Parsing Large CRLs in Java

Karl.Scheibelhofer@iaik.tugraz.at
IAIK, Graz University of Technology
June 2006

Introduction

This article shows how to use IAIK-JCE to parse large X.509 CRLs, which can grow up to several megabytes.

Certificate validation usually requires revocation checking. Even though OCSP is the revocation checking mechanism of the future, the most widespread one is still based on certificate revocation lists, short CRLs. On the first look, these lists are simple. They are a signed sequence of the serial numbers of all revoked certificates of a CRL. For small and medium CAs, the number of revoked certificates is limited and the size of CRLs are usually some kilobytes. For large CAs that issue millions of certificates, these sizes can grow up to several megabytes easily. For example, the CRL of the VeriSign Commercial Software Publishers CA has a size of 777 kB as of this writing and a CRL of the e-mail CAs of the Department of Defense has 11.6 MB.

The Approach with a `CertificiateFactory`

Large CRLs require advanced programming techniques. The `java.security.cert.CertificateFactory` provides a method for parsing CRLs from a stream, but the whole content of it is stored in memory. This entails a significant memory consumption of the application. The parsed CRL object usually requires a multiple of the memory of the underlying CRL. Typically, a Java VM with a maximum heap size of 64 MB runs out of memory, if the application tries to parse a CRL with a size of 4 MB. The parsed CRL object will consume up to 20 times more memory than the actual CRL. This number depends on the actual `CertificateFactory` implementation.

Here is an example, which uses a `CertificateFactory` to parse a CRL and to do the revocation checking.

```
CertificateFactory certFactory =
    CertificateFactory.getInstance("X.509");
URL crlURL = new URL(crlURLString);
InputStream crlStream = crlURL.openStream();
X509CRL crl = (X509CRL) certFactory.generateCRL(crlStream);
crl.verify(issuerCertificate.getPublicKey());
boolean revoked = crl.isRevoked(certificate);
```

You may try to run this code with SUN Java 5 and the default heap size of 64 MB. Parsing the CRL of VeriSign Commercial Software Publishers CA (http://crl.verisign.com/Class3SoftwarePublishers.crl) will usually succeed. At the time of this writing, it has a size of 777 kB and contains 22691 entries. On the heap, the parsed

CRL object takes about 25 MB. If we try to parse a CRL with 4 MB, like the CRL of the Department of Defense CA-4 with 188387 entries (ldap://ds-4.c3pki.den.disa.mil/cn%3dDOD%20CLASS%203%20CA-4%2cou%3dPKI%2cou%3dDoD%2co%3dU.S.%20Government%2cc%3dUS?certificaterevocationlist;binary), a Java VM with a 64 MB heap runs out of memory.

The Streaming CRL Classes of IAIK-JCE

To parse a CRL of this size, we can use the streaming CRL classes of the IAIK-JCE toolkit (version 3.12 or newer). These classes are in the `iaik.x509.stream` package. They support stream processing a CRL. This means that the application specifies in advance, in which certificates it is interested in, and the CRL object will browse through the data stream without buffering the CRL in memory. This reduces the memory consumption significantly and supports parsing CRLs of virtually any size.

Here is a piece of code that uses the mentioned CRL streaming classes.

```
X509Certificate[] consideredCertificates =
    new X509Certificate[] {certificate};
RevokedCertificatesCRLListener listener =
    new RevokedCertificatesCRLListener(consideredCertificates,
        issuerCertificate.getPublicKey());
X509CRLStream crlStreamHandler = new X509CRLStream(listener);
URL crlURL = new URL(crlURLString);
InputStream crlStream = crlURL.openStream();
crlStreamHandler.parse(crlStream);
Hashtable revocationEntriesTable = listener.getRevokedCertificates();
RevokedCertificate revocationEntry = (RevokedCertificate)
    revocationEntriesTable.get(certificate);
boolean revoked = (revocationEntry != null);
```

You can try to run this code with huge CRLs like the one of the Department of Defense E-Mail CA-6 (ldap://email-ds-4.c3pki.den.disa.mil/cn%3dDOD%20CLASS%203%20EMAIL%20CA-6%2cou%3dPKI%2cou%3dDoD%2co%3dU.S.%20Government%2cc%3dUS?certificaterevocationlist;binary), which has 11.6 MB and 551434 entries. The peak of memory consumption during parsing will not depend on the size of the CRL. It is typically a few kilobytes. This streaming class processes the entries sequentially. Before it proceeds to the next entry, it releases the memory of the previous entry. The processing time will only increase linearly with the size of the CRL.

Summary

This article shows how to use the `iaik.x509.stream` package of the IAIK-JCE toolkit to parse even very large CRLs. A typical `CertificateFactory` is unable to handle

CRLs of such sizes. Moreover, the memory consumption and performance of these streaming classes are much better.

References

1. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile,
   http://www.ietf.org/rfc/rfc3280.txt
2. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP
   http://www.ietf.org/rfc/rfc2560.txt
3. IAIK-JCE Toolkit
   http://jce.iaik.tugraz.at/products/core_crypto_toolkits/jca_jce