```

Place the config under /etc/cbom/sensors/ (chmod 640, owned root:cbom-scanner) and create the log directory.

Step 4: Validate

```
sudo -u cbom-scanner cbom-sensor scan \
  --config /etc/cbom/sensors/discover-files-linux-truststore.yaml --dry-run --verbose
```

Confirm certificate files are found per path, then verify under Assets → Certificates filtered by Source: Discover_Files and your host, with subject/issuer/expiry/algorithm populated.

Common Errors

Permission denied reading certificate directory

Cause: cbom-scanner lacks read/execute on the directory.

Resolution: chmod o+rx the directory and confirm with sudo -u cbom-scanner ls.

Authentication token invalid or expired (401)

Cause: The auth_token is wrong, expired, or revoked.

Resolution: Generate a new sensor token (Settings → Sensor Tokens), update the config, and restart.

No certificate files found in path

Cause: Wrong distribution paths, or files use other extensions.

Resolution: Disable non-applicable targets and add observed extensions to file_patterns.

Security Recommendations

- The cbom-scanner account must have no sudo, no shell, and access only to certificate directories.

- Never point the sensor at private key directories (/etc/ssl/private/, /etc/pki/tls/private/).
- Protect the config file (root-owned, mode 640) — it holds the platform token.
- Rotate sensor tokens every 90 days.
- Use auditd to log cbom-scanner file access.

Conclusion

The Discover_Files sensor provides read-only discovery of all X.509 certificates in the Linux system trust store across Debian and RHEL families, under a dedicated low-privilege account — delivering full visibility into CA certs, custom trust anchors, and weak/expired certificates in CBOM Secure.