SIDR
Internet-Draft
Obsoletes: 6487 (if approved)
Intended status: Standards Track
Expires: September 10, 2015

G. Huston
G. Michaelson
APNIC
R. Loomans
Suncorp
A. Newton
ARIN
R. Hansen
BBN

March 9, 2015

A Profile for X.509 PKIX Resource Certificates
draft-rhansen-sidr-rfc6487bis-00

Abstract

This document defines a standard profile for X.509 certificates for the purpose of supporting validation of assertions of "right-of-use" of Internet Number Resources (INRs). The certificates issued under this profile are used to convey the issuer's authorization of the subject to be regarded as the current holder of a "right-of-use" of the INRs that are described in the certificate. This document contains the normative specification of Certificate and Certificate Revocation List (CRL) syntax in the Resource Public Key Infrastructure (RPKI). This document also specifies profiles for the format of certificate requests and specifies the Relying Party RPKI certificate path validation procedure.

This document obsoletes RFC 6487.

Status of This Memo

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

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

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

This Internet-Draft will expire on September 10, 2015.

Table of Contents

1. Introduction

This document defines a standard profile for X.509 certificates [X.509] for use in the context of certification of Internet Number Resources (INRs), i.e., IP Addresses and Autonomous System (AS) numbers. Such certificates are termed "resource certificates". A resource certificate is a certificate that conforms to the PKIX profile [RFC5280], and that conforms to the constraints specified in this profile. A resource certificate attests that the issuer has granted the subject a "right-of-use" for a listed set of IP addresses and/or Autonomous System numbers.

This document is referenced by Section 7 of the "Certificate Policy (CP) for the Resource Public Key Infrastructure (RPKI)" [RFC6484]. It is an integral part of that policy and the normative specification for certificate and Certificate Revocation List (CRL) syntax used in the RPKI. The document also specifies profiles for the format of certificate requests, and the relying party (RP) RPKI certificate path validation procedure.

Resource certificates are to be used in a manner that is consistent with the RPKI Certificate Policy (CP) [RFC6484]. They are issued by entities that assign and/or allocate public INRs, and thus the RPKI is aligned with the public INR distribution function. When an INR is allocated or assigned by a number registry to an entity, this allocation can be described by an associated resource certificate. This certificate is issued by the number registry, and it binds the certificate subject's key to the INRs enumerated in the certificate.

One or two critical extensions, the IP Address Delegation or AS Identifier Delegation Extensions [RFC3779], enumerate the INRs that were allocated or assigned by the issuer to the subject.

Relying party (RP) validation of a resource certificate is performed in the manner specified in Section 7.1. This validation procedure differs from that described in Section 6 of [RFC5280], such that:

- o additional validation processing imposed by the INR extensions is required,
- o a confirmation of a public key match between the CRL issuer and the resource certificate issuer is required, and
- o the resource certificate is required to conform to this profile.

This profile defines those fields that are used in a resource certificate that MUST be present for the certificate to be valid. Any extensions not explicitly mentioned MUST be absent. The same applies to the CRLs used in the RPKI, that are also profiled in this document. A Certification Authority (CA) conforming to the RPKI CP MUST issue certificates and CRLs consistent with this profile.

1.1. Terminology

It is assumed that the reader is familiar with the terms and concepts described in "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile" [RFC5280], and "X.509 Extensions for IP Addresses and AS Identifiers" [RFC3779].

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. Describing Resources in Certificates

The framework for describing an association between the subject of a certificate and the INRs currently under the subject’s control is described in [RFC3779]. This profile further requires that:

- o Every resource certificate MUST contain either the IP Address Delegation or the Autonomous System Identifier Delegation extension, or both.
- o These extensions MUST be marked as critical.
- o The sorted canonical format describing INRs, with maximal spanning ranges and maximal spanning prefix masks, as defined in [RFC3779],

MUST be used for the resource extension field, except where the "inherit" construct is used instead.

When validating a resource certificate, an RP MUST verify that the INRs described in the issuer's resource certificate encompass the INRs of the resource certificate being validated. In this context, "encompass" allows for the issuer's INRs to be the same as, or a strict superset of, the subject's INRs.

3. End-Entity (EE) Certificates and Signing Functions in the RPKI

As noted in [RFC6480], the primary function of end-entity (EE) certificates in the RPKI is the verification of signed objects that relate to the usage of the INRs described in the certificate, e.g., Route Origin Authorizations (ROAs) and manifests.

The private key associated with an EE certificate is used to sign a single RPKI signed object, i.e., the EE certificate is used to validate only one object. The EE certificate is embedded in the object as part of a Cryptographic Message Syntax (CMS) signed-data structure [RFC6488]. Because of the one-to-one relationship between the EE certificate and the signed object, revocation of the certificate effectively revokes the corresponding signed object.

An EE certificate may be used to validate a sequence of signed objects, where each signed object in the sequence overwrites the previous instance of the signed object in the repository publication point, such that only one instance of the signed object is published at any point in time (e.g., an EE certificate MAY be used to sign a sequence of manifests [RFC6486]). Such EE certificates are termed "sequential use" EE certificates.

EE certificates used to validate only one instance of a signed object, and are not used thereafter or in any other validation context, are termed "one-time-use" EE certificates.

4. Resource Certificates

A resource certificate is a valid X.509 public key certificate, consistent with the PKIX profile [RFC5280], containing the fields listed in this section. Only the differences from [RFC5280] are noted below.

Unless specifically noted as being OPTIONAL, all the fields listed here MUST be present, and any other fields MUST NOT appear in a conforming resource certificate. Where a field value is specified here, this value MUST be used in conforming resource certificates.

4.1. Version

As resource certificates are X.509 version 3 certificates, the version MUST be 3 (i.e., the value of this field is 2).

RPs need not process version 1 or version 2 certificates (in contrast to [RFC5280]).

4.2. Serial Number

The serial number value is a positive integer that is unique for each certificate issued by a given CA.

4.3. Signature Algorithm

The algorithm used in this profile is specified in [RFC6485].

4.4. Issuer

The value of this field is a valid X.501 distinguished name.

An issuer name MUST contain one instance of the CommonName attribute and MAY contain one instance of the serialNumber attribute. If both attributes are present, it is RECOMMENDED that they appear as a set. The CommonName attribute MUST be encoded using the ASN.1 type PrintableString [X.680]. Issuer names are not intended to be descriptive of the identity of issuer.

The RPKI does not rely on issuer names being globally unique, for reasons of security. However, it is RECOMMENDED that issuer names be generated in a fashion that minimizes the likelihood of collisions. See Section 8 for (non-normative) suggested name-generation mechanisms that fulfill this recommendation.

4.5. Subject

The value of this field is a valid X.501 distinguished name [RFC4514], and is subject to the same constraints as the issuer name.

In the RPKI, the subject name is determined by the issuer, not proposed by the subject [RFC6481]. Each distinct subordinate CA and EE certified by the issuer MUST be identified using a subject name that is unique per issuer. In this context, "distinct" is defined as an entity and a given public key. An issuer SHOULD use a different subject name if the subject’s key pair has changed (i.e., when the CA issues a certificate as part of re-keying the subject.) Subject names are not intended to be descriptive of the identity of subject.

4.6. Validity

The certificate validity period is represented as a SEQUENCE of two dates: the date on which the certificate validity period begins (notBefore) and the date on which the certificate validity period ends (notAfter).

While a CA is typically advised against issuing a certificate with a validity period that spans a greater period of time than the validity period of the CA's certificate that will be used to validate the issued certificate, in the context of this profile, a CA MAY have valid grounds to issue a subordinate certificate with a validity period that exceeds the validity period of the CA's certificate.

4.6.1. notBefore

The "notBefore" time SHOULD be no earlier than the time of certificate generation.

In the RPKI, it is valid for a certificate to have a value for this field that pre-dates the same field value in any superior certificate. Relying Parties SHOULD NOT attempt to infer from this time information that a certificate was valid at a time in the past, or that it will be valid at a time in the future, as the scope of an RP's test of validity of a certificate refers specifically to validity at the current time.

4.6.2. notAfter

The "notAfter" time represents the anticipated lifetime of the current resource allocation or assignment arrangement between the issuer and the subject.

It is valid for a certificate to have a value for this field that post-dates the same field value in any superior certificate. The same caveats apply to RP's assumptions relating to the certificate's validity at any time other than the current time.

4.7. Subject Public Key Info

The algorithm used in this profile is specified in [RFC6485].

4.8. Resource Certificate Extensions

The following X.509 v3 extensions MUST be present in a conforming resource certificate, except where explicitly noted otherwise. Each extension in a resource certificate is designated as either critical or non-critical. A certificate-using system MUST reject the

certificate if it encounters an extension not explicitly mentioned in this document. This is in contrast to [RFC5280] which allows non-critical extensions to be ignored.

4.8.1. Basic Constraints

The Basic Constraints extension field is a critical extension in the resource certificate profile, and MUST be present when the subject is a CA, and MUST NOT be present otherwise.

The issuer determines whether the "cA" boolean is set.

The Path Length Constraint is not specified for RPKI certificates, and MUST NOT be present.

4.8.2. Subject Key Identifier

This extension MUST appear in all resource certificates. This extension is non-critical.

The Key Identifier used for resource certificates is the 160-bit SHA-1 hash of the value of the DER-encoded ASN.1 bit string of the Subject Public Key, as described in Section 4.2.1.2 of [RFC5280].

4.8.3. Authority Key Identifier

This extension MUST appear in all resource certificates, with the exception of a CA who issues a "self-signed" certificate. In a self-signed certificate, a CA MAY include this extension, and set it equal to the Subject Key Identifier. The authorityCertIssuer and authorityCertSerialNumber fields MUST NOT be present. This extension is non-critical.

The Key Identifier used for resource certificates is the 160-bit SHA-1 hash of the value of the DER-encoded ASN.1 bit string of the issuer’s public key, as described in Section 4.2.1.1 of [RFC5280].

4.8.4. Key Usage

This extension is a critical extension and MUST be present.

In certificates issued to certification authorities only, the keyCertSign and CRLSign bits are set to TRUE, and these MUST be the only bits set to TRUE.

In EE certificates, the digitalSignature bit MUST be set to TRUE and MUST be the only bit set to TRUE.

4.8.5. Extended Key Usage

The Extended Key Usage (EKU) extension MUST NOT appear in any CA certificate in the RPKI. This extension also MUST NOT appear in EE certificates used to verify RPKI objects (e.g., ROAs or manifests. The extension MUST NOT be marked critical.

The EKU extension MAY appear in EE certificates issued to routers or other devices. Permitted values for the EKU OIDs will be specified in Standards Track RFCs issued by other IETF working groups that adopt the RPKI profile and that identify application-specific requirements that motivate the use of such EKUs.

4.8.6. CRL Distribution Points

This extension MUST be present, except in "self-signed" certificates, and it is non-critical. In a self-signed certificate, this extension MUST be omitted.

In this profile, the scope of the CRL is specified to be all certificates issued by this CA issuer.

The CRL Distribution Points (CRLDP) extension identifies the location(s) of the CRL(s) associated with certificates issued by this issuer. The RPKI uses the URI [RFC3986] form of object identification. The preferred URI access mechanism is a single rsync URI ("rsync://") [RFC5781] that references a single inclusive CRL for each issuer.

In this profile, the certificate issuer is also the CRL issuer, implying that the CRLIssuer field MUST be omitted, and the distributionPoint field MUST be present. The Reasons field MUST be omitted.

The distributionPoint MUST contain the fullName field, and MUST NOT contain a nameRelativeToCRLIssuer. The form of the generalName MUST be of type URI.

The sequence of distributionPoint values MUST contain only a single DistributionPoint. The DistributionPoint MAY contain more than one URI value. An rsync URI [RFC5781] MUST be present in the DistributionPoint and MUST reference the most recent instance of this issuer’s CRL. Other access form URIs MAY be used in addition to the rsync URI, representing alternate access mechanisms for this CRL.

4.8.7. Authority Information Access

In the context of the RPKI, this extension identifies the publication point of the certificate of the issuer of the certificate in which the extension appears. In this profile, a single reference to the publication point of the immediate superior certificate MUST be present, except for a "self-signed" certificate, in which case the extension MUST be omitted. This extension is non-critical.

This profile uses a URI form of object identification. The preferred URI access mechanisms is "rsync", and an rsync URI [RFC5781] MUST be specified with an accessMethod value of id-ad-caIssuers. The URI MUST reference the point of publication of the certificate where this Issuer is the subject (the issuer's immediate superior certificate). Other accessMethod URIs referencing the same object MAY also be included in the value sequence of this extension.

A CA MUST use a persistent URL name scheme for CA certificates that it issues [RFC6481]. This implies that a reissued certificate overwrites a previously issued certificate (to the same subject) in the publication repository. In this way, certificates subordinate to the reissued (CA) certificate can maintain a constant Authority Information Access (AIA) extension pointer and thus need not be reissued when the parent certificate is reissued.

4.8.8. Subject Information Access

In the context of the RPKI, this Subject Information Access (SIA) extension identifies the publication point of products signed by the subject of the certificate.

4.8.8.1. SIA for CA Certificates

This extension MUST be present and MUST be marked non-critical.

This extension MUST have an instance of an accessMethod of id-ad-caRepository, with an accessLocation form of a URI that MUST specify an rsync URI [RFC5781]. This URI points to the directory containing all published material issued by this CA, i.e., all valid CA certificates, published EE certificates, the current CRL, manifest, and signed objects validated via EE certificates that have been issued by this CA [RFC6481]. Other accessDescription elements with an accessMethod of id-ad-caRepository MAY be present. In such cases, the accessLocation values describe alternate supported URI access mechanisms for the same directory. The ordering of URIs in this accessDescription sequence reflect the CA's relative preferences for access methods to be used by RPs, with the first element of the sequence being the most preferred by the CA.

This extension MUST have an instance of an AccessDescription with an accessMethod of id-ad-rpkiManifest,

```
id-ad OBJECT IDENTIFIER ::= { id-pkix 48 }

id-ad-rpkiManifest OBJECT IDENTIFIER ::= { id-ad 10 }
```

with an rsync URI [RFC5781] form of accessLocation. The URI points to the CA's manifest of published objects [RFC6486] as an object URL. Other accessDescription elements MAY exist for the id-ad-rpkiManifest accessMethod, where the accessLocation value indicates alternate access mechanisms for the same manifest object.

4.8.8.2. SIA for EE Certificates

This extension MUST be present and MUST be marked non-critical.

This extension MUST have an instance of an accessMethod of id-ad-signedObject,

```
id-ad-signedObject OBJECT IDENTIFIER ::= { id-ad 11 }
```

with an accessLocation form of a URI that MUST include an rsync URI [RFC5781]. This URI points to the signed object that is verified using this EE certificate [RFC6481]. Other accessDescription elements may exist for the id-ad-signedObject accessMethod, where the accessLocation value indicates alternate URI access mechanisms for the same object, ordered in terms of the EE's relative preference for supported access mechanisms.

Other AccessMethods MUST NOT be used for an EE certificates's SIA.

4.8.9. Certificate Policies

This extension MUST be present and MUST be marked critical. It MUST include exactly one policy, as specified in the RPKI Certificate Policy (CP) [RFC6484]. Exactly one policy qualifier MAY be included. If a policy qualifier is included, the policyQualifierId MUST be the Certification Practice Statement (CPS) pointer qualifier type (id-qt-cps).

As noted in [RFC5280], Section 4.2.1.4: "Optional qualifiers, which MAY be present, are not expected to change the definition of the policy." In this case, any optional policy qualifier that MAY be present in a resource certificate MUST NOT change the definition of the RPKI CP [RFC6484].

4.8.10. IP Resources

Either the IP Resources extension, or the AS Resources extension, or both, MUST be present in all RPKI certificates, and if present, MUST be marked critical.

This extension contains the list of IP address resources as per [RFC3779]. The value may specify the "inherit" element for a particular Address Family Identifier (AFI) value. In the context of resource certificates describing public number resources for use in the public Internet, the Subsequent AFI (SAFI) value MUST NOT be used.

This extension MUST either specify a non-empty set of IP address records, or use the "inherit" setting to indicate that the IP address resource set of this certificate is inherited from that of the certificate’s issuer.

4.8.11. AS Resources

Either the AS Resources extension, or the IP Resources extension, or both, MUST be present in all RPKI certificates, and if present, MUST be marked critical.

This extension contains the list of AS number resources as per [RFC3779], or it may specify the "inherit" element. Routing Domain Identifier (RDI) values are NOT supported in this profile and MUST NOT be used.

This extension MUST either specify a non-empty set of AS number records, or use the "inherit" setting to indicate that the AS number resource set of this certificate is inherited from that of the certificate’s issuer.

5. Resource Certificate Revocation Lists

Each CA MUST issue a version 2 CRL that is consistent with [RFC5280]. RPs are NOT required to process version 1 CRLs (in contrast to [RFC5280]). The CRL issuer is the CA. CRLs conforming to this profile MUST NOT include Indirect or Delta CRLs. The scope of each CRL MUST be all certificates issued by this CA.

The issuer name is as in Section 4.4 above.

Where two or more CRLs are issued by the same CA, the CRL with the highest value of the "CRL Number" field supersedes all other CRLs issued by this CA.

The algorithm used in CRLs issued under this profile is specified in [RFC6485].

The contents of the CRL are a list of all non-expired certificates that have been revoked by the CA.

An RPKI CA MUST include the two extensions, Authority Key Identifier and CRL Number, in every CRL that it issues. The Authority Key Identifier extension MUST follow the same restrictions as in Section 4.8.3 above. RPs MUST be prepared to process CRLs with these extensions. No other CRL extensions are allowed. The extensions mentioned above MUST NOT appear more than once each.

For each revoked resource certificate, only the two fields, Serial Number and Revocation Date, MUST be present, and all other fields MUST NOT be present. No CRL entry extensions are supported in this profile, and CRL entry extensions MUST NOT be present in a CRL.

6. Resource Certificate Requests

A resource certificate request MAY use either of PKCS#10 or Certificate Request Message Format (CRMF). A CA MUST support certificate issuance in PKCS#10 and a CA MAY support CRMF requests.

Note that there is no certificate response defined in this profile. For CA certificate requests, the CA places the resource certificate in the repository, as per [RFC6484]. No response is defined for EE certificate requests.

6.1. PCKS#10 Profile

This profile refines the specification in [RFC2986], as it relates to resource certificates. A Certificate Request Message object, formatted according to PKCS#10, is passed to a CA as the initial step in issuing a certificate.

With the exception of the SubjectPublicKeyinfo and the SIA extension request, the CA is permitted to alter any field in the request when issuing a certificate.

6.1.1. PKCS#10 Resource Certificate Request Template Fields

This profile applies the following additional requirements to fields that MAY appear in a CertificationRequestInfo:

Version

This field is mandatory and MUST have the value 0.

Subject
This field SHOULD be empty (i.e., NULL), in which case the CA MUST generate a subject name that is unique in the context of certificates issued by this CA. This field is allowed to be non-empty only for a re-key/reissuance request, and only if the CA has adopted a policy (in its Certificate Practice Statement (CPS)) that permits reuse of names in these circumstances.

SubjectPublicKeyInfo
This field specifies the subject's public key and the algorithm with which the key is used. The algorithm used in this profile is specified in [RFC6485].

Attributes
[RFC2986] defines the attributes field as key-value pairs where the key is an OID and the value's structure depends on the key.

The only attribute used in this profile is the extensionRequest attribute as defined in [RFC2985]. This attribute contains certificate extensions. The profile for extensions in certificate requests is specified in Section 6.3.

This profile applies the following additional constraint to fields that MAY appear in a CertificationRequest Object:

signatureAlgorithm
The signatureAlgorithm value is specified in [RFC6485].

6.2. CRMF Profile

This profile refines the Certificate Request Message Format (CRMF) specification in [RFC4211], as it relates to resource certificates. A Certificate Request Message object, formatted according to the CRMF, is passed to a CA as the initial step in certificate issuance.

With the exception of the SubjectPublicKeyinfo and the SIA extension request, the CA is permitted to alter any requested field when issuing the certificate.

6.2.1. CRMF Resource Certificate Request Template Fields

This profile applies the following additional requirements to fields that may appear in a Certificate Request Template:

version
This field SHOULD be omitted. If present, it MUST specify a request for a version 3 Certificate.

serialNumber
This field MUST be omitted.

signingAlgorithm
This field MUST be omitted.

issuer
This MUST be omitted in this profile.

Validity
This field MAY be omitted. If omitted, the CA will issue a Certificate with Validity dates as determined by the CA. If specified, then the CA MAY override the requested values with dates as determined by the CA.

Subject
This field MAY be omitted. If present, the value of this field SHOULD be empty (i.e., NULL), in which case the CA MUST generate a subject name that is unique in the context of certificates issued by this CA. This field is allowed to be non-empty only for a re-key/reissuance request, and only if the CA has adopted a policy (in its CPS) that permits the reuse of names in these circumstances.

PublicKey
This field MUST be present.

extensions
The profile for extensions in certificate requests is specified in Section 6.3.

6.2.2. Resource Certificate Request Control Fields

The following control fields are supported in this profile:

Authenticator Control
The intended model of authentication of the subject is a "long term" model, and the guidance offered in [RFC4211] is that the Authenticator Control field be used.

6.3. Certificate Extension Attributes in Certificate Requests

The following extensions MAY appear in a PKCS#10 or CRMF Certificate Request. Any other extensions MUST NOT appear in a Certificate Request. This profile places the following additional constraints on these extensions:

BasicConstraints
: If this is omitted, then the CA will issue an EE certificate (hence no BasicConstraints extension will be included).

  The pathLengthConstraint is not supported in this profile, and this field MUST be omitted.

  The CA MAY honor the cA boolean if set to TRUE (CA Certificate Request). If this bit is set, then it indicates that the subject is requesting a CA certificate.

  The CA MUST honor the cA bit if set to FALSE (EE Certificate Request), in which case the corresponding EE certificate will not contain a Basic Constraints extension.

KeyUsage
: The CA MAY honor KeyUsage extensions of keyCertSign and cRLSign if present, as long as this is consistent with the BasicConstraints SubjectType sub-field, when specified.

ExtendedKeyUsage
: The CA MAY honor ExtendedKeyUsage extensions in requests for EE certificates that are issued to routers or other devices, consistent with values specified in Standards Track RFCs that adopt this profile and that identify application-specific requirements that motivate the use of such EKUs.

SubjectInformationAccess
: This field MUST be present, and the field value SHOULD be honored by the CA if it conforms to the requirements set forth in Section 4.8.8. If the CA is unable to honor the requested value for this field, then the CA MUST reject the Certificate Request.

7. Resource Certificate Validation

This section describes the resource certificate validation procedure. This refines the generic procedure described in Section 6 of [RFC5280].

7.1. Resource Extension Validation

The IP Resources and AS Resources extensions [RFC3779] define critical extensions for INRs. These are ASN.1 encoded representations of the IPv4 and IPv6 address range and an AS number set.

Valid resource certificates MUST have a valid IP address and/or AS number resource extension. In order to validate a resource certificate, the resource extension MUST also be validated. This validation process relies on definitions of comparison of resource sets:

more specific
: Given two contiguous IP address ranges or two contiguous AS number ranges, A and B, A is "more specific" than B if range B includes all IP addresses or AS numbers described by range A, and if range B is larger than range A.

equal
: Given two contiguous IP address ranges or two contiguous AS number ranges, A and B, A is "equal" to B if range A describes precisely the same collection of IP addresses or AS numbers described by range B. The definition of "inheritance" in [RFC3779] is equivalent to this "equality" comparison.

encompass
: Given two IP address and AS number sets, X and Y, X "encompasses" Y if, for every contiguous range of IP addresses or AS numbers elements in set Y, the range element is either "more specific" than or "equal" to a contiguous range element within the set X.

Validation of a certificate's resource extension in the context of a certification path (see Section 7.2 entails that for every adjacent pair of certificates in the certification path (certificates 'x' and 'x + 1'), the number resources described in certificate 'x' "encompass" the number resources described in certificate 'x + 1', and the resources described in the trust anchor information "encompass" the resources described in the first certificate in the certification path.

7.2. Resource Certification Path Validation

Validation of signed resource data using a target resource certificate consists of verifying that the digital signature of the signed resource data is valid, using the public key of the target resource certificate, and also validating the resource certificate in the context of the RPKI, using the path validation process. This path validation process verifies, among other things, that a prospective certification path (a sequence of n certificates) satisfies the following conditions:

1. for all 'x' in {1, ..., n-1}, the subject of certificate 'x' is the issuer of certificate ('x' + 1);
2. certificate '1' is issued by a trust anchor;
3. certificate 'n' is the certificate to be validated; and
4. for all 'x' in {1, ..., n}, certificate 'x' is valid.

Certificate validation entails verifying that all of the following conditions hold, in addition to the certification path validation criteria specified in Section 6 of [RFC5280]:

1. The certificate can be verified using the issuer's public key and the signature algorithm
2. The current time lies within the certificate's Validity From and To values.
3. The certificate contains all fields that MUST be present, as defined by this specification, and contains values for selected fields that are defined as allowable values by this specification.
4. No field, or field value, that this specification defines as MUST NOT be present is used in the certificate.
5. The issuer has not revoked the certificate. A revoked certificate is identified by the certificate's serial number being listed on the issuer's current CRL, as identified by the CRLDP of the certificate, the CRL is itself valid, and the public key used to verify the signature on the CRL is the same public key used to verify the certificate itself.

6. The resource extension data is "encompassed" by the resource extension data contained in a valid certificate where this issuer is the subject (the previous certificate in the context of the ordered sequence defined by the certification path).

7. The certification path originates with a certificate issued by a trust anchor, and there exists a signing chain across the certification path where the subject of Certificate 'x' in the certification path matches the issuer in Certificate 'x + 1' in the certification path, and the public key in Certificate 'x' can verify the signature value in Certificate 'x+1'.

A certificate validation algorithm MAY perform these tests in any chosen order.

Certificates and CRLs used in this process MAY be found in a locally maintained cache, maintained by a regular synchronization across the distributed publication repository structure [RFC6481].

There exists the possibility of encountering certificate paths that are arbitrarily long, or attempting to generate paths with loops as means of creating a potential denial-of-service (DOS) attack on an RP. An RP executing this procedure MAY apply further heuristics to guide the certification path validation process to a halt in order to avoid some of the issues associated with attempts to validate such malformed certification path structures. Implementations of resource certificate validation MAY halt with a validation failure if the certification path length exceeds a locally defined configuration parameter.

8. Design Notes

The following notes provide some additional commentary on the considerations that lie behind some of the design choices that were made in the design of this certificate profile. These notes are non-normative, i.e., this section of the document does not constitute a formal part of the profile specification, and the interpretation of key words as defined in RFC 2119 are not applicable in this section of the document.

Certificate Extensions:

This profile does not permit the use of any other critical or non-critical extensions. The rationale for this restriction is that the resource certificate profile is intended for a specific defined use. In this context, having certificates with additional non-critical extensions that RPs may see as valid certificates without understanding the extensions is inappropriate, because if the RP were in a position to

understand the extensions, it would contradict or qualify this original judgment of validity in some way. This profile takes the position of minimalism over extensibility. The specific goal for the associated RPKI is to precisely match the INR allocation structure through an aligned certificate structure that describes the allocation and its context within the INR distribution hierarchy. The profile defines a resource certificate that is structured to meet these requirements.

Certification Authorities and Key Values:
: This profile uses a definition of an instance of a CA as a combination of a named entity and a key pair. Within this definition, a CA instance cannot rollover a key pair. However, the entity can generate a new instance of a CA with a new key pair and roll over all the signed subordinate products to the new CA [RFC6489].

  This has a number of implications in terms of subject name management, CRL Scope, and repository publication point management.

CRL Scope and Key Values:
: For CRL Scope, this profile specifies that a CA issues a single CRL at a time, and the scope of the CRL is all certificates issued by this CA. Because the CA instance is bound to a single key pair, this implies that the CA's public key, the key used to validate the CA's CRL, and the key used to validate the certificates revoked by that CRL are all the same key value.

Repository Publication Point:
: The definition of a CA affects the design of the repository publication system. In order to minimize the amount of forced re-certification on key rollover events, a repository publication regime that uses the same repository publication point for all CA instances that refers to the same entity, but with different key values, will minimize the extent of re-generation of certificates to only immediate subordinate certificates. This is described in [RFC6489].

Subject Name:
: This profile specifies that subject names must be unique per issuer, and does not specify that subject names must be globally unique (in terms of assured uniqueness). This is due to the nature of the RPKI as a distributed PKI, implying that there is no ready ability for certification authorities to coordinate a simple RPKI-wide unique name space without resorting to additional critical external dependencies. CAs

are advised to use subject name generation procedures that minimize the potential for name clashes.

One way to achieve this is for a CA to use a subject name practice that uses the CommonName component of the Distinguished Name as a constant value for any given entity that is the subject of CA-issued certificates, and set the serialNumber component of the Distinguished Name to a value that is derived from the hash of the subject public key value.

If the CA elects not to use the serialNumber component of the DistinguishedName, then it is considered beneficial that a CA generates CommonNames that have themselves a random component that includes significantly more than 40 bits of entropy in the name. Some non-normative recommendations to achieve this include:

1) Hash of the subject public key (encoded as ASCII HEX).
   example: cn="999d99d564de366a29cd8468c45ede1848e2cc14"

2) A Universally Unique IDentifier (UUID) [RFC4122]
   example: cn="6437d442-6fb5-49ba-bbdb-19c260652098"

3) A randomly generated ASCII HEX encoded string of length 20 or greater:
   example: cn="0f8fcc28e3be4869bc5f8fa114db05e1">
   (A string of 20 ASCII HEX digits would have 80-bits of entropy)

4) An internal database key or subscriber ID combined with one of the above
   example: cn="<DBkey1> (6437d442-6fb5-49ba-bbdb-19c2606520980)"
   (The issuing CA may wish to be able to extract the database key or subscriber ID from the commonName. Since only the issuing CA would need to be able to parse the commonName, the database key and the source of entropy (e.g., a UUID) could be separated in any way that the CA wants, as long as it conforms to the rules for PrintableString. The separator could be a space character, parenthesis, hyphen, slash, question mark, etc.

9. Operational Considerations for Profile Agility

This profile requires that relying parties reject certificates or CRLs that do not conform to the profile. (Through the remainder of this section, the term "certificate" is used to refer to both certificates and CRLs.) This includes certificates that contain

extensions that are prohibited, but that are otherwise valid as per [RFC5280]. This means that any change in the profile (e.g., extensions, permitted attributes or optional fields, or field encodings) for certificates used in the RPKI will not be backward compatible. In a general PKI context, this constraint probably would cause serious problems. In the RPKI, several factors minimize the difficulty of effecting changes of this sort.

Note that the RPKI is unique in that every relying party (RP) requires access to every certificate issued by the CAs in this system. An important update of the certificates used in the RPKI must be supported by all CAs and RPs in the system, lest views of the RPKI data differ across RPs. Thus, incremental changes require very careful coordination. It would not be appropriate to introduce a new extension, or authorize use of an extant, standard extension, for a security-relevant purpose on a piecemeal basis.

One might imagine that the "critical" flag in X.509 certificate extensions could be used to ameliorate this problem. However, this solution is not comprehensive and does not address the problem of adding a new, security-critical extension. (This is because such an extension needs to be supported universally, by all CAs and RPs.) Also, while some standard extensions can be marked either critical or non-critical, at the discretion of the issuer, not all have this property, i.e., some standard extensions are always non-critical. Moreover, there is no notion of criticality for attributes within a name or optional fields within a field or an extension. Thus, the critical flag is not a solution to this problem.

In typical PKI deployments, there are few CAs and many RPs. However, in the RPKI, essentially every CA in the RPKI is also an RP. Thus the set of entities that will need to change in order to issue certificates under a new format is the same set of entities that will need to change to accept these new certificates. To the extent that this is literally true, it says that CA/RP coordination for a change is tightly linked anyway. In reality, there is an important exception to this general observation. Small ISPs and holders of provider-independent allocations are expected to use managed CA services, offered by Regional Internet Registries (RIRs) and potentially by wholesale Internet Service Providers (ISPs). This reduces the number of distinct CA implementations that are needed and makes it easier to effect changes for certificate issuance. It seems very likely that these entities also will make use of RP software provided by their managed CA service provider, which reduces the number of distinct RP software implementations. Also note that many small ISPs (and holders of provider-independent allocations) employ default routes, and thus need not perform RP validation of RPKI data, eliminating these entities as RPs.

Widely available PKI RP software does not cache large numbers of certificates, an essential strategy for the RPKI. It does not process manifest or ROA data structures, essential elements of the RPKI repository system. Experience shows that such software deals poorly with revocation status data. Thus, extant RP software is not adequate for the RPKI, although some open source tools (e.g., OpenSSL and cryptlib) can be used as building blocks for an RPKI RP implementation. Thus, it is anticipated that RPs will make use of software that is designed specifically for the RPKI environment and is available from a limited number of open sources. Several RIRs and two companies are providing such software today. Thus it is feasible to coordinate change to this software among the small number of developers/maintainers.

If the resource certificate profile is changed in the future, e.g., by adding a new extension or changing the allowed set of name attributes or encoding of these attributes, the following procedure will be employed to effect deployment in the RPKI. The model is analogous to that described in [RPKI-ALG], but is simpler.

A new document will be issued as an update to this RFC. The CP for the RPKI [RFC6484] will be updated to reference the new certificate profile. The new CP will define a new policy OID for certificates issued under the new certificate profile. The updated CP also will define a timeline for transition to the new certificate (CRL) format. This timeline will define 3 phases and associated dates:

1. At the end of phase 1, all RPKI CAs MUST be capable of issuing certificates under the new profile, if requested by a subject. Any certificate issued under the new format will contain the new policy OID.

2. During phase 2, CAs MUST issue certificates under the new profile, and these certificates MUST coexist with certificates issued under the old format. (CAs will continue to issue certificates under the old OID/format as well.) The old and new certificates MUST be identical, except for the policy OID and any new extensions, encodings, etc. The new certificates, and associated signed objects, will coexist in the RPKI repository system during this phase, analogous to what is required by an algorithm transition for the RPKI [RPKI-ALG]. Relying parties MAY make use of the old or the new certificate formats when processing signed objects retrieved from the RPKI repository system. During this phase, a relying party that elects to process both formats will acquire the same values for all certificate fields that overlap between the old and

new formats. Thus if either certificate format is verifiable, the relying party accepts the data from that certificate. This allows CAs to issue certificates under the new format before all relying parties are prepared to process that format.

3. At the beginning of phase 3, all relying parties MUST be capable of processing certificates under the new format. During this phase, CAs will issue new certificates ONLY under the new format. Certificates issued under the old OID will be replaced with certificates containing the new policy OID. The repository system will no longer require matching old and new certificates under the different formats.

At the end of phase 3, all certificates under the old OID will have been replaced. The resource certificate profile RFC will be replaced to remove support for the old certificate format, and the CP will be replaced to remove reference to the old policy OID and to the old resource certificate profile RFC. The system will have returned to a new, steady state.

10. Security Considerations

The Security Considerations of [RFC5280] and [RFC3779] apply to resource certificates. The Security Considerations of [RFC2986] and [RFC4211] apply to resource certificate certification requests.

A resource certificate PKI cannot in and of itself resolve any forms of ambiguity relating to uniqueness of assertions of rights of use in the event that two or more valid certificates encompass the same resource. If the issuance of resource certificates is aligned to the status of resource allocations and assignments, then the information conveyed in a certificate is no better than the information in the allocation and assignment databases.

This profile requires that the key used to sign an issued certificate be the same key used to sign the CRL that can revoke the certificate, implying that the certification path used to validate the signature on a certificate is the same as that used to validate the signature of the CRL that can revoke the certificate. It is noted that this is a tighter constraint than required in X.509 PKIs, and there may be a risk in using a path validation implementation that is capable of using separate validation paths for a certificate and the corresponding CRL. If there are subject name collisions in the RPKI as a result of CAs not following the guidelines provided here relating to ensuring sufficient entropy in constructing subject names, and this is combined with the situation that an RP uses an implementation of validation path construction that is not in

conformance with this RPKI profile, then it is possible that the subject name collisions can cause an RP to conclude that an otherwise valid certificate has been revoked.

11. IANA Considerations

This document has no actions for IANA.

12. Acknowledgements

The authors would like to particularly acknowledge the valued contribution from Stephen Kent in reviewing this document and proposing numerous sections of text that have been incorporated into the document. The authors also acknowledge the contributions of Sandy Murphy, Robert Kisteleki, Randy Bush, Russ Housley, Ricardo Patara, and Rob Austein in the preparation and subsequent review of this document. The document also reflects review comments received from Roque Gagliano, Sean Turner, and David Cooper.

13. References

13.1. Normative References

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

[RFC2986] Nystrom, M. and B. Kaliski, "PKCS #10: Certification Request Syntax Specification Version 1.7", RFC 2986, November 2000.

[RFC3779] Lynn, C., Kent, S., and K. Seo, "X.509 Extensions for IP Addresses and AS Identifiers", RFC 3779, June 2004.

[RFC4211] Schaad, J., "Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)", RFC 4211, September 2005.

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

[RFC5781] Weiler, S., Ward, D., and R. Housley, "The rsync URI Scheme", RFC 5781, February 2010.

[RFC6484] Kent, S., Kong, D., Seo, K., and R. Watro, "Certificate Policy (CP) for the Resource Public Key Infrastructure (RPKI)", BCP 173, RFC 6484, February 2012.

[RFC6485] Huston, G., "The Profile for Algorithms and Key Sizes for Use in the Resource Public Key Infrastructure (RPKI)", RFC 6485, February 2012.

[X.509] ITU-T, , "Recommendation X.509: The Directory - Authentication Framework", 2000.

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

13.2. Informative References

[RFC2985] Nystrom, M. and B. Kaliski, "PKCS #9: Selected Object Classes and Attribute Types Version 2.0", RFC 2985, November 2000.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.

[RFC4122] Leach, P., Mealling, M., and R. Salz, "A Universally Unique IDentifier (UUID) URN Namespace", RFC 4122, July 2005.

[RFC4514] Zeilenga, K., "Lightweight Directory Access Protocol (LDAP): String Representation of Distinguished Names", RFC 4514, June 2006.

[RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, February 2012.

[RFC6481] Huston, G., Loomans, R., and G. Michaelson, "A Profile for Resource Certificate Repository Structure", RFC 6481, February 2012.

[RFC6486] Austein, R., Huston, G., Kent, S., and M. Lepinski, "Manifests for the Resource Public Key Infrastructure (RPKI)", RFC 6486, February 2012.

[RFC6487] Huston, G., Michaelson, G., and R. Loomans, "A Profile for X.509 PKIX Resource Certificates", RFC 6487, February 2012.

[RFC6488] Lepinski, M., Chi, A., and S. Kent, "Signed Object Template for the Resource Public Key Infrastructure (RPKI)", RFC 6488, February 2012.

[RFC6489] Huston, G., Michaelson, G., and S. Kent, "Certification Authority (CA) Key Rollover in the Resource Public Key Infrastructure (RPKI)", BCP 174, RFC 6489, February 2012.

[RFC7318] Newton, A. and G. Huston, "Policy Qualifiers in Resource Public Key Infrastructure (RPKI) Certificates", RFC 7318, July 2014.

[RPKI-ALG] Gagliano, R., Kent, S., and S. Turner, "Algorithm Agility Procedure for RPKI", Work in Progress, November 2011.

Appendix A. Example Resource Certificate

The following is an example resource certificate.

Certificate Name: 9JfgAEcq7Q-47IwMC5CJIJr6EJs.cer

```
Data:
  Version: 3 (0x2)
  Serial: 1500 (0x5dc)
  Signature Algorithm: SHA256WithRSAEncryption
  Issuer: CN=APNIC Production-CVPQSgUkLy7pOXdNeVWGvnFX_0s
  Validity
   Not Before: Oct 25 12:50:00 2008 GMT
    Not After : Jan 31 00:00:00 2010 GMT
  Subject: CN=A91872ED
  Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (2048 bit)
    Modulus (2048 bit):
      00:bb:fb:4a:af:a4:b9:dc:d0:fa:6f:67:cc:27:39:
      34:d1:80:40:37:de:88:d1:64:a2:f1:b3:fa:c6:7f:
      bb:51:df:e1:c7:13:92:c3:c8:a2:aa:8c:d1:11:b3:
      aa:99:c0:ac:54:d3:65:83:c6:13:bf:0d:9f:33:2d:
      39:9f:ab:5f:cd:a3:e9:a1:fb:80:7d:1d:d0:2b:48:
      a5:55:e6:24:1f:06:41:35:1d:00:da:1f:99:85:13:
      26:39:24:c5:9a:81:15:98:fb:5f:f9:84:38:e5:d6:
      70:ce:5a:02:ca:dd:61:85:b3:43:2d:0b:35:d5:91:
      98:9d:da:1e:0f:c2:f6:97:b7:97:3e:e6:fc:c1:c4:
      3f:30:c4:81:03:25:99:09:4c:e2:4a:85:e7:46:4b:
      60:63:02:43:46:51:4d:ed:fd:a1:06:84:f1:4e:98:
      32:da:27:ee:80:82:d4:6b:cf:31:ea:21:af:6f:bd:
      70:34:e9:3f:d7:e4:24:cd:b8:e0:0f:8e:80:eb:11:
      1f:bc:c5:7e:05:8e:5c:7b:96:26:f8:2c:17:30:7d:
      08:9e:a4:72:66:f5:ca:23:2b:f2:ce:54:ec:4d:d9:
      d9:81:72:80:19:95:57:da:91:00:d9:b1:e8:8c:33:
      4a:9d:3c:4a:94:bf:74:4c:30:72:9b:1e:f5:8b:00:
      4d:e3
    Exponent: 65537 (0x10001)
  X509v3 extensions:
    X509v3 Subject Key Identifier:
      F4:97:E0:00:47:2A:ED:0F:B8:EC:8C:0C:0B:90:89:
      20:9A:FA:10:9B

    X509v3 Authority Key Identifier:
      keyid:09:53:D0:4A:05:24:2F:2E:E9:39:77:4D:79:
      55:86:BE:71:57:FF:4B

    X509v3 Key Usage: critical
```

```
        Certificate Sign, CRL Sign

      X509v3 Basic Constraints: critical
        CA:TRUE

      X509v3 CRL Distribution Points:
        URI:rsync://rpki.apnic.net/repository/A3C38A24
            D60311DCAB08F31979BDBE39/CVPQSgUkLy7pOXdNe
            VWGvnFX_0s.crl

      Authority Information Access:
         CA Issuers - URI:rsync://rpki.apnic.net/repos
            itory/8BDFC7DED5FD11DCB14CF4B1A703F9B7/CVP
            QSgUkLy7pOXdNeVWGvnFX_0s.cer

      X509v3 Certificate Policies: critical
         Policy: 1.3.6.1.5.5.7.14.2

      Subject Information Access:
         CA Repository - URI:rsync://rpki.apnic.net/mem
             ber_repository/A91872ED/06A83982887911DD81
             3F432B2086D636/
         Manifest - URI:rsync://rpki.apnic.net/member_r
             epository/A91872ED/06A83982887911DD813F432
             B2086D636/9JfgAEcq7Q-47IwMC5CJIJr6EJs.mft

       sbgp-autonomousSysNum: critical
         Autonomous System Numbers:
           24021
           38610
           131072
           131074

       sbgp-ipAddrBlock: critical
         IPv4:
           203.133.248.0/22
           203.147.108.0/23

   Signature Algorithm: sha256WithRSAEncryption
       51:4c:77:e4:21:64:80:e9:35:30:20:9f:d8:4b:88:60:b8:1f:
       73:24:9d:b5:17:60:65:6a:28:cc:43:4b:68:97:ca:76:07:eb:
       dc:bd:a2:08:3c:8c:56:38:c6:0a:1e:a8:af:f5:b9:42:02:6b:
       77:e0:b1:1c:4a:88:e6:6f:b6:17:d3:59:41:d7:a0:62:86:59:
       29:79:26:76:34:d1:16:2d:75:05:cb:b2:99:bf:ca:c6:68:1b:
       b6:a9:b0:f4:43:2e:df:e3:7f:3c:b3:72:1a:99:fa:5d:94:a1:
       eb:57:9c:9a:2c:87:d6:40:32:c9:ff:a6:54:b8:91:87:fd:90:
       55:ef:12:3e:1e:2e:cf:c5:ea:c3:4c:09:62:4f:88:00:a0:7f:
       cd:67:83:bc:27:e1:74:2c:18:4e:3f:12:1d:ef:29:0f:e3:27:
```

```
00:ce:14:eb:f0:01:f0:36:25:a2:33:a8:c6:2f:31:18:22:30:
cf:ca:97:43:ed:84:75:53:ab:b7:6c:75:f7:2f:55:5c:2e:82:
0a:be:91:59:bf:c9:06:ef:bb:b4:a2:71:9e:03:b1:25:8e:29:
7a:30:88:66:b4:f2:16:6e:df:ad:78:ff:d3:b2:9c:29:48:e3:
be:87:5c:fc:20:2b:df:da:ca:30:58:c3:04:c9:63:72:48:8c:
0a:5f:97:71
```

Appendix B. Example Certificate Revocation List

The following is an example Certificate Revocation List.

```
   CRL Name: q66IrWSGuBE7jqx8PAUHAlHCqRw.crl
   Data:
     Version: 2
     Signature Algorithm:
       Hash: SHA256, Encryption: RSA
     Issuer: CN=Demo Production APNIC CA - Not for real use,
       E=ca@apnic.net
     This Update: Thu Jul 27 06:30:34 2006 GMT
     Next Update: Fri Jul 28 06:30:34 2006 GMT
     Authority Key Identifier: Key Identifier:
       ab:ae:88:ad:64:86:b8:11:3b:8e:ac:7c:3c:05:
       07:02:51:c2:a9:1c
     CRLNumber: 4
     Revoked Certificates: 1
       Serial Number: 1
       Revocation Date: Mon Jul 17 05:10:19 2006 GMT
       Serial Number: 2
       Revocation Date: Mon Jul 17 05:12:25 2006 GMT
       Serial Number: 4
       Revocation Date: Mon Jul 17 05:40:39 2006 GMT
     Signature:
       b2:5a:e8:7c:bd:a8:00:0f:03:1a:17:fd:40:2c:46:
       0e:d5:64:87:e7:e7:bc:10:7d:b6:3e:39:21:a9:12:
       f4:5a:d8:b8:d4:bd:57:1a:7d:2f:7c:0d:c6:4f:27:
       17:c8:0e:ae:8c:89:ff:00:f7:81:97:c3:a1:6a:0a:
       f7:d2:46:06:9a:d1:d5:4d:78:e1:b7:b0:58:4d:09:
       d6:7c:1e:a0:40:af:86:5d:8c:c9:48:f6:e6:20:2e:
       b9:b6:81:03:0b:51:ac:23:db:9f:c1:8e:d6:94:54:
       66:a5:68:52:ee:dd:0f:10:5d:21:b8:b8:19:ff:29:
       6f:51:2e:c8:74:5c:2a:d2:c5:fa:99:eb:c5:c2:a2:
       d0:96:fc:54:b3:ba:80:4b:92:7f:85:54:76:c9:12:
       cb:32:ea:1d:12:7b:f8:f9:a2:5c:a1:b1:06:8e:d8:
       c5:42:61:00:8c:f6:33:11:29:df:6e:b2:cc:c3:7c:
       d3:f3:0c:8d:5c:49:a5:fb:49:fd:e7:c4:73:68:0a:
       09:0e:6d:68:a9:06:52:3a:36:4f:19:47:83:59:da:
       02:5b:2a:d0:8a:7a:33:0a:d5:ce:be:b5:a2:7d:8d:
       59:a1:9d:ee:60:ce:77:3d:e1:86:9a:84:93:90:9f:
       34:a7:02:40:59:3a:a5:d1:18:fb:6f:fc:af:d4:02:
       d9
```

Appendix C. Differences from RFC 6487

The following changes were made since [RFC6487]:

- The changes from all verified errata have been incorporated.
- The changes from [RFC7318] have been incorporated.

- Section 4.8 was corrected to say that certificate extensions not mentioned in this document are rejected. The text now agrees with the text in Section 1 and Section 8.

- Section 5 now specifies the format of the Authority Key Identifier extension.

Authors' Addresses

Geoff Huston
APNIC

EMail: gih@apnic.net
URI: http://www.apnic.net

George Michaelson
APNIC

EMail: ggm@apnic.net
URI: http://www.apnic.net

Robert Loomans
Suncorp

EMail: Robert.Loomans@suncorp.com.au
URI: http://www.suncorpgroup.com.au/

Andrew Lee Newton
American Registry for Internet Numbers
3635 Concorde Parkway
Chantilly, VA 20151
USA

EMail: andy@arin.net
URI: http://www.arin.net

Richard Hansen
BBN Technologies
10 Moulton St
Cambridge, MA 02138-1119
USA

EMail: rhansen@bbn.com