Internet Engineering Task Force	P. Hallam-Baker
Internet-Draft	Comodo Group Inc.
Intended status: Standards Track	R. Stradling
Expires: March 30, 2012	Comodo CA Ltd.
	September 27, 2011

TOC

The DIGEST URI Scheme
draft-hallambaker-digesturi-00

Abstract

A URI scheme for referencing static data abjects by means of a cryptographic digest mechanism is specified. The format is designed to resist content type substitution attacks and supports a choice of digest algorithms.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as “work in progress.”

This Internet-Draft will expire on March 30, 2012.

Copyright Notice

Table of Contents

1. Definitions

TOC

1.1. Requirements Language

TOC

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 **RFC 2119** [RFC2119].

1.2. Defined Terms

TOC

The following terms are used in this document:

Abstract Syntax Notation One (ASN.1)
: A notation for describing abstract types and values, as specified in **X.680** [X.680].

2. The DIGEST URI Type.

TOC

Provides a strong reference to a static data object.

Does not provide a means of resolution.

Allows an authenticated data source to provide an authenticated reference to a static data object.

Intended applications include creating references from

Web pages delivered over HTTP/TLS

DNS resource records signed using DNSSEC

Data values embedded in certificates, CRLs, OCSP tokens and other signed data objects.

2.1. The DIGEST URI TYPE

TOC

The DIGEST URI Type has the following format:

DIGEST:< Base64 (Object Digest Value Specifier) >

2.2. Use in binary formats.

TOC

The URI encoding of the Object Digest Value Specifier is compatible with ASCII encoding formats and MAY be used in any situation where a URI is specified.

In a binary format such as an ASN.1 signed object, a direct encoding of the data without the BASE64 encoding MAY be more convenient.

TOC

3. Object Digest Value Specifier

An Object Digest is an ASN.1 structure with three components:

An ASN.1 Object Identifier specifying the object type of the referenced object

An ASN.1 Object Identifier specifying the digest algorithm.

Either:

An ASCII MIME Content type specifier.

An ASN.1 **DER** [X.690] encoded data field containing the digest value of the referenced object processed using the specified digest algorithm.

The ASN.1 structure is defined by the following schema:

```
DIGESTURI DEFINITIONS ::=

BEGIN

 ObjectDigestIdentifier ::= SEQUENCE {
   CHOICE {
     OIDtype        OBJECT IDENTIFIER,
     MIMEtype       IA5String
     }
   digestAlgorithm  OBJECT IDENTIFIER,
   digest           OCTET STRING
 }

END
```

The Object Digest Identifier construction is designed to facilitate implementation in applications that already require ASN.1 handling mechanisms (i.e. most cryptographic applications) without causing an undue coding burden in cases where ASN.1 code is not already supported. Appendix C provides all the necessary information to create a fully compliant Object Digest Identifier implementation.

3.1. Example: CA Certificate A

The ODI of CA Certificate A (specified in Appendix B.1) is calculated as follows:

ASN.1 Sequence tag: `3032`

ASN.1 OID id-at-cACertificate (2.5.4.37): `0603550425`

ASN.1 OID sha256 (2.16.840.1.101.3.4.2.1): `0609608648016503040201`

SHA-256 Digest Value:
`042017cc980f6a84fb15e5da3f32afea62360f4ca29627feed68739a13062defe804`

The DIGEST URI is
DIGEST:MDIGA1UEJQYJYIZIAWUDBAIBBCAXzJgPaoT7FeXaPzKv6mI2D0yilif+7WhzmhMGLe/oBA==.

3.2. Example: Text File

The Digest URI of the text file "Hello World" is computed as follows:

ASN.1 Sequence tag `3039`

ASN.1 IA5String 'text/plain' `160A746578742f706c61696e`

ASN.1 OID 'SHA-256' `0609608648016503040201`

SHA-256 Digest Value

a591a6d40bf420404a011733cfb7b190d62c65bf0bcda32b57b277d9ad9f146e

The DIGEST URI is
DIGEST:MDkWCnRleHQvcGxhaW4GCWCGSAFlAwQCAaWRptQL9CBASgEXM8+3sZDWLGW/C82jK1eyd9mtnxRu.

4. Security Considerations

TOC

4.1. Integrity

TOC

No secret information is required to generate a DIGEST URI. Therefore a DIGEST URI only provides a proof of integrity for the referenced object and the proof of integrity provided is only as good as the proof of integrity for the DIGEST URI value.

4.2. Confidentiality

TOC

Disclosure of a DIGEST URI value does not necessarily entail disclosure of the referenced object but may enable an attacker to determine the contents of the referenced object by reference to a search engine or other data repository.

4.3. Weak Digest Algorithm

TOC

[The digest algorithm MUST be strong]

[For most use cases collision resistance is a requirement]

5. IANA Considerations

TOC

[Assign the DIGEST URI type.]

6. References

TOC

6.1. Normative References

TOC

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

[RFC4055] Schaad, J., Kaliski, B., and R. Housley, "Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile," RFC 4055, June 2005 (TXT).

[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 (TXT).

[X.509] International Telecommunication Union, "ITU-T Recommendation X.509 (11/2008): Information technology - Open systems interconnection - The Directory: Public-key and attribute certificate frameworks," ITU-T Recommendation X.509, November 2008.

[X.680] International Telecommunication Union, "ITU-T Recommendation X.680 (11/2008): Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation," ITU-T Recommendation X.680, November 2008.

[X.690] International Telecommunication Union, "ITU-T Recommendation X.690 (11/2008): Information technology - Abstract Syntax Notation One (ASN.1): Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)," ITU-T Recommendation X.690, November 2008.

6.2. Non Normative References

TOC

[NIST

[NIST-ALGS] National Institute of Standards and Technology, "Cryptographic Algorithm Registration," March 2009.

[RFC3642] Legg, S., "Common Elements of Generic String Encoding Rules (GSER) Encodings," RFC 3642, October 2003 (TXT).

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings," RFC 4648, October 2006 (TXT).

Appendix A. Example Certificates

TOC

The following certificates are used in the examples.

A.1. CA Certificate A

TOC

CA Certificate A is a self signed certificate signed with a 2048 bit RSA key:

```
-----BEGIN CERTIFICATE-----
MIIDATCCAeugAwIBAgIBATALBgkqhkiG9w0BAQUwKDERMA8GA1UEChMIQWNtZSBJ
bmMxEzARBgNVBAMTCkV4YW1wbGUgQ0EwHhcNMTAxMTExMTgxMjAzWhcNMjAxMTA4
MTgxMjAzWjAoMREwDwYDVQQKEwhBY21lIEluYzETMBEGA1UEAxMKRXhhbXBsZSBD
QTCCAR8wCwYJKoZIhvcNAQEBA4IBDgAwggEJAoIBALHvos3yEe0ugR6Ae2rPATXA
pBYGK6BMzGTLkXCg6MZaG9CZpfleZTZ/EgIKBwRJlIXvWdKwjMZ7GBByT+fdMDZp
7zkx64UZ4+CJm98NRjdugxovl8HhscIBXnhCHERgamp0U/f8Ho5W8eAxYLZ1XcIG
mB7mVknvolaN9EqlEmYn+qHexGJPlpWFmR4NKhVAATE6B1a9z5PCmoOgW9p0Vqic
SJ6CdAHKaa7JZS+sqNQDx57H8Q6R9lh52XXmJVVficxBp2K7C+Wvht45t68FG6f1
sXWuWDRYc6iUmOxZbzDDvIoFU0pAXESTdMOWvXKI8ZUaYBoZ7/YnSSTaseiW86sC
AwEAAaM9MDswDgYDVR0PAQEBBAQDAgAEMA8GA1UdEwEBAQQFMAMBAQEwGAYDVR0g
BBEwDzANBgsrBgEEAYKUTYUaATALBgkqhkiG9w0BAQUDggEBAGcNiaQXdyiI9Y5e
Ps+XEYdKiWYvmSnRIfbUZuQWaQpPcj5cHzMe91CUZipGDNJYXwqWhIUtQAAGmtrq
ZGa4F9Yh0cPFAHBXPHXKGeM1hMtAR7Mv9kHu4DFIhb822O0n4DdBIit8FNas5t/5
CbM6crDpWB5hjAsD37U+GZGvTJmag059VWjnjv90NcfCQ6YJ6AA5VKnmrV695VnL
dSPaN9VS5RN6heJqU9tcbqPkAEP3MuJtd1QxB8Q34f9e1kTYXxc/dBJK1RQ0F4nc
Jc4NbJzakvFq+QcbzEqkhDMiXvjDV0JJt+GkFZrsREi6IgQY4DQHPv65OIvbr3uW
329dd+g=
-----END CERTIFICATE-----
```

In binary form, the certificate data is:

```
0000  30 82 03 01 30 82 01 eb  a0 03 02 01 02 02 01 01
0010  30 0b 06 09 2a 86 48 86  f7 0d 01 01 05 30 28 31
0020  11 30 0f 06 03 55 04 0a  13 08 41 63 6d 65 20 49
0030  6e 63 31 13 30 11 06 03  55 04 03 13 0a 45 78 61
0040  6d 70 6c 65 20 43 41 30  1e 17 0d 31 30 31 31 31
0050  31 31 38 31 32 30 33 5a  17 0d 32 30 31 31 30 38
0060  31 38 31 32 30 33 5a 30  28 31 11 30 0f 06 03 55
0070  04 0a 13 08 41 63 6d 65  20 49 6e 63 31 13 30 11
0080  06 03 55 04 03 13 0a 45  78 61 6d 70 6c 65 20 43
0090  41 30 82 01 1f 30 0b 06  09 2a 86 48 86 f7 0d 01
00a0  01 01 03 82 01 0e 00 30  82 01 09 02 82 01 00 b1
00b0  ef a2 cd f2 11 ed 2e 81  1e 80 7b 6a cf 01 35 c0
00c0  a4 16 06 2b a0 4c cc 64  cb 91 70 a0 e8 c6 5a 1b
00d0  d0 99 a5 f9 5e 65 36 7f  12 02 0a 07 04 49 94 85
00e0  ef 59 d2 b0 8c c6 7b 18  10 72 4f e7 dd 30 36 69
00f0  ef 39 31 eb 85 19 e3 e0  89 9b df 0d 46 37 6e 83
0100  1a 2f 97 c1 e1 b1 c2 01  5e 78 42 1c 44 60 6a 6a
0110  74 53 f7 fc 1e 8e 56 f1  e0 31 60 b6 75 5d c2 06
0120  98 1e e6 56 49 ef a2 56  8d f4 4a a5 12 66 27 fa
0130  a1 de c4 62 4f 96 95 85  99 1e 0d 2a 15 40 01 31
0140  3a 07 56 bd cf 93 c2 9a  83 a0 5b da 74 56 a8 9c
0150  48 9e 82 74 01 ca 69 ae  c9 65 2f ac a8 d4 03 c7
0160  9e c7 f1 0e 91 f6 58 79  d9 75 e6 25 55 5f 89 cc
0170  41 a7 62 bb 0b e5 af 86  de 39 b7 af 05 1b a7 f5
0180  b1 75 ae 58 34 58 73 a8  94 98 ec 59 6f 30 c3 bc
0190  8a 05 53 4a 40 5c 44 93  74 c3 96 bd 72 88 f1 95
01a0  1a 60 1a 19 ef f6 27 49  24 da b1 e8 96 f3 ab 02
01b0  03 01 00 01 a3 3d 30 3b  30 0e 06 03 55 1d 0f 01
01c0  01 01 04 04 03 02 00 04  30 0f 06 03 55 1d 13 01
01d0  01 01 04 05 30 03 01 01  01 30 18 06 03 55 1d 20
```

```
01e0  04 11 30 0f 30 0d 06 0b  2b 06 01 04 01 82 94 4d
01f0  85 1a 01 30 0b 06 09 2a  86 48 86 f7 0d 01 01 05
0200  03 82 01 01 00 67 0d 89  a4 17 77 28 88 f5 8e 5e
0210  3e cf 97 11 87 4a 89 66  2f 99 29 d1 21 f6 d4 66
0220  e4 16 69 0a 4f 72 3e 5c  1f 33 1e f7 50 94 66 2a
0230  46 0c d2 58 5f 0a 96 84  85 2d 40 00 06 9a da ea
0240  64 66 b8 17 d6 21 d1 c3  c5 00 70 57 3c 75 ca 19
0250  e3 35 84 cb 40 47 b3 2f  f6 41 ee e0 31 48 85 bf
0260  36 d8 ed 27 e0 37 41 22  2b 7c 14 d6 ac e6 df f9
0270  09 b3 3a 72 b0 e9 58 1e  61 8c 0b 03 df b5 3e 19
0280  91 af 4c 99 9a 83 4e 7d  55 68 e7 8e ff 74 35 c7
0290  c2 43 a6 09 e8 00 39 54  a9 e6 ad 5e bd e5 59 cb
02a0  75 23 da 37 d5 52 e5 13  7a 85 e2 6a 53 db 5c 6e
02b0  a3 e4 00 43 f7 32 e2 6d  77 54 31 07 c4 37 e1 ff
02c0  5e d6 44 d8 5f 17 3f 74  12 4a d5 14 34 17 89 dc
02d0  25 ce 0d 6c 9c da 92 f1  6a f9 07 1b cc 4a a4 84
02e0  33 22 5e f8 c3 57 42 49  b7 e1 a4 15 9a ec 44 48
02f0  ba 22 04 18 e0 34 07 3e  fe b9 38 8b db af 7b 96
0300  df 6f 5d 77 e8
```

The SHA-256 digest of the certificate data is:

```
17cc980f6a84fb15e5da3f32afea62360f4ca29627feed68739a13062defe804
```

Appendix B. ASN.1 Values (Non-Normative)

TOC

Although the Object Digest Identifier form employs ASN.1 DER encoding only a small subset of ASN.1 features are used and a full ASN.1 stack is not necessary.

This appendix provides sufficient information to implement an Object Digest Identifier constructor or parser.

B.1. DER Sequence Encoding

TOC

In DER encoding, the enclosing SEQUENCE will always be represented by the type identifier x30 followed by the length specifier. Since the total length of the following data fields will almost certainly be less than 127 bytes, the single byte encoding mechanism in which bit 7 is clear and the length value is encoded in the lower 7 bits will be required.

B.2. Object Identifiers for Certificate Types

TOC

OIDs have been defined in connection with the X.500 directory for user certificates, certification authority certificates, revocations of certification authority, and revocations of user certificates. The following table lists the OIDs, their DER encoding, and their type identifier and length-prefixed hex format for use in Object Digest Identifiers.

```
id-at  OBJECT IDENTIFIER  ::=  { joint-iso-itu-t(2) ds(5) 4 }

id-at-userCertificate  OBJECT IDENTIFIER  ::=  { id-at 36 }
                                                 -- 06 03 55 04 24
  id-at-cACertificate  OBJECT IDENTIFIER  ::=  { id-at 37 }
                                                 -- 06 03 55 04 25
 TBS-PUBLIC-KEY-VALUE  OBJECT IDENTIFIER  ::=  { ??? }
                                                 -- 06 xx xx xx xx
```

B.3. Object Identifiers for Digest Algorithms

TOC

OIDs have been assigned by NIST for the SHA-2 digest algorithms **[NIST-ALGS] [RFC4055]**

Use of the SHA-1 digest algorithm is not recommended due to concerns for the security of the algorithm.

```
hashAlgs  OBJECT IDENTIFIER  ::=  { joint-iso-itu-t(2)
             country(16) us(840) organization(1) gov(101) csor(3)
             nistAlgorithm(4) 2 }

id-sha256  OBJECT IDENTIFIER  ::=  { hashAlgs 1 }
                                     -- 06 09 60 86 48 01 65 03 04 02 01
id-sha384  OBJECT IDENTIFIER  ::=  { hashAlgs 2 }
                                     -- 06 09 60 86 48 01 65 03 04 02 02
id-sha512  OBJECT IDENTIFIER  ::=  { hashAlgs 3 }
                                     -- 06 09 60 86 48 01 65 03 04 02 03
id-sha224  OBJECT IDENTIFIER  ::=  { hashAlgs 4 }
                                     -- 06 09 60 86 48 01 65 03 04 02 04
```

B.4. DER Data Encoding Prefixes

TOC

The rules of ASN.1 encoding state that every data value is preceded by a data type identifier and a length identifier. In the case of an Object Digest Identifier the data type identifier is always OCTET STRING (04) and the length for all currently defined digest algorithms will be less than 128 bytes (1024 bits) and thus use the single byte encoding form in which bit 7 is set to 0 and the lower 7 bits specify the length.

The length prefixes for commonly used digest lengths in hexadecimal notation are thus:

160 bits
    04 14
224 bits
    04 1C
256 bits
    04 20
384 bits
    04 30
512 bits
    04 40

Authors' Addresses

TOC

Phillip Hallam-Baker
Comodo Group Inc.
Email: philliph@comodo.com

Rob Stradling
Comodo CA Ltd.
Email: rob.stradling@comodo.com