Independent Submission
Request for Comments: 7093
Category: Informational
ISSN: 2070-1721

S. Turner
IECA
S. Kent
BBN
J. Manger
Telstra

December 2013

Additional Methods for Generating Key Identifiers Values

Abstract

This document specifies additional example methods for generating Key Identifier values for use in the AKI (Authority Key Identifier) and SKI (Subject Key Identifier) certificate extensions.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This is a contribution to the RFC Series, independently of any other RFC stream. The RFC Editor has chosen to publish this document at its discretion and makes no statement about its value for implementation or deployment. Documents approved for publication by the RFC Editor are not a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7093.

1. Introduction

[RFC5280] defines the AKI (Authority Key Identifier) and SKI (Subject Key Identifier) certificate extensions. [RFC5280] describes two example mechanisms for generating AKI and 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 not be critical to security. This document defines three additional mechanisms for generating Key Identifier values using SHA-256, SHA-384, and SHA-512 [SHS] that are similar to those examples defined in [RFC5280] as well as one based on hashing the certificate's Subject Public Key Info field.

2. Additional Methods for Generating Key Identifiers

[RFC5280] specifies two examples for generating key identifiers from public keys. Four additional mechanisms are as follows:

1) The keyIdentifier is composed of the leftmost 160-bits of the SHA-256 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 leftmost 160-bits of the SHA-384 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 leftmost 160-bits of the SHA-512 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 hash of the DER encoding of the SubjectPublicKeyInfo value.

3. Examples

This section provides some examples. The keys and SKIs are presented in hexadecimal (two hex digits per byte).

Given the following DER-encoded SubjectPublicKeyInfo value holding an P-256 ECDSA (Elliptic Curve Digital Signature Algorithm) key:

```
  30 59
     30 13
        06 07 2A8648CE3D0201    -- id-ecPublicKey
        06 08 2A8648CE3D030107  -- secp256r1
     03 42 00
           04 7F7F35A79794C950060B8029FC8F363A
              28F11159692D9D34E6AC948190434735
              F833B1A66652DC514337AFF7F5C9C75D
              670C019D95A5D639B72744C64A9128BB
```

The SHA-256 hash of the 65 bytes 047F7F...BB is:
BF37B3E5808FD46D54B28E846311BCCE1CAD2E1A62AA9092EF3EFB3F11451F44

The SHA-1 hash of these 65 bytes is:
6FEF9162C0A3F2E7608956D41C37DA0C8E87F0AE

The SHA-256 hash of the 91 bytes 305930...BB is:
6D20896AB8BD833B6B66554BD59B20225D8A75A296088148399D7BF763D57405

Using method 1 from Section 2, the subjectKeyIdentifier would be:

```
  30 1D
     06 03 551D0E -- id-ce-subjectKeyIdentifier
     04 16
        04 14 BF37B3E5808FD46D54B28E846311BCCE1CAD2E1A
```

Using method 4 from Section 2 with SHA-256 and no truncation, the subjectKeyIdentifier extensions would be:

```
  30 29
     06 03 551D0E -- id-ce-subjectKeyIdentifier
     04 22
        04 20 6D20896AB8BD833B6B66554BD59B2022
              5D8A75A296088148399D7BF763D57405
```

4. Security Considerations

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

While hash algorithms provide preimage resistance, second-preimage resistance, and collision resistance, none of these properties are needed for key identifiers.

5. Acknowledgements

The authors wish to thank Santosh Chokhani, Stephen Farrell, Tom Gindin, Peter Gutmann, Henry Holtz, David Kemp, Timothy Miller, Michael StJohns, Stefan Santesson, Jim Schaad, Rene Struik, Koichi Sugimoto, and Carl Wallace for taking the time to participate in the discussions about this document. The discussions resulted in numerous editorial and technical changes to the document.

6. Normative References

[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.

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

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
USA

EMail: kent@bbn.com

James Manger
Telstra
6 / 150 Lonsdale Street
Melbourne, Victoria 3000
Australia

EMail: james.h.manger@team.telstra.com