Additional Methods for Generating Key Identifiers
draft-turner-additional-methods-4kis-01.txt

Abstract

   This document specifies additional methods for generating key
   identifiers from a public key. This document also specifies an
   extension to identify the algorithms used to generate the key
   identifiers.

Status of this Memo

Copyright Notice

1. Introduction

   [RFC5280] defines the AKI (Authority Key Identifier) and SKI (Subject
   Key Identifier) certificate extensions. These extensions allow one
   certificate to refer to another certificate via the matching of these
   corresponding values. The principal use of this mechanism is to
   enable a relying party to disambiguate between two CA (Certification
   Authority) certificates with the same Subject name, located in the
   same directory entry. These identifiers are used during
   certification path construction in support of heuristics to reduce
   relying party workload. These identifiers are not used during
   certificate path validation. These key identifiers are used by PKI-
   enabled security protocols, such as CMP (Certificate Management
   Protocol) [RFC4210] and CMS (Cryptographic Message Syntax) [RFC5652],
   to identify the certificate used to protect a message, a session,
   etc.

   [RFC5280] describes two mechanisms for generating AKI/SKI values: a
   160-bit SHA-1 (Secure Hash Algorithm) hash of the public key and a
   four-bit type field with the value 0100 followed by the least
   significant 60 bits of the SHA-1 hash. Both of these mechanisms were
   designed to be non-security critical. That is, the use a hash
   algorithm was intended to provide a high probability (but not a
   guarantee) of uniqueness. [RFC5280] allows for additional
   mechanisms. (This is consistent with the fact that the SKI and AKI
   extensions are always marked non-critical.)

   This document defines four additional mechanisms for generating key
   identifier values, using SHA-224, SHA-256, SHA-384, and SHA-512
   [SHS]. Sample code for SHA-224, SHA-256, SHA-384, and SHA-512 can be
   found in [RFC6234]. The motivation for defining these additional
   means of generating AKI/SKI values is to accommodate use of
   additional, standard one-way hash functions that are becoming more
   widely used in PKI contexts.

   The additional key identifier generation mechanisms described in this
   document maintain the 160-bit value size, to avoid adversely
   affecting relying party code. With these additional mechanisms, CAs
   can omit code for algorithms that are otherwise unwanted or unused.
   For example, a CA that issues certificates hashed with SHA-256 and
   signed with ECDSA on the P-256 curve [RFC5480] might no longer need
   to implement SHA-1 as part of their CA application.

1.1. Requirements Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in [RFC2119].

1.2. ASN.1

The extension is defined using ASN.1 [X.680], [X.681], [X.682], and
[X.683].

2. Additional Methods for Generating Key Identifiers

As specified in [RFC5280], both authority and subject key identifiers
SHOULD be derived from the public key. Four additional mechanisms
CAs can use to identify public keys are as follows:

1) The keyIdentifier is composed of the least significant 160-bits
   of the SHA-224 hash of the value of the BIT STRING
   subjectPublicKey (excluding the tag, length, and number of unused
   bits).

2) The keyIdentifier is composed of the least significant 160-bits
   of the SHA-256 hash of the value of the BIT STRING
   subjectPublicKey (excluding the tag, length, and number of unused
   bits).

3) The keyIdentifier is composed of the least significant 160-bits
   of the SHA-384 hash of the value of the BIT STRING
   subjectPublicKey (excluding the tag, length, and number of unused
   bits).

4) The keyIdentifier is composed of the least significant 160-bits
   of the SHA-512 hash of the value of the BIT STRING
   subjectPublicKey (excluding the tag, length, and number of unused
   bits).

3. **Key Identifier Algorithm and Length Extension**

   The key identifier algorithm and length extension indicates the hash
   algorithm used to compute the key identifier and the length of the
   resulting key identifier. This extension MAY, at the option of the
   certificate issuer, be either critical or non-critical. This
   extension is identified by id-pe-skiAlgAndLen.

```
   ext-kiAlgAndLen EXTENSION ::= {
     SYNTAX KIAlgAndLens
     IDENTIFIED BY id-pe-kiAlgAndLen }

   id-pe-kiAlgAndLen OBJECT IDENTIFIER ::= { id-pe TBD }

   KIAlgAndLens ::= SEQUNENCE {
     aki  [0] KIAlgAndLen OPTIONAL,
     ski  [1] KIAlgAndLen OPTIONAL }

   KIAlgAndLen ::= SEQUNENCE {
     kiAlgorithm   AlgorithmIdentifier
                     { DIGEST-ALGORITHM, {DigestAlgorithmSet} },
     kiLength      INTEGER OPTIONAL }
```

   If this extension is included in a certificate, then either the aki,
   ski, or both MUST be included. KIAlgAndLen has two fields:

   o kiAlgorithm indicates the algorithm used to generate the key
     identifier. For example, if the CA wanted to indicate that one
     of the algorithms listed in Section 2 was used, then it would
     include OIDs (Object Identifiers) from [RFC5758].

   o kiLength indicates the length of the key identifier. It MUST be
     included if the key identifier is truncated (i.e., the length is
     shorter than the output of the hash algorithm); otherwise, it is
     OPTIONAL. For example, the output lengths of the hash algorithms
     defined in Section 2 are for 224, 256, 384, and 512-bits but all
     would indicate 160 to truncated to 160-bits for SHA-224, SHA-256,
     SHA-384, and SHA-512 bits.

4. **Security Considerations**

      The security considerations of [RFC5280] apply to certificates.
      The security considerations of [RFC5758] apply to the hash
      algorithms. The security considerations of [RFC5912] apply to
      the ASN.1.

      While hash algorithms provide collision resistance, this property
      is not needed for key identifiers.

5. IANA Considerations

   None.

6. References

6.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
             Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S.,
             Housley, R., and W. Polk, "Internet X.509 Public Key
             Infrastructure Certificate and Certificate Revocation List
             (CRL) Profile", RFC 5280, May 2008.

   [RFC5758] Dang, Q., Santesson, S., Moriarty, K., Brown, D., and T.
             Polk, "Internet X.509 Public Key Infrastructure: Additional
             Algorithms and Identifiers for DSA and ECDSA", RFC 5758,
             January 2010.

   [RFC5912] Hoffman, P. and J. Schaad, "New ASN.1 Modules for the
             Public Key Infrastructure Using X.509 (PKIX)", RFC 5912,
             June 2010.

   [SHS]   National Institute of Standards and Technology (NIST), FIPS
             Publication 180-3: Secure Hash Standard, October 2008.

   [X.680]   ITU-T Recommendation X.680 (2002) | ISO/IEC 8824-1:2002,
             Information technology - Abstract Syntax Notation One
             (ASN.1): Specification of basic notation.

   [X.681]   ITU-T Recommendation X.681 (2002) | ISO/IEC 8824-2:2002,
             Information Technology - Abstract Syntax Notation One:
             Information Object Specification.

   [X.682]   ITU-T Recommendation X.682 (2002) | ISO/IEC 8824-3:2002,
             Information Technology - Abstract Syntax Notation One:
             Constraint Specification.

   [X.683]   ITU-T Recommendation X.683 (2002) | ISO/IEC 8824-4:2002,
             Information Technology - Abstract Syntax Notation One:
             Parameterization of ASN.1 Specifications.

6.2. Informative References

   [RFC4210] Adams, C., Farrell, S., Kause, T., and T. Mononen,
             "Internet X.509 Public Key Infrastructure Certificate
             Management Protocol (CMP)", RFC 4210, September 2005.

   [RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70,
             RFC 5652, September 2009.

   [RFC5480] Turner, S., Brown, D., Yiu, K., Housley, R., and T. Polk,
             "Elliptic Curve Cryptography Subject Public Key
             Information", RFC 5480, March 2009.

   [RFC6234] Eastlake 3rd, D. and T. Hansen, "US Secure Hash Algorithms
             (SHA and SHA-based HMAC and HKDF)", RFC 6234, May 2011.

Appendix A   ASN.1 Module

```
KIAlgAndLen-2012
   { iso(1) identified-organization(3) dod(6) internet(1)
     security(5) mechanisms(5) pkix(7) id-mod(0)
     id-mod-skiAlgAndLen(TBD) }

DEFINITIONS EXPLICIT TAGS ::=
BEGIN
IMPORTS

-- Imports are all from [RFC5912]

id-pe
  FROM PKIX1Explicit-2009
    { iso(1) identified-organization(3) dod(6) internet(1)
      security(5) mechanisms(5) pkix(7) id-mod(0)
      id-mod-pkix1-explicit-02(51) }

AlgorithmIdentifier{}, DIGEST-ALGORITHM
  FROM AlgorithmInformation-2009
    { iso(1) identified-organization(3) dod(6) internet(1)
      security(5) mechanisms(5) pkix(7) id-mod(0)
      id-mod-algorithmInformation-02(58) }

mda-sha224, mda-sha256, mda-sha384, mda-sha512
  FROM PKIX1-PSS-OAEP-Algorithms-2009
    { iso(1) identified-organization(3) dod(6) internet(1)
      security(5) mechanisms(5) pkix(7) id-mod(0)
      id-mod-pkix1-rsa-pkalgs-02(54) } ;

ext-skiAlgAndLen EXTENSION ::= {
  SYNTAX SKIAlgAndLen
  IDENTIFIED BY id-pe-skiAlgAndLen }

id-pe-skiAlgAndLen OBJECT IDENTIFIER ::= { id-pe TBD }

KIAlgAndLens ::= SEQUNENCE {
  aki  [0] KIAlgAndLen OPTIONAL,
  ski  [1] KIAlgAndLen OPTIONAL }

KIAlgAndLen ::= SEQUNENCE {
  kiAlgorithm   AlgorithmIdentifier
                  { DIGEST-ALGORITHM, { KIHashAlgs } },
  kiLength      INTEGER OPTIONAL }

KIHashAlgs DIGEST-ALGORITHM ::= {
  mda-224, mda-256, mda-384, mda-512, ... }
```

          END

Authors' Addresses

     Sean Turner
     IECA, Inc.
     3057 Nutley Street, Suite 106
     Fairfax, VA 22031
     USA

     EMail: turners@ieca.com

     Stephen Kent
     BBN Technologies
     10 Moulton St.
     Cambridge, MA 02138

     EMail: kent@bbn.com