Windows Certificate Stores Setup

CBOM Secure · WinCerts Sensor (Discover_WinCerts)

Overview

The CBOM Secure Discover_WinCerts sensor performs read-only discovery of cryptographic material in the Windows Local Machine certificate stores - certificates, public keys, chains, and metadata (subject, issuer, validity, key algorithm and length). It reads:

- MY - Personal certificates (including those with associated private keys)
- ROOT - Trusted Root Certification Authorities
- CA - Intermediate Certification Authorities
- TRUST - Enterprise Trust store
- TrustedPeople - certificates for trusted individuals
- TrustedPublisher - certificates for trusted software publishers

Prerequisites

- Administrative access to the target Windows machine or domain.
- Windows Server 2016+ or Windows 10/11.
- ADUC (domain accounts) or Local Users and Groups (local accounts).
- The CBOM sensor package downloaded and extracted on the target.
- PowerShell 5.1+, certlm.msc, and certutil.exe available.
- Network connectivity to the CBOM platform, plus its URL and a valid API token.

Step-by-Step Guide

Step 1: Create a Dedicated Service Account

Create a least-privilege account (never Domain Admin). Domain: ADUC > New > User svc-cbom-wincerts. Local: Computer Management > Local Users and Groups > New User svc-cbom-wincerts. Set a strong password with Password never expires and User cannot change password.

Step 2: Grant Read Access to the Certificate Stores

For MY-store private keys, open certlm.msc, and for each certificate with a private key: All Tasks > Manage Private Keys > Add svc-cbom-wincerts > grant Read only. Confirm the other stores are readable:

```
$stores = @("ROOT","CA","TRUST","TrustedPeople","TrustedPublisher")
foreach ($s in $stores) {
  $st = New-Object
System.Security.Cryptography.X509Certificates.X509Store($s,'LocalMachine')
  try { $st.Open('ReadOnly'); Write-Host "$s: Accessible"; $st.Close() }
  catch { Write-Host "$s: NOT accessible - $_" } }
```

Grant Log on as a service via secpol.msc > User Rights Assignment > Log on as a service > add the account.

Step 3: Install and Restrict the Sensor Files

```
mkdir "C:\Program Files\CBOM\Logs"
icacls "C:\Program Files\CBOM\Logs" /grant "svc-cbom-wincerts:(OI)(CI)M"
icacls "C:\Program Files\CBOM\Sensors\WinCerts" /inheritance:r
icacls "C:\Program Files\CBOM\Sensors\WinCerts" /grant "Administrators:(OI)(CI)F"
icacls "C:\Program Files\CBOM\Sensors\WinCerts" /grant "svc-cbom-wincerts:(OI)(CI)RX"
```

Step 4: Configure the CBOM Secure Sensor

```
sensor:
  name: Discover_WinCerts
  description: "Windows Local Machine Certificate Store Discovery"
platform:
  url: "https://cbom.example.com"
  api_token: "${CBOM_API_TOKEN}"
  environment: "production"
target:
  hostname: "WIN-APPSERVER01.corp.example.com"
  os: "windows"
  scan_interval_hours: 24
certificate_stores:
  scope: "LocalMachine"
  stores:
    - { name: MY, enabled: true, include_private_key_metadata: true }
    - { name: ROOT, enabled: true }
    - { name: CA, enabled: true }
    - { name: TRUST, enabled: true }
    - { name: TrustedPeople, enabled: true }
    - { name: TrustedPublisher, enabled: true }
  discovery:
    collect_subject: true; collect_issuer: true; collect_validity_dates: true
    collect_key_algorithm: true; collect_key_length: true; collect_chain: true
service_account:
  username: "CORP\\svc-cbom-wincerts"
```

Note The api_token grants CBOM ingestion - source it from a secrets manager, and restrict the config file to Administrators and the service account.

Step 5: Deploy as a Windows Service (or Scheduled Task)

```
sc create "CBOM-WinCerts" ^
  binPath= "\"C:\Program Files\CBOM\Sensors\WinCerts\cbom-sensor-runner.exe\" --config
\"C:\Program Files\CBOM\Sensors\WinCerts\sensor-config.yaml\"" ^
  DisplayName= "CBOM WinCerts Sensor" start= auto ^
  obj= "CORP\svc-cbom-wincerts" password= "<ServiceAccountPassword>"
sc failure "CBOM-WinCerts" reset= 86400 actions=
restart/60000/restart/120000/restart/300000
sc start "CBOM-WinCerts" && sc query "CBOM-WinCerts"
```

Alternatively register a daily scheduled task (Register-ScheduledTask) running as the service account at RunLevel Highest.

Step 6: Validate

```
certutil -store MY && certutil -store ROOT && certutil -store CA
```

```
type "C:\Program Files\CBOM\Logs\wincerts-sensor.log"     # look for Scan complete + HTTP 200
```

In the CBOM platform, go to Assets > Certificates and filter by Source Discover_WinCerts and the target host to confirm metadata is populated correctly.

Common Errors

Access Denied Opening the MY Store (Keyset does not exist)

Cause: The account lacks read on a private-key ACL, or Log on as a service was not granted.

Resolution: Add the account with Read via Manage Private Keys on each MY cert, confirm Log on as a service, and restart the service.

Report Submission - HTTP 401 Unauthorized

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

Resolution: Generate/confirm the token under Settings > API Tokens, update the config, and restart the service.

Service Fails to Start - Error 1069 Logon Failure

Cause: The service account password is wrong, changed, or expired since registration.

Resolution: Verify the password, update it on the service Log On tab (or sc config ... password=), and start the service.

Security Recommendations

- Least privilege: grant only Read on store and private-key ACLs; never add the account to Administrators or Domain Admins.
- Protect sensor-config.yaml (it holds the API token) - restrict to Administrators and the service account, never world-readable.
- Rotate the API token every 90 days and restart the service after each rotation.
- Audit service-account logons (Event IDs 4624/4625) for the account.
- On Manage Private Keys, assign Read only - never Full Control - so the account cannot export or delete keys.

Conclusion

With a least-privilege service account, scoped store and private-key ACLs, and deployment as a managed service or scheduled task, Discover_WinCerts feeds certificate metadata - algorithms, key lengths, validity, subject/issuer, and chains - from every Windows Local Machine store into CBOM Secure for centralized cryptographic asset management and compliance reporting.