SIDR
Internet-Draft
Intended status: Standards Track
Expires: April 17, 2011

G. Huston
G. Michaelson
R. Loomans
APNIC

October 14, 2010

# A Profile for X.509 PKIX Resource Certificates
## draft-ietf-sidr-res-certs-19

## Abstract

This document defines a standard profile for X.509 certificates for the purposes of supporting validation of assertions of "right-of-use" of Resources (INRs). The certificates issued under this profile are used to convey the Issuer's authorisation 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). The document also specifies profiles for the format of certificate requests. The document also specifies the Relying Party RPKI certificate path validation procedure.

## 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 April 17, 2011.

## Copyright Notice

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) [ID.sidr-cp]. 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 [ID.sidr-cp]. 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.

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:

- additional validation processing imposed by the INR extensions is required,
- a conformation of a public key match between the CRL issuer and the Resource Certificate issuer is required, and
- 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 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 RFC 2119.

## 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:

- 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, a 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 [ID.sidr-arch], 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. There are two types of EE certificates defined within the RPKI framework: single-use and multi-use.





### 3.1. Single-Use EE Certificates

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

### 3.2. Multi-Use EE Certificates

If the private key associated with an EE certificate is intended to be used to validate more than one RPKI signed object, then the

certificate is termed a "multi-use" EE certificate. All objects that can be verified under a multi-use EE certificate are revoked when the certificate is revoked.

## 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 field 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 [ID.sidr-rpki-algs].

### 4.4. Issuer

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

An Issuer name MUST contain one instance of the Common Name attribute and MAY contain one instance of the Serial Number attribute. If both attributes are present, it is RECOMMENDED that they appear as a set. The Common Name attribute MUST be encoded as a printable string. 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 fulfil this recommendation.

## 4.5. Subject

The value of this field is a valid X.501 distinguished name, 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 [ID.sidr-repos-struct]. 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 rekeying the Subject.) Subject names are not intended to be descriptive of the identity of Subject.

## 4.6. Valid From

The "Valid From" 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 will be valid at a time in the future, as the scope of a relying party's test of validity of a certificate refers specifically to





validity at the current time.

## 4.7. Valid To

The Valid To 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.

While a CA is typically advised against issuing a certificate with a validity interval that exceeds the validity interval 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 certificate with a validity interval that exceeds the validity interval of its certificate.

## 4.8. Subject Public Key Info

The algorithm used in this profile is specified in [ID.sidr-rpki-algs].

## 4.9. 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 a critical extension it does not recognise; however, a non-critical extension MAY be ignored if it is not recognised [RFC5280].

### 4.9.1. 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.

### 4.9.2. 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].

### 4.9.3. Authority Key Identifier

This extension MUST appear in all Resource Certificates, with the exception of a CA who issues a "self-signed" certificate. 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].

### 4.9.4. 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.

### 4.9.5. 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.

### 4.9.6. 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 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 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.

### 4.9.7. 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 [ID.sidr-repos-struct]. This implies that a re-issued certificate overwrites a previously issued certificate (to the same Subject) in the publication repository. In this way certificates





subordinate to the re-issued (CA) certificate can maintain a constant Authority Information Access (AIA) extension pointer and thus need not be re-issued when the parent certificate is re-issued.

### 4.9.8. Subject Information Access

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

#### 4.9.8.1. SIA for CAs

This extension MUST be present, and is 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 material issued by this CA. i.e., all valid CA certificates, multi-use EE certificates, the current CRL, manifest and signed objects signed by single-use EE certificates that have been issued by this CA [ID.sidr-repos-struct]. 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 relying parties, with he 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 [ID.sidr-rpki-manifests] 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.

#### 4.9.8.2. SIA for Multi-use EEs

This extension MUST be present, and is non-critical.

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





id-ad-signedObjectRepository OBJECT IDENTIFIER ::= { id-ad 9 }

with an accessLocation form of a URI that MUST specify an RSYNC URI [RFC5781]. This URI points to the directory containing all signed objects that are verified using this EE certificate [ID.sidr-repos-struct]. Other accessDescription elements MAY exist for the id-ad-signedObjectRepository accessMethod, where the accessLocation value indicates alternate supported access mechanisms for the same directory, ordered in terms of the EE's relative preference for supported access mechanisms.

This extension MUST have an instance of an AccessDescription with an accessMethod of id-ad-rpkiManifest, with the same specification as the CA's manifest.

#### 4.9.8.3. SIA for Single-use EEs

This extension MUST be present, and is 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 a RSYNC URI

[RFC5781]. This URI points to the signed object that is verified using this EE certificate [ID.sidr-repos-struct]. 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 a single-use EE's SIA.

### 4.9.9. Certificate Policies

This extension MUST be present, and MUST be marked critical. It MUST include exactly one policy, as specified in the RPKI CP [ID.sidr-cp]

### 4.9.10. 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 AFI value. In the context of resource certificates





describing public number resources for use in the public Internet, the SAFI value MUST NOT be used.

This extension MUST either specify a non-empty set 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.

### 4.9.11. 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 may specify the "inherit" element. RDI values are NOT supported in this profile and MUST NOT be used.

This extension MUST either specify a non-empty set 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 Certificate Revocation List (CRL), 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 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 [ID.sidr-rpki-algs].

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. RPs MUST be prepared to process CRLs with these extensions. No other CRL extensions are allowed.





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 and multi-use EE certificate requests, the CA places the Resource Certificate in the repository, as per [ID.sidr-cp]. No response is defined for single-use 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 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 rekey/reissuance request, and only if the CA has adopted a policy (in its Certificate Practice Statement (CPS)) that permits name reuse 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 [ID.sidr-rpki-algs].

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 constraints to fields that MAY appear in a CertificationRequest Object:

signatureAlgorithm
: The signatureAlgorithm value is specified in [ID.sidr-rpki-algs].

## 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. It

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 rekey/reissuance request, and only if the CA has adopted a policy (in its CPS) that permits name reuse 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 honour 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 honour 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 honour 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 honour ExtendedKeyUsage extensions of keyCertSign and cRLSign if present, as long as this is consistent with the BasicConstraints SubjectType sub field, when specified.

SubjectInformationAccess
: This field MUST be present, and the field value SHOULD be honoured by the CA if it conforms to the requirements set forth in Section 4.9.8. If the CA is unable to honour 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 definitions [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 IP address or AS number contiguous 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 IP address or AS number contiguous ranges, A and B, A is "equal" to B if range A describes precisely the same collection of IP addresses or AS numbers as 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. That 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 synchronisation across the distributed publication repository structure [ID.sidr-repos-struct].

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 DOS attack on a RP. A RP executing this procedure MAY apply further heuristics to guide halting the certification path validation process 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 RFC2119 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 define use. In this context it is not seen as being appropriate to be in the position of having certificates with additional non-critical extensions that RPs may see as valid certificates without understanding the extensions, but were the RP in a position to understand the extensions, would contradict or qualify in some way this original judgment of validity. 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 [ID.sidr-keyroll].

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 [ID.sidr-keyroll].

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 wanted, as long as it conformed to the rules for PrintableString. The separator could be a space character, parenthesis, hyphen, slash, question mark, etc.

## 9. 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 is the same key used to sign the CRL that can revoke the certificate, implying that the certificate path used to validate a signature on a certificates is the same as that used to validate a signatures the CRL that revokes the certificate. It is noted that this is a higher 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.

## 10. IANA Considerations

[Note to IANA, to be removed prior to publication: there are no IANA considerations stated in this document.]

## 11. 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 text. 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.

## 12. References

### 12.1. Normative References

[ID.sidr-cp]
Kent, S., Kong, D., Seo, K., and R. Watro, "Certificate Policy (CP) for the Resource PKI (RPKI)", Work in progress: Internet Drafts draft-ietf-sidr-c-13.txt, September 2010.

[ID.sidr-rpki-algs]
Huston, G., "A Profile for Algorithms and Key Sizes for

use in the Resource Public Key Infrastructure", Work in progress: Internet
Drafts draft-ietf-sidr-rpki-algs-00.txt, August 2009.

[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.

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

## 12.2. Informative References

[ID.sidr-arch]
Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", Work in progress: Internet Drafts draft-ietf-sidr-arch-04.txt, November 2008.

[ID.sidr-keyroll]
Huston, G., Michaelson, G., and S. Kent, "CA Key Rollover in the RPKI", draft-ietf-sidr-keyroll-02.txt (work in progress), October 2010.

[ID.sidr-repos-struct]
Huston, G., Loomans, R., and G. Michaleson, "A Profile for Resource Certificate Repository Structure", draft-ietf-sidr-repos-struct-04.txt (work in progress), May 2010.

[ID.sidr-rpki-manifests]

Austein, R., Huston, G., Kent, S., and M. Lepinski,
"Manifests for the Resource Public Key Infrastructure",
Work in progress: Internet
Drafts draft-ietf-sidr-rpki-manifests-04.txt,
October 2008.





[ID.sidr-signed-object]
Lepinski, M., Chi, A., and S. Kent, "Signed Object
Template for the Resource Public Key Infrastructure",
draft-ietf-sidr-signed-object-01.txt (work in progress),
October 2010.

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

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

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

## 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
  Authority Key Identifier: Key Identifier g(AKI):
    q66IrWSGuBE7jqx8PAUHAlHCqRw
```

```
    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
```





## 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
APNIC

Email: robertl@apnic.net
URI: http://www.apnic.net