

How-to Enable (D)TLS Certificate Revocation Checking?

By default SNMP4J does not perform certification revocation checking when establishing or accepting DTLS or TLS connections.

To enable revocation checking, there are basically three options:

1. Provide a CRL file for the cert path checking
2. Use OCSP revocation checking
3. Other cert patch checking

CRL

CRL File (OpenSSL Format)

```
// Enable revocation checking in Java PKI (cannot be changed again in runtime):
System.setProperty("com.sun.net.ssl.checkRevocation", "true");

TLSTM tlstmCommandSender = new TLSTM();
tlstmCommandSender.setServerEnabled(false);
// Activate CRL checking for command sender:
tlstmCommandSender.setX09CertificateRevocationListURI(getClass().getResource("tls/crl_servers.pem").toURI().
toString());

TLSTM tlstmCommandResponder = new TLSTM(new TlsAddress("127.0.0.1/0"));
// Activate CRL checking for command responder:
tlstmCommandResponder.setX09CertificateRevocationListURI(getClass().getResource("tls/crl_clients.pem").toURI().
toString());
```

OCSP

OCSP Revocation Checking

```
System.setProperty("com.sun.net.ssl.checkRevocation", "true");
Security.setProperty("ocsp.enable", "true");
```

Cert Path Validation

Custom Cert Path Checking

```
/**
 * Creates a default revocation checker with CRL check only (no OCSP) and check is limited to end entity
 * only.
 * @return
 *   a simple revocation checker to be used with {@link #setPKIXRevocationChecker(PKIXRevocationChecker)}.
 */
public PKIXRevocationChecker createDefaultPKIXRevocationChecker() {
    CertPathBuilder cpb;
    try {
        cpb = CertPathBuilder.getInstance(TrustManagerFactory.getDefaultAlgorithm());
    } catch (NoSuchAlgorithmException e) {
        throw new RuntimeException(e);
    }
    PKIXRevocationChecker revocationChecker = (PKIXRevocationChecker)cpb.getRevocationChecker();
    // Relaxed checking - avoid OCSP because of 33% overhead on TLS connection creation:
    revocationChecker.setOptions(EnumSet.of(
        PKIXRevocationChecker.Option.PREFER_CRLS, // prefer CLR over OCSP
        PKIXRevocationChecker.Option.ONLY_END_ENTITY,
        PKIXRevocationChecker.Option.NO_FALLBACK)); // do not fall back to OCSP
    return revocationChecker;
}

...

TLSTM tlstm = new TLSTM();
tlstm.setServerEnabled(false);
tlstm.setPKIXRevocationChecker(createDefaultPKIXRevocationChecker());
...
```