Internet X.509 Public Key Infrastructure -- Certificate Image

Abstract

This document specifies a method to bind a visual representation of a certificate in the form of a certificate image to a public key certificate as defined in RFC 5280, by defining a new "otherLogos" image type according to RFC 3709.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in 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/rfc6170.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.
1. Introduction

This standard specifies how to bind a certificate image to a certificate (defined in [RFC5280]), providing a visual representation of that certificate using the Logotype extension defined in [RFC3709] and specifying the certificate image as a new "otherLogos" type.

The purpose of the certificate image is to aid human interpretation of a certificate by providing meaningful visual information to the user interface (UI).

Typical situations when a human needs to examine the visual representation of a certificate are:

- A person establishes a secured channel with an authenticated service. The person needs to determine the identity of the service based on the authenticated credentials.

- A person validates the signature on critical information, such as signed executable code, and needs to determine the identity of the signer based on the signer's certificate.

- A person is required to select an appropriate certificate to be used when authenticating to a service or Identity Management infrastructure. The person needs to see the available certificates in order to distinguish between them in the selection process.
The display of certificate information to humans is challenging due to lack of well-defined semantics for critical identity attributes. Unless the application has out-of-band knowledge about a particular certificate, the application will not know the exact nature of the data stored in common identification attributes such as serialNumber, organizationName, country, etc. Consequently, the application can display the actual data, but faces the problem of labeling that data in the UI and informing the human about the exact nature (semantics) of that data. It is also challenging for the application to determine which identification attributes are important to display and how to organize them in a logical order.

**RFC 3709** [RFC3709] defines a certificate extension for binding images to a certificate, such as a community logo and issuer logo, enhancing the display of certificate information. The syntax is extensible and allows inclusion of new image types using the otherLogos structure. This standard defines how to include a complete certificate image using the extensibility mechanism of **RFC 3709**.

1.1. 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].

2. Certificate Image

This section defines the certificate image as a new otherLogos type according to Section 4.1 of [RFC3709].

The certificate image otherLogos type is identified by the Object Identifier (OID) id-logo-certimage.

```
  id-pkix OBJECT IDENTIFIER ::= {
    iso(1) identified-organization(3) dod(6) internet(1)
    security(5) mechanisms(5) pkix(7) }

  id-logo OBJECT IDENTIFIER ::= { id-pkix 20 }

  id-logo-certimage OBJECT IDENTIFIER ::= { id-logo 3 }
```

When present, the certificate image MUST be a complete visual representation of the certificate. This means that the display of this certificate image represents all information about the certificate that the issuer subjectively defines as relevant to show to a typical human user within the typical intended use of the certificate, giving adequate information about at least the following three aspects of the certificate:
Certificate Context information is visual marks and/or textual information that helps the typical user to understand the typical usage and/or purpose of the certificate.

It is up to the issuer to decide what information -- in the form of text, graphical symbols, and elements -- represents a complete visual representation of the certificate. However, the visual representation of Certificate Subject and Certificate Issuer information from the certificate MUST have the same meaning as the textual representation of that information in the certificate itself.

Applications providing a Graphical User Interface (GUI) to the certificate user MAY present a certificate image according to this standard in any given application interface, as the only visual representation of a certificate.

3. LogotypeImageInfo

The optional LogotypeImageInfo structure is defined in [RFC3709] and is included here for convenience:

```
LogotypeImageInfo ::= SEQUENCE {
  type          [0] LogotypeImageType DEFAULT color,
  fileSize      INTEGER,  -- In octets
  xSize         INTEGER,  -- Horizontal size in pixels
  ySize         INTEGER,  -- Vertical size in pixels
  resolution    LogotypeImageResolution OPTIONAL,
  language      [4] IA5String OPTIONAL }  -- RFC 3066 Language Tag
```

NOTE: The referenced RFC 3066 in the structure above (from RFC 3709) is obsolete and is currently replaced by RFC 5646 [RFC5646]. The language tag may carry information about the language used to express any textual elements within the image as well as any audio information associated with the image.

When the optional LogotypeImageInfo is included with a certificate image, the parameters shall be used with the following semantics and restrictions.

xSize and ySize represent the recommended display size for the image. When a value of 0 (zero) is present, no recommended display size is specified. When non-zero values are present and these values differ
from corresponding size values in the referenced image file, then the referenced image SHOULD be scaled to fit within the size parameters of LogotypeImageInfo, while keeping the x and y ratio intact.

The resolution parameter is redundant for all image formats that are relevant for certificate images and MUST NOT be specified.

4. Embedded Images

The certificate image otherLogos type defined in this specification and all logotype types defined in RFC 3709 [RFC3709] MAY be stored within the logotype extension using the "data" URL scheme defined in RFC 2397 [RFC2397] if the logotype image is provided through direct addressing, i.e., the image is referenced using the LogotypeDetails structure.

The syntax of Logotype details defined in RFC 3709 is included here for convenience:

```
LogotypeDetails ::= SEQUENCE {
  mediaType       IA5String, -- MIME media type name and optional
                  -- parameters (see Section 5)
  logotypeHash    SEQUENCE SIZE (1..MAX) OF HashAlgAndValue,
  logotypeURI     SEQUENCE SIZE (1..MAX) OF IA5String }
```

The syntax of the "data" URL scheme defined in RFC 2397 is included here for convenience:

```
dataurl := "data:" [ mediatype ] [ ";base64" ] "," data
mediatype := [ type "/" subtype ] *( ";" parameter )
data := *urlchar
parameter := attribute "=" value
```

When including the image data in the logotype extension using the "data" URL scheme, the following conventions apply.

- The value of mediaType in LogotypeDetails MUST be identical to the media type value in the "data" URL.

- The hash of the image MUST be included in logotypeHash and MUST be calculated over the same data as it would have been, had the image been referenced through a link to an external resource.

NOTE: As the "data" URL scheme is processed as a data source rather than as a URL, the image data is typically not limited by any URL length limit settings that otherwise apply to URLs in general.
NOTE: Implementations need to be cautious about the size of images included in a certificate in order to ensure that the size of the certificate does not prevent the certificate from being used as intended.

5. Certificate Image Formats

Implementations of this specification MUST support JPEG and GIF as defined in RFC 3709 [RFC3709]. In addition to these mandatory-to-implement formats, this specification specifies the use of the Portable Document Format (PDF), Scalable Vector Graphics (SVG), and Portable Network Graphics (PNG) as image formats.

5.1. PDF

A certificate image MAY be provided in the form of a Portable Document Format (PDF) document according to [ISO32000] and following the conventions defined in this section. When a certificate image is formatted as a PDF document, it MUST also be formatted according to the profile PDF/A [ISO19005].

When including a PDF document as a certificate image, the following MIME media type as specified in [RFC3778] MUST be used as mediaType in LogotypeDetails:

application/pdf

5.2. SVG

A certificate image MAY be provided in the form of a Scalable Vector Graphics (SVG) image, which MUST follow the SVG Tiny profile [SVGT] with the following amendments:

- The SVG image MUST NOT contain any Internationalized Resource Identifier (IRI) references to information stored outside of the SVG image of type B, C, or D, according to Section 14.1.4 of SVG Tiny 1.2 [SVGT].

- The SVG image MUST NOT contain any 'script' element, according to Section 15.2 of SVG Tiny 1.2 [SVGT].

- The XML structure in the SVG file MUST use <LF> (linefeed 0x0A) as the end-of-line (EOL) character when calculating a hash over the SVG image.

The referenced SVG file MAY be provided in GZIP-compressed [RFC1952] form as an SVGZ file. In this case, the extension 'svgz' is used as an alias for 'svg.gz' [RFC1952], i.e., octet streams of type
image/svg+xml, subsequently compressed with gzip as specified in [SVGR]. The hash over the SVGZ file is calculated over the decompressed SVG content with canonicalized EOL characters (<LF>) as specified above.

The following MIME media type, defined in Appendix M of [SVGT], MUST be included as mediaType in LogotypeDetails for all SVG and SVGZ images:

   image/svg+xml

When the SVG image is embedded using the "data" URL scheme as defined in Section 4, SVG image data MUST be provided in SVGZ (GZIP compressed) form (i.e., it MUST NOT be provided in uncompressed SVG form).

Compliant implementations of this specification SHOULD be able to process SVG images that are formatted according to this section.

5.3. PNG

If a certificate image is provided as a bitmapped image, the PNG [ISO15948] format SHOULD be used.

PNG images are identified by the following mediaType in LogotypeDetails:

   image/png

6. Security Considerations

This document is based on and inherits all security considerations from RFC 3709 [RFC3709]. In particular, RFC 3709 discusses several issues a Certificate Authority (CA) should take into consideration when evaluating a request to issue a certificate with a certificate image.

Images incorporated according to RFC 3709 provide an additional possibility for a CA with bad intentions or bad security procedures to include false, conflicting, or malicious information to relying parties. Such a CA may, for example:

- include information in graphical form that is in conflict with information in provided text-based attributes or other name forms, and

- include malicious data that could exploit known security bugs in common software libraries used to render graphical images.
This underlines the necessity for CAs to provide reliable services, and the relying party’s responsibility and need to carefully select which CAs are trusted to provide public key certificates.

This also underlines the general necessity for relying parties to use up-to-date software libraries to render or dereference data from external sources (such as certificates), to minimize risks related to processing potentially malicious data before the data has been adequately verified and validated.

Referenced image files are hashed in order to bind the image to the signature of the certificate. Some image types, such as SVG, allow part of the image to be collected from an external source by incorporating a reference to an external image file. If this feature were used within a certificate image file, the hash of the image file would only cover the URI reference to the external image file, but not the referenced image data. Clients SHOULD verify that SVG images meet all requirements listed in Section 5.2 and reject images that contain references to external data.

CAs issuing certificates with embedded certificate images should be cautious when accepting graphics from the certificate requestor for inclusion in the certificate if the hash algorithm used to sign the certificate is vulnerable to collision attacks. In such a case, the accepted image may contain data that could help an attacker to obtain colliding certificates with identical certificate signatures.

Certificates, and hence their certificate images, are commonly public objects and as such usually will not contain privacy-sensitive information. However, when a certificate image that is referenced from a certificate contains privacy-sensitive information, appropriate security controls should be in place to protect the privacy of that information. Details of such controls are outside the scope of this document.

7. Acknowledgements

The authors recognize valuable contributions from members of the PKIX working group, the CA Browser Forum, and James Manger, for their review and sample data.
8. References

8.1. Normative References


8.2. Informative References


[SVGR] "Media Type Registration for image/svg+xml", http://dev.w3.org/SVG/profiles/1.1F2/master/mimereg.html.
Appendix A.  ASN.1 Module

CERT-IMAGE-MODULE { iso(1) identified-organization(3) dod(6)
    internet(1) security(5) mechanisms(5) pkix(7) id-mod(0)
    id-mod-logotype-certimage(68) }

DEFINITIONS EXPLICIT TAGS ::= BEGIN

EXPORTS ALL;  -- export all items from this module

    id-logo-certImage OBJECT IDENTIFIER ::= 
        { iso(1) identified-organization(3) dod(6) internet(1)
          security(5) mechanisms(5) pkix(7) id-log(20) 3 }

END
APPENDIX B

Example

The following example stores an embedded svgz-encoded SVG image using the "data" URL scheme.

data:image/svg+xml;base64,H4sICLXutU0AA0NlcnRJbWFnZURlbW8uc3ZnANVaW2/bOBZ+ii19BqBiwdoS7x9jmeapB0EWHQHsez2WZ0oRitZMi5qi5vvr95CSL7gi1E
m8C9di9ERSPOd85x8r1c5y0uy1TPi3h13ygfPmlrjJ7TeOv/661myCFBHvck
pnmmL50sd34b/TIa6yVois+da11yUs5Wjkqj2/k41lG6DbyNyUMST+s+eq1YDz1sul+
6u9xU9P9rU67QPRk51KqCk635Wwumu+a/UGYw9eEigh0qVrLHSwjcjjdG1GHQ
914on/sKL1M4fWvctkJr6X7/sGEydNA5L/WeRT5UDSmfLVWNoo2c9NCih1XyoKN8Ns
ux0jivW8Q1bOF0VQIP16oeme6awqPeIg37x9n9gCzyYVxWlUwpFLCwCgoLp8sy
n8wzRgkR4t6eFEnhMBaCn7DERk0pOBjM8mIVqPEYGTa+Bv0MsF2E5uF1W7gR7F7iyV
NFElrWnOlpiV3111VpGntMrW9n6KRdf0+9BrFrMebVeWhtzdbWbRV6ULobYPvB6WJX
Bk7six2xH6vnXco5xB7yUsVFpV6OGHQh6fSxg1mPkScflFvdmD4M10c1owov
81F2WzhN1VumgEFRIimDD3PhPGyTgUUMC61KgKAxjaptq1b0UvQFSvi+LOJYxZkP
E/v/CwHuAmxmoj1AarnRbatzkgbv7cK5s2oRFm/wvHosG1U2RqozXnDPKZw25tVj5z
IhFK00B6D6hARXSD6Fzqq7H7mQeJAOq1USPvFIRUHoFuuizFrFi5DvYeAmpcOocb9
u63vLjaek4YX4ryFifrTa2SI1miYyd+ceWEagADML2LH6SH4R9xyRyAp16g3YF02+F
zI1A+czSKh66gSIVa80fsQnMnWqZpmsXwAPyNoNaQ59uNIgasKPwhxGzQ2zMMIZIi
KBAkmLl11470zfSjJWnn+kvplvQ9l113yRic8gukz0uysEcakKdy3KMk+150XO1oX
1cL7LTeI1UzFP4tm70k8xk6cYS629WmwF1POte0p4ndYb7/j/kaWySN0ey0/1
qbiCxsDJKMvwo37QekBcAUFSPPGKCPnUJF5wawbacDK5G1Ep4BC21YoJcRNNGVC7dI
q7X4iKSp1rAG8mL8whRejQ9eEmmIAoz3s/s9Xp4RZEzugqzb7c3Q89u3WQKY9ae
qbsA/AUJB/bJs6fpyJ5BHFRuk5DlIT2OH5u25ThlUsd/qJOG6RmSiy1mHTAqLr6A6iB
AQMAhnhHNlsd61WtDUv3juhrkedQvB7L9Bv9hjQe4wLyBOWMOMiyQAXNB8sm
9Gx2qgdGgKMGGw68aL6f4dfmVc+pP/e/Z/VnZos8ClOqbz/gg/LdLxCIDC1D6DFP
2D98ctx8VqF57ECa7Ax62yaDF6C4S1yWMQcmztq8p+umChAm1A63qHqO3M3283s8
b/XO6LYsFzpGV5v1L+DmdWvnY+NaKor0/IJJIL3dBz72Aug+0+Nxm53XstShleogfo
v/J35BTQaQ/cmFUE2mp9p1OoP2cJW00K2o7ah0uH1BDL3G6m6i6aj4bqZo2iCmN
FxD1R/x2mC5s+HKh1AU3Ne3P81qg0P0qjffj9uu+HWWXSFwNoH4qUgTAmTadYMc0C7y
EEd9EuhkVEOEOCDSHQ=ynh5VUHR8miQ02FKSTCJW2ZGKB8i8pHud16wApncVtozjI
FAj97Qdxca+addOTiaza1xJvD0qMrxX+Ydaj6iwJQGOvaSDYWp4T4v4WVPRA3260NOj
unEiEkrVMhujpa2+wPmQ9RWFQAF9h9BBqGZfeXEX+WwOnxQ8pV35F2acdm0pGebeMC70
QgKae0DKEAf1k/9hUKEGBVwuce4aJnj/LBYU09hWYfX01H8Bc21byIuYF5808p+
addMwUt9YaoX/yIrTAC9NAdAyBuEB3V5r9u8/TTGYx/Xzj8bbP3/F9Bo9sgBk+4x
fitDgrw0GECQ6pdqV4PPRvE+26idSmGpm8RwRZqcT18EMPoor히+av4JeC7j0b
/vi/puzEmMKEA6F/7FVHHJ8hBhdrRQ3YmcgZFrVjwfmOnN3GxWU+FzhcPmpQiptHca
yatC28T7j3sCo/3/LQowo6j2Yp45R8SasobjFegusoJhVQNSv2RcpqASviq+gbk
BywETPNt+12apT6UerR1xzMwU1ZTVF51n5basOlfk2u4a8sb1kJi8qLu3cyDw7Bi
nxexMcSThQnJHFpfGkiK2N3mo7/7H5ev40icpSYS+6GkCM9al1Qv3EchCUA+5H
TcboQK6F14Yp+Yn7GwmzwpZCD5FD1XT97B6rdHApdb7QmLVqjgZIlWN61jz
Q7/c703hm9846u8Zn0ZMAmDlm71G2ieXfUtUL1UEs9NEUHibE9Nv9/2/2j8JGZqMk
3y7ykjJ0lIv1XjbsDPCpmg0e6XlHR3KSQVKqp+WiQammmu7Bjtqk5Mh4y/o5/sPf
LhDCXgq8uORL+DLPcKRxq6VxDv6pV1p+gg0R0ryBkoF91KZvrWGAKVrdz3cuucovqO/q
emC1PMYEFEH70i+vpkex4s4bCM9qK+3MhQUD1Dp/e/7py/2/6PW31kXPeyagP04ePv
1cE50Umy6IQZeQM7+01742+eHfpHNc70jffq/HeV0X8BopoDKeGkAAA=
Authors’ Addresses

Stefan Santesson
3xA Security (AAA-sec.com)
Bjornstorp 744
247 98 Genarp
Sweden
EMail: sts@aaa-sec.com

Russell Housley
Vigil Security, LLC
918 Spring Knoll Drive
Herndon, VA  20170
USA
EMail: housley@vigilsec.com

Siddharth Bajaj
Symantec Corp.
350 Ellis Street
Mountain View, CA  94043
USA
EMail: siddharthietf@gmail.com

Leonard Rosenthol
3533 Sunset Way
Huntingdon Valley, PA  19006
USA
EMail: leonardr@adobe.com