Workgroup: Network Working Group
Internet-Draft: draft-ietf-lamps-rfc3709bis-03
Obsoletes: 3709, 6170 (if approved)
Published: 23 June 2022
Intended Status: Standards Track
Expires: 25 December 2022

Authors: S. Santesson R. Housley
IDsec Solutions Vigil Security
T. Freeman L. Rosenthol
Amazon Web Services Adobe

Internet X.509 Public Key Infrastructure: Logotypes in X.509 Certificates

Abstract

This document specifies a certificate extension for including logotypes in public key certificates and attribute certificates. This document obsoletes RFC 3709 and RFC 6170.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This specification supplements [RFC5280], which profiles public-key certificates and certificate revocation lists (CRLs) for use in the Internet, and it supplements [RFC5755] which profiles attribute certificates for use in the Internet.

This document obsoletes RFC 3709 [RFC3709] and RFC 6170 [RFC6170]. Appendix C provides a summary of the changes since the publication of RFC 3709 and RFC 6170.

The basic function of a certificate is to bind a public key to the identity of an entity (the subject). From a strictly technical viewpoint, this goal could be achieved by signing the identity of the subject together with its public key. However, the art of Public Key Infrastructure (PKI) has developed certificates far beyond this functionality in order to meet the needs of modern global networks and heterogeneous information technology structures.

Certificate users must be able to determine certificate policies, appropriate key usage, assurance level, and name form constraints. Before a relying party can make an informed decision whether a particular certificate is trustworthy and relevant for its intended usage, a certificate may be examined from several different perspectives.

Systematic processing is necessary to determine whether a particular certificate meets the predefined prerequisites for an intended usage. Much of the information contained in certificates is appropriate and effective for machine processing; however, this information is not suitable for a corresponding human trust and recognition process.

Humans prefer to structure information into categories and symbols. Most humans associate complex structures of reality with easily recognizable logotypes and marks. Humans tend to trust things that they recognize from previous experiences. Humans may examine information to confirm their initial reaction. Very few consumers actually read all terms and conditions they agree to in accepting a service, rather they commonly act on trust derived from previous experience and recognition.

A big part of this process is branding. Service providers and product vendors invest a lot of money and resources into creating a strong relation between positive user experiences and easily recognizable trademarks, servicemarks, and logotypes.

Branding is also pervasive in identification instruments, including identification cards, passports, driver's licenses, credit cards, gasoline cards, and loyalty cards. Identification instruments are intended to identify the holder as a particular person or as a member of the community. The community may represent the subscribers of a service or any other group. Identification instruments, in physical form, commonly use logotypes and symbols, solely to enhance human recognition and trust in the identification instrument itself. They may also include a registered trademark to allow legal recourse for unauthorized duplication.

Since certificates play an equivalent role in electronic exchanges, we examine the inclusion of logotypes in certificates. We consider certificate-based identification and certificate selection.

1.1. Certificate-based Identification

The need for human recognition depends on the manner in which certificates are used and whether certificates need to be visible to human users. If certificates are to be used in open environments and in applications that bring the user in conscious contact with the result of a certificate-based identification process, then human recognition is highly relevant, and may be a necessity.

Examples of such applications include:

- Web server identification where a user identifies the owner of the web site.
- Peer e-mail exchange in B2B, B2C, and private communications.
- Exchange of medical records, and system for medical prescriptions.
- Unstructured e-business applications (i.e., non-EDI applications).
- Wireless client authenticating to a service provider.

Most applications provide the human user with an opportunity to view the results of a successful certificate-based identification process. When the user takes the steps necessary to view these results, the user is presented with a view of a certificate. This solution has two major problems. First, the function to view a certificate is often rather hard to find for a non-technical user. Second, the presentation of the certificate is too technical and is not user friendly. It contains no graphic symbols or logotypes to enhance human recognition.

Many investigations have shown that users of today's applications do not take the steps necessary to view certificates. This could be due to poor user interfaces. Further, many applications are structured to hide certificates from users. The application designers do not want to expose certificates to users at all.

1.2. Selection of Certificates

One situation where software applications must expose human users to certificates is when the user must select a single certificate from a portfolio of certificates. In some cases, the software application can use information within the certificates to filter the list for suitability; however, the user must be queried if more than one certificate is suitable. The human user must select one of them.

This situation is comparable to a person selecting a suitable plastic card from his wallet. In this situation, substantial assistance is provided by card color, location, and branding.

In order to provide similar support for certificate selection, the users need tools to easily recognize and distinguish certificates. Introduction of logotypes into certificates provides the necessary graphic.

1.3. Combination of Verification Techniques

The use of logotypes will, in many cases, affect the users decision to trust and use a certificate. It is therefore important that there be a distinct and clear architectural and functional distinction between the processes and objectives of the automated certificate verification and human recognition.

Since logotypes are only aimed for human interpretation and contain data that is inappropriate for computer based verification schemes, the logotype extension **MUST NOT** be an active component in automated certification path validation as specified in Section 6 of [RFC5280].

Automated certification path verification determines whether the end-entity certificate can be verified according to defined policy. The algorithm for this verification is specified in [RFC5280].

The automated processing provides assurance that the certificate is valid. It does not indicate whether the subject is entitled to any particular information, or whether the subject ought to be trusted to perform a particular service. These are authorization decisions. Automatic processing will make some authorization decisions, but others, depending on the application context, involve the human user.

In some situations, where automated procedures have failed to establish the suitability of the certificate to the task, the human user is the final arbitrator of the post certificate verification authorization decisions. In the end, the human will decide whether or not to accept an executable email attachment, to release personal information, or follow the instructions displayed by a web browser. This decision will often be based on recognition and previous experience.

The distinction between systematic processing and human processing is rather straightforward. They can be complementary. While the systematic process is focused on certification path construction and verification, the human acceptance process is focused on recognition and related previous experience.

There are some situations where systematic processing and human processing interfere with each other. These issues are discussed in the Section 9.

1.4. Terminology

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Different Types of Logotypes in Certificates

This specification defines the inclusion of three standard logotype types:

*Community logotype

*Issuer organization logotype

*Subject organization logotype

The community logotype is the general mark for a community. It identifies a service concept for entity identification and certificate issuance. Many issuers may use a community logotype to co-brand with a global community in order to gain global recognition of its local service provision. This type of community branding is very common in the credit card business, where local independent card issuers include a globally recognized brand (such as VISA and MasterCard). Certificate issuers may include more than one community logotype to indicate participation in more than one global community.

Issuer organization logotype is a logotype representing the organization identified as part of the issuer name in the certificate.

Subject organization logotype is a logotype representing the organization identified in the subject name in the certificate.

In addition to the standard logotype types, this specification accommodates inclusion of other logotype types where each class of logotype is defined by an object identifier. The object identifier can be either locally defined or an identifier defined in Section 4.4 of this document.

3. Logotype Data

This specification defines two types of logotype data: image data and audio data. Implementations **MUST** support image data; however, support for audio data is **OPTIONAL**.

Image and audio data for logotypes can be remote by including a URI that identifies the location to the logotype data and a one-way hash of the referenced data in the certificate. The privacy-related properties for remote logotype data depend on four parties: the certificate relying parties that use the information in the certificate extension to fetch the logotype data, the certificate issuers that populate the certificate extension, certificate subscribers that request certificates that include the certificate extension, and server operators that provides the logotype data.

Alternatively, embedding the logotype data in the certificate with direct addressing (as defined in Section 4.3) provides improved privacy properties and depends upon fewer parties. However, this approach can significantly increase the size of the certificate.

Several image objects, representing the same visual content in different formats, sizes, and color palates, may represent each logotype image. At least one of the image objects representing a logotype **SHOULD** contain an image with a width between of 60 pixels and 200 pixels and a height between 45 pixels and 150 pixels.

Several instances of audio data may further represent the same audio sequence in different formats, resolutions, and languages. At least one of the audio objects representing a logotype **SHOULD** provide text-based audio data suitable for processing by text-to-speech software.

A typical use of text based audio data is inclusion in web applications where the audio text is placed as the "alt" atttribute value of an html image (img) element and the language value obtained from LogotypeAudioInfo is included as the "lang" attribute of that image.

If a logotype of a certain type (as defined in Section 2) is represented by more than one image object, then each image objects

MUST contain variants of roughly the same visual content. Likewise, if a logotype of a certain type is represented by more than one audio object, then the audio objects **MUST** contain variants of the same audio information. A spoken message in different languages is considered a variation of the same audio information. Compliant applications **MUST NOT** display more than one of the image objects and **MUST NOT** play more than one of the audio object for any logotype type (see Section 2) at the same time.

A client **MAY** simultaneously display multiple logotypes of different logotype types. For example, it may display one subject organization logotype while also displaying a community logotype, but it **MUST NOT** display multiple image variants of the same community logotype.

Each logotype present in a certificate **MUST** be represented by at least one image data object.

Client applications **SHOULD** enhance processing and off-line functionality by caching logotype data.

4. Logotype Extension

This section specifies the syntax and semantics of the logotype certificate extension.

4.1. Extension Format

The logotype extension **MAY** be included in public key certificates [RFC5280] or attribute certificates [RFC5755]. The logotype extension **MUST** be identified by the following object identifier:

```
id-pe-logotype  OBJECT IDENTIFIER  ::=
   { iso(1) identified-organization(3) dod(6) internet(1)
     security(5) mechanisms(5) pkix(7) id-pe(1) 12 }
```

This extension **MUST NOT** be marked critical.

Logotype data may be referenced through either direct or indirect addressing. Client applications **SHOULD** support both direct and indirect addressing. Certificate issuing applications **MUST** support direct addressing, and certificate issuing applications **SHOULD** support indirect addressing.

The direct addressing includes information about each logotype in the certificate, and URIs point to the image and audio data object. Direct addressing supports cases where just one or a few alternative images and audio objects are referenced.

The indirect addressing includes one reference to an external hashed data structure that contains information on the type, content, and location of each image and audio object. Indirect addressing supports cases where each logotype is represented by many alternative audio or image objects.

Both direct and indirect addressing accommodate alternative URIs to obtain exactly the same logotype data. This opportunity for replication is intended to improve availability. Therefore, if a client is unable to fetch the item from one URI, the client **SHOULD** try another URI in the sequence. All direct addressing URIs **SHOULD** use either the HTTP scheme (http://...) or the HTTPS scheme (https://...) or the DATA scheme (data://...) [RFC3986]. However, the "data" URI scheme **MUST NOT** be used with the indirect addressing. Clients **MUST** support retrieval of referenced LogoTypeData with the HTTP [I-D.ietf-httpbis-semantics] and the HTTP with TLS [RFC8446], or subsequent versions of these protocols. Client applications **SHOULD** also support the "data" URI scheme [RFC2397] for direct addressing with embedded logotype data within the extension.

The logotype extension **MUST** have the following syntax:

```
LogotypeExtn ::= SEQUENCE {
   communityLogos  [0] EXPLICIT SEQUENCE OF LogotypeInfo OPTIONAL,
   issuerLogo      [1] EXPLICIT LogotypeInfo OPTIONAL,
   subjectLogo     [2] EXPLICIT LogotypeInfo OPTIONAL,
   otherLogos      [3] EXPLICIT SEQUENCE OF OtherLogotypeInfo
                          OPTIONAL }

LogotypeInfo ::= CHOICE {
   direct          [0] LogotypeData,
   indirect        [1] LogotypeReference }

LogotypeData ::= SEQUENCE {
   image           SEQUENCE OF LogotypeImage OPTIONAL,
   audio           [1] SEQUENCE OF LogotypeAudio OPTIONAL }

LogotypeImage ::= SEQUENCE {
   imageDetails    LogotypeDetails,
   imageInfo       LogotypeImageInfo OPTIONAL }

LogotypeAudio ::= SEQUENCE {
   audioDetails    LogotypeDetails,
   audioInfo       LogotypeAudioInfo OPTIONAL }

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

LogotypeImageInfo ::= SEQUENCE {
   type            [0] LogotypeImageType DEFAULT color,
   fileSize        INTEGER,  -- In octets, 0=unspecified
   xSize           INTEGER,  -- Horizontal size in pixels
   ySize           INTEGER,  -- Vertical size in pixels
   resolution      LogotypeImageResolution OPTIONAL,
   language        [4] IA5String OPTIONAL }  -- RFC 5646 Language Tag

LogotypeImageType ::= INTEGER { grayScale(0), color(1) }

LogotypeImageResolution ::= CHOICE {
   numBits         [1] INTEGER,   -- Resolution in bits per pixel
   tableSize       [2] INTEGER }  -- Number of colors or grey tones

LogotypeAudioInfo ::= SEQUENCE {
   fileSize        INTEGER,  -- In octets, 0=unspecified
   playTime        INTEGER,  -- In milliseconds, 0=unspecified
   channels        INTEGER,  -- 0=unspecified,
                             -- 1=mono, 2=stereo, 4=quad
   sampleRate      [3] INTEGER OPTIONAL,  -- Samples per second
   language        [4] IA5String OPTIONAL }  -- RFC 5646 Language Tag
```

```
OtherLogotypeInfo ::= SEQUENCE {
   logotypeType    OBJECT IDENTIFIER,
   info            LogotypeInfo }

LogotypeReference ::= SEQUENCE {
   refStructHash   SEQUENCE SIZE (1..MAX) OF HashAlgAndValue,
   refStructURI    SEQUENCE SIZE (1..MAX) OF IA5String }
                    -- Places to get the same LogotypeData
                    -- image or audio object

HashAlgAndValue ::= SEQUENCE {
   hashAlg         AlgorithmIdentifier,
   hashValue       OCTET STRING }
```

When using indirect addressing, the URI (refStructURI) pointing to the external data structure **MUST** point to a resource that contains the DER-encoded data with the syntax LogotypeData.

At least one of the optional elements in the LogotypeExtn structure **MUST** be present.

When using direct addressing, at least one of the optional elements in the LogotypeData structure **MUST** be present.

The LogotypeReference and LogotypeDetails structures explicitly identify one or more one-way hash functions employed to authenticate referenced image or audio objects. CAs **MUST** include a hash value for each referenced object, calculated on the whole object. CAs **SHOULD** include a hash value that computed with the one-way hash function associated with the certificate signature, and CAs **MAY** include other hash values. Clients **MUST** compute a one-way hash value using one of the identified functions, and clients **MUST** discard the logotype data if the computed hash value does not match the hash value in the certificate extension.

A MIME type is used to specify the format of the image or audio object containing the logotype data. The mediaType field **MUST** contain a string that is constructed according to the ABNF [RFC5234] provided in Section 4.2 of [RFC6838]. MIME types **MAY** include parameters.

Image format requirements are specified in Section 7, and audio format requirements are specified in Section 8.

When language is specified, the language tag **MUST** use the [RFC5646] syntax.

Logotype types defined in this specification are:

> Community Logotype: If communityLogos is present, the logotypes **MUST** represent one or more communities with which the certificate issuer is affiliated. The communityLogos **MAY** be present in an end entity certificate, a CA certificate, or an attribute certificate. The communityLogos contains a sequence of Community Logotypes, each representing a different community. If more than one Community logotype is present, they **MUST** be placed in order of preferred appearance. Some clients **MAY** choose to display a subset of the present community logos; therefore the placement within the sequence aids the client selection. The most preferred logotype **MUST** be first in the sequence, and the least preferred logotype **MUST** be last in the sequence.
>
> Issuer Organization Logotype: If issuerLogo is present, the logotype **MUST** represent the issuer's organization. The logotype

MUST be consistent with, and require the presence of, an organization name stored in the organization attribute in the issuer field (for either a public key certificate or attribute certificate). The issuerLogo **MAY** be present in an end entity certificate, a CA certificate, or an attribute certificate.

Subject Organization Logotype: If subjectLogo is present, the logotype **MUST** represent the subject's organization. The logotype **MUST** be consistent with, and require the presence of, an organization name stored in the organization attribute in the subject field (for either a public key certificate or attribute certificate). The subjectLogo **MAY** be present in an end entity certificate, a CA certificate, or an attribute certificate.

The relationship between the subject organization and the subject organization logotype, and the relationship between the issuer and either the issuer organization logotype or the community logotype, are relationships asserted by the issuer. The policies and practices employed by the issuer to check subject organization logotypes or claims its issuer and community logotypes is outside the scope of this document.

4.2. Conventions for LogotypeImageInfo

When the optional LogotypeImageInfo is included with a logotype image, the parameters **MUST** be used with the following semantics and restrictions.

The xSize and ySize fields represent the recommended display size for the logotype 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 object, then the referenced image **SHOULD** be scaled to fit within the size parameters of LogotypeImageInfo, while preserving the x and y ratio.

The resolution field is redundant for all logotype image formats listed in Section 7. The optional resolution field **SHOULD** be omitted when the image format already contains this information.

4.3. Embedded Images

If the logotype image is provided through direct addressing, then the image **MAY** be stored within the logotype certificate extension using the "data" scheme [RFC2397]. The syntax of the "data" URI scheme defined 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" URI 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" URI 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.

4.4. Other Logotypes

Logotypes identified by otherLogos (as defined in Section 4.1) can be used to enhance the display of logotypes and marks that represent partners, products, services, or any other characteristic associated with the certificate or its intended application environment when the standard logotype types are insufficient.

The conditions and contexts of the intended use of these logotypes are defined at the discretion of the local client application.

Three other logotype types are defined in the follow subsections.

4.4.1. Loyalty Logotype

When a loyalty logotype appears in the otherLogos, it **MUST** be identified by the id-logo-loyalty object identifier.

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

id-logo-loyalty    OBJECT IDENTIFIER ::= { id-logo 1 }
```

A loyalty logotype, if present, **MUST** contain a logotype associated with a loyalty program related to the certificate or its use. The relation between the certificate and the identified loyalty program is beyond the scope of this document. The logotype extension **MAY** contain more than one Loyalty logotype.

If more than one loyalty logotype is present, they **MUST** be placed in order of preferred appearance. Some clients **MAY** choose to display a subset of the present loyalty logotype data; therefore the placement within the sequence aids the client selection. The most preferred loyalty logotype data **MUST** be first in the sequence, and the least preferred loyalty logotype data **MUST** be last in the sequence.

4.4.2. Certificate Background Logotype

When a certificate background logotype appears in the otherLogos, it **MUST** be identified by the id-logo-background object identifier.

```
id-logo-background OBJECT IDENTIFIER ::= { id-logo 2 }
```

The certificate background logotype, if present, **MUST** contain a graphical image intended as a background image for the certificate, and/or a general audio sequence for the certificate. The background image **MUST** allow black text to be clearly read when placed on top of the background image. The logotype extension **MUST NOT** contain more than one certificate background logotype.

4.4.3. Certificate Image Logotype

When a certificate image logotype appears in the otherLogos, it **MUST** be identified by the id-logo-certImage object identifier.

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

The certificate image logotype, if present, aids human interpretation of a certificate by providing meaningful visual information to the user interface (UI). The logotype extension **MUST NOT** contain more than one certificate image logotype.

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.

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

*Certificate Issuer

*Certificate Subject

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 as the only visual representation of a certificate; however, the certificate user **SHOULD** be able to easily obtain the details of the certificate content.

5. Type of Certificates

Logotypes **MAY** be included in public key certificates and attribute certificates at the discretion of the certificate issuer; however, the relying party **MUST NOT** use the logotypes as part of certification path validation or automated trust decision. The sole purpose of logotypes is to enhance the display of a particular certificate, regardless of its position in a certification path.

6. Use in Clients

All PKI implementations require relying party software to have some mechanism to determine whether a trusted CA issues a particular certificate. This is an issue for certification path validation, including consistent policy and name checking.

After a certification path is successfully validated, the replying party trusts the information that the CA includes in the certificate, including any certificate extensions. The client software can choose to make use of such information, or the client software can ignore it. If the client is unable to support a provided logotype, the client **MUST NOT** report an error, rather the client **MUST** behave as though no logotype extension was included in the certificate. Current standards do not provide any mechanism for cross-certifying CAs to constrain subordinate CAs from including private extensions (see Section 9).

Consequently, if relying party software accepts a CA, then it should be prepared to (unquestioningly) display the associated logotypes to its human user, given that it is configured to do so. Information about the logotypes is provided so that the replying party software can select the one that will best meet the needs of the human user. This choice depends on the abilities of the human user, as well as the capabilities of the platform on which the replaying party software is running. If none of the provided logotypes meets the needs of the human user or matches the capabilities of the platform, then the logotypes can be ignored.

A client **MAY**, subject to local policy, choose to display none, one, or any number of the logotypes in the logotype extension. In many cases, a client will be used in an environment with a good network connection and also used in an environment with little or no network connectivity. For example, a laptop computer can be docked with a high-speed LAN connection, or it can be disconnected from the network altogether. In recognition of this situation, the client **MUST** include the ability to disable the fetching of logotypes. However, locally cached logotypes can still be displayed when the user disables the fetching of additional logotypes.

A client **MAY**, subject to local policy, choose any combination of audio and image presentation for each logotype. That is, the client **MAY** display an image with or without playing a sound, and it **MAY** play a sound with or without displaying an image. A client **MUST NOT** play more than one logotype audio sequence at the same time.

The logotype is to be displayed in conjunction with other identity information contained in the certificate. The logotype is not a replacement for this identity information.

Care is needed when designing replying party software to ensure that an appropriate context of logotype information is provided. This is especially difficult with audio logotypes. It is important that the human user be able to recognize the context of the logotype, even if other audio streams are being played.

If the relying party software is unable to successfully validate a particular certificate, then it **MUST NOT** display any logotype data associated with that certificate.

7. Image Formats

Animated images **SHOULD NOT** be used.

The following table lists many commons image formats and their corresponding MIME type. The table also indicates the support requirements these image formats. The filename extensions commonly used for each of these formats is also provided. Implementations **MAY** support other image formats.

Format	MIME Type	.ext	References	Implement?
JPEG	image/jpeg	.jpg .jpeg	[JPEG] [RFC2046]	**MUST** support
GIF	image/gif	.gif	[GIF] [RFC2046]	**MUST** support
SVG	image/svg+xml	.svg	[SVGT] [SVGR]	**SHOULD** support
SVG + GZIP	image/svg+xml+gzip	.svgz .svg.gz	[SVGT] [SVGZR]	**MUST** support
PNG	image/png	.png	[ISO15948] [PNGR]	**SHOULD** support
PDF	application/pdf	.pdf	[ISO32000] [ISO19005] [RFC8118]	**MAY** support

Table 1: Image Formats

NOTE: The image/svg+xml-compressed media type is widely implemented, but it has not yet been registered with IANA.

When a Scalable Vector Graphics (SVG) image is used, whether the image is compressed or not, the SVG Tiny profile [SVGT] **MUST** be followed, with these additional restrictions:

- 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 [SVGT].

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

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

When a GZIP-compressed SVG image is fetched with HTTP, the client will receive response that includes these headers:

```
Content-Type: image/svg+xml
Content-Encoding: gzip
```

In this case, the octet stream of type image/svg+xml is compressed with GZIP [RFC1952] as specified in [SVGR].

When a uncompressed SVG image is fetched with HTTP, the client will receive response with the same Content-Type header, but no Content-Encoding header.

Whether the SVG image is GZIP-compressed or uncompressed, the hash value for the SVG image is calculated over the uncompressed SVG content with canonicalized EOL characters as specified above.

When a SVG image is embedded in the certificate extension using the "data" URL scheme, the SVG image data **MUST** be provided in GZIP-compressed form, and the XML structure, prior to compression, **SHOULD** use linefeed (0x0A) as the end-of-line (EOL) character.

When a bitmapped image is used, the PNG [ISO15948] format **SHOULD** be used.

When a Portable Document Format (PDF) document according to [ISO32000] is used, it **MUST** also be formatted according to the profile PDF/A [ISO19005].

8. Audio Formats

Implementations that support audio **MUST** support the MP3 audio format [MP3] with a MIME type of "audio/mpeg" [RFC3003]. Implementations **SHOULD** support text-based audio data with a MIME type of "text/

plain;charset=UTF-8". Implementations **MAY** support other audio formats.

Text-based audio data using the MIME type of "text/plain;charset=UTF-8" is intended to be used by text-to-speech software. When this audio type is used, the following requirements apply:

- LogotypeAudioInfo **MUST** be present and specify the language of the text.

- The fileSize, playTime, and channels elements of LogotypeAudioInfo **MUST** have the value of 0.

- The sampleRate element of LogotypeAudioInfo **MUST** be absent.

9. Security Considerations

Implementations that simultaneously display multiple logotype types (subject organization, issuer, community, or other), **MUST** ensure that there is no ambiguity as to the binding between the image and the type of logotype that the image represents. "Logotype type" is defined in Section 1.1, and it refers to the type of entity or affiliation represented by the logotype, not the of binary format if the image or audio.

Logotypes are very difficult to securely and accurately define. Names are also difficult in this regard, but logotypes are even worse. It is quite difficult to specify what is, and what is not, a legitimate logotype of an organization. There is an entire legal structure around this issue, and it will not be repeated here. However, issuers should be aware of the implications of including images associated with a trademark or servicemark before doing so. As logotypes can be difficult (and sometimes expensive) to verify, the possibility of errors related to assigning wrong logotypes to organizations is increased.

This is not a new issue for electronic identification instruments. It is already dealt with in a number of similar situations in the physical world, including physical employee identification cards. In addition, there are situations where identification of logotypes is rather simple and straightforward, such as logotypes for well-known industries and institutes. These issues should not stop those service providers who want to issue logotypes from doing so, where relevant.

It is impossible to prevent fraudulent creation of certificates by dishonest or badly performing issuers, containing names and logotypes that the issuer has no claim to or has failed to check correctly. Such certificates could be created in an attempt to socially engineer a user into accepting a certificate. The premise used for the logotype work is thus that logotype graphics in a certificate are trusted only if the certificate is successfully validated within a valid path. It is thus imperative that the representation of any certificate that fails to validate is not enhanced in any way by using the logotype data.

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, including logotype data in certificates, to minimize risks related to processing potentially malicious data before it has been adequately verified and validated.

Referenced image objects 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 file that contains the image. If this feature were used within a logotype image, the hash of the image 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 7 and reject images that contain references to external data.

CAs issuing certificates with embedded logotype 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.

Certification paths may also impose name constraints that are systematically checked during certification path processing, which, in theory, may be circumvented by logotypes.

Certificate path processing as defined in [RFC5280] does not constrain the inclusion of logotype data in certificates. A parent CA can constrain certification path validation such that subordinate CAs cannot issue valid certificates to end-entities outside a limited name space or outside specific certificate polices. A malicious CA can comply with these name and policy requirements and still include inappropriate logotypes in the certificates that it issues. These certificates will pass the certification path validation algorithm, which means the client will trust the logotypes in the certificates. Since there is no technical mechanism to prevent or control subordinate CAs from including the logotype extension or its contents, where appropriate, a parent CA could employ a legal agreement to impose a suitable restriction on the subordinate CA. This situation is not unique to the logotype extension.

When a relying party fetches remote logotype data, a mismatch between the media type provided in the mediaType field of the LogotypeDetails and the Content-Type HTTP header of the retrieved object **MUST** be treated as a failure and the fetched logotype data should not be presented to the user. However, if more than one location for the remote logotype data is provided in the certificate extension, the relying party **MAY** try to fetch the remote logotype data from an alternate location to resolve the failure.

When a subscriber requests the inclusion of remote logotype data in a certificate, the CA cannot be sure that any logotype data will be available at the provided URI for the entire validity period of the certificate. To mitigate this concern, the CA may provide the logotype data from a server under its control, rather than a subscriber-controlled server.

The controls available to a parent CA to protect itself from rogue subordinate CAs are non-technical. They include:

*Contractual agreements of suitable behavior, including terms of liability in case of material breach.

*Control mechanisms and procedures to monitor and follow the behavior of subordinate CAs, including Certificate Transparency [RFC9162].

*Use of certificate policies to declare an assurance level of logotype data, as well as to guide applications on how to treat and display logotypes.

*Use of revocation functions to revoke any misbehaving CA.

There is not a simple, straightforward, and absolute technical solution. Rather, involved parties must settle some aspects of PKI outside the scope of technical controls. As such, issuers need to clearly identify and communicate the associated risks.

10. Privacy Considerations

Certificates are commonly public objects, so the inclusion of privacy-sensitive information in certificates should be avoided. The more information that is included in a certificate, the greater the likelihood that the certificate will reveal privacy-sensitive information. The inclusion of logotype data needs to be considered in this context.

Logotype data might be fetched from a server when it is needed. By watching activity on the network, an observer can determine which clients are making use of certificates that contain particular logotype data. Since clients are expected to locally cache logotype data, network traffic to the server containing the logotype data will not be generated every time the certificate is used. Further, when logotype data is not cached, activity on the network would reveal certificate usage frequency. Even when logotype data is cached, regardless of whether direct or indirect addressing is employed, network traffic monitoring could reveal when logotype data is fetched for the first time. Implementations **MAY** encrypt fetches of logotype data using HTTPS and pad them to a common size to reduce visibility into the data that is being fetched. Likewise, servers **MAY** reduce visibility into the data that is being returned by encrypting with HTTPS and padding to a few common sizes.

Similarly, when fetching logotype data from a server, the server operator can determine which clients are making use of certificates that contain particular logotype data. As above, locally caching logotype data will eliminate the need to fetch the logotype data each time the certificate is used, and lack of caching would reveal usage frequency. Even when implementations cache logotype data is cached, regardless of whether direct or indirect addressing is employed, the server operator could observe when logotype data is fetched for the first time.

When the "data" URI scheme is used with direct addressing, there is no network traffic to fetch logotype data, which avoids the observations of network traffic or server operations described above. To obtain this benefit, the certificate will be larger than one that contains a URL. Due to the improved privacy posture, the "data" URI scheme with direct addressing will be the only one that is supported by some CAs. Privacy-aware certificate subscribers **MAY** wish to insist on their logotype data being embedded in the certificate with the "data" URI scheme with direct addressing.

In cases where logotype data is cached by the relying party, the cache index should include the hash values of the associated logotype data with the goal of fetching the logotype data only once, even when it is referenced by multiple URIs. The index should include hash values for all supported hash algorithms. The cached data should include the media type as well as the logotype data. Implementations should give preference to logotype data that is already in the cache when multiple alternatives are offered in the LogotypeExtn certificate extension.

When the "data" URI scheme is used, the relying party **MAY** add the embedded logotype data to the local cache, which could avoid the need to fetch the logotype data if it is referenced by a URL in another certificate.

When fetching remote logotype data, relying parties should use the most privacy-preserving options that are available to minimize the opportunities for servers to "fingerprint" clients. For example, avoid cookies, e-tags, and client certificates.

When a relying party encounters a new certificate, the lack of network traffic to fetch logotype data might indicate that a certificate with references to the same logotype data has been previously processed and cached.

TLS 1.3 [RFC8446] includes the ability to encrypt the server's certificate in the TLS handshake, which helps hide the server's identity from anyone that is watching activity on the network. If the server's certificate includes remote logotype data, the client fetching that data might disclose the otherwise protected server identity.

11. IANA Considerations

For the new ASN.1 Module in Appendix A.2, IANA is requested to assign an object identifier (OID) for the module identifier. The OID for the module should be allocated in the "SMI Security for PKIX Module Identifier" registry (1.3.6.1.5.5.7.0).

12. Acknowledgments

12.1. Acknowledgments from RFC 3709

This document is the result of contributions from many professionals. The authors appreciate contributions from all members of the IETF PKIX Working Group. We extend a special thanks to Al Arsenault, David Cross, Tim Polk, Russel Weiser, Terry Hayes, Alex Deacon, Andrew Hoag, Randy Sabett, Denis Pinkas, Magnus Nystrom, Ryan Hurst, and Phil Griffin for their efforts and support.

Russ Housley thanks the management at RSA Laboratories, especially Burt Kaliski, who supported the development of this specification. The vast majority of the work on this specification was done while Russ was employed at RSA Laboratories.

12.2. Acknowledgments from RFC 6170

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.

12.3. Additional Acknowledgments

Combining RFC 3709 and RFC 6170 has produced an improved specification. The authors appreciate contributions from all members of the IETF LAMPS Working Group. We extend a special thanks to Alexey Melnikov for his guidance on media types. We extend a special thanks to Corey Bonnell and Daniel Kahn Gillmor for their careful review and comments.

13. References

13.1. Normative References

[GIF] CompuServe Incorporated, "Graphics Interchange Format", Version 89a, 31 July 1990, <https://www.w3.org/Graphics/GIF/spec-gif89a.txt>.

[I-D.ietf-httpbis-semantics] Fielding, R. T., Nottingham, M., and J. Reschke, "HTTP Semantics", Work in Progress, Internet-Draft, draft-ietf-httpbis-semantics-19, 12 September 2021, <https://www.ietf.org/archive/id/draft-ietf-httpbis-semantics-19.txt>.

[ISO15948] ISO/IEC, "Information technology -- Computer graphics and image processing -- Portable Network Graphics (PNG): Functional specification", ISO/IEC 15948:2004, 2004.

[JPEG] ITU-T, "Information technology -- Digital compression and coding of continuous-tone still images: JPEG File Interchange Format (JFIF)", ITU-T Recommendation T.871, ISO/IEC 10918-5:2013, May 2011.

[MP3] ISO/IEC, "Information technology -- Generic coding of moving pictures and associated audio information -- Part 3: Audio", ISO/IEC 13818-3:1998, 1998.

[NEW-ASN1] ITU-T, "Information technology -- Abstract Syntax Notation One (ASN.1): Specification of basic notation", ITU-T Recommendation X.680, ISO/IEC 8824-1:2021, February 2021, <https://www.itu.int/rec/T-REC-X.680>.

[RFC1952] Deutsch, P., "GZIP file format specification version 4.3", RFC 1952, DOI 10.17487/RFC1952, May 1996, <https://www.rfc-editor.org/info/rfc1952>.

[RFC2046] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046, DOI 10.17487/RFC2046, November 1996, <https://www.rfc-editor.org/info/rfc2046>.

[RFC2119]
Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2397] Masinter, L., "The "data" URL scheme", RFC 2397, DOI 10.17487/RFC2397, August 1998, <https://www.rfc-editor.org/info/rfc2397>.

[RFC3003] Nilsson, M., "The audio/mpeg Media Type", RFC 3003, DOI 10.17487/RFC3003, November 2000, <https://www.rfc-editor.org/info/rfc3003>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <https://www.rfc-editor.org/info/rfc5234>.

[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, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/info/rfc5280>.

[RFC5646] Phillips, A., Ed. and M. Davis, Ed., "Tags for Identifying Languages", BCP 47, RFC 5646, DOI 10.17487/RFC5646, September 2009, <https://www.rfc-editor.org/info/rfc5646>.

[RFC5755] Farrell, S., Housley, R., and S. Turner, "An Internet Attribute Certificate Profile for Authorization", RFC 5755, DOI 10.17487/RFC5755, January 2010, <https://www.rfc-editor.org/info/rfc5755>.

[RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type Specifications and Registration Procedures", BCP 13, RFC

6838, DOI 10.17487/RFC6838, January 2013, <https://www.rfc-editor.org/info/rfc6838>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[SVGT] World Wide Web Consortium, "Scalable Vector Graphics (SVG) Tiny 1.2 Specification", W3C PR-SVGTiny12-20081117, 17 November 2008, <http://www.w3.org/TR/2008/PR-SVGTiny12-20081117>.

13.2. Informative References

[ISO19005] ISO, "Document management -- Electronic document file format for long-term preservation -- Part 1: Use of PDF 1.4 (PDF/A-1)", ISO 19005-1:2005, 2005.

[ISO32000] ISO, "Document management -- Portable document format -- Part 1: PDF 1.7", ISO 32000-1:2008, 2008.

[OLD-ASN1] CCITT, "Specification of Abstract Syntax Notation One (ASN.1)", CCITT Recommendation X.208, November 1988, <https://www.itu.int/rec/T-REC-X.208/en>.

[PNGR] World Wide Web Consortium, "Media Type Registration for image/png", <https://www.iana.org/assignments/media-types/image/png>.

[RFC3709] Santesson, S., Housley, R., and T. Freeman, "Internet X.509 Public Key Infrastructure: Logotypes in X.509 Certificates", RFC 3709, DOI 10.17487/RFC3709, February 2004, <https://www.rfc-editor.org/info/rfc3709>.

[RFC5912] Hoffman, P. and J. Schaad, "New ASN.1 Modules for the Public Key Infrastructure Using X.509 (PKIX)", RFC 5912, DOI 10.17487/RFC5912, June 2010, <https://www.rfc-editor.org/info/rfc5912>.

[RFC6170] Santesson, S., Housley, R., Bajaj, S., and L. Rosenthol, "Internet X.509 Public Key Infrastructure -- Certificate Image", RFC 6170, DOI 10.17487/RFC6170, May 2011, <https://www.rfc-editor.org/info/rfc6170>.

[RFC6268] Schaad, J. and S. Turner, "Additional New ASN.1 Modules for the Cryptographic Message Syntax (CMS) and the Public

Key Infrastructure Using X.509 (PKIX)", RFC 6268, DOI 10.17487/RFC6268, July 2011, <https://www.rfc-editor.org/info/rfc6268>.

[RFC8118] Hardy, M., Masinter, L., Markovic, D., Johnson, D., and M. Bailey, "The application/pdf Media Type", RFC 8118, DOI 10.17487/RFC8118, March 2017, <https://www.rfc-editor.org/info/rfc8118>.

[RFC9162] Laurie, B., Messeri, E., and R. Stradling, "Certificate Transparency Version 2.0", RFC 9162, DOI 10.17487/RFC9162, December 2021, <https://www.rfc-editor.org/info/rfc9162>.

[RFC9216] Gillmor, D. K., Ed., "S/MIME Example Keys and Certificates", RFC 9216, DOI 10.17487/RFC9216, April 2022, <https://www.rfc-editor.org/info/rfc9216>.

[SVGR] World Wide Web Consortium, "Media Type Registration for image/svg+xml", <https://www.iana.org/assignments/media-types/image/svg+xml>.

[SVGZR] "A separate MIME type for svgz files is needed", <https://github.com/w3c/svgwg/issues/701>.

Appendix A. ASN.1 Modules

A.1. ASN.1 Modules with 1988 Syntax

This appendix contains two ASN.1 modules, both using the old syntax [OLD-ASN1].

The first ASN.1 module provides the syntax for the Logotype certificate extension. Only comments have changed in the module from RFC 3709, and the IMPORTS now come from [RFC5280].

The second ASN.1 module provides the Certificate Image object identifier. The module is unchanged from RFC 6170.

```
<CODE BEGINS>
LogotypeCertExtn
  { iso(1) identified-organization(3) dod(6) internet(1)
    security(5) mechanisms(5) pkix(7) id-mod(0)
    id-mod-logotype(22) }

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

IMPORTS
   AlgorithmIdentifier FROM PKIX1Explicit88 -- RFC 5280
     { iso(1) identified-organization(3) dod(6) internet(1)
       security(5) mechanisms(5) pkix(7) id-mod(0)
       id-pkix1-explicit(18) };

-- Logotype Extension OID

id-pe-logotype  OBJECT IDENTIFIER  ::=
   { iso(1) identified-organization(3) dod(6) internet(1)
     security(5) mechanisms(5) pkix(7) id-pe(1) 12 }


-- Logotype Extension Syntax

LogotypeExtn ::= SEQUENCE {
   communityLogos  [0] EXPLICIT SEQUENCE OF LogotypeInfo OPTIONAL,
   issuerLogo      [1] EXPLICIT LogotypeInfo OPTIONAL,
   subjectLogo     [2] EXPLICIT LogotypeInfo OPTIONAL,
   otherLogos      [3] EXPLICIT SEQUENCE OF OtherLogotypeInfo
                          OPTIONAL }

-- Note: At least one of the OPTIONAL components MUST be present

LogotypeInfo ::= CHOICE {
   direct          [0] LogotypeData,
   indirect        [1] LogotypeReference }

LogotypeData ::= SEQUENCE {
   image           SEQUENCE OF LogotypeImage OPTIONAL,
   audio           [1] SEQUENCE OF LogotypeAudio OPTIONAL }

-- Note: At least one of the OPTIONAL components MUST be present

LogotypeImage ::= SEQUENCE {
   imageDetails    LogotypeDetails,
   imageInfo       LogotypeImageInfo OPTIONAL }

LogotypeAudio ::= SEQUENCE {
   audioDetails    LogotypeDetails,
   audioInfo       LogotypeAudioInfo OPTIONAL }
```

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

LogotypeImageInfo ::= SEQUENCE {
   type            [0] LogotypeImageType DEFAULT color,
   fileSize        INTEGER,  -- In octets, 0=unspecified
   xSize           INTEGER,  -- Horizontal size in pixels
   ySize           INTEGER,  -- Vertical size in pixels
   resolution      LogotypeImageResolution OPTIONAL,
   language        [4] IA5String OPTIONAL }  -- RFC 5646 Language Tag

LogotypeImageType ::= INTEGER { grayScale(0), color(1) }

LogotypeImageResolution ::= CHOICE {
   numBits         [1] INTEGER,   -- Resolution in bits per pixel
   tableSize       [2] INTEGER }  -- Number of colors or grey tones

LogotypeAudioInfo ::= SEQUENCE {
   fileSize        INTEGER,  -- In octets, 0=unspecified
   playTime        INTEGER,  -- In milliseconds, 0=unspecified
   channels        INTEGER,  -- 0=unspecified,
                             -- 1=mono, 2=stereo, 4=quad
   sampleRate      [3] INTEGER OPTIONAL,  -- Samples per second
   language        [4] IA5String OPTIONAL }  -- RFC 5646 Language Tag

OtherLogotypeInfo ::= SEQUENCE {
   logotypeType    OBJECT IDENTIFIER,
   info            LogotypeInfo }

LogotypeReference ::= SEQUENCE {
   refStructHash   SEQUENCE SIZE (1..MAX) OF HashAlgAndValue,
   refStructURI    SEQUENCE SIZE (1..MAX) OF IA5String }
                    -- Places to get the same LogotypeData
                    -- image or audio object

-- Note: The referenced LogotypeData binary file contains a
--       DER-encoded LogotypeData type

HashAlgAndValue ::= SEQUENCE {
   hashAlg         AlgorithmIdentifier,
   hashValue       OCTET STRING }

-- Other logotype type OIDs

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

```
id-logo-loyalty    OBJECT IDENTIFIER ::= { id-logo 1 }

id-logo-background OBJECT IDENTIFIER ::= { id-logo 2 }

END


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-logo(20) 3 }

END

<CODE ENDS>
```

A.2. ASN.1 Module with 2002 Syntax

Some developers like to use the latest version of ASN.1 standards. This appendix provides an ASN.1 module to assist in that goal. It uses the ASN.1 syntax defined in [NEW-ASN1], and it follows the conventions established in [RFC5912] and [RFC6268].

This ASN.1 module incorporates the module from RFC 3709 and the module from RFC 6170.

Note that [NEW-ASN1] was published in 2021, and all of the features used in this module are backward compatible with the specification that was published in 2002.

```
<CODE BEGINS>
LogotypeCertExtn
  { iso(1) identified-organization(3) dod(6) internet(1)
    security(5) mechanisms(5) pkix(7) id-mod(0)
    id-mod-logotype(TBD) }

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

IMPORTS
  EXTENSION
  FROM PKIX-CommonTypes-2009  -- RFC 5912
    { iso(1) identified-organization(3) dod(6) internet(1)
      security(5) mechanisms(5) pkix(7) id-mod(0)
      id-mod-pkixCommon-02(57) }

  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) } ;


-- Logotype Extension

ext-logotype EXTENSION ::= {
   SYNTAX LogotypeExtn
   IDENTIFIED BY id-pe-logotype }

-- Logotype Extension OID

id-pe-logotype  OBJECT IDENTIFIER  ::=
   { iso(1) identified-organization(3) dod(6) internet(1)
     security(5) mechanisms(5) pkix(7) id-pe(1) 12 }

-- Logotype Extension Syntax

LogotypeExtn ::= SEQUENCE {
   communityLogos  [0] EXPLICIT SEQUENCE OF LogotypeInfo OPTIONAL,
   issuerLogo      [1] EXPLICIT LogotypeInfo OPTIONAL,
   subjectLogo     [2] EXPLICIT LogotypeInfo OPTIONAL,
   otherLogos      [3] EXPLICIT SEQUENCE OF OtherLogotypeInfo
                          OPTIONAL }
      -- At least one of the OPTIONAL components MUST be present
      ( WITH COMPONENTS { ..., communityLogos PRESENT } |
        WITH COMPONENTS { ..., issuerLogo PRESENT } |
        WITH COMPONENTS { ..., subjectLogo PRESENT } |
        WITH COMPONENTS { ..., otherLogos PRESENT } )

LogotypeInfo ::= CHOICE {
   direct          [0] LogotypeData,
   indirect        [1] LogotypeReference }

LogotypeData ::= SEQUENCE {
   image           SEQUENCE OF LogotypeImage OPTIONAL,
   audio           [1] SEQUENCE OF LogotypeAudio OPTIONAL }
      -- At least one image component MUST be present
      ( WITH COMPONENTS { ..., image PRESENT } )

LogotypeImage ::= SEQUENCE {
   imageDetails    LogotypeDetails,
   imageInfo       LogotypeImageInfo OPTIONAL }

LogotypeAudio ::= SEQUENCE {
   audioDetails    LogotypeDetails,
   audioInfo       LogotypeAudioInfo OPTIONAL }

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

LogotypeImageInfo ::= SEQUENCE {
   type            [0] LogotypeImageType DEFAULT color,
   fileSize        INTEGER,  -- In octets, 0=unspecified
   xSize           INTEGER,  -- Horizontal size in pixels
   ySize           INTEGER,  -- Vertical size in pixels
   resolution      LogotypeImageResolution OPTIONAL,
   language        [4] IA5String OPTIONAL }  -- RFC 5646 Language Tag

LogotypeImageType ::= INTEGER { grayScale(0), color(1) }

LogotypeImageResolution ::= CHOICE {
   numBits         [1] INTEGER,   -- Resolution in bits
   tableSize       [2] INTEGER }  -- Number of colors or grey tones

LogotypeAudioInfo ::= SEQUENCE {
   fileSize        INTEGER,  -- In octets, 0=unspecified
   playTime        INTEGER,  -- In milliseconds, 0=unspecified
   channels        INTEGER,  -- 0=unspecified
                             -- 1=mono, 2=stereo, 4=quad
   sampleRate      [3] INTEGER OPTIONAL,  -- Samples per second
   language        [4] IA5String OPTIONAL }  -- RFC 5646 Language Tag

OtherLogotypeInfo ::= SEQUENCE {
   logotypeType    OBJECT IDENTIFIER,
   info            LogotypeInfo }

LogotypeReference ::= SEQUENCE {
   refStructHash   SEQUENCE SIZE (1..MAX) OF HashAlgAndValue,
   refStructURI    SEQUENCE SIZE (1..MAX) OF IA5String }
                    -- Places to get the same LogotypeData
                    -- image or audio object

-- Note: The referenced LogotypeData binary file contains a
--       DER-encoded LogotypeData type

HashAlgAndValue ::= SEQUENCE {
   hashAlg         AlgorithmIdentifier{DIGEST-ALGORITHM, {...}},
   hashValue       OCTET STRING }

-- Other logotype type OIDs

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

id-logo-loyalty    OBJECT IDENTIFIER ::= { id-logo 1 }

id-logo-background OBJECT IDENTIFIER ::= { id-logo 2 }

id-logo-certImage  OBJECT IDENTIFIER  ::= { id-logo 3 }

END

<CODE ENDS>
```

Appendix B. Examples

B.1. Example from RFC 3709

The following example displays a logotype extension containing one Issuer logotype using direct addressing. The issuer logotype image is of the type image/gif. The logotype image is referenced through one URI and the image is hashed with SHA-1. This example is unchanged from RFC 3709, except that shallow indenting is used to keep the example within traditional margins. The use of SHA-1 was reasonable at the time that RFC 3709 was published, but many better choices are available today.

The values on the left are the ASN.1 tag (in hexadecimal) and the length (in decimal).

```
30 106: SEQUENCE {
06   8:  OBJECT IDENTIFIER logotype (1 3 6 1 5 5 7 1 12)
04  94:  OCTET STRING, encapsulates {
30  92:   SEQUENCE {
A1  90:    [1] {
A0  88:     [0] {
30  86:      SEQUENCE {
30  84:       SEQUENCE {
30  82:        SEQUENCE {
16   9:         IA5String 'image/gif'
30  33:         SEQUENCE {
30  31:          SEQUENCE {
30   7:           SEQUENCE {
06   5:            OBJECT IDENTIFIER sha1 (1 3 14 3 2 26)
      :             }
04  20:           OCTET STRING
      :            8F E5 D3 1A 86 AC 8D 8E 6B C3 CF 80 6A D4 48 18
      :            2C 7B 19 2E
      :            }
      :           }
30  34:         SEQUENCE {
16  32:          IA5String 'http://logo.example.com/logo.gif'
      :           }
      :          }
      :         }
      :        }
      :       }
      :      }
      :     }
      :    }
      :   }
```

B.2. Issuer Logotype Example

The following example displays a logotype extension containing one Issuer logotype using direct addressing. The issuer logotype image is of the type image/jpeg. The logotype image is referenced through one URI and the image is hashed with SHA-256.

The values on the left are the ASN.1 tag (in hexadecimal) and the length (in decimal).

```
30 124: SEQUENCE {
06   8:  OBJECT IDENTIFIER logotype (1 3 6 1 5 5 7 1 12)
04 112:  OCTET STRING, encapsulates {
30 110:   SEQUENCE {
A1 108:    [1] {
A0 106:     [0] {
30 104:      SEQUENCE {
30 102:       SEQUENCE {
30 100:        SEQUENCE {
16  10:         IA5String 'image/jpeg'
30  49:         SEQUENCE {
30  47:          SEQUENCE {
30  11:           SEQUENCE {
06   9:            OBJECT IDENTIFIER
      :             sha-256 (2 16 840 1 101 3 4 2 1)
      :             }
04  32:           OCTET STRING
      :            1E 8F 96 FD D3 50 53 EF C6 1C 9F FC F0 00 2E 53
      :            B4 9C 24 9A 32 C5 E9 0C 2C 39 39 D3 AD 6D A9 09
      :            }
      :           }
30  35:         SEQUENCE {
16  33:          IA5String 'http://logo.example.com/logo.jpeg'
      :           }
      :          }
      :         }
      :        }
      :       }
      :      }
      :     }
      :    }
      :   }
```

B.3. Embedded Image Example

The following example displays a logotype extension containing one Subject logotype using direct addressing. The subject logotype image uses image/svg+xml-compressed. The logotype image is embedded in the certificate extension with a "data:" URI and the image is hashed by SHA-256. This technique produces a large certificate extension, but offers reduced latency and improved privacy.

The values on the left are the ASN.1 tag (in hexadecimal) and the length (in decimal).

```
30 2160: SEQUENCE {
06    8:  OBJECT IDENTIFIER logotype (1 3 6 1 5 5 7 1 12)
04 2146:  OCTET STRING, encapsulates {
30 2142:   SEQUENCE {
A2 2138:    [2] {
A0 2134:     [0] {
30 2130:      SEQUENCE {
30 2126:       SEQUENCE {
30 2122:        SEQUENCE {
16   24:         IA5String 'image/svg+xml-compressed'
30   49:         SEQUENCE {
30   47:          SEQUENCE {
30   11:           SEQUENCE {
06    9:            OBJECT IDENTIFIER
       :             sha-256 (2 16 840 1 101 3 4 2 1)
       :             }
04   32:           OCTET STRING
       :           C5 AC 94 1A 0A 25 1F B3 16 6F 97 C5 52 40 9B 49
       :           9E 7B 92 61 5A B0 A2 6C 19 BF B9 D8 09 C5 D9 E7
       :            }
       :           }
30 2041:         SEQUENCE {
16 2037:          IA5String
       :          'data:image/svg+xml-compressed;base64,H4sICIGpy2E'
       :          'AA2xvZ28tY29weS5zdmcApVbbbhs3EH3nV0y3Lw2Q9fK2JLe'
       :          'wHDROUBRo2iBxW+RRlTa2UFkypIWV5ut7zlB2UqF9cuLlUkt'
       :          'yLmfOzPD8xafbtdyPu/1qu5k17sw2sp/mm+V8vd2Ms2azbV5'
       :          'cmPNvXv16efXh7WvZ31/L299e/vzTpTRt1/0RLrvu1dUref/'
       :          '7j+KtdXawsete/9IYaW6m6e77rjscDmeHcLbdXXdX7zpu6t6'
       :          '9vmxxon08AREdRDt7tpyWDRRSz7+tgp2b/ew/hEKI5WGoPKy'
       :          'W082s8SmeWf13NzVyM66ub6ZZk+xXH+9X4+Hl9tOssWLly35'
       :          '53ARpd7txP+7uxx/2d+NiejefVttZ8+nNavkBj9yO40RLb8d'
       :          'pvpxP8wtzuRvn07iUP/+Wu+20my9GcWfOPpfDbjVN44YLb8d'
       :          'p3Mn7cb3aXGNCAICCc+a8+yLo/FpwfLP/uN3dzhqdriH5uwf'
       :          'bnj9a+Uz2i/maK66utA+zZ435uFqvZ823R38Q1t32Lw3pZqT'
       :          'hd/PpRpaz5o2LNkocvCzaIm0vrQvSpog359lLy3my0ga+e3H'
       :          'p+B4InjVFPD9awdhnrGEFW30Sl/Pnpvta2QBVxUEVxFbJ2VU'
       :          'FfYC01pUs+O4GK84V/k6CHUFyhvhiDVQF8Y5aPDbmnsrXbS7'
       :          '4DANjguwgENZLPwjUYVTRJQgEpiLR0ctiWj+Ig8rCvZAArxK'
       :          'ExEEWMJLqMA1F+ggnsQDXgpQeomJPCVhtCRycNrAWxgAI+g1'
       :          'Qsr6IUxlomBswjydYBEgOeVCDoRreBjiFjX2SdSA60BP5DgQ'
       :          'M63xoPlWHbNq+egAEeAzxyNAdCQz+sDEMOhaGisKJdSlS6gt'
       :          'WWm4M1rQwP0egEBIhhFLoXuCJhR4mT5RJBaiLKqqFROUEzYr'
       :          '1idG0gahwCzEnk+AMJLdp0FevQQ6VZ+SKOwGlOIJOh1MVjo0'
       :          'eB6DRA10SRpSY6il/eFFKAm+MKSIWNFqSo4OFnORfwH5wJHC'
       :          'MNM0qlDRlcIwUEkDlgiSBhiEpBgMKOx5FdAYqI3KYewKKkAI'
       :          'tTABTkp5khI86kgbOgRywEBR0VGcwAjf8t9wqvdUMG6gLAbI'
       :          '0QQ8CbzCTtCSn/DEhCbm++duQaiRG1mQkdWHnminHA+r5wpL'
       :          'vsJbCALUKsDW5NAj43J+AD5vpfamUzJqiRJACmCWwIMhQq4H'
:          'mYGKaiiJPmIvpS80UzTtAjdSraApQZogslgFcJHw0y5WoEXD'
:          'Yr/aTqfxk2qhcg3z6ETQL+S18llvHOZQvlEOVEVpzqCozE9V'
:          '6JZhh/lCslg7mUFY4AR7IlcApmgV6gz3DCSDe56fQ0SRS7el'
:          '0NJWO8mQ6mkc6ylPpaL7QUZ5IR/M/dEwoJiEp+L6iT4cdSyI'
:          'p4ljDkoaZpQlgMoz0ApahjTiTWbZYu9v+MUqVjY61j2Bxr68'
:          'bPF3uS1232qAyAQDMhr4MRyVZq5l2QcuwgY/oTozbgoIKycH'
:          '+yQxhzQsPJQ/ne9OmRKvYH1AeKA/EQRtzrmaYUiHUhpJOW4b'
:          'reSaxZ/TVc3ZAQJKOagAJiw6pRHVkBMIBa5E+SUMWi0ZNW1R'
:          'fn/xQXywHXyMHN5G8WF6gZ2IVjANHMIJQ1lAJQE8MJjZHJiU'
:          'tQZAWzmkisDywTVWSqLkkQG2NNB3wwyaerqRGLNKpvwUOhaQ'
:          'FiYcqviSjvp1n8WnRRzXFs9IXDxiiDd8HU/ROoAGn9+QgTPE'
:          'Vu6HaN6i0VPuv1SCzwyZeHwBA1EjFYoAk2jJ3OFeJ5Gp1E+3'
:          'Dlf3Aj70bbvmag5oyKHunVyGPq6+EnvTua/JUn3iadMHlqUa'
:          'psK2T8SwCBJUF1JnEmhu0ntBthJoQpZqumsBk5mA1hRc0LR5'
:          'ZFerdjksaCqt3IUWXcXW16vb6xdWyHLTgCaKXWKUKK1kOp9H'
:          'K5B3ELjSdXb0loB5RYtS01L6h9yTPW51Wpqwgosr5I927aw6'
:          '401+YfwDria4WoQwAAA=='
:           }
:          }
:         }
:        }
:       }
:      }
:     }
:    }
:   }
```

B.4. Embedded Certificate Image Example

The following example displays a logotype extension containing one Certificate Image logotype using direct addressing. The Certificate Image logotype uses image/svg+xml-compressed. The logotype image is embedded in the certificate extension with a "data:" URI and the image is hashed by SHA-256. This example contains the image from Appendix B of RFC 6170, however, the media type used here is explicit about the use of GZIP compression [RFC1952].

The values on the left are the ASN.1 tag (in hexadecimal) and the length (in decimal).

```
30 2910: SEQUENCE {
06    8:  OBJECT IDENTIFIER logotype (1 3 6 1 5 5 7 1 12)
04 2896:  OCTET STRING, encapsulates {
30 2892:   SEQUENCE {
A3 2888:    [3] {
30 2884:     SEQUENCE {
30 2880:      SEQUENCE {
06    8:       OBJECT IDENTIFIER '1 3 6 1 5 5 7 20 3'
A0 2866:       [0] {
30 2862:        SEQUENCE {
30 2858:         SEQUENCE {
16   24:          IA5String 'image/svg+xml-compressed'
30   49:          SEQUENCE {
30   47:           SEQUENCE {
30   11:            SEQUENCE {
06    9:             OBJECT IDENTIFIER
       :              sha-256 (2 16 840 1 101 3 4 2 1)
       :              }
04   32:            OCTET STRING
       :           83 14 B3 26 9B D3 8B 0B 2A E6 6E 42 74 E2 A7 57
       :           7A 40 B7 E1 2E 53 42 44 CC 7C AE 14 68 1B 0E B6
       :             }
       :            }
30 2777:          SEQUENCE {
16 2773:           IA5String
       :          'data:image/svg+xml-compressed;base64,H4sICLXutU0'
       :          'AA0NlcnRJbWFnZURlbW8uc3ZnANVaW2/bOBZ+n19BqBigwdo'
       :          'S7xK9jmeapB0EWHQHzez2WZZoR1tZMiQ5jvvr95CSL7Gl1Em'
       :          '8C9d9iERSPOd85+O5EB3+9jhL0YMuyiTPLh3iYgfpLMrjJJt'
       :          'eOv/661M/cFBZhVkcpnmmL50sd34b/TIsH6YoiS+da11UySS'
       :          'Jwkqj21k41Q6CDbNyUMSTS+e+quYDz1sul+6SuXkx9YhSysP'
       :          'Uo7QPK/rlKqvCx35Wvmu+a/uGYow9EOigh0Qvr/LHSwcjjDj'
       :          'GiGHQ914n0/sKlMf4Vwctk7i6X7/sGEYdNA5L/WeRT5IUDKm'
       :          'SbLVWNoo2cqNCh1XyoKN8Nsuz0iqwVW8Qb1fOF0Vqp+PI06m'
       :          'e6awqPeISzxn9goYzXYVxWIUWpfWLCMwcGoLpgy83n8wzGkb'
       :          'R4GtefENmMBznC7DEroKpOBpM8mIWVqPEYGtA+BvoMfS2E5u'
       :          'F1Wqu7R6FLvNFEelWReNolpiV3l2VpGntMW9nk6RKdf0+9Br'
       :          'FrMbeVuWhtzbHvMR6UlobPyVpBWjXBk7six2vH5nCwY6nXCo'
       :          '5xb7YusvFVPqCOGh16fSxSxglmPkScLfvmDDmC4FlDc1wov8'
       :          'IF2WZhNlVumgEPRliimDD3PhGPyTgUUMC6lKqKAjxaptq1bo'
       :          'UJvQFsvi+LOJyxZkPE/vCwHuAmXmoj1AarnRBatzqkbv7cK5'
       :          'Ls2ORfwM/vsOG5lURZqXxOnDXPKZw5t5jVzIhFKO0B6D6hAR'
       :          'SXDR6Fzqq7H7mQeJAOQiUSPvFIrUHOfuui3zrFI5dYVeAmpc'
       :          'OcOb9u63vLjae4kYX4yRifYPrTa2SlMigYdO+cEWeGADMLZL'
       :          'H96SH4R9xRYApl6q3Y02f+NzlRAl+cZSKhB6qSIVa80fsqMn'
       :          'WOqZJpmsXwAPoyNaQ95uNIGasKPwhxGzQzOXzMIIzBKabmLI'
       :          'il470zfSjWWn+kvpvLQ9g1l3yRIc8gukz0uysEcakcDfy3KM'
       :          'k+l0SOXlOopltJL7EPtUlzZfP4tnM70k8xkKCySt92MwfIXP'
       :          'oTe0pnu4dYbp7hJ/kxWySN0ey0o/1qbiCsxDXJMWWo37QekB'
:          'cAUFPSGkPCnUJF5wwBacDK5cGlEp4BC2lYoJcrNNGVc7DzIq'
:          'xT4CKsPlrAG8mL8whRejiQe9EmImIAoz3sds9NxP4RZEzugq'
:          'zb7c3Q89u3WQKY9aegbsA/AUJB/bJs6pfJt9BHFEuk5DWITz'
:          'OH5uZSThLUsDjQ5GE6RMsyihMTaQLfA6BIiAQMAhnHHN1sd6'
:          '1WtUhDVJiuhkrdBXd740+hLB9Vm1HjQe4ywLOBLWOMMiyQAX'
:          'NB8sm9Gx2qdGgGkMG6wY8aLfqgH4dfnmrVc+pPrE/Z/QnZOs'
:          '8C1Okb2/ggwLdxlDC1D6DFPZDD98txv8xQf5TEc7Ax6ZyaDf'
:          '6BC4SylWKCMqtizp80+UMchATal63qHq0M3ZTs83Ob/XO6LY'
:          'sFzpGVY5+iLxdWvwY+NaKoR/0iJIXL3dBjT2hG+wO+NXm53X'
:          'StSh1eogfeojV35BTOaqh/cmPUe2Mdp91pQp2CjWOO2k7Oam'
:          'hjU1HB3DLGm66n6iajz4bqn2oICmNFxDR/x2mC5s+rKhlkUA'
:          '3Ne3P8lgP0qJfjf9uvu+HWXSfFwNoH4uqGUmTadYMtOc7yjE'
:          'Ed9EUhkwEEOcDSHKQ+yhnSvUYRH8miQo2FK5TCjWZZGWKB8i'
:          'HPud16wApnCvTOzjIFAj9TQdCxa+ddOTizaa1xJvD0qMrKx+'
:          'Ydaj6iwJQG0vaSdYWpTv4HwVRAP3Z6ONjOJunEIeKRVmhujp'
:          'A2+wPmQR9WFQAFhh9bGQzFEXX+WwOnXq8pV35P2Acdn0pGeb'
:          'cMg7OgQKaEdOKEAkFlk/9HuEKGBVwucc4AjnJ/LBYU09hVwW'
:          'Y1F0HlBUC2lbyIuYF58O8p+adMwUt9YAoX/IwRtAC9NAdBAy'
:          'GuEB3VR59u8/TGYx9/Xjz8bPB/Z/F9B0SghBK+4xxfiwtr0G'
:          'XECqedQQ9PRVpEAQ+26MidbGSmPm8RwRzcQsT17EPSmoorH3'
:          '+av4Jcj78O/vIp/uzMEkHKAE6/F7VHHSj8HddR0Q3ymcGZfR'
:          'VjwfmOnNn3GuWR+FzhcPmPqiptHcayacT28T8j3Cs0/LQCwo'
:          '6J2iYxP4R58AsobjFegusoJhuq7VNS2evRPcqASvQki+gbkB'
:          'YwETNPt/1A2pT6UErR1zMzUITZRvF5Lp5basO1fk2U4aBSjk'
:          'ji8quL3cDyW7TpI3unxezMcSTNhQJhfpGctKgKN2Amo7/7Sh'
:          'Sev4oXicPSYS+6GkCm9a1Qw3VEchCUA+z5HtTcbQhK6F14YF'
:          'Up+Yn7WgmzwpZCDf5DDiXT9B7U6RdHAHpdb7IqmLVjqZSLnT'
:          'W61zjQ7/G7D3hm9E846uTDZoNMADmLlm7IG2ieXfUtu1US9T'
:          'eNGUHibE9Nv//2jRJGZfQmK3v7ykJJOv1IXjBsDCPpmgWppe'
:          '6sHxR3KVSQKqp+WIqammuJbtqkxZmMHry4oS/9pLhdCXKq8u'
:          'R0R+LDEqCKRxqc5VXdvPvIP+ggwR0RkyBfO9iKZvrWGAKVdz'
:          '31cuocvoO/qemClFMYEFEH7oI+vpkek4s4bCMBqK+5mHQUlD'
:          'pE/oylpy+2/6pWXK31PEYagP04epV1cE50UMy6IQZeQM7+Ol'
:          '74Z+eHfpHNc7OjffQ/HeV0X8BopoDkGEkAAA='
:            }
:           }
:          }
:         }
:        }
:       }
:      }
:     }
:    }
:   }
```

B.5. Full Certificate Example

The following example contains a certificate for Alice; it is essentially a renewal of the certificate that appears in [RFC9216]. Of course, the serial number and issue dates are different. In addition, Alice's certificate now has a logotype extension. The extension contains URLs for two community logotype images, both at fictional URLs. The extension also contains URLs for two subject logotype images, both at fictional URLs. An implementation would display at most three of these images, both of the community logotype images and one of the subject logotype images. Direct addressing is used for all of the images, and the images are hashed by SHA-256.

```
-----BEGIN CERTIFICATE-----
MIIFnTCCBIWgAwIBAgITN0EFee11f0Kpolw69Phqzpqx1zANBgkqhkiG9w0BAQ0F
ADBVMQ0wCwYDVQQKEwRJRVRGMREwDwYDVQQLEwhMQU1QUyBXRzExMC8GA1UEAxMo
U2FtcGxlIExBTVBTIFJTQSBDZXJ0aWZpY2F0aW9uIEF1dGhvcml0eTAgFw0yMjA2
MTUxODE4MThaGA8yMDUyMDkyNzA2NTQxOFowOzENMAsGA1UEChMESUVURjERMA8G
A1UECxMITEFNUFMgV0cxFzAVBgNVBAMTDkFsaWNlIExvdmVsYWNlMIIBIjANBgkq
hkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAtPSJ6Fg4Fj5Nmn9PkrYo0jTkfCv4TfA/
pdO/KLpZbJOAEr0sI7AjaO7B1GuMUFJeSTulamNfCwDcDkY63PQWl+DILs7GxVwX
urhYdZlaV5hcUqVAckPvedDBc/3rz4D/esFfs+E7QMFtmd+K04s+A8TCNO12DRVB
DpbP4JFD9hsc8prDtpGmFk7rd0q8gqnhxBW2RZAeLqzJOMayCQtws1q7ktkNBR2w
ZX5ICjecF1YJFhX4jrnHwp/iELGqqaNXd3/Y0pG7QFecN7836IPPdfTMSiPR+peC
rhJZwLSewbWXLJe3VMvbvQjoBMpEYlaJBUIKkO1zQ1Pq90njlsJLOwIDAQABo4IC
fDCCAngwDAYDVR0TAQH/BAIwADAXBgNVHSAEEDAOMAwGCmCGSAFlAwIBMAEwHgYD
VR0RBBcwFYETYWxpY2VAc21pbWUuZXhhbXBsZTATBgNVHSUEDDAKBggrBgEFBQcD
BDAOBgNVHQ8BAf8EBAMCBsAwHQYDVR0OBBYEFLv2zLItHQYSHJeuKWqQENMgZmZz
MB8GA1UdIwQYMBaAFJEwjnwHFwyn8QkoZTYaZxxodvRZMIIByAYIKwYBBQUHAQwE
ggG6MIIBtqCB3zCB3KBtMGswaRYKaW1hZ2UvanBlZzAxMC8wCwYJYIZIAWUDBAIB
BCCv/BAWRstWJbSZfeWJPq46hG9aAtOC1tqO1O74fL0d7TAoFiZodHRwOi8vd3d3
LmV4YW1wbGUubmV0L2ltYWdlcy9sb2dvLmpwZ6BrMGkwZxYJaW1hZ2UvZ2lmMDEw
LzALBglghkgBZQMEAgEEIIiQgYGt+2auL2bQSaBNjqDsTqhkQjhbNkq/LIvS6elm
MCcWJWh0dHA6Ly93d3cuZXhhbXBsZS5vcmcvbG9nby1pbWFnZS5naWaigdGggc4w
gcswYxYJaW1hZ2UvZ2lmMDEwLzALBglghkgBZQMEAgEEIGpYUC5ZZ/nd0Yr+vQ2x
/mClExvfD7K+8LVzRVC6G78ZMCMWIWh0dHA6Ly93d3cuc21pbWUuZXhhbXBsZS9s
b2dvLmdpZjBkFgppbWFnZS9qcGVnMDEwLzALBglghkgBZQMEAgEEIL3Le3VybYwb
M6Qs3qx5ctpK2fJ5hApYWGrOLwKA6telMCMWIWh0dHA6Ly93d3cuc21pbWUuZXhh
bXBsZS9sb2dvLmpwZzANBgkqhkiG9w0BAQ0FAAOCAQEAqwgkXOqK9JDy3ZCyC+Zu
xXX+SaPc7LUEruUif4KFvFUoMOdWyelUeDxZpgOA/6uMdavtAWy31/ObDtJ3CV1U
RXHXUC84ActoNaCAZIozlM0RWtquV5QMFcsLWl4zT/znfYZF8nf9wX3xap6XJ0i4
w0a5MnHGoCdb8hnjVZ7qoKBiQyAmVsW7KZDvQf3nYkRCrwaHb5zdUNB2uf0MhCRh
6sy4FuSJogrOTOd1yf1l+/FF9r8qD35gGQm9NRYsT04TZ2bf0z5+kwmukrG701sJ
TiXiWMnwp/UuoZRc7xjCCxmUUCbAdufC1FX7fdbjfHizuPQO78OAq/KhkVZuy/Qv
Aw==
-----END CERTIFICATE-----
```

The following displays the logotype extension from Alice's certificate. The values on the left are the ASN.1 tag (in hexadecimal) and the length (in decimal).

```
30 438: SEQUENCE {
A0 223:  [0] {
30 220:   SEQUENCE {
A0 109:    [0] {
30 107:     SEQUENCE {
30 105:      SEQUENCE {
16  10:       IA5String 'image/jpeg'
30  49:       SEQUENCE {
30  47:        SEQUENCE {
30  11:         SEQUENCE {
06   9:          OBJECT IDENTIFIER sha-256 (2 16 840 1 101 3 4 2 1)
      :           }
04  32:         OCTET STRING
      :          AF FC 10 16 46 CB 56 25 B4 99 7D E5 89 3E AE 3A
      :          84 6F 5A 02 D3 82 D6 DA 8E D4 EE F8 7C BD 1D ED
      :          }
      :         }
30  40:       SEQUENCE {
16  38:        IA5String 'http://www.example.net/images/logo.jpg'
      :         }
      :        }
      :       }
      :      }
A0 107:    [0] {
30 105:     SEQUENCE {
30 103:      SEQUENCE {
16   9:       IA5String 'image/gif'
30  49:       SEQUENCE {
30  47:        SEQUENCE {
30  11:         SEQUENCE {
06   9:          OBJECT IDENTIFIER sha-256 (2 16 840 1 101 3 4 2 1)
      :           }
04  32:         OCTET STRING
      :          88 90 81 81 AD FB 66 AE 2F 66 D0 49 A0 4D 8E A0
      :          EC 4E A8 64 42 38 5B 36 4A BF 2C 8B D2 E9 E9 66
      :          }
      :         }
30  39:       SEQUENCE {
16  37:        IA5String 'http://www.example.org/logo-image.gif'
      :         }
      :        }
      :       }
      :      }
      :     }
      :    }
A2 209:  [2] {
A0 206:   [0] {
30 203:    SEQUENCE {
30  99:     SEQUENCE {
```

```
16   9:      IA5String 'image/gif'
30  49:      SEQUENCE {
30  47:       SEQUENCE {
30  11:        SEQUENCE {
06   9:         OBJECT IDENTIFIER sha-256 (2 16 840 1 101 3 4 2 1)
      :          }
04  32:        OCTET STRING
      :         6A 58 50 2E 59 67 F9 DD D1 8A FE BD 0D B1 FE 60
      :         A5 13 1B DF 0F B2 BE F0 B5 73 45 50 BA 1B BF 19
      :         }
      :        }
30  35:      SEQUENCE {
16  33:       IA5String 'http://www.smime.example/logo.gif'
      :        }
      :       }
30 100:     SEQUENCE {
16  10:      IA5String 'image/jpeg'
30  49:      SEQUENCE {
30  47:       SEQUENCE {
30  11:        SEQUENCE {
06   9:         OBJECT IDENTIFIER sha-256 (2 16 840 1 101 3 4 2 1)
      :          }
04  32:        OCTET STRING
      :         BD CB 7B 75 72 6D 8C 1B 33 A4 2C DE AC 79 72 DA
      :         4A D9 F2 79 84 0A 58 58 6A CE 2F 02 80 EA D7 A5
      :         }
      :        }
30  35:      SEQUENCE {
16  33:       IA5String 'http://www.smime.example/logo.jpg'
      :        }
      :       }
      :      }
      :     }
      :    }
      :   }
```

Appendix C. Changes Since RFC 3709 and RFC 6170

This appendix summarizes the changes since RFC 3709. The changes are:

- Combine RFC 3709 and RFC 6170 into one document, and encourage implementers to support the "data" URI scheme (data:...) that was originally specified in RFC 6170. Merging RFC 3709 and RFC 6170 lead to many editoral changes throughout the document.

- Drop SHA-1 as the mandatory-to-implement hash algorithm, and encourage use of the one-way hash function that is employed by the certificate signature algorithm.

- RFC 3709 required client applications to support both direct and indirect addressing. This requirement is changed to **SHOULD** support both direct and indirect addressing to allow implementations to be more privacy preserving.

- Update the reference for language tags to be RFC 5646 instead of the now obsolete RFC 3066.

- Update the reference for the URI Generic Syntax to be RFC 3986 instead of the now obsolete RFC 2396.

- Update the reference for the application/pdf media type to be RFC 8118 instead of the now obsolete RFC 3778.

- No longer require support for the FTP scheme (ftp://...) URI.

- Require support for the HTTP scheme (http://...) URI and the HTTPS scheme (https://...) URI.

- Require support for the compressed SVG image format with the image/svg+xml+gzip media type.

- Media types **MUST** follow the ABNF [RFC5234] that is provided in Section 4.2 of [RFC6838]. This change resolves Errata ID 2679.

- Remove the requirement that the LogotypeData file name have a file extension of ".LTD". This change resolves Errata ID 2325.

- Encourage, instead of requiring, each logotype to be represented by at least one image.

- Encourage the inclusion of text-based audio data suitable for processing by a text-to-speech software using the MIME type of "text/plain;charset=UTF-8".

- Require that the logotype extension not contain more than one certificate image logotype.

- Privacy-related topics that were previously discussed in the Security Considerations section are now covered in a separate Privacy Considerations section. Additional topics are covered in both sections.

- Provide ASN.1 modules for both the older syntax [OLD-ASN1] and the most recent ASN.1 syntax [NEW-ASN1].

- Provide additional references.

- Provide additional examples.

- Several editorial changes to improve clarity.

Authors' Addresses

Stefan Santesson
IDsec Solutions AB
Forskningsbyn Ideon
SE-223 70 Lund
Sweden

Email: sts@aaa-sec.com

Russ Housley
Vigil Security, LLC
516 Dranesville Road
Herndon, VA, 20170
United States of America

Email: housley@vigilsec.com

Trevor Freeman
Amazon Web Services
1918 8th Ave
Seattle, WA, 98101
United States of America

Email: frtrevor@amazon.com

Leonard Rosenthol
Adobe
345 Park Avenue
San Jose, CA, 95110
United States of America

Email: lrosenth@adobe.com